Contacting BMC Software

Several methods are available for contacting BMC Software.

You can access the BMC Software website at http://www.bmc.com. From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

United States and Canada

Address  
BMC SOFTWARE INC  
2103 CITYWEST BLVD  
HOUSTON TX 77042-2827 USA

Telephone  
1 713 918 8800  
or  
1 800 841 2031

Fax  
1 713 918 8000

Outside United States and Canada

Telephone  
+01 713 918 8800  
Fax  
+01 713 918 8000

© Copyright 1994-2016 BMC Software, Inc.

BMC, BMC Software, the BMC logo, the BMC Software logo, and other BMC marks are the exclusive properties of BMC Software, Inc. and are registered or may be registered in the U.S. and in other countries. BladeLogic and other BladeLogic marks are the exclusive properties of BladeLogic, Inc. and are registered or may be registered in the U.S. and in other countries. All other trademarks or registered trademarks are the property of their respective owners.

DB2, FlashCopy, IBM, IBM z, MVS, RACF, RETAIN, System z, and z/OS are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

R/3 and SAP are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

UNIX is the registered trademark of The Open Group in the US and other countries.

The information included in this documentation is the proprietary and confidential information of BMC Software, Inc., its affiliates, or licensors. Your use of this information is subject to the terms and conditions of the applicable End User License agreement for the product and to the proprietary and restricted rights notices included in the product documentation.

Restricted rights legend

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED—RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Section 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC SOFTWARE INC, 2103 CITYWEST BLVD, HOUSTON TX 77042-2827, USA. Any contract notices should be sent to this address.
Customer support

Support website
You can obtain technical support from BMC 24 hours a day, 7 days a week at [http://www.bmc.com/support](http://www.bmc.com/support). From this website, you can:

- Read overviews about support services and programs that BMC offers
- Find the most current information about BMC products
- Search a database for problems similar to yours and possible solutions
- Order or download product documentation
- Download products and maintenance
- Report a problem or ask a question
- Subscribe to receive proactive e-mail alerts
- Find worldwide BMC support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by telephone or e-mail
In the United States and Canada, if you need technical support and do not have access to the web, call 1 800 537 1813 or send an e-mail message to customer_support@bmc.com. (In the subject line, enter SupID:yourSupportContractID, such as SupID:12345). Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC
Have the following information available so that Customer Support can begin working on your issue immediately:

- Product information
  - Product name
  - Product version (release number)
  - License number and password (trial or permanent)
- Operating system and environment information
  - Machine type
  - Operating system type, version, and service pack or other maintenance level such as PUT or PTF
  - System hardware configuration
  - Serial numbers
  - Related software (database, application, and communication) including type, version, and service pack or maintenance level
- Sequence of events leading to the problem
- Commands and options that you used
- Messages received (and the time and date that you received them)
  - Product error messages
  - Messages from the operating system
  - Messages from related software
License key and password information

If you have questions about your license key or password, contact Customer Support through one of the following methods:

- Send an e-mail message to customer_support@bmc.com. (In the Subject line, enter SupID:yourSupportContractID, such as SupID:12345.)
- In the United States and Canada, call 1 800 537 1813. Outside the United States and Canada, contact your local support center for assistance.
Contents

About this book 23
Related publications ................................................................. 23
Conventions ........................................................................ 24
Syntax diagrams .................................................................... 24
Summary of changes .......................................................... 25
Summary of changes Version 12.1.00 December 2016 .......... 26
  Support for IBM DB2 Version 12 ........................................ 26
  RECOVER PLUS for DB2 name change ............................ 26
  COPY PLUS for DB2 name change .................................... 26
  Recovery Progress Reports renamed ................................. 26
  REBUILD INDEX SHRLEVEL CHANGE ........................... 26
  Forward recovery of indexes to a timestamp .................... 27
  New ARMBGRP syntax ................................................... 27
  ARMBGPV populates the RECOVERY_EXCEPTIONS table .... 27
  ARMBSR populates the RECOVERY_EXCEPTIONS table .... 28
  ARMGA001 GROUP Authorization panel ............................ 28
  ARMBGRP supports group name patterns for QUERY GROUP INFO,
  QUERY GROUP OBJECTS, and QUERY GROUP AUTH .......... 28
  Simulate Recovery option for Progress Reports ................ 28
  Dynamic grouping .......................................................... 28
  Exporting and migrating groups ........................................ 29
  Option to recover work file database ............................... 29
  BACKOUT AUTO time estimation .................................... 30
  Delimited object set names ............................................ 30
  RECOVER_SCOPE and REBUILD_SCOPE options .......... 30
  Removal of restrictions on recovery simulation ............... 31
  New &PART4 symbolic variable ...................................... 31
  Symbolic variable substrings ......................................... 31
  Multi-job JCL generation .............................................. 32
  Reports with ANSI printer carriage control characters ...... 32
  SEPARATE_BY_PARTITION option ................................. 32
  Object set management in BMC Workbench .................. 32
Summary of changes Version 11.2.00 May 2015 .................. 33
Summary of changes Version 11.1.00 June 2013 .................. 35
RECOVERY MANAGER changes 11.1.00 ............................... 35
Changes for COPY PLUS 11.1.00 ....................................... 41
Changes for RECOVER PLUS 11.1.00 .................................................................41
Changes for Recovery Management 11.1.00 .......................................................41

Part 1 RECOVERY MANAGER for DB2

Chapter 1 Introducing RECOVERY MANAGER

Overview of RECOVERY MANAGER .................................................................45
The BMC Recovery Management for DB2 solution .................................................46
RECOVERY MANAGER concepts and functionality ............................................46
   How RECOVERY MANAGER works .................................................................47
   Creating application groups for backup or recovery .........................................49
   Specifying utilities and syntax options ...........................................................49
   Validating and revalidating application groups ...............................................50
   Maintaining backing up and recovering system resources ..............................50
Generating backup and recovery JCL ..................................................................0
Output data sets, job cards, and symbolic variables ............................................0
RECOVERY MANAGER batch programs ..............................................................0
System setup .......................................................................................................0

Chapter 2 Getting started with RECOVERY MANAGER

For first time users of RECOVERY MANAGER .................................................0
RECOVERY MANAGER task overview ...............................................................0
Before using RECOVERY MANAGER ...............................................................0
   Adding a DB2 subsystem to RECOVERY MANAGER .......................................0
   RECOVERY MANAGER and PACLOG option set considerations .....................0
   Sharing BMC tables .......................................................................................0
   Sharing solution common code (SCC) ............................................................0
   Setting up DB2 group attach names ..............................................................0
   Creating required temporary tables ..............................................................0
   Preparation for archive logs greater than 64K tracks ......................................0
   Safeguarding the repository ..........................................................................0
   Utilities supported by RECOVERY MANAGER .............................................0
   DB2 system resource maintenance and recovery ..........................................0
   Performance considerations ..........................................................................0
   Avoid RUNSTATS on BMCLGRNX ...............................................................0
   Improving recovery time ..............................................................................0
   Multiple job optimization .............................................................................0
   Improving catalog and directory recovery ....................................................0
   Enhancing JCL generation and performance ................................................0
   Interactive versus batch processing ................................................................0
   Interactive versus batch group creation ......................................................0

6 RECOVERY MANAGER for DB2 User Guide
Setting group authorizations ................................................................. 128
Renaming groups .................................................................................. 129
Validating the objects in a group ............................................................. 130
  Verifying object availability to DB2 ................................................ 131
  Verifying eligibility for backup or recovery .................................... 131
  Verifying object recoverability ......................................................... 132
  Responding to unsatisfactory object status ...................................... 132
Group recovery revalidation and reporting ............................................. 133
  Revalidating and reporting on groups in batch .............................. 134
  Using plan and package impact analysis and reporting ............... 139
Using RECOVERY MANAGER object sets for backup and recovery backup and 
recovery setup ....................................................................................... 143

Chapter 4    Backing up groups  .................................................. 157
About backup options ........................................................................... 157
Displaying, updating, and deleting backup options for standard copies ................................................................. 158
Setting backup options for Instant Snapshots ........................................ 159
Setting backup options for online consistent copies ............................ 161
Setting backup options for cabinet copies ............................................ 162
Generating a backup job interactively .................................................... 163
Generating a backup job in batch ........................................................... 164
Generating a batch ARMBGEN job interactively .................................. 164
Restarting a failed backup job ................................................................. 166
Backup option descriptions ................................................................. 167
  General backup options ................................................................... 167
  NGT Copy-specific options .............................................................. 169
  DB2 COPY (DSNUTILB) options ...................................................... 180
  NGT Recover (OUTCOPY) options .................................................. 183
  Output copy data set options .......................................................... 186
About incremental index copies ............................................................. 193
About Instant Snapshots ....................................................................... 193
  Allocation of Instant Snapshots ....................................................... 194
  Registration of Instant Snapshots .................................................... 195
  Command option restrictions for Instant Snapshots ....................... 196
  OUTPUT command options applied to Instant Snapshots ............. 196
  COPY command options ignored for Instant Snapshots ............... 197
  DSNUM and Instant Snapshots ......................................................... 197
  SHRLEVEL and Instant Snapshots .................................................... 197
  Instant Snapshot use by other BMC utilities .................................. 198

Chapter 5    Recovering groups .......................................................... 199
Preparing recovery jobs ................................................................. 199
Supporting OBJECTSET ................................................................. 200
BACKOUT recovery ................................................................. 201
LOB and XML object recovery ......................................................... 202
NOT LOGGED object recovery ........................................................ 204
Timestamp recovery ................................................................. 205
Log mark recovery ................................................................. 206
Recovery simulation for application spaces ............................................... 206
Optimized recovery JCL ................................................................. 206
Choosing interactive or batch generation ............................................... 208
Generating recovery JCL interactively .................................................... 208
Generating recovery JCL in batch ........................................................ 210
Using ARMBGEN in disaster recovery planning ......................................... 211
Generating batch recovery JCL interactively ................................................. 211
Options for recovery JCL generation .................................................... 213
Actions on objects after recovery point selection .......................................... 220
Object status after recovery point selection ................................................ 221
Submitting your recovery job ............................................................. 221
Restarting failed recovery jobs .............................................................. 222
Restarting a recovery for a set of concurrent jobs ............................................ 223
Restarting jobs that recover application data ................................................ 224
Restarting system resource recovery (ARMBSRR) jobs .................................... 226
Restarting synchronized jobs .............................................................. 227
Working with recovery options ............................................................ 229
Maximizing concurrency of key sorts ......................................................... 230
Displaying, updating, and deleting recovery options ......................................... 231
General recovery options ........................................................................ 233
NGT Recover options ........................................................................ 238
Alternate recovery resource options ........................................................ 238
DB2 RECOVER options ........................................................................ 249
Work file option descriptions ................................................................ 251
Output data set option descriptions .......................................................... 254
Symbolic variables in post-recovery image copy data set names ................................ 259

Chapter 6    Exporting and migrating groups  261
Task title ......................................................................................... 261

Chapter 7    Managing DB2 system resources  265
System resources ........................................................................... 265
DB2 subsystem status ........................................................................ 266
Authorizations required to access system resources ........................................ 266
## Chapter 10 Modeling the DB2 logging environment

- About the logging environment modeling tool ................................................................. 369
- Viewing and modeling logging environment statistics .................................................... 370
  
  **Log environment model field descriptions** ................................................................. 373
- Viewing active log information ...................................................................................... 379
  
  **Active log field descriptions** .................................................................................... 380
- Viewing archive log information .................................................................................. 380
  
  **Archive log 1 and 2 detail field descriptions** ............................................................... 381
- Audit synchronization ..................................................................................................... 382

## Chapter 11 Accessing online Progress Reports

- About Progress Reports .................................................................................................. 383
- Information in Progress Reports .................................................................................... 384
- Viewing Progress Report information .......................................................................... 385
  
  **Viewing Progress Reports for recovered, changed, and copied objects** ................ 386
- Generating JCL for ARMBRPR .................................................................................... 389
- Viewing information for backup time estimates ......................................................... 389
- Viewing detailed information for recover time estimates ............................................ 392
- Viewing and saving backup or recover time estimates ................................................ 394
- Entry field descriptions for the Progress Report panels ............................................... 395

## Part 2 RECOVERY MANAGER batch programs

### Chapter 12 ARMBACT—Initialize active logs with DSNJLOGF

- About ARMBACT .......................................................................................................... 403
- Authorizations ............................................................................................................... 403
- Building the ARMBACT JCL ....................................................................................... 403
  
  **Specifying the JOB statement** ................................................................................ 404
- Specifying the EXEC statement ................................................................................... 404
- Specifying the STEPLIB DD statement ...................................................................... 405
- Specifying the ARMBACT data set DD statements ..................................................... 405
- Sample JCL .................................................................................................................. 406
- Sample output ............................................................................................................... 406
- Executing the JCL ......................................................................................................... 407

### Chapter 13 ARMBARC—Archive log data sets

- About ARMBARC ......................................................................................................... 409
- Authorizations ............................................................................................................... 410
- Building the ARMBARC JCL ....................................................................................... 410
Specifying the JOB statement ................................................................. 411
Specifying the EXEC statement .............................................................. 411
Specifying the STEPLIB DD statement ................................................... 412
Specifying the ARMBARC data set DD statements ............................... 412
Control cards and syntax ........................................................................ 413
HISTONLY .............................................................................................. 415
Copy specifications ................................................................................ 415
Archive options ...................................................................................... 416
Filter options ......................................................................................... 419
Global options ....................................................................................... 420
Sample JCL ............................................................................................. 421
Sample output ......................................................................................... 422
Executing the JCL .................................................................................. 424

Chapter 14          ARMBCRC—Conditional recovery to a timestamp 425
About ARMBCRC .................................................................................... 425
Establishing a recovery point ................................................................. 426
Authorizations ....................................................................................... 426
Building the ARMBCRC JCL ................................................................. 426
   Specifying the JOB statement .......................................................... 427
   Specifying the EXEC statement ....................................................... 427
   Specifying the STEPLIB DD statement ............................................ 428
   Specifying the ARMBCRC data set DD statements ......................... 428
Sample JCL ............................................................................................ 429
Sample output ....................................................................................... 429
Executing the JCL .................................................................................. 430

Chapter 15          ARMBGEN—Backup and recovery JCL 431
About ARMBGEN .................................................................................... 431
   About XUNCHANGED processing in local subsystem recovery .......... 432
   Using ARMBGEN in full subsystem recovery .................................. 433
   Using ARMBGEN in disaster recovery planning ............................... 433
   About BACKOUT recovery ............................................................... 434
Authorizations ....................................................................................... 435
Building the ARMBGEN JCL ................................................................. 436
   Specifying the JOB statement ....................................................... 436
   Specifying the EXEC statement ...................................................... 436
   Specifying the STEPLIB DD statement for ARMBGEN JCL .......... 437
   Specifying the ARMBGEN data set DD statements ....................... 438
Building the ARMBGNR JCL ................................................................. 441
   Specifying the EXEC statement ...................................................... 441
Specifying the STEPLIB DD statement for ARMBGNR JCL ........................................... 442
Specifying the ARMBGNR data set DD statements ......................................................... 442
ARMBGEN syntax and option descriptions ...................................................................... 443
Syntax diagrams ................................................................................................................. 443
Option descriptions ............................................................................................................. 446
Sample JCL .......................................................................................................................... 457
Sample output ....................................................................................................................... 460
Executing the JCL .................................................................................................................. 463

Chapter 16 ARMBGIM—Impact analysis 465
About ARMBGIM .................................................................................................................. 465
Authorizations ...................................................................................................................... 465
Building the ARMBGIM JCL .............................................................................................. 466
  Specifying the JOB statement .......................................................................................... 466
  Specifying the EXEC statement ....................................................................................... 466
  Specifying the STEPLIB DD statement .......................................................................... 467
  Specifying the ARMBGIM data set DD statements .......................................................... 467
ARMBGIM syntax and option descriptions ........................................................................ 468
  SET CURRENT SQLID ..................................................................................................... 469
  REPORT GROUP ............................................................................................................... 469
Sample JCL .......................................................................................................................... 470
Sample output ....................................................................................................................... 470
Executing the JCL .................................................................................................................. 471

Chapter 17 ARMBGPS—Subsystem group split 473
About ARMBGPS .................................................................................................................. 473
  Delta groups ...................................................................................................................... 474
  How ARMBGPS builds multiple groups ........................................................................... 474
  Revalidating ARMBGPS groups ....................................................................................... 479
Authorizations ...................................................................................................................... 479
Building the JCL ................................................................................................................... 479
  Specifying the JOB statement .......................................................................................... 480
  Specifying the EXEC statement ....................................................................................... 480
  Specifying the STEPLIB DD statement .......................................................................... 480
  Specifying the ARMBGPS data set DD statements .......................................................... 481
ARMBGPS syntax and option descriptions ........................................................................ 481
  SET CURRENT SQLID ..................................................................................................... 483
  BUILD GROUPS ............................................................................................................... 483
Sample JCL .......................................................................................................................... 485
Sample output ....................................................................................................................... 486
Executing the JCL .................................................................................................................. 487
Syntax for updating group options ................................................................. 546
UPDATE GROUP option description ............................................................... 559
RENAME GROUP option descriptions ........................................................... 560
Syntax for renaming groups ................................................................. 560
RENAME GROUP option descriptions .......................................................... 561
DELETE GROUP option descriptions ............................................................ 561
Syntax for deleting groups ................................................................. 561
DELETE GROUP option descriptions ............................................................. 562
REPORT GROUP option descriptions ......................................................... 562
Syntax for reporting on groups ................................................................. 563
REPORT GROUP option descriptions ............................................................ 563
QUERY option descriptions ................................................................. 564
Syntax for querying groups ................................................................. 565
QUERY ................................................................. 565
EXPORT GROUP option descriptions ........................................................... 567
Syntax for exporting groups ................................................................. 568
EXPORT GROUP command ................................................................. 568
MIGRATE_TO_SSID option descriptions ..................................................... 568
Syntax for migrating groups ................................................................. 568
MIGRATE_TO_SSID option descriptions ..................................................... 569
RECOVER_SCOPE option descriptions ....................................................... 571
Syntax for RECOVER_SCOPE ................................................................. 571
REBUILD_SCOPE option descriptions ....................................................... 573
Syntax for REBUILD_SCOPE ................................................................. 573
Sample JCL and output .................................................................................. 573
CREATE GROUP ................................................................. 573
CREATE GROUP VIA SQL ................................................................. 577
CREATE GROUP (Volume, VVDS) ................................................................. 580
CREATE GROUP (volume, catalog search method) ........................................ 581
UPDATE GROUP ................................................................. 582
RENAME GROUP ................................................................. 584
DELETE GROUP ................................................................. 585
REPORT GROUP ................................................................. 585
QUERY ................................................................. 590
EXPORT GROUP and MIGRATE_TO_SSID .................................................. 593
REBUILD_SCOPE and RECOVER_SCOPE .................................................. 595
Executing the JCL .................................................................................. 595

Chapter 20         ARMBLGR—Log range analysis 597
Chapter 21  ARMBLOG—Archive log creation 603
About ARMBLOG ............................................................................................................603
Authorizations ..................................................................................................................603
Building the JCL ...............................................................................................................604
  Specifying the JOB statement .............................................................. 604
  Specifying the EXEC statement .......................................................... 604
  Specifying the STEPLIB DD statement ............................................. 605
  Specifying the ARMBLOG data set DD statements ....................... 605
Sample JCL ........................................................................................................................606
Sample output .................................................................................................................. 607
Executing the JCL .............................................................................................................607

Chapter 22  ARMBLRD—Log range formatting 609
About ARMBLRD ............................................................................................................609
Authorizations ..................................................................................................................609
Building the JCL ...............................................................................................................609
  Specifying the JOB statement .............................................................. 610
  Specifying the EXEC statement .......................................................... 610
  Specifying the STEPLIB DD statement ............................................. 611
  Specifying the ARMBLRD data set DD statements ....................... 611
Sample JCL ........................................................................................................................612
Sample output .................................................................................................................. 613
Executing the JCL .............................................................................................................613

Chapter 23  ARMBRDC—Recovery data collection report 615
About ARMBRDC ............................................................................................................615
Authorizations ..................................................................................................................615
Building the JCL ...............................................................................................................616
  Specifying the JOB statement .............................................................. 616
  Specifying the EXEC statement .......................................................... 616
### Chapter 24  ARMBRID—Recover indoubt threads  
**About ARMBRID**  ................................................................. 623  
**Authorizations**  ........................................................................ 623
**Building the JCL**  ........................................................................ 623  
  - Specifying the JOB statement  .................................................. 624
  - Specifying the EXEC statement  .................................................. 624
  - Specifying the STEPLIB DD statement  ................................... 625
  - Specifying the ARMBRID data set DD statements  ........................ 625  
**Sample JCL**  ........................................................................... 626
**Sample output**  ........................................................................ 626
**Executing the JCL**  ................................................................. 626

### Chapter 25  ARMBRPR—Progress Reports  
**About ARMBRPR**  ...................................................................... 629
**Authorizations**  ........................................................................ 630
**Building the JCL**  ........................................................................ 630  
  - Specifying the JOB statement  .................................................. 631
  - Specifying the EXEC statement  .................................................. 631
  - Specifying the STEPLIB DD statement  ................................... 631
  - Specifying the ARMBRPR data set DD statements  ...................... 632  
**Syntax diagrams**  ...................................................................... 633
**Option descriptions**  .................................................................. 635
**Sample ARMBRPR JCL**  ............................................................ 637
**Sample ARMBRPR output**  ......................................................... 638
**Executing the JCL**  ................................................................. 650

### Chapter 26  ARMBSDR—Extend recovery point at disaster recovery site  
**About ARMBSDR**  ...................................................................... 651
**Authorizations**  ........................................................................ 652
**Specifying the ARMBSDR data set DD statements**  ...................... 652
**Sample JCL**  ........................................................................... 653
**Sample output**  ........................................................................ 654
**Executing the JCL**  ................................................................. 657
Chapter 27  ARMBSET—OBJECTSET processing 659

About ARMBSET ............................................................................................................. 659
Authorizations .................................................................................................................. 660
Building the JCL ............................................................................................................... 660
  Specifying the JOB statement ..................................................................................... 661
  Specifying the EXEC statement .................................................................................. 661
  Specifying the STEPLIB DD statement ...................................................................... 661
  Specifying the ARMBSET data set DD statements .................................................... 662
Syntax diagrams ............................................................................................................... 662
  Option descriptions ...................................................................................................... 665
Sample ARMBSET JCL ..................................................................................................... 669
Sample ARMBSET output ............................................................................................... 670
Executing the JCL ............................................................................................................ 671

Chapter 28  ARMBSRR—System resource recovery 673

About ARMBSRR ............................................................................................................. 673
  Recovery simulation ..................................................................................................... 674
  Recovery estimation ..................................................................................................... 675
  Hardware mirroring support ....................................................................................... 675
  Extending the recovery point at the disaster recovery site .......................................... 676
About JES support .......................................................................................................... 676
  How ARMBSRR selects a subsystem recovery point .................................................. 677
ARMBSRR jobs ................................................................................................................. 678
  Phase 1 jobs—Initialization ......................................................................................... 679
  Phase 2 jobs—Recovery JCL generation ...................................................................... 680
Tasks performed for job generation ............................................................................. 681
Functions performed by recovery jobs ........................................................................... 683
ARMBSRR-generated JCL ............................................................................................... 687
Authorizations .................................................................................................................. 688
Building the JCL ............................................................................................................... 688
  Specifying the JOB statement .................................................................................... 689
  Specifying the EXEC statement ................................................................................ 689
  Specifying the STEPLIB DD statement ..................................................................... 690
  Specifying the ARMBSRR data set DD statements ................................................ 691
ARMBSRR syntax and option descriptions ..................................................................... 694
Sample JCL ....................................................................................................................... 711
Sample output .................................................................................................................. 712
Executing the JCL ............................................................................................................ 716
Troubleshooting an ARMBSRR job ............................................................................. 716

Contents  19
# Chapter 29  ARMBTSI—Time stamp insertion

- **About ARMBTSI** .............................................................................................................. 719
- **Authorizations** .................................................................................................................. 719
- **Building the JCL** ............................................................................................................... 720
  - Specifying the JOB statement .......................................................................................... 720
  - Specifying the EXEC statement ...................................................................................... 720
  - Specifying the STEPLIB DD statement ............................................................................ 721
  - Specifying the ARMBTSI data set DD statements .......................................................... 721
- **ARMBTSI syntax and option descriptions** ....................................................................... 722
- **Sample JCL** ........................................................................................................................ 723
- **Sample output** .................................................................................................................. 723
- **Executing the JCL** ............................................................................................................. 724

# Chapter 30  ARMBWDC—System recovery data collection

- **About ARMBWDC** ............................................................................................................. 725
- **Authorizations** .................................................................................................................. 725
- **Building the JCL** ............................................................................................................... 726
  - Specifying the JOB statement .......................................................................................... 726
  - Specifying the EXEC statement ...................................................................................... 726
  - Specifying the STEPLIB DD statement ............................................................................ 727
  - Specifying the ARMBWDC data set DD statements ........................................................ 727
- **ARMBWDC syntax and option descriptions** ...................................................................... 728
- **Sample JCL** ........................................................................................................................ 731
- **Sample output** .................................................................................................................. 732
- **Executing the JCL** ............................................................................................................. 732

# Appendix A  Option sets and configuration options

- **Overview of DB2 Product Configuration** ......................................................................... 733
- **DB2 Product Configuration interface** ............................................................................... 734
  - Menu bar and drop-down menus ...................................................................................... 736
  - Action bar ....................................................................................................................... 738
  - Commands for the Command line .................................................................................... 738
  - Action Selection Menu .................................................................................................... 739
  - Interface tools .................................................................................................................. 740
  - Input panels for option sets ............................................................................................ 741
- **Option set names** .............................................................................................................. 742
- **Section defaults** ............................................................................................................... 742
- **Accessing DB2 Product Configuration** ........................................................................... 744
- **Defining an option set** ..................................................................................................... 745
RECOVERY MANAGER option sets, Product Configuration panels, and configuration options ................................................................. 747
BMC utility configuration options ............................................................................................................................... 748
TSO DB2 and user step library information .................................................................................................................. 748
DB2 subsystem resource information ............................................................................................................................ 749
Operating system resource information ....................................................................................................................... 749
Alphabetical list of configuration options ..................................................................................................................... 750
Configuration option categories ....................................................................................................................................... 755
Common options ............................................................................................................................................................... 755
Subsystem options ............................................................................................................................................................. 757

Appendix B  BMC Common DB2 repository tables  767
Naming conventions ............................................................................................................................................................... 767
OBJSETS table ..................................................................................................................................................................... 767
OBJSET_DEF table ............................................................................................................................................................... 768
OBJSET_SQL table ............................................................................................................................................................... 771
GRPOPTS table ..................................................................................................................................................................... 771
PRODREG table ..................................................................................................................................................................... 772
GROUPAUTH table ............................................................................................................................................................... 772

Appendix C  RECOVERY MANAGER repository  773
RECOVERY MANAGER repository tables ............................................................................................................................ 773
Coordinated disaster recovery information: CRRDRPT table ............................................................................................. 773
Recovery history: UTILITY_RUN table ........................................................................................................................................ 774
Recovery history: JOB table .................................................................................................................................................... 776
Recovery history: PHASE table ................................................................................................................................................ 785
Recovery history: TS table ........................................................................................................................................................ 788
Recovery history: TS_PART table ............................................................................................................................................. 790
Recovery history: KEYSORT table .......................................................................................................................................... 796
Recovery history: TSPSORT table ........................................................................................................................................... 798
Recovery history: IX_PART table ............................................................................................................................................... 799
Recovery history: IXPSORT table ........................................................................................................................................... 805
Job history: JOB_RESTART table .......................................................................................................................................... 806
Job history: BMCARM.BMCRMD_RECOVERY_EXCEPTIONS table ......................................................................................... 807
Data collection SQL statements ............................................................................................................................................. 808

Appendix D  Common utility tables  811
Overview of common utility tables ....................................................................................................................................... 811
Warnings and considerations for common utility tables ........................................................................................................ 813
Managing common utility tables ........................................................................................................................................... 814
BMCDICT table ...................................................................................................................................................................... 816
About this book

This book contains detailed information about the associated product or products. This preface explains the special conventions that the book uses, and how to access related publications.

If applicable, the preface also summarizes the major changes included in the latest release of the product.

Related publications

From the BMC Support Central website, you can use the following methods to access related publications that support your product or solution:


- View Quick Course videos (short overviews of selected product concepts, tasks, or features), which are available from the following locations:
  
  — Documentation Center (primary center and secured center)
  
  — Support Central (at http://www.bmc.com/support/mainframe-demonstrations)
  
  — BMC Mainframe YouTube channel (https://www.youtube.com/user/BMCSoftwareMainframe)


Products with online interfaces also offer online Help via the F1 key or, for graphical user interfaces (GUIs), via a Help button.
**Tip**
If you prefer hardcopy documentation, you can order it from your BMC sales representative or from Support Central. Also, from Support Central you can subscribe to receive proactive e-mail alerts when BMC issues notices.

---

## Conventions

This document uses the following special conventions:

- **All syntax, operating system terms, and literal examples are presented in this typeface.**

- Variable text in path names, system messages, or syntax is displayed in italic text: `testsyst/instance/fileName`

- This document uses a symbol to show menu sequences. For example, **Actions => Create Test** instructs you to choose the **Create Test** command from the **Actions** menu.

---

## Syntax diagrams

The following figure shows the standard format for syntax diagrams:
The following example illustrates the syntax for a hypothetical DELETE statement. Because the FROM keyword, alias variable, and WHERE clause are optional, they appear below the main command line. In contrast, the tableName variable appears on the command line because the table name is required. If the statement includes a WHERE clause, the clause must contain a search condition or a CURRENT OF clause. (The searchCondition variable appears on the main line for the WHERE clause, indicating that this choice is required.)

The following guidelines provide additional information about syntax diagrams:

- Read diagrams from left to right and from top to bottom
- A recursive (left-pointing) arrow above a stack indicates that you may choose more than one item in the stack
- An underlined item is a default option
- If a diagram shows punctuation marks, parentheses, or similar symbols, you must enter them as part of the syntax. Asterisks are exceptions. An asterisk in a diagram indicates a reference note
- In general, IBM MVS commands, keywords, clauses, and data types are displayed in uppercase letters. However, if an item can be shortened, the minimum portion of the MVS command or keyword might be displayed in uppercase letters with the remainder of the word in lowercase letters (for example, CANcel)
- The following conventions apply to variables in syntax diagrams:
  - Variables typically are displayed in lowercase letters and are always italicized
  - If a variable is represented by two or more words, initial capitals distinguish the second and subsequent words (for example, databaseName)

## Summary of changes

This section summarizes changes to the functionality of the product, listing the changes by product version and release date. The summary includes enhancements to the product and any major changes to the documentation. All significant changes to this book since the last release are marked with a revision bar.
Summary of changes Version 12.1.00 December 2016

This release includes the following product enhancements and changes:

Support for IBM DB2 Version 12

This release adds support for DB2 Version 12.

This release supports:

- DB2 Version 12
- DB2 Version 11
- DB2 Version 10 in new-function mode (NFM) only

Note

RECOVERY MANAGER always uses the real SYSLGRNX for IBM DB2 Version 10.

RECOVER PLUS for DB2 name change

Starting with this release, the name of the RECOVER PLUS for DB2 product has changed to the BMC Next Generation Technology Recover for DB2 for z/OS (or NGT Recover) product. Affected product panels and documentation use the new name.

COPY PLUS for DB2 name change

Starting with this release, the name of the COPY PLUS for DB2 product has changed to the BMC Next Generation Technology Copy for DB2 for z/OS (or NGT Copy) product. Affected product panels and documentation use the new name.

Recovery Progress Reports renamed

Starting with this release, Recovery Progress Reports have been renamed Progress Reports.

REBUILD INDEX SHRLEVEL CHANGE

RECOVERY MANAGER now supports JCL generation for the REBUILD INDEX SHRLEVEL CHANGE option in the NGT Recover product.
Forward recovery of indexes to a timestamp

NGT Recover can now perform a forward recovery of indexes to a timestamp by using the RECOVER INDEX command with OPTION RECOVERYPOINT.

New ARMBGRP syntax

This release includes the following changes to ARMBGRP syntax:

- Added the BASE and ONLY settings to the INCLUDELOB, INCLUDEXML, INCLUDEHISTORY, and INCLUDEARCHIVE options

- Added the DSNUM option to the CREATE GROUP command for the TABLESPACE, INDEX, and INDEXSPACE options. This addition enables you to create groups by table space, index, or index space by partition.
  Valid values for DSNUM are:
  - n for partition number
  - n.m for a range of partitions starting with n and ending with m, where n and m are partition numbers
  - ALL to include all partitions

- Added the EXCLUDE DSNUM option to the CREATE GROUP command for the TABLESPACE, INDEX, and INDEXSPACE options. This addition enables you to exclude groups by table space, index, or index space by partition.
  Valid values for EXCLUDE DSNUM are:
  - n—a partition number.
  - n.m—a range of partitions starting with n and ending with m, where n and m are partition numbers.

ARMBGPMV populates the RECOVERY.Exceptions table

If you have a BMC Recovery for DB2 solution password, RECOVERY MANAGER writes all recoverability exceptions to the new RECOVERY.Exceptions table for later reporting to BMC Workbench for DB2.

For more information, see “Job history: BMCARM.BMCRMD_RECOVERY_EXCEPTIONS table” on page 807.
ARMBSRR populates the RECOVERY_EXCEPTIONS table

If you have a BMC Recovery for DB2 solution password, RECOVERY MANAGER writes the validation report to the new RECOVERY_EXCEPTIONS table for later reporting to BMC Workbench for DB2.

For more information, see “Job history: BMCARM.BMCRMD_RECOVERY_EXCEPTIONS table” on page 807.

ARMGA001 GROUP Authorization panel

BMC is deprecating the ARMGA001 GROUP Authorization panel. Although this release continues to include the panel, BMC plans to remove the panel in a future release.

ARMBGRP supports group name patterns for QUERY GROUP INFO, QUERY GROUP OBJECTS, and QUERY GROUP AUTH

This release adds the following options to the QUERY command:
  ■ GROUP INFO RDAJ*.*
  ■ GROUP OBJECTS RMD.*
  ■ GROUP AUTH PUBLIC.ACP*

These options allow wildcard values and delimited group names. For the existing options, *.* is the default value.

Simulate Recovery option for Progress Reports

The online panel ARMRPR01 has a new option to report progress for SIMULATED and INDEP OUTSPACE recoveries instead of actual recoveries. The new SIMULATED_RECOVERY syntax for program ARMBRPR provides the same support in batch.

Dynamic grouping

Starting with this release, when you create groups online or use the ARMBGRP batch program based on exceptions, you can enable dynamic grouping. If you enable dynamic grouping, you can create an empty group (a group for which no objects are found that meet the definition). After you create the objects and then open the group, the group is automatically populated.
This release added the **Create group as dynamic** field to the **Exception Status Specification for Object List Generation** panel (ARMEX001).

### Exporting and migrating groups

RECOVERY MANAGER now enables you to convert group definitions into CREATE GROUP syntax and write them to a specified data set. You can then either keep the data set as a backup or use it to migrate the converted definitions to another IBM DB2 subsystem.

The export and migration of group definitions involves the following steps:

1. Use the online interface to prepare ARMBGRP syntax and JCL.
2. Execute the ARMBGRP batch program.

This change adds:

- **ARMEXPRT DD**
- **EXPORT GROUP** and **MIGRATE_TO_SSID** commands
- The following options on the Group List panel (ARMGP001):
  - **X** to export the specified groups
  - **XA** to export all groups
- **Batch Group Export/Migrate JCL Generation panels**

### Option to recover work file database

Adds an option to ARMBSRR to recover work file database (DSND807) objects during local or remote subsystem recovery. If you choose not to recover work file database objects, this step is omitted from the recovery job.

You can also specify the library that contains DSNTEP2.

These options include the following additions:

- **Additions to the online interface** on panels ARMUFS4A and ARMDR004A
- **Addition of RECOVER WORKFILE** and **DSNTEP2LIB dataSetName** syntax to ARMBSRR (see “ARMBSRR syntax and option descriptions” on page 694)
BACKOUT AUTO time estimation

This release adds BACKOUT AUTO time estimation to Progress Report functionality. BACKOUT AUTO time estimation is the sum of the following times:

- The time estimate for recovering objects that can be backed out
- The forward recovery time estimate for recovering objects that cannot be backed out

If objects cannot be recovered via backout or forward recovery, a message indicates that the objects are not eligible for BACKOUT AUTO recovery. You can view a list of BACKOUT AUTO exceptions; these are the same as backout and forward recovery exceptions.

This feature includes the following additions:

- **A=Backout Auto Exceptions** on the ARMRPR06 panel
- **Auto** on the ARMBRPR ARMSUMRY DD recovery estimation report
- A new ARMAUERR DD for BACKOUT AUTO estimation exceptions

Delimited object set names

BMC is deprecating support for delimited object set names. Although this release continues to support delimited object set names, BMC plans not to end this support in a future release.

Delimited names are those enclosed by double quotation marks and can contain special characters (for example, "creator"."name").

**Note**
The terms *group* and *object set* are synonymous.

RECOVER_SCOPE and REBUILD_SCOPE options

This release of NGT Recover adds the RECOVER_SCOPE and REBUILD_SCOPE options to the NGT Recover OPTIONS command.

The following options enable RECOVERY MANAGER to recover or rebuild objects based on specified criteria:

- **RECOVER_SCOPE (ALL)** recovers all specified spaces, including those that have changed.
- **REBUILD_SCOPE (ALL)** rebuilds all specified spaces, including those that have changed since the specified recovery point.

- **RECOVER_SCOPE (UPDT)** recovers only spaces that have changed since a specified recovery point.

- **RECOVER_SCOPE (status1,status2,...)** and **REBUILD_SCOPE (status1,status2,...)** select objects based on their specified statuses.

- **REBUILD_SCOPE (PEND)** processes only indexes that are in a RBDP, PSRBD, RBDP*, RECP, or AREO* status.

This release of RECOVERY MANAGER adds the **Recover scope** and **Recover/Rebuild scope status** fields to panel ARMOR41.

See “**RECOVER_SCOPE syntax and option descriptions**” on page 569 and “**REBUILD_SCOPE syntax and option descriptions**” on page 571.

### Removal of restrictions on recovery simulation

This release now supports recovery simulation of:

- Recovery from Instant Snapshot copies
- BACKOUT recovery
- LOGONLY or LOGAPPLY SCANONLY
- INLINE image copies

### New &PART4 symbolic variable

You can use a new symbolic variable, &PART4, in NGT Recover and NGT Copy to generate partitions for data set allocation. You can use &PART4 for any data set.

### Symbolic variable substrings

You can use the following NGT Copy and NGT Recover (OUTCOPY) output data set names specified by OUTCOPYDSN, RECOVERYDSN, and INCOPY MODEL

dataSetName:

- &DB(s,l)
- &TS(s,l)
- &USERID(s,l)
- &USER(s,l)
- &UID(s,l)

The s variable represents the substring, and l represents the length.
See “Job card (J CARD) specification” on page 52.

Multi-job JCL generation

You cannot use SIMULATE or ESTIMATE options with multi-job JCL generation.

Reports with ANSI printer carriage control characters

Starting with this release, RECOVERY MANAGER reports are no longer produced with ANSI printer carriage control characters.

SEPARATE_BY_PARTITION option

This option is valid for creating classic NGT Copy syntax only when you have specified OBJECTSET NO in ARMBGEN; this option is invalid for the COPY TABLESPACE OBJECTSET group name.

Object set management in BMC Workbench

The BMC Workbench for DB2 product now enables you to manage object sets, which are used by the RECOVERY MANAGER and DASD MANAGER PLUS products, and by certain NGT utilities.

Note

In BMC Workbench, you can edit only object sets that were created by using BMC Workbench. However, you can resolve, copy, and view object sets from all sources.

For more information, view the Quick Course "Workbench for DB2 - Managing object sets," the BMC Workbench for DB2 User Guide, and the BMC Workbench online Help.

Users who have a license for any of the following simplified solutions can access and use BMC Workbench to manage object sets:

- BMC High Speed Utilities for DB2
- BMC Object Administration for DB2
- BMC Recovery for DB2
- BMC Performance for DB2SQL
- BMC Performance for DB2 Databases
- BMC Utility Management for DB2
- BMC Large Object Management for DB2
Summary of changes Version 11.2.00 May 2015

This release of RECOVERY MANAGER (RMGR) for DB2 includes the following product enhancements and changes.

RECOVERY MANAGER changes 11.2.00

- No longer supports DB2 Version 9 — References to DB2 version 9 have been removed throughout this book. In addition, RECOVERY MANAGER supports DB2 Version 10 only in new-function mode.

- DB2 Version 11 support:
  - Adds support for Persistent Read Only (PRO) and Read-or-Replication Only (RREPL) statuses for Exception Groups.
  - Adds support for Archive-Enabled Tables by providing an INCLUDE option when creating groups/object sets online and in batch. This option includes related archive tables in the group/objectset. See “CREATE GROUP” on page 530.
  - Adds support for enhancements and restrictions to point-in-time (PIT) recoveries.
  - Adds support for RBALRSN_CONVERSION for catalog/directory Rebuild Index.

- Online creation of groups by INDEXSPACE Name — Adds option to create groups online by INDEXSPACE name. RECOVERY MANAGER and DASD MANAGER PLUS share the same repository for groups/object sets. Creating groups by indexspace gives RECOVERY MANAGER the same functionality of the DASD Manager interface.

- Adds support for additional statuses when creating Exception Groups — See new statuses in VIA EXCEPTION in “CREATE GROUP option descriptions (catalog search)” on page 530.

- Method to copy groups from one DB2 to another — The GA line command on the group list panel will generate JCL for program ARMBGRP to create all of the groups in the list. By changing the SSID, this job can be used to create the same groups on another subsystem. This JCL can also be used as a backup in case one or more groups are inadvertently deleted.

- ARMBGRP changes — The following change has been implemented for ARMBGRP:
  - IVP now reports synonyms
- Adds support for VARBINARY and PENDINGDDL for recoverability — These statuses can be seen on the group object list panel and in the ARMBGPV Exception report. See “RECOVERY MANAGER object exception status” on page 839.

- Enhances CHECK DATA support after recovery — Table spaces that have parents or hash tables will be selected to have CHECK DATA after they are recovered and when the group option Check Pend Action is CHECK.

- ARMBSRR changes — The following changes have been implemented for ARMBSRR:

  — Bypass Quiesce supports VALIDATE FAIL/WARN.

  — New syntax BSDS HOURSLIMIT and BSDS DAYSLIMIT provide alternate methods to specify the maximum number of logs that you want processed by ARMBSRR.

  — New syntax LIMIT HOURS and LIMIT DAYS provide alternate methods to specify how many log data sets are to be restored for each recovery-site archive log copy.

  — Adds the option to initialize active logs with DSNJLOGF — This option improves performance by avoiding initialization overhead the first time the active log is used by DB2.

  See “ARMBSRR—System resource recovery” on page 673 for changes.

- Adds new batch program ARMBACT — The ARMBACT program allows you to initialize all active logs for a specified SSID by calling DSNJLOGF. See “ARMBACT—Initialize active logs with DSNJLOGF” on page 403.

- Extends functionality of Progress Reports and the batch program ARMBRPR. New functions include:

  — ARMBRPR issues command to externalize Real Time Statistics

  — 'What if' scenarios for Backup elapsed time

  — 'What if' scenarios for Recover elapsed time estimates

  New functionality requires a BMC Recovery for DB2 password. See “Accessing online Progress Reports” on page 383 and “ARMBRPR — Progress Reports” on page 629.

- RECOVERY MANAGER reports are no longer produced with ANSI printer carriage control characters.
RECOVER PLUS changes 11.2.00

- Adds support for MAXPRIM and AUTOSIZE for OUTCOPY. See “NGT Recover (OUTCOPY) options” on page 183.

- Adds support for TOTIMESTAMP and TOLOGMARK for OUTCOPY — New options TOTIMESTAMP and TOLOGMARK offer more flexibility when specifying OUTCOPY ONLY by allowing you to set a migration point to any point in time, not just the last copy. See “NGT Recover (OUTCOPY) options” on page 183.

- Adds support for the &PART5 Variable for Output Data Set Names.

- Adds Alternate Resource SB (System Backup) — RECOVER PLUS supports recovering individual spaces using system backups. RECOVERY MANAGER adds SB (System Backup) to the Alternate Resource Selection panel (ARMR005C). See “Alternate recovery resource options” on page 248.

COPY PLUS changes 11.2.00

- Adds support for the &PART5 Variable for Output Data Set Names.

- Shrlevel Change Consistent replaces PGM=NSCMAIN — In previous releases, Online Consistent Copies were made by executing the program NSCMAIN. Starting with this release, Online Consistent Copies can be made by execution ACPMAIN with the CONSISTENT YES option. NSCMAIN is still be supported for this release for compatibility with previous releases, but it will be deprecated in the next release. See “NGT Copy backup options” on page 170 and “Setting backup options for online consistent copies” on page 161.

Summary of changes Version 11.1.00 June 2013

This release of RECOVERY MANAGER (RMGR) for DB2 includes the following product enhancements and changes.

RECOVERY MANAGER changes 11.1.00

- No longer supports DB2 Version 8 — References to DB2 version 8 have been removed throughout this book.

- Implements the use of the DB2 Product Configuration technology for the option set and configuration option default values. This change includes the following items:
Recasts “Option sets and configuration options” on page 733 to accommodate the implementation of the DB2 Product Configuration technology. This change includes the change of "control information" to "configuration option" throughout this book. Also, "control file" has been changed to "option set" throughout this book.

Adds option 5. **Product Option Sets - Set RECOVERY MANAGEMENT**

**Product options** to the RMGR Main Menu and also adds the field **Option Set** for entry of an option set name. The default option set is ARM$OPTS.

Adds the default option set name, ARM$OPTS, to EXEC PARM throughout this book.

Uses the ARMOPTS DD to override configuration option values at runtime.

Removes the DB2V option — The DB2 version is now obtained when connecting to DB2. If not connected to DB2, the DB2 version is obtained from the DSNHDECP module in the STEPLIB.

Removes the following configuration options from “Option sets and configuration options” on page 733:

- ssid.XBMVRSN
- ssid.ACKVRSN
- ssid.ACPVRSN
- ssid.AFRVRSN
- ssid.ALMVRSN
- ssid.ALPVRSN
- ssid.XBMVRSN

RECOVERY MANAGER now checks the xxxVRSN load module from the STEPLIB. If the xxxVRSN module is not found for a product, RECOVERY MANAGER does not produce informational message. If the xxxVRSN is found, RECOVERY MANAGER produces an informational message if the version is not supported.

Adds the configuration option to get the option set name for PACLOG.

Removes ARMBSDR, ACASTINST, APTLOAD, and APTPLAN from “Option sets and configuration options” on page 733.

Add the Recovery Progress Report and the new batch program ARMBRPR— The Recovery Progress Report and the new ARMBRPR program provide a way to gauge the progress of recoveries by DATABASE.TABLESPACE pattern and the group level. See the following information:

- “Accessing online Progress Reports” on page 383
- ARMBRPR — Progress Reports on page 629
■ Shares repository with DASD MANAGER PLUS — RECOVERY MANAGER now shares the BMC Common DB2 repository. This repository is used to work with groups/object sets.

■ Adds support for the generation of standalone JCL for IBM’s BACKUP SYSTEM and RESTORE SYSTEM ("Using RECOVERY MANAGER for BACKUP SYSTEM and RESTORE SYSTEM” on page 364).

■ Removes the restriction that does not allow TIMESTAMP recovery JCL (TOTIMESTAMP option) to be generated by RECOVERY MANAGER on a non-data-sharing system ("GENJCL” on page 452).

■ Adds support for external security, such as ACF2 or RACF, for opening and saving a group ("RACF authorization” on page 89.

■ For unchanged table spaces, COPY NO indexes are not recovered. The status for these indexes is now set to TS STAT. This status is shown on panel ARMOB001 (Object List) ("Adding objects to a group” on page 122).

■ Adds options to the Group List panel (ARMGP001) — To allow you to make changes without the overhead of opening the group, RECOVERY MANAGER adds the following options on the Group List panel:

  — **H** to display group authorizations
  — **O** to display group utility options
  — **G** to generate ARMBGRP CREATE JCL based on the group definition of the selected group
  — **P** to display the Recovery Progress Report panel

■ Adds support for the Deactivate/Destroy/Restore function for data sharing members — RECOVERY MANAGER adds support for the Deactivate/Destroy/Restore function for data sharing members by adding a new panel (ARMRBS — 3.4.12 from the Main Menu).

  Support for this feature requires DB2 Version 10 new-function mode (NFM) or higher. (IBM APAR PM42528 introduced this function.)

  See “Deactivate/Destroy/Restore for data sharing members” on page 282.

■ Adds DSSNAP to the Output Options panel (ARMRO03D) — See the following DSSNAP additions:

  — “Output copy data set options” on page 186
  — “Syntax for updating group options” on page 546
  — “Output copy options” on page 888
Summary of changes

- Adds Work prefix to the Update/Browse Work File Options panel (ARMRO002) — RECOVERY MANAGER now allows you to specify a work prefix in the group options (see “Work file option descriptions” on page 251).

  Work prefix is now reported in REPORT GROUP output under WORK FILE OPTION DESCRIPTIONS (“REPORT GROUP” on page 563).

  WORKFILE_WORKPREFIX is also a new option for ARMBGRP (“Syntax for updating group options” on page 546).

- Adds support to ARMBSRR and ARMBTSI to handle quiesced and deactivated data sharing members. This support includes the following additions:

  - Additions to the online interface on panels ARMDR01A, ARMDR06A, and ARMUFS4 to exclude data sharing members, bypass quiesced members, or bypass deactivated members.

  - Addition of EXCLUDE MEMBERS, BYPASS QUIESCED, and BYPASS DEACT syntax to ARMBSRR and ARMBTSI (“ARMBSRR syntax and option descriptions” on page 694 and “ARMBTSI syntax and option descriptions” on page 722).

  **Note**

  For ARMBSRR, EXCLUDE MEMBERS replaces QUIESCED MEMBERS and works as QUIESCED MEMBERS worked.

- ARMBGRP changes — The following changes have been implemented for ARMBGRP:

  - Allows multiple VIA statements for group creation — Multiple VIA statements are allowed in CREATE GROUP syntax with the exception of VIA EXCEPTION and VIA VOLUMES, which are static groups. (“Creating groups” on page 507)

  - Adds EXCLUDE and EXCLUDEIX syntax for VIA VOLUMES — You can now use EXCLUDE and EXCLUDEIX for groups created with VIA VOLUMES (“CREATE GROUP option descriptions (catalog search)” on page 530).

  - Adds VIA GROUP syntax — You can now specify VIA GROUP syntax for group creation (“CREATE GROUP syntax and option descriptions” on page 520).

  - Adds VIA INDEXSPACE syntax — You can now specify VIA INDEXSPACE syntax for group creation (“CREATE GROUP syntax and option descriptions” on page 520).

  - Adds VIA PACKAGE syntax — You can now specify VIA PACKAGE syntax for group creation (“CREATE GROUP syntax and option descriptions” on page 520).
— Adds VIA PLAN syntax — You can now specify VIA PLAN syntax for group creation (“CREATE GROUP syntax and option descriptions” on page 520). The following DB2 version-specific items now apply to the creation of groups by plan:

■ When running on DB2 Version 9, RECOVERY MANAGER will include objects with plan and package dependencies for groups built by plan name.

■ When running on DB2 Version 10 or later, RECOVERY MANAGER will include objects with package dependencies for groups built by plan name.

— Adds VIA STOGROUP syntax— You can now specify VIA STOGROUP syntax for group creation (“CREATE GROUP syntax and option descriptions” on page 520).

— Allows dynamic SQL inline in VIA SQL syntax -- You can now enter SQL in VIA SQL syntax using #BEGINSQL and #ENDSQL for multiple SQL statements per group (“CREATE GROUP syntax and option descriptions” on page 520).

— Allows dynamic SQL subselects — You can now enter subselects in the SQL used to create groups.

— Supports ‘SG’ in dynamic SQL— ‘SG’ for storage group is now supported in the dynamic SQL used to create groups (“Specifying objects for a new group” on page 107, “Creating a new group” on page 113).

— Adds the new report QUERY OBJECTS — The report provides what groups have certain table spaces or indexes by object name (Syntax for querying groups on page 565).

■ ARMBGPS changes — RMGR adds support for including or excluding groups by providing the ability to specify group name patterns to populate new groups. This enhancement includes the following items:

— Includes changes for the RMGR online interface to panel ARMUFS1A to add the following new entry fields:

   — GROUP OWNER.NAME Include names or patterns
   — GROUP OWNER.NAME Exclude names or patterns (optional)

   These fields are accessed using 2.10 from the RMGR Main Menu and with your desired entries on panel ARMUFS1 to display panel ARMUFS1A.

— Adds INCLUDE_GROUP and EXCLUDE_GROUP syntax to ARMBGPS (“ARMBGPS syntax and option descriptions” on page 481).

■ ARMBSET changes — The following changes have been implemented for ARMBSET:
— Adds the new command SET CURRENT SQLID (“SET CURRENT SQLID” on page 665)

— Adds the new command for subsystem RESET_GRECP_LPL (“SET CURRENT SQLID” on page 665)

A START command is issued for each object in LPL or GRECP status. This action is done for the entire subsystem and not for the object set.

— Adds the following new commands for object sets:

— QUIESCE WRITE (“SET CURRENT SQLID” on page 665)

— RUNSTATS (“SET CURRENT SQLID” on page 665)

— Changes in ARMBSET for all syntax options noted as requiring numbers as values — These options now also accept the alphabetic option value associated with the number (“Syntax diagrams” on page 662).

**Note**

For ARMBSET, the use of numbers for the option values for those options that also have alphabetic option values will be deprecated in a future release of RECOVERY MANAGER and only alphabetic option values will be accepted.

- ARMBGPV changes — ARMBGPV now writes SAMS RESTORE commands to the new ARMRSTOR DD. You can use the SAMS RESTORE commands as input to restore migrated data sets. You do not set an option to have RMGR create the ARMRSTOR DD (“Specifying the ARMBGPV data set DD statements” on page 493).

- Adds the following DD statement for ARMBSDR and ARMBCOR (“Specifying the ARMBSDR data set DD statements” on page 652 and “Sample JCL” on page 653) due to changes in their method of communicating with each other:

```
//ARMCOMM DD DISP=SHR,
//            DSN=PRODUCT.CNTL.LIBS(ARMBSDR)
```

In RECOVERY MANAGER versions earlier than version 11.1.00, the ARMBCOR and ARMBSDR batch programs communicated with each other by using the ARMBSDR option in ARMSOPTS during Phase 1 execution of disaster recovery JCL for data sharing. With the conversion of ARMSOPTS to use DB2 Product Configuration (LGC), this method of communication is no longer valid. The new DD statement shown above will be added to ARMBCOR and ARMBSDR JCL.

The ARMBSDR member in the CNTL data set, as seen in the ARMCOMM DD statement, will now be used to communicate.

This change does not affect the how ARMBCOR or ARMBSDR work. Only the method of communicating with each other has changed.
Changes for COPY PLUS 11.1.00

RECOVERY MANAGER has been enhanced to support the following items for COPY PLUS:

- Adds support for the COPY PLUS SNAP option--The SNAP option enables COPY PLUS to make VSAM copies, even if the data set is not on a snappable disk. ("UPDATE GROUP syntax and option descriptions" on page 545 and "NGT Recover recover options" on page 851)

- Adds a second value, the maximum total tasks (MAX_TASK2), for MAXTASKS -- MAXTASKS ("NGT Recover copy options" on page 870) is now valued as MAXTASKS (maximum tape tasks, maximum total tasks). Appropriate changes have been made to the ARMC0023 panel in the RMGR online interface to accommodate the second value.

- Adds the &UNIQ (or &UQ) symbolic variable to generate unique image copy data set names — If you use &UNIQ, COPY PLUS generates a 1- to 8-character value that is based on the system clock. The first character is always an uppercase letter. Each remaining character is either an uppercase letter or a numeral from 0 through 9. ("Symbolic variables in image copy data set names" on page 191)

Changes for RECOVER PLUS 11.1.00

RECOVERY MANAGER has been enhanced to support the following items for RECOVER PLUS:

- Adds support for the RECOVER PLUS SNAP option--The SNAP option enables RECOVER PLUS to recover VSAM copies, even if the data set is not on a snappable disk. ("UPDATE GROUP syntax and option descriptions" on page 545 and "NGT Recover copy options" on page 870)

- Adds the &UNIQ (or &UQ) symbolic variable to generate unique image copy data set names — If you use &UNIQ, RECOVER PLUS generates a 1- to 8-character value that is based on the system clock. The first character is always an uppercase letter. Each remaining character is either an uppercase letter or a numeral from 0 through 9. ("Symbolic variables in image copy data set names" on page 191)

Changes for Recovery Management 11.1.00

The following changes in RECOVERY MANAGER affect the Recovery Management solution:

- Adds TOTIMESTAMP recovery for non-data-sharing environments -- TOTIMESTAMP recovery was previously allowed only for data sharing
subsystems. For this release, this type of recovery is also allowed for non-data-sharing subsystems.

This restriction has been removed in section about inflight resolution technology and timestamp recovery in the *Recovery Management for DB2 User Guide*.

- Uses RESOLVE INFLIGHTS with TOLOGMARK recovery — See the section about Inflight resolution technology and timestamp recovery and the section about Recovering to a log mark in the *Recovery Management for DB2 User Guide*.
RECOVERY MANAGER for DB2

This part presents descriptions of features and user instructions for using the RECOVERY MANAGER for DB2 online interface.

This part presents the following chapters:

Chapters:

- Introducing RECOVERY MANAGER
- Getting started with RECOVERY MANAGER
- Creating and working with groups
- Backing up groups
- Recovering groups
- Exporting and migrating groups
- Managing DB2 system resources
- Recovering from a DB2 system disaster
- Full subsystem recovery
- Modeling the DB2 logging environment
- Accessing online Progress Reports
Introducing RECOVERY MANAGER

This chapter introduces RECOVERY MANAGER for DB2.

Overview of RECOVERY MANAGER

The RECOVERY MANAGER product automates the backup and recovery of DB2 objects in both data sharing and non-data-sharing environments and ensures fast and successful task completion.

RECOVERY MANAGER enables you to plan for various recovery scenarios including volume failure and disaster recovery. You can predefine groups of objects and the appropriate backup and recovery options, revalidate recovery points and objects, enforce recovery rules, and test your recovery plans before failures occur.

As a solution for recovery management problems, RECOVERY MANAGER provides the following major benefits:

- Reduced costs and time savings in initiating a recovery because of the immediate availability and readiness of a plan that is already tailored to fit the recovery scenario
- Reduced costs in ad hoc backups and recoveries due to the product’s ease of use and ability to automatically generate accurate optimized JCL
- Reduced costs and time savings in all backup and recovery operations because of — Accurate optimized JCL that guarantees success
  — Automatic data set sizing that prevents failure during execution
- Reduced costs and time savings in performing automated full subsystem backups and recoveries, a feature that is extremely useful for enterprise resource planning (ERP) applications such as SAP
- Reduced costs and time savings in providing the ability (in conjunction with NGT Copy, NGT Recover, and SNAPSHOT UPGRADE FEATURE) to automate the hardware-based Instant Snapshot solution in backup and recovery operations
Reduced costs and time savings in disaster recovery by providing the following items:

— JCL for recovering system resources at a recovery site
— JCL for recovering applications at a recovery site
— Procedures for contingency planning at the local site
— Procedures for recovery at the recovery site

An additional benefit, which is equally important but less tangible, is the peace of mind that is provided to the DBA by RECOVERY MANAGER.

The BMC Recovery Management for DB2 solution

RECOVERY MANAGER is a standalone product, but it is also a component of the BMC Recovery Management for DB2 solution. The BMC Recovery Management for DB2 solution integrates the features of the following products:

— RECOVERY MANAGER for DB2
— BMC Next Generation Technology Recover for DB2 for z/OS
— BMC Next Generation Technology Copy for DB2 for z/OS
— Log Master for DB2
— SNAPSHOT UPGRADE FEATURE
— R+/CHANGE ACCUM for DB2

Customers who acquire this solution benefit from all of the features of these individual products, as well as exclusive solution-only features. For more information, see the Recovery Management for DB2 User Guide.

RECOVERY MANAGER concepts and functionality

The BMC RECOVERY MANAGER product for DB2 automates the entire recovery process and generates optimized job streams to get DB2 data back.
RECOVERY MANAGER can perform the following tasks:

- Coordinates recoveries among multiple DB2 subsystems
- Automates definition of application groups
- Generates optimized backup and recovery JCL
- Provides log analysis of unchanged objects
- Audits recoverability of applications
- Avoids backup and recovery for unchanged objects
- Identifies valid recovery points

You can use RECOVERY MANAGER to prepare backup and recovery JCL for many DB2 application failure scenarios and for system resource maintenance and recovery scenarios. Use RECOVERY MANAGER to prepare for disaster recovery and to recover applications and system resources at a recovery site in both data sharing and non-data-sharing situations with JCL that specifies one or more IBM or BMC utilities.

You can perform most functions interactively through the RECOVERY MANAGER ISPF menu-driven user interface. You also have the option of performing many recovery functions in batch mode.

**How RECOVERY MANAGER works**

The figure below shows the resources that RECOVERY MANAGER for DB2 can use when you create maintenance, backup, or recovery JCL. When you group application or system objects and validate them, RECOVERY MANAGER uses the DB2 catalog and directory and operating system catalog as necessary.

If you are saving or retrieving a group, RECOVERY MANAGER also accesses a repository (a group of DB2 tables).
To recover or maintain system resources (such as the logs, bootstrap data set, or the catalog and directory), you interact with the RECOVERY MANAGER online interface to specify the required information.

Figure 1: How recovery management works
Creating application groups for backup or recovery

Using RECOVERY MANAGER, you can create application groups in any of the following ways:
■ By volume
■ By plan (obsolete with DB2 Version 10)
■ By package
■ By full DB2 subsystem
■ By table space set
■ By storage group
■ By index
■ By owner ID
■ By certain exception statuses
■ By user-defined SQL

You can use wildcards in most cases to provide fast and flexible object identification.

For more information about creating groups, see the following chapters:
■ “Creating and working with groups” on page 105
■ “Full subsystem recovery” on page 347
■ “ARMBGRP—Group creation and maintenance” on page 507

Specifying utilities and syntax options

You can use the backup and recovery features of RECOVERY MANAGER independently of one another.

You can specify all options (including the utilities that are used to perform the selected tasks) separately for backup or recovery. You can also cause the options to default in different ways for optimal processing. You can specify group options by using either the online interface or the ARMBGRP batch program.

For more information, see the following chapters:
■ “Backing up groups” on page 157
■ “Recovering groups” on page 199
■ “Exporting and migrating groups” on page 261
■ “ARMBGRP—Group creation and maintenance” on page 507
Validating and revalidating application groups

After you have created a group, all objects in the group are validated for their availability to DB2 and their eligibility for recovery.

If RECOVERY MANAGER encounters a problem with an object, an appropriate status is returned for that object. These unsatisfactory statuses are handled differently in a recovery situation or in a backup situation. See “Responding to unsatisfactory object status” on page 132 for more information.

After initial validation is complete, you can save the group to the repository and then periodically revalidate it to determine whether any changes that affect the recoverability of the group (as it was initially defined) have taken place. If changes have occurred, they are reported and can be implemented automatically.

Note

With the introduction of dynamic grouping in RECOVERY MANAGER version 9.2.00, revalidation no longer refers to the process of running the group definitions again to populate the group with an updated list of objects based on the current system. This type of revalidation is no longer needed with dynamic grouping. Revalidation now refers to checking the recoverability of the objects in the group.

RECOVERY MANAGER does not revalidate catalog and directory resources.

Whenever a recovery point is selected, the objects are also validated for recoverability to that point.

Revalidation can be performed online or in batch. For more information, see:

- “Validating the objects in a group” on page 130
- “ARMBGPV—Group recovery revalidation” on page 489
- “ARMBGIM—Impact analysis” on page 465

Maintaining backing up and recovering system resources

You can generate backup and recovery JCL for the DB2 catalog and directory, the BSDSs, active and archive logs, and the repository in both data sharing and non-data-sharing environments.

For more information, see:

- “Recovering from a DB2 system disaster” on page 291
- “Full subsystem recovery” on page 347
- “ARMBSRR—System resource recovery” on page 673
Generating backup and recovery JCL

RECOVERY MANAGER performs one check for both backup and recovery JCL to verify general object validity.

For recovery JCL generation, RECOVERY MANAGER checks all objects in the group for recoverability to the recovery point that you specify. RECOVERY MANAGER reports unsuitable object status, and you can often use an alternate recovery point or recovery method to recover objects with unsuitable status. You can then save the group or generate JCL. If you save the group to the repository, you can retrieve it later, and then generate JCL.

You can also use the batch program ARMBGEN to generate JCL for one or more existing groups. ARMBGEN can be executed when needed to generate up-to-date backup and recovery JCL for the specified group or groups.

When RECOVERY MANAGER creates recovery JCL, it can build multiple jobs that run concurrently to speed the recovery. RECOVERY MANAGER optimizes and synchronizes the jobs and takes into account user-defined constraints, logical constraints on common resources, and object interdependencies.

For backup JCL, RECOVERY MANAGER generates a single job. Using the ARMBGPS program for full subsystem recovery, RECOVERY MANAGER can split the table spaces in a system into multiple, balanced groups for backup and recovery concurrency.

RECOVERY MANAGER automatically analyzes copies on stacked tapes in order to determine the most efficient order of recovery. It also performs automatic sizing of output copy data sets and sort work data sets for index recovery, check data, and check LOB functions.

Output data sets, job cards, and symbolic variables

When you specify a recovery job for one or more groups, you must supply the data set names into which the output JCL should be placed and the job card information for that JCL. Both items can include symbolic variables.

When you do not specify job card information for recovery job generation, RECOVERY MANAGER uses a default. You can view or update the defaults by:

- Selecting **Product Option Sets** in the RECOVERY MANAGER Main Menu
- Selecting the option set
- Expanding **Common Options**
- Expanding **Job Card Options**
You can also edit the default during job creation; the edited information then becomes the new default for your user ID.

**Output data set (JCLOUT) specification**

The name of the data set into which the JCL is placed must be the name of an existing cataloged partitioned data set or sequential data set.

You can use the following symbolic variables when you specify the data set name.

### Table 1: Output data set specification

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;SSID</td>
<td>ID of the DB2 subsystem</td>
</tr>
<tr>
<td>&amp;USERID, &amp;USER, &amp;UID</td>
<td>Time Sharing Option (TSO) user ID</td>
</tr>
<tr>
<td>&amp;SUIDnm</td>
<td>Substring of your user ID, starting with character n for a length of m characters</td>
</tr>
<tr>
<td>&amp;DATE</td>
<td>Current date (in the form YYMMDD)</td>
</tr>
<tr>
<td>&amp;TIME</td>
<td>Current time (in the form HHMMSS)</td>
</tr>
<tr>
<td>&amp;SEQ</td>
<td>(NGT Copy only) sequence number that increments with each reference The sequence number restarts at 1 for each job step and is used to provide unique output data set names.</td>
</tr>
</tbody>
</table>

Symbols with a numeric result must be prefixed by at least one alphabetic character.

**Job card (JCARD) specification**

The job card information (JCARD1 through JCARD5) must include symbolic variables to enable RECOVERY MANAGER to be able to utilize the multiple job optimization feature.

The job name must include one of the following symbolics in the job name:

&—the least significant digit of the job number

&##—the two least significant digits of the job number

&###—the three least significant digits of the job number

&####—all four digits of the job number

If you do not use this symbolic variable, RECOVERY MANAGER issues an error message.
Example

//SZJB&## generates job names SZJB01 and SZJB02.
//SZJB&### generates job names SZJB001 and SZJB002.

You can also use substrings of symbolic variables in job names in the format `symbolicVariable(n,m)`. The substring starts with character `n` for a length of `m` characters. In the following example, 2 is the substring starting position and 4 is the substring length:

\[
&DBNAME(2,4)
\]

The complete list of symbolic variables that are available to use in your job card specification is shown in Table 2 on page 53.

Table 2: Job card specification

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;SSID</td>
<td>ID of the DB2 subsystem</td>
</tr>
<tr>
<td>&amp;USERID, &amp;USER, &amp;UID</td>
<td>TSO user ID</td>
</tr>
<tr>
<td>&amp;SUIDnm</td>
<td>Substring of your user ID, starting with character n for a length of m characters</td>
</tr>
<tr>
<td>&amp;####</td>
<td>Generated numeric digits</td>
</tr>
</tbody>
</table>

- The maximum total length allowed for a data set name is 44 bytes.
- WARNING: You must include enough variables (#) in this value to handle the maximum number of jobs that you request RECOVERY MANAGER to generate. Otherwise, RECOVERY MANAGER might generate duplicate job names. (For example, if you want 10 jobs, you must specify at least &##:. If you want 100 jobs, you must specify &###:.)

RECOVERY MANAGER batch programs

The following table describes the batch programs provided by RECOVERY MANAGER for DB2 to perform backup and recovery.

The programs are listed in alphabetical order.
### Table 3: RECOVERY MANAGER batch programs

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARMBACT</td>
<td>Use the ARMBACT program to initialize all active logs for a specified SSID by calling DSNJLOGF. For more information, see “ARMBACT—Initialize active logs with DSNJLOGF” on page 403.</td>
</tr>
<tr>
<td>ARMBARC</td>
<td>Use the ARMBARC program for disaster recovery planning to create recovery site copies of the archive log data sets and to identify image copies on the log. For more information, see “ARMBARC—Archive log data sets” on page 409.</td>
</tr>
<tr>
<td>ARMBCOR a</td>
<td>RECOVERY MANAGER uses this program for the disaster recovery extend feature in a data sharing environment. ARMBCOR manipulates the value of the ARMBSDR member in the CNTL data set to ensure that all members are processed. For more information, see “ARMBSRR—System resource recovery” on page 673.</td>
</tr>
<tr>
<td>ARMBCRC</td>
<td>RECOVERY MANAGER uses this disaster recovery program to translate a timestamp to a relative byte address (RBA) or log range sequence number (LRSN) value. This translation provides a recovery point for disaster recoveries across members of a data sharing system and across multiple DB2 subsystems. For more information, see “ARMBCRC—Conditional recovery to a timestamp” on page 425. Note: The timestamp insertion program, ARMBTSI, inserts a row containing a user-specified timestamp into the RECOVERY MANAGER CRRDRPT table. ARMBCRC can then convert this timestamp to an RBA or LRSN. See “ARMBTSI —Time stamp insertion” on page 719 for more information.</td>
</tr>
<tr>
<td>ARMBEOL4</td>
<td>The ARMBEOL program is used in the JCL generated by ARMBSRR for non-data-sharing systems to truncate archive logs to assist in coordinated disaster recoveries or in recoveries to a user-specified time.</td>
</tr>
<tr>
<td>ARMBGEN</td>
<td>Use the ARMBGEN program to generate backup and recovery JCL for one or more application groups. For more information, see “ARMBGEN—Backup and recovery JCL” on page 431.</td>
</tr>
<tr>
<td>ARMBGIM</td>
<td>Use the ARMBGIM program to generate impact analysis reports for one or more application groups. For more information, see “ARMBGIM—Impact analysis” on page 465.</td>
</tr>
<tr>
<td>ARMBGNR</td>
<td>The ARMBGNR program copies to a permanent data set the JCL that ARMBGEN generates. For more information, see “Building the ARMBGNR JCL” on page 441.</td>
</tr>
<tr>
<td>ARMBGPS</td>
<td>Use the ARMBGPS program to create multiple groups that you can use for backup and recovery of the non-system objects. ARMBGPS creates or replaces the delta group (group 00). The delta group contains those spaces created or added since the last execution of ARMBGPS. For more information, see “ARMBGPS—Subsystem group split” on page 473.</td>
</tr>
<tr>
<td>Program</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ARMBGPV</td>
<td>Use the ARMBGPV program to revalidate the recoverability of a group and generate reports for object recoverability, recovery resources, and pick list. For more information, see “ARMBGPV—Group recovery revalidation” on page 489.</td>
</tr>
<tr>
<td>ARMBGRP</td>
<td>Use the ARMBGRP program to create, rename, update options, delete and report on application groups in batch mode. For more information, see “ARMBGRP—Group creation and maintenance” on page 507.</td>
</tr>
<tr>
<td>ARMBLGR</td>
<td>The ARMBLGR performs log range analysis for a subsystem. ARMBGEN requires this analysis when it performs a local point in time recovery via a conditional restart. For more information, see “ARMBLGR—Log range analysis” on page 597.</td>
</tr>
<tr>
<td>ARMBLOG</td>
<td>Use the ARMBLOG program during full subsystem recovery to issue an archive log command and wait for the completion of the archive log offload process. For more information, see “ARMBLOG—Archive log creation” on page 603.</td>
</tr>
<tr>
<td>ARMBLPL</td>
<td>Use the ARMBLPL program to check logical partitions of indexes and issue a START DB if the space is in LPL status. The program waits for the LPL status to clear.</td>
</tr>
<tr>
<td>ARMBLRD</td>
<td>Use the ARMBLRD program to read the log range file and then print it in a readable format to the TRACE file. This program is used for diagnostic purposes only. For more information, see “ARMBLRD—Log range formatting” on page 609.</td>
</tr>
<tr>
<td>ARMBMJO</td>
<td>The ARMBMJO program is used for controlling and restarting failed jobs generated online and by ARMBGEN for application object sets (groups). (ARMBMJO is not used to control or restart failed jobs generated by ARMBSRR.)</td>
</tr>
<tr>
<td>ARMRRDC</td>
<td>Use the ARMRRDC program to report data collected during actual, simulated, and estimated disaster recoveries. It is only available with the Recovery Management for DB2 solution. For more information, see “ARMRRDC—Recovery data collection report” on page 615.</td>
</tr>
<tr>
<td>ARMBRID</td>
<td>The recover indoubt threads program, ARMBRID, displays indoubt threads, parses the output, and executes Recover Indoubt commands as needed at the DR site. For more information, see “ARMBRID—Recover indoubt threads” on page 623.</td>
</tr>
<tr>
<td>ARMRRPR</td>
<td>Use the ARMRRPR program to print reports to gauge the progress of recoveries at the subsystem level and the group level. For more information, see “ARMRRPR — Progress Reports” on page 629.</td>
</tr>
<tr>
<td>Program</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ARMBSDR</td>
<td>The DREXTEND YES causes ARMBSRR to generate JCL to run ARMBSDR at the disaster recovery site as part of the Phase 1 job. ARMBSDR finds the most recent BSDS and archive log at the disaster recovery site (for each member if data sharing), updates the BSDS, and adds a new conditional restart control record to the BSDS. For more information, see “ARMBSDR—Extend recovery point at disaster recovery site” on page 651.</td>
</tr>
<tr>
<td>ARMBSET</td>
<td>The ARMBSET program uses stored information, pulls objects based on the specified OBJECTSET, and issues CHECK and REPAIR commands. The generation of the RECOVERY MANAGER job JCL completes with the automatic addition of ARMBSET without any input or changes necessary on your part. The ARMBSET syntax is generated from information that is already available. You can also use ARMBSET to issue LEVELID, START, and STOP commands against an object set or object partitions.</td>
</tr>
<tr>
<td>ARMBSRR</td>
<td>Use ARMBSRR to create batch jobs at the local site to restore DB2 system resources at the recovery site before you recover applications. For more information, see “ARMBSRR—System resource recovery” on page 673.</td>
</tr>
<tr>
<td>ARMBSTP&lt;sup&gt;a&lt;/sup&gt;</td>
<td>The ARMBSTP program is used in the JCL generated by ARMBSRR to stop and start spaces as required.</td>
</tr>
<tr>
<td>ARMBSYN&lt;sup&gt;a&lt;/sup&gt;</td>
<td>The ARMBSYN program is used during parallel processing to synchronize multiple jobs.</td>
</tr>
<tr>
<td>ARMBTRM&lt;sup&gt;a&lt;/sup&gt;</td>
<td>The ARMBTRM program is used in the JCL generated by ARMBSRR to terminate DB2 utilities running against the catalog and directory at the recovery site (except for COPY, REORG and LOAD).</td>
</tr>
<tr>
<td>ARMBTSI</td>
<td>Use the ARMBTSI program to insert a timestamp into the table that maps timestamps to relative byte addresses (RBAs) for DB2 conditional restart. The data is used by ARMBSRR and updated by ARMBCRC. For more information, see “ARMBTSI—Time stamp insertion” on page 719.</td>
</tr>
<tr>
<td>ARMBUTL&lt;sup&gt;a&lt;/sup&gt;</td>
<td>The ARMBUTL program is used in the JCL generated by ARMBSRR to terminate BMC utilities.</td>
</tr>
<tr>
<td>ARMBWDC</td>
<td>Use the ARMBWDC program to collect information about recovery start and end times for actual, estimated, and simulated disaster recoveries of system resources. It is only available with the Recovery Management for DB2 solution. For more information, see “ARMBWDC—System recovery data collection” on page 725.</td>
</tr>
</tbody>
</table>

<sup>a</sup> This book does not provide JCL or syntax for this program because it is only generated by RECOVERY MANAGER to perform internal functions.
System setup

For installation, RECOVERY MANAGER is supplied on the BMC distribution tape for products for DB2.

For installation information, see the BMC Installation System User Guide. For configuration information, see the BMC Products and Solutions for DB2 Configuration Guide.

Review this section for recommendations and requirements before you use RECOVERY MANAGER.

DB2 support

This version of RECOVERY MANAGER supports DB2 Versions 10 and 11.

Note

RECOVERY MANAGER does not support mixed mode DB2s.

System requirements

This version of RECOVERY MANAGER requires:

- z/OS Version 1.7 or later
- ISPF Version 3.1 or later

BMC product and component requirements

This version of RECOVERY MANAGER has the following requirements for BMC products and components:

- For RECOVERY MANAGER, version 9.2.00 is the minimum supported release for the following BMC products for DB2:
  - NGT Copy
  - Log Master
  - NGT Recover

If you try to use NGT Copy or NGT Recover versions earlier than 9.2.00 with RECOVERY MANAGER versions 10.1 or later, RECOVERY MANAGER online and ARMGEN JCL generation for both backup and recovery issues message BMC80324E and ends with a return code 8.

Because versions earlier than 9.2.00 use a different repository and do not support OBJECTSET syntax, RECOVERY MANAGER cannot generate JCL for NGT Copy or NGT Recover for those versions.
- BMC Solution Common Code (SCC) version 11.1.00 or later
  SCC is a set of technologies that provide common processes for several BMC products for DB2 including RECOVERY MANAGER.
  This component is installed during RECOVERY MANAGER installation but is maintained separately from RECOVERY MANAGER.

- BMC DB2 Component Services (DBC) version 10.1.00 or later
  Using DBC, you can type a question mark (?) in the first position in the Subsystem/group attach name field on the RECOVERY MANAGER Main Menu to display the available SSIDs. If you do this without DBC running, RECOVERY MANAGER issues an error message indicating that DBC was not found.
  DBC is also used to work with option sets and the DB2 Product Configuration technology.
  To use BMC DB2 Component Services (DBC) with RECOVERY MANAGER, you must have a DBC started task running.
  For more information about DBC, see the BMC Global Infrastructure Components Administration Guide.

- BMC DB2 Product Configuration technology (LGC) version 10.1.00 or later
  DB2 Product Configuration technology separates product (or solution) installation from configuration.
  Through its online interface, DB2 Product Configuration simplifies configuration. You can accept the default option values or make changes to them, if needed.
  DB2 Product Configuration panels simplify navigation by allowing you to expand or contract sections as needed. Also, you can link to DB2 Product Configuration from within your product or solution, thus maintaining a consistent look and feel, and retaining your changes from version to version.
  For more information, see “Option sets and configuration options” on page 733 and the BMC Global Infrastructure Components Administration Guide.

- BMC EXTENDED BUFFER MANAGER (XBM) or SNAPSHOT UPGRADE FEATURE (SUF) version 5.6 with PTF BPE0313 or BMC EXTENDED BUFFER MANAGER (XBM) or SNAPSHOT UPGRADE FEATURE (SUF) version 6.1
  If you want to offload eligible processing to a zIIP, you must have installed a minimum of version 5.6 with PTF BPE0313 of either XBM or SUF.

  **Note**
  To enable DB2 Version 10 support, XBM and SUF also require PTF BPE0311.

  If you use the XBMID option to specify a particular XBM subsystem, that subsystem must be at this maintenance level. If you do not specify a particular XBM subsystem and ZIIP ENABLED is in effect, RECOVERY MANAGER searches for an XBM subsystem at this level.
Additional information about BMC products and DB2

If you want the enhanced performance that is provided by the BMC utilities for DB2, you should have the NGT Recover, NGT Copy, CHECK PLUS, EXTENDED BUFFER MANAGER, and PACLOG products installed. The following table shows the recommended minimum version of these utilities required for full support and exploitation of each supported version of DB2.

In situations where an installed BMC utility does not support the DB2 recovery scenario, RECOVERY MANAGER uses DSNUTILB.

Table 4: BMC utility version requirements for DB2 support

<table>
<thead>
<tr>
<th>Product</th>
<th>DB2 Version 10</th>
<th>DB2 Version 11</th>
</tr>
</thead>
<tbody>
<tr>
<td>NGT Copy</td>
<td>10.1.00</td>
<td>11.1.00</td>
</tr>
<tr>
<td>CHECK PLUS</td>
<td>10.1.00</td>
<td>11.1.00</td>
</tr>
<tr>
<td>NGT Recover</td>
<td>10.1.00</td>
<td>11.1.00</td>
</tr>
<tr>
<td>R+/CHANGE ACCUM</td>
<td>10.1.00</td>
<td>11.1.00</td>
</tr>
<tr>
<td>PACLOG</td>
<td>10.1.00</td>
<td>11.1.00</td>
</tr>
<tr>
<td>Log Master</td>
<td>10.1.00</td>
<td>11.1.00</td>
</tr>
<tr>
<td>High-speed Apply Engine</td>
<td>10.1.00</td>
<td>11.1.00</td>
</tr>
</tbody>
</table>

a  This document lists product versions for informational purposes only. BMC does not necessarily support all of the listed versions. For information about supported versions, see the BMC website at http://www.bmc.com.

b  To use R+/CHANGE ACCUM, you must have NGT Recover installed.
This chapter describes how to get started with RECOVERY MANAGER.

For first time users of RECOVERY MANAGER

If you are using the RECOVERY MANAGER product for the first time, you should familiarize yourself with the information in this chapter, as follows:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Description</th>
<th>See</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installation considerations</td>
<td>Prerequisites for installation</td>
<td>“Before using RECOVERY MANAGER” on page 63</td>
</tr>
<tr>
<td>Task flow and panels</td>
<td>This chapter describes getting started with RECOVERY MANAGER. processes</td>
<td>“RECOVERY MANAGER online interface” on page 85</td>
</tr>
<tr>
<td>Authorizations</td>
<td>DB2 plan authorization required to access This chapter describes getting started with RECOVERY MANAGER.</td>
<td>“Authorizations” on page 89</td>
</tr>
<tr>
<td>DB2 subsystem library and resources</td>
<td>Display and change subsystem library and resource information</td>
<td>“RECOVERY MANAGER option sets and configuration options” on page 94</td>
</tr>
<tr>
<td>Supported utilities</td>
<td>Support for both IBM and BMC utilities</td>
<td>“Utilities supported by RECOVERY MANAGER” on page 66</td>
</tr>
<tr>
<td>System resource maintenance and</td>
<td>Automates system resource maintenance and recovery</td>
<td>“DB2 system resource maintenance and recovery” on page 68</td>
</tr>
<tr>
<td>recovery</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application backup and recovery</td>
<td>Tailor backups and recoveries by object or by application group</td>
<td>“Backing up groups” on page 157 and “Recovering groups” on page 199</td>
</tr>
<tr>
<td>information</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
RECOVERY MANAGER task overview

The tasks that you can perform with RECOVERY MANAGER fall into the following major categories.

The figure below shows the general flow of those tasks.

- Creating groups of DB2 recoverable objects (table spaces and indexes)
- Generating backup or recovery JCL and reports for those groups
- Generating maintenance or recovery JCL for DB2 system resources
- Making contingency preparations for disaster recovery

Figure 2: Processing objects and resources for backup and recovery
Before using RECOVERY MANAGER

This section provides information about aspects of RECOVERY MANAGER that require special consideration and may require additional actions before you start using RECOVERY MANAGER for backup and recovery purposes.

Adding a DB2 subsystem to RECOVERY MANAGER

When you install RECOVERY MANAGER for the first time, the BMC Installation System:

■ Generates the RECOVERY MANAGER option set (the ARM$OPTS file)
■ Creates the repository tables for the specified DB2 subsystem
■ Creates synonyms to R+/CHANGE ACCUM and Log Master if required

To add a subsystem to RECOVERY MANAGER after the initial installation, see the BMC Installation System User Guide.

RECOVERY MANAGER and PACLOG option set considerations

RECOVERY MANAGER and PACLOG use the DB2 Product Configuration technology (LGC) for option sets.

Option sets provide values for the configuration options for RECOVERY MANAGER and PACLOG. The default option sets are

■ ARM$OPTS for RECOVERY MANAGER
■ ALM$OPTS for PACLOG

You should set the option values in ARM$OPTS and ALM$OPTS to the same values for an SSID.

Using the same values in the option sets ensures that both products use the same archive history file, time stamp value, and work data sets, all of which are specified in the option sets.

For more information, see “Option sets and configuration options” on page 733.
Sharing BMC tables

RECOVERY MANAGER shares tables with other BMC products. For example, RECOVERY MANAGER uses the BMCLGRNX table during recovery point analysis and the BMCXCOPY table for index backups and Instant Snapshot copies.

The following tables should be shared with other BMC products:

- BMCUTIL
- BMCSYNC
- BMCXCOPY
- BMCLGRNX

Sharing solution common code (SCC)

The Solution Common Code (SCC) is a component of RECOVERY MANAGER.

SCC is a supporting software component that provides common processes for several BMC products for DB2. The SCC component requires no action from you except when maintenance is required. You can view the version of the SCC code on RECOVERY MANAGER reports or by selecting the About option on the RECOVERY MANAGER Main Menu. You can also see a list of applied fixes by using the About option on the RECOVERY MANAGER Main Menu.

Setting up DB2 group attach names

If you are operating in a data sharing environment, you can define DB2 group attach names during RECOVERY MANAGER installation. You must also define other control information before you can use those group attach names.

Creating required temporary tables

RECOVERY MANAGER requires declared DB2 global temporary tables and uses them when working with groups and in several other functions throughout the product.

DB2 Solution Common Code (SCC) also uses global temporary tables for dynamic grouping.

To ensure that you have enough space allocated for processing, set up the temporary tables, as follows:
DB2 Version 10 and DB2 Version 11 systems

DB2 uses the work file database to dynamically allocate the global temporary tables. For each DB2 subsystem and for each member of a DB2 data sharing system, you must ensure that the work file database contains at least one DB2 storage-group-defined table space with a page size of 32 KB.

IBM recommends that at least one 32 KB storage-group-defined table space be defined for global, temporary tables, and indexes. (See information for a new zparm WFDBSEP in IBM APAR PM02528 WORKFILE DATABASE USABILITY ENHANCEMENT.)

Preparation for archive logs greater than 64K tracks

The following procedure describes preparation for archive logs greater than 64K tracks.

To successfully use archive logs greater than 64K tracks which are available with DB2, you must set up some SMS rules.

1 Create an SMS DATACLAS with a Data Set Name Type LARGE. This value assigns a DSORG type of PS-L to the data set. The simplest way to accomplish this is to make assignments based on a data set name filter. For example:

   WHEN (&DSN = DSNDXW.DXW2.ARCLG1L.A0*)  
   SET &DATACLAS = 'DCLARGE'

2 Create a DATACLAS rule to accommodate the temporary files created by some RECOVERY MANAGER programs when processing archive logs. These files are identified with .Z0* and should also be allocated as DSNTYPE=LARGE. For example:

   WHEN (&DSN = DSNDXW.DXW2.ARCLG1L.Z0*)  
   SET &DATACLAS = 'DCLARGE'

3 The archive log files and the temporary files can be extremely large, so you might want to set up a STORCLAS rule and a STORGRP rule to direct the data sets to a specific SMS Storage group. For example:

   WHEN (&DATACLAS = 'DCLARGE')  
   SET &STORCLAS = 'DXWSMS'

and

   WHEN (&STORCLAS = 'DXWSMS')  
   SET &STORGRP = 'DXWSMS'
Safeguarding the repository

You should make backup copies of the repository table spaces as soon as possible after installing RECOVERY MANAGER.

You can use the online interface to create the JCL to do this (“Creating a repository group for backup or recovery” on page 289). You should continue to make backups at intervals that are compatible with your standard practices.

In the event of a DB2 failure that involves the repository, you can continue to use RECOVERY MANAGER to create groups and generate JCL. However, you cannot save group information or retrieve information that has already been saved until the repository failure is resolved.

When you perform a recovery at a disaster recovery site, you must recover objects in the correct order. Use the ARMBSRR program to make available and to recover the catalog and directory, the repository, and the R+/CHANGE ACCUM repository in the correct sequence. For more information, see “Recovering from a DB2 system disaster” on page 291 and “ARMBSRR—System resource recovery” on page 673.

Utilities supported by RECOVERY MANAGER

The following are the utilities that are supported by RECOVERY MANAGER.

For version information, see “BMC product and component requirements” on page 57.

Note

When used as a component of the BMC Recovery Management for DB2 solution, RECOVERY MANAGER uses the BMC utilities as the default for recover and copy functions. When used alone, RECOVERY MANAGER uses the IBM DB2 utilities as defaults for recover, copy, and check functions.

Utilities for performing or assisting in the recovery function

The following utilities are used when performing recoveries:

- The BMC NGT Recover utility
- The BMC R+/CHANGE ACCUM utility (requires the corresponding version of NGT Recover)
- The IBM DB2 RECOVER (DSNUTILB) utility
- The IBM IDCAMS utility for deleting/redefining data sets prior to recovery
- The BMC NGT Copy utility
- The BMC PACLOG utility

RMGR can use PACLOG to move archive logs to DASD at a recovery site in preparation for disaster recovery

- The BMC Log Master for DB2 utility (with the High-speed Apply Engine (formerly APPLY PLUS))
- The BMC EXTENDED BUFFER MANAGER (XBM) SNAPSHOT UPGRADE FEATURE utility

**Utilities for generating a recovery point**

The IBM DB2 QUIESCE (DSNUTILB) utility is used for generating a recovery point.

**Utilities for performing integrity checking of DB2 objects**

The following utilities are used for performing integrity checking of DB2 objects:
- The BMC CHECK PLUS utility
- The IBM DB2 CHECK (DSNUTILB) utility

**Utilities for performing post-recovery image copy functions**

The following utilities are used for performing post-recovery image copy functions:
- The BMC NGT Copy utility
- The IBM DB2 COPY (DSNUTILB) utility
- The BMC NGT Recover utility (using the OUTCOPY option)

**Utility for performing repair functions**

The IBM DB2 REPAIR (DSNUTILB) utility performs repair functions on individual objects to reset check or rebuild pending status when required.

**Utilities for performing backup functions**

The BMC NGT Copy utility is used for performing backup functions.
When making Instant Snapshot copies with NGT Copy, you need either EXTENDED BUFFER MANAGER (XBM) or SNAPSHOT UPGRADE FEATURE (SUF):

- The IBM DB2 COPY (DSNUTILB) utility
- The BMC NGT Recover (using the OUTCOPY option)
- The online consistent copy technology (available with the BMC Recovery Management for DB2 solution)

Utilities for performing BSDS maintenance and recovery

The following utilities are used for performing BSDS maintenance and repair:

- The IBM DB2 Change Log Inventory (DSNJU003) utility
- The IBM Print Log Map (DSNJU004) utility
- The IBM operating system utilities (IEBGENER, IDCAMS, and IKJEFT01)

DB2 system resource maintenance and recovery

You can use RECOVERY MANAGER to automate the maintenance or recovery of DB2 system resources.

In general, DB2 does not need to be active to perform system resource maintenance and recovery tasks because these tasks do not require access to the DB2 catalog or to the repository. Exceptions are the recovery of the repository itself and the generation of disaster recovery JCL—both require DB2 to be active.

If DB2 is not active when you generate the maintenance or recovery JCL, RECOVERY MANAGER must get some of the bootstrap data set and archive log information from the control information records and some of it from you. This requirement is also true if the SCCAUTH module is not accessible for some reason (see “APF authorization” on page 90).

If you are authorized to access RECOVERY MANAGER, you can generate system resource maintenance and recovery JCL. However, you must also have authority to execute the DB2 utilities that are invoked by the JCL (see “System resource authorization” on page 92).

All of the system resource maintenance and recovery tasks are accessed through the System resources option on the Main Menu and are described in Table 5 on page 69.
<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backup and recovery of the DB2 catalog and directory (DSNDB01 and</td>
<td>RECOVERY MANAGER generates JCL to back up or recover the catalog and directory spaces in the correct sequence. RECOVERY MANAGER uses DSNUTILB for these tasks. See “DB2 catalog and directory” on page 266.</td>
</tr>
<tr>
<td>DSNDB06)</td>
<td></td>
</tr>
<tr>
<td>Active and archive log recovery</td>
<td>RECOVERY MANAGER generates JCL to recover the active and archive logs. The procedures depend on whether single or dual logs are used and on whether the archive logs are cataloged. See “DB2 active logs” on page 269 and “DB2 archive logs” on page 273.</td>
</tr>
<tr>
<td>BSDS maintenance and recovery</td>
<td>RECOVERY MANAGER generates JCL to maintain or recover the BSDS. The procedures depend on whether DB2 is active and the type of failure. Maintenance tasks generate Change Log Inventory jobs. See “DB2 BSDS recovery and maintenance” on page 277.</td>
</tr>
<tr>
<td>View physical data set attributes</td>
<td>RECOVERY MANAGER uses the IDCAMS LISTCAT utility to obtain information about the physical attributes of both system and nonsystem data sets. For more information, see “Physical data set attributes” on page 286.</td>
</tr>
<tr>
<td>Repository backup and recovery</td>
<td>RECOVERY MANAGER generates JCL to back up or recover the BMC Common DB2 repository, RECOVERY MANAGER repository, the R+/CHANGE ACCUM repository (if installed), and the Log Master repository (if installed). The BMC Common DB2 repository and the RECOVERY MANAGER repository store all information relating to application groups and RECOVERY MANAGER options that you have created. See “The repository” on page 288.</td>
</tr>
<tr>
<td>Disaster recovery planning assistance</td>
<td>RECOVERY MANAGER automates procedures for disaster recovery preparation. For more information about using RECOVERY MANAGER for disaster recovery planning, see “Recovering from a DB2 system disaster” on page 291.</td>
</tr>
<tr>
<td>Logging environment planning information</td>
<td>The RECOVERY MANAGER logging environment modeling tool allows you to examine and view different logging scenarios for a selected DB2 subsystem in order to optimize the amount of DASD space required by the active and archive logs. For more information, see “Modeling the DB2 logging environment” on page 369.</td>
</tr>
</tbody>
</table>
Performance considerations

This section provides information about improving performance when using BMC Recovery Management for DB2.

Avoid RUNSTATS on BMCLGRNX

Do not run RUNSTATS against the RECOVERY MANAGER BMCLGRNX table space.

Doing so causes DB2 to perform an unnecessary table space scan on SYSLGRNX when doing unchanged analysis. Instead, if you have statistics in the DB2 catalog for BMCLGRNX, you should reset the values to -1. A sample job is located in the ARMRESET member in the .ARMSAMP data set that was created during installation. For more information about DB2 statistics, see the IBM DB2 Administration Guide.

Improving recovery time

To recover an application as quickly as possible after a failure, perform as much group processing as possible before recovery becomes necessary.

In particular, perform the following tasks:

- Whenever possible, create object groups before a potential recovery and routinely update and revalidate their recoverability. In the case of a recovery to a prior point in time, you should ensure that indexes and objects that are related by referential integrity are included in the group.

- Run the RUNSTATS utility as often as needed to keep DB2 catalog statistics current. Out-of-date catalog statistics can adversely affect both the accuracy of the JCL and the time that is taken to generate it. Alternatively, keep BMCSTATS information up-to-date using DASD MANAGER PLUS or NGT Copy, and set the appropriate sizing option to instruct RECOVERY MANAGER to use those statistics instead of catalog statistics.

**WARNING**

Do not execute the RUNSTATS utility against the BMCLGRNX table. Doing so will cause RECOVERY MANAGER to perform a table space scan on SYSLGRNX when performing unchanged analysis.

- Run the RUNSTATS utility on the repository to improve performance if the online performance seems slow.
In general, when the recovery of a group becomes necessary, always revalidate the recoverability of the group and then generate and submit the JCL. JCL that is generated at an earlier time may be inaccurate.

Use multiple job optimization wherever possible to provide the best overall recovery time. See “Multiple job optimization” on page 71 for more information.

When recovering an entire table space, do not split it by partition unless you make copies by partition and one of the following scenarios applies:

— You are using DSNUTILB as the recover utility.

— You are using multiple job optimization.

— You are using the RECOVERY MANAGER UNLOADKEYS/BUILINDEX strategy.

Limit SYSIBM.SYSCOPY searches to avoid unneeded input/output (I/O) operations and memory use. You can set the value of the LIMIT_SYSCOPY_SEARCH option to limit the period that RECOVERY MANAGER searches the table for a requested copy or quiesce point. If you do not set this value, RECOVERY MANAGER does not limit the search. For more information, see “General recovery options” on page 847.

Always consider generating recovery JCL in batch mode instead of performing this task online. Doing so frees up your terminal during processing.

Include indexes and objects that are related by referential integrity only for point-in-time recoveries or when you generate JCL for disaster recovery.

Choosing BMC utilities can provide a significant improvement compared with IBM utilities. Also, choosing NGT Recover as your recover utility lets you perform recovery simulation.

Avoid backing up objects with no changes since the previous backup. Avoid recovering objects that have not changed since the recovery point in time. Use the XUNCHANGED option in ARMBGEN or the Unchanged option on the online panels.

**Multiple job optimization**

RECOVERY MANAGER provides the Multiple Job Optimization feature that creates multiple jobs for the recovery of a group.

If you are recovering using the BMC NGT Recover product, this feature can take advantage of the UNLOADKEYS/BUILINDEX strategy and also, in combination with batch JCL generation, provides a new and powerful tool for offsite recovery.
When RECOVERY MANAGER creates multiple jobs, it saves them in a single member of a partitioned data set or sequential file by default. You can optionally save them to separate members, which enables you to control job submission although it decreases the performance benefits (see “Separating jobs from a multi-job batch job stream” on page 77).

The number of jobs that are created is controlled by the following factors:

- The value that you provide at the Max concurrent jobs prompt in the subsystem or group Recovery Options panel sets the maximum number of jobs that can run concurrently during recovery of a given group. RECOVERY MANAGER might use less than this number depending on other conditions.

- Groups created using the ARMBGPS program are designed to have one recovery job per group. ARMBGPS automatically sets the Max concurrent jobs option to 1 when the groups are created.

**WARNING**

If you provide a value that is higher than the number of initiators that are available, an unending wait situation might arise during recovery JCL execution.

- If a noticeable disparity exists among the sizes of objects in a group, the number of jobs that RECOVERY MANAGER creates may be less than the specified maximum. This situation may occur when RECOVERY MANAGER finds that the job with the largest estimated execution time can no longer be split into multiple jobs. Estimated recovery time is relative to the objects in the group and may be influenced by the following factors:
  - Number of pages to be restored from an image copy
  - Number of pages to be copied (after the recovery)
  - Amount of work space required for index unload or build
  - Amount of work space required for check data
  - Utilities used in the recovery

The exact formula is proprietary and was the result of extensive benchmark testing.

- If the group includes objects that require a resource that cannot be shared, the objects will be recovered in the same job. Examples of such resources include tape volumes that contain stacked image copies, archive logs on tape, or change accumulation files on tape. In this situation, RECOVERY MANAGER may limit the number of jobs that it creates to less than the specified maximum. RECOVERY MANAGER uses the following configuration options to determine if resources are on tape:
  - Primary Arc on Tape
  - Alternate Arc on Tape
  - Change Accum on Tape
Important recommendations

This topic describes the important recommendations regarding RECOVERY MANAGER.

If possible, limit the size of these groups to no more than a few hundred objects, both table spaces and indexes. One large group requires more time for JCL generation than the time required for the same set of objects when divided into smaller groups. Use ARMBGPS to split all objects in a subsystem into multiple groups.

Also, if possible, limit the SYSCOPY search in the group or subsystem options.

Using multiple job optimization with NGT Recover

If you are recovering with the BMC NGT Recover product, the creation of multiple jobs for the recovery of a group uses the UNLOADKEYS/BUILDINDEX strategy.

To take advantage of this strategy, the following criteria must be met:

- Image copies of the partitions must exist on separate tape volumes.
- You must select the objects by partition when you build the group.
- You must use NGT Recover as the recovery utility.
- You must select UNLOADKEYS/BUILDINDEX in the subsystem or on the Group Recovery Options panel.

---

Note

UNLOADKEYS/BUILDINDEX and MAXKSORT > 1 are mutually exclusive options.

---

Using multiple job optimization in offsite recovery

Using both the ARMBGEN batch JCL generation program and multiple job optimization, RECOVERY MANAGER can produce a complete set of JCL for the recovery of your application data at your recovery site.

This JCL can be designed and optimized to meet your specific recovery site needs. To take the fullest advantage of this capability, BMC recommends that you perform the following steps:

- Specify the RECOVER TORESTARTRBA syntax option when you code the JCL for the ARMBGEN batch program. When you specify this option, the system resource recovery program, ARMBSRR, provides the restart relative byte address (RBA) value to ARMBGEN to ensure that the recovery of your DB2 system objects and application objects at the recovery site are correctly synchronized.
Create a set of groups specifically for use in recovery site JCL generation. Place objects in these groups to reflect the sequence in which you want them to be recovered at the recovery site.

The following steps are an example of a procedure to follow for multiple job optimization:

**To optimize multiple jobs**

1. Create a group called OFFSITE_PRIORITY_01 containing all of the objects (both table spaces and related indexes) that you want to have the highest priority for recovery at the recovery site. Then create another group, OFFSITE_PRIORITY_02 for the next lower priority level, and so on.

   **Tip**
   
   If possible, limit the size of these groups to no more than a few hundred objects. One large group requires more time for JCL generation than is required for the same set of objects when divided into smaller groups. Also, performing data set sizing separately and storing in the repository can speed JCL generation time. For more information, see “Data set sizing” on page 80. Also, if possible, limit the SYSCOPY search (for more information, see “General recovery options” on page 233).

2. Set the value of **Max concurrent jobs** (on the Recovery Options Specification panel) for these groups to the number of initiators that will be available for DB2 recoveries at your recovery site.

3. Run the ARMBGEN program to create a fully optimized set of offsite recovery JCL for each of these groups after you have run the ARMBSRR system resource recovery program.

4. Send the generated recovery JCL offsite along with the JCL that is created by the system resource recovery program, ARMBSRR.

   For more information about planning for offsite recovery, see “Recovering from a DB2 system disaster” on page 291.

**Optimized recovery job processing**

RECOVERY MANAGER has the following paths for optimizing recovery for a set of jobs:

- For jobs generated online and by ARMBGEN for application data, RECOVERY MANAGER uses ARMBMJO (Table 3 on page 54) and the JOB_RESTART table (“Job history: JOB_RESTART table” on page 806) to control and restart failed jobs. For more information, see “Restarting jobs that recover application data” on page 224.
For ARMBSRR jobs for system resource recovery, RECOVERY MANAGER uses a synchronization file to restart failed jobs. For more information, see “Restarting system resource recovery (ARMBSRR) jobs” on page 226.

With multiple job optimization, recovery JCL is placed in a single user-specified data set or member, unless you specifically separate the jobs into separate members (see “Separating jobs from a multi-job batch job stream” on page 77). The JCL consists of up to n jobs, where the variable n is the user-specified maximum plus two. In the example shown in Figure 3 on page 76, multiple jobs have been generated and execute as described in the following steps.
For jobs generated by ARMBSRR, two of these jobs will have the same job name (JOB1).

**Figure 3: Optimized recovery job processing**

The sequence of events is as follows:

1. JOB0 performs the following tasks, and then submits the remainder of the jobs:
   - For jobs generated by ARMBSRR, allocates a job synchronization file.
   - For online and RMBGEN jobs, initializes the JOB_RESTART table.
2. For ARMBSRR jobs, submit restart JOB1.
3. For ARMBSRR jobs, synchronize files.
4. Recovery task synchronization.
5. For ARMBSRR jobs, delete the synchronization file.
   - For online and ARMBGEN jobs, run CLEAR_TABLE found in ARMBMJO$ SAMPLIB to delete all rows in JOB_RESTART table.
6. Job set complete?
   - No
   - Yes
7. Successful completion?
   - No
   - Yes
8. Enable restart for set of jobs.
For jobs generated online and by ARMBGEN, initializes the JOB_RESTART table.

2 JOB1 performs the following tasks:

3 For ARMBSRR jobs, the first JOB1 submits another JOB1 (a restart job), which will determine whether the set of jobs completes successfully. The first JOB1 then performs initial recovery steps.

4 For jobs generated online and by ARMBGEN, JOB1 runs concurrently with JOB2.

5 JOB2 through JOBn perform recovery tasks to ensure that these tasks are performed in the correct sequence.

6 For ARMBSRR jobs, the jobs are under the control of the job synchronization file.

7 For ARMBGEN jobs, the jobs are under the control of ARMBMJO and the JOB_RESTART table.

8 When the recovery jobs are completed, the following tasks are next:

9 For ARMBSRR jobs, the restart JOB1 determines whether the jobs have completed successfully. If so, the synchronization file is deleted. If not, this job is used to restart the set of jobs.

10 For jobs generated online and by ARMBGEN, if any of the jobs failed, ARMBMJO uses the JOB_RESTART table to determine which jobs and steps to execute when you resubmit a set of jobs or a failed job. You can clear the JOB_RESTART table by running CLEAR_TABLE found in the ARMBMJO$ SAMPLIB.

For more information, see “Restarting a recovery for a set of concurrent jobs” on page 223.

Separating jobs from a multi-job batch job stream

You can optionally generate certain types of multi-job batch JCL into separate members of a partitioned data set.

This option enables you to run the generated jobs separately instead of running in a single execution.

**WARNING**

When you separate the jobs in a multi-job job stream, you lose the performance benefits that come from running jobs concurrently. You also cannot take advantage of the automatic job synchronization provided by RECOVERY MANAGER when creating a single job stream.
This feature is only valid for:

- Batch recovery job streams for which you have specified SYNC=NO and MAX_CONCURRENT_JOB greater than 1
- The ARMBGPS subsystem group split batch job stream (which does not require synchronization steps)

This feature is incompatible with UNLOADKEYS_BUILDINDEX=YES, which requires synchronization steps. If you have specified UNLOADKEYS_BUILDINDEX=YES or SYNC=YES, RECOVERY MANAGER overrides the parameter and generates the jobs into a single member.

To save the jobs into separate members:

- (online) When generating batch jobs, specify Yes at the If output data set partitioned, one job per member (batch) field on the batch JCL generation panel
- (batch) code the MEMBER=YES parameter in the ARMBGNR execution statement. For example:

```plaintext
//ARMDD003 EXEC PGM=ARMBGNR,COND=(4,LT),PARM='MEMBER'
//              REGION=0M
```

---

**Improving catalog and directory recovery**

The information in this section applies to cdisaster recoveries or full subsystem local point-in-time recoveries only.

Use the ARMBARC program or PACLOG to collect image copy information from the log and store it in the archive history file. If you do not want to make copies of the archive logs, run ARMBARC with the keyword HISTONLY to gather the image copy information. When ARMBSRR generates the recovery JCL for catalog and directory recovery, it checks SYSIBM.SYSCOPY and the archive history file for copy information. If copies are registered for all spaces and stacked tape is detected, ARMBSRR generates the necessary DD statements to retain the tape or tapes. Tape retention saves elapsed time by avoiding unnecessary mounting and positioning of tapes.

**Note**

For ARMBSRR to generate JCL that minimizes tape movement, you must run ARMBARC or PACLOG to store information about the catalog and directory copies.

ARMBSRR can also split the catalog recovery into parallel jobs to further speed the recovery process. The initial set of catalog and directory spaces must be recovered serially. These spaces are:
The image copies for those spaces can be stacked on tape. The remaining table spaces and their indexes can be recovered in separate parallel jobs if they are not stacked on the same tape or tapes. To improve the overall elapsed time of the catalog recovery, review the remaining table space sizes in your shop and consider putting some of the largest ones on separate tapes for parallel recovery. The following spaces are typically large:

- DSNDB01.SCT02
- DSNDB01.SPT01
- DSNDB06.SYSPLAN

To speed the recovery, use the MAXCATJOBS option in ARMBSRR to specify the maximum number of jobs or tasks in a single job to use to recover the catalog. (This value should be based on the number of tape drives that are available.)

Enhancing JCL generation and performance

The following subsections discuss methods of enhancing JCL generation and performance.

DB2 statistics

To get the best recovery performance, RECOVERY MANAGER JCL generation uses the statistics stored in the DB2 catalog.

Out-of-date catalog statistics can adversely affect both the accuracy of the JCL and the time taken to generate it. Use the RUNSTATS utility as often as necessary to keep DB2 catalog statistics current. NGT Copy can be used to update table space level statistics. RECOVERY MANAGER also has the option to use the BMCSTATS catalog statistics that are produced by DASD MANAGER PLUS or NGT Copy instead of DB2’s catalog statistics.
When to split by partition

When recovering an entire table space, split it by partition when one of the following applies:

- You made copies by partition and you are using DSNUTILB as the recover utility.
- You made copies by partition and are requesting multiple job generation. See “Multiple job optimization” on page 71.
- You are using the NGT Recover UNLOADKEYS/BUILDINDEX strategy.

Note
Splitting by partition can increase the time needed to generate JCL.

Data set sizing

For JCL generation, RECOVERY MANAGER performs a number of sizing calculations. The following types of data set sizing are performed:

- Sort work file sizing for index recovery
- Image copy sizing for disk copies
- Work file sizing for CHECK data
- Table space sizing for multiple job optimization
- IDCAMS data set allocation size

RECOVERY MANAGER uses statistics from the DB2 catalog when available, as well as information from the ICF catalog to size objects. Because sizing calculations can involve a significant amount of DB2 access and processor time, RECOVERY MANAGER has options to reduce the use of those resources during JCL generation.

RECOVERY MANAGER performs sizing calculations at the time of JCL generation.

Data set sizing options

The following options are available for the Dataset Sizing field on the General Recovery Options panel.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catalog</td>
<td>Use the DB2 and ICF catalog information for sizing purposes. This is the default and requires sizing calculations to be performed at the time of JCL generation.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Defaults</td>
<td>Use existing default sizing information from the Work File options established in the Recovery options specification. This option bypasses the sizing calculations by using the primary and secondary space values set in the Work File Options.</td>
</tr>
<tr>
<td>BMCSTATS</td>
<td>Use sizing information that is generated when you run BMCSTATS. These statistics are collected by DASD MANAGER PLUS and optionally by NGT Copy.</td>
</tr>
</tbody>
</table>

**Note**

The IDCAMS data set allocation (performed when you have requested that the VCAT-defined space be deleted and redefined) uses ICF catalog information regardless of the Data Set Sizing option setting. If ICF information is not available, RECOVERY MANAGER uses the chosen setting for the Data Set Sizing option.

**To set data set sizing options for a subsystem**

1. From the RECOVERY MANAGER Main Menu, select option **Subsystem options**.
2. Specify Action **2. Update** and Utility Type **1. Recovery**.
3. Specify **1. General recovery options** and press **Enter** twice to display the second General Recovery Options panel (Figure 4 on page 81).

**Figure 4: General Recovery Options panel**

```
ARMROG02 ====== Browse General Recovery Options SUB SYSTEM DEDL ===============
Command ===> _________________________________________________________________
With defaults and overrides. Press Enter to continue.

Lvl (S=System  G=Group  Blank=RMGR default)
Delete STOGROUP objs... 2 1. Yes 2. No (Prior to recovery)
REUSE .................. 2 1. No 2. Yes 3. NOSCR (NOSCR for R+ only)
Max concurrent jobs ... 1
Dataset Sizing ......... 1 1. Catalog 2. Defaults 3. BMCSTATS
Always rebuild indexes... 2 1. Yes 2. No
Use INDEX ALL recover .. 2 1. Yes 2. No

Mirror Strategy
BSOS. ................. 1. Both 2. Copy 1 3. Copy 2
Active Logs ............ 1. Both 2. Copy 1 3. Copy 2
Catalog and Directory . 2 1. Yes 2. No
Rmgr Repository ....... 2 1. Yes 2. No
Log Master Repository . 2 1. Yes 2. No
```

**Note**

Although the **Data Set Sizing** option is located with the General Recovery Options, it is also used when generating backup JCL with any copy utility other than NGT Copy.
To set data set sizing options for a group

1. Display a group.

2. From the Group Edit Panel, specify 1. **Utility options**.


4. Specify 1. **General recovery options** and press **Enter** twice to display the second General Recovery Options panel (Figure 4 on page 81).

INDEX ALL recovery

Using the INDEX ALL option can significantly reduce the amount of time required for JCL generation for groups with a large number of indexes (for example, ERP applications such as SAP/R3).

When you select the INDEX ALL option, the ARMBGEN program does not search for indexes that are related to objects in a group during JCL generation because the inclusion of those indexes is implied. Without the INDEX ALL option, the ARMBGEN program searches for indexes related to the objects within the application group and automatically performs data set sizing.

**Note**

If you use INDEX ALL, you must set the primary and secondary allocations in the work file options for the group.

**WARNING**

Do not use this option if your application group includes indexes or if the group was created by partition because it would cause multiple recoveries of the same index.

Interactive versus batch processing

Objects can be processed quickly and efficiently either interactively or in batch mode.

The interactive method makes use of the convenience of the online interface. The batch method is more powerful and often faster. Extremely large groups (those containing more than a few hundred objects) should always be processed in batch mode.
Interactive versus batch group creation

You can create application groups either interactively or in batch mode. For performance reasons, you should create groups that will contain extremely large numbers of objects in batch mode.

For information about creating groups interactively, see “Creating and working with groups” on page 105. For information about building application groups in batch mode, see “ARMBGRP—Group creation and maintenance” on page 507.

Interactive versus batch revalidation of recoverability

You can revalidate the recoverability of groups one at a time using the online interface.

You can revalidate the recoverability of many groups concurrently when you use batch revalidation. For performance reasons, you should revalidate extremely large groups (those containing more than a few hundred objects) in batch mode.

For information about revalidating the recoverability of a group interactively, see “Group recovery revalidation and reporting” on page 133. For information about batch revalidation, see “ARMBGPV—Group recovery revalidation” on page 489.

Interactive versus batch JCL generation

You can generate backup or recovery JCL using the online interface.

You can specify recovery points, recovery options at the group level, revalidate the recoverability of a group, and specify such things as alternate recovery points for individual objects when necessary in a single TSO session. Apart from providing a job statement for the generated backup or recovery job, you are not required to code any JCL.

You can also use the TSO session to generate backup and recovery JCL in batch mode, although you can also completely avoid using a TSO session if you code your own JCL and use the ARMBGEN program to generate the backup and recovery jobs. In addition to releasing the RECOVERY MANAGER online interface for other work, ARMBGEN provides better automation for point-in-time recovery following an application failure and more automation for disaster recovery preparation.
Note
You can set group backup and recovery options that will remain in effect only for the duration of the JCL generation by using the UPDATE option of the ARMBGEN program. You can permanently change group backup or recovery options prior to JCL generation by using the ARMBGRP program.

For more information, see the following:

- “Generating batch recovery JCL interactively” on page 211.
- “Generating a backup job in batch” on page 164.
- “Generating recovery JCL interactively” on page 208.
- “Generating recovery JCL in batch” on page 210.
- “ARMBGEN—Backup and recovery JCL” on page 431.

RECOVERY MANAGER CLIST

As you install RECOVERY MANAGER, one of the installation jobs places a CLIST named ARMISPF in the library named HLQ.DBCLIB (or copies it to a different library that you specify during installation).

The HLQ value represents a qualifier that is assigned in your environment during installation.

The ARMISPF CLIST gives you the ability to specify both a unique application ID and the DB2 subsystem on which RECOVERY MANAGER will run. The application ID is used as a prefix for the applidPROF member of the ISPFPROF data set, and enables you to store and access multiple profile members that contain settings and options to be used with RECOVERY MANAGER.

In the following examples,

- clistLibName is the name of the data set where ARMISPF exists
- ssid is the identifier of the DB2 subsystem where the product will run
- applid is the 1 - 4 character application identifier that is used as the prefix of the applidPROF member of the ISPFPROF data set. The default is ARM.

To run the online interface, execute ARMISPF in one of the following ways:
To execute from ISPF Option 6, enter one of the following TSO commands, depending on whether you want to pass the DB2 subsystem ID (SSID), the application ID (APPLID), or both:

- `EX 'clistLibName(ARMISPF)'`
- `EX 'clistLibName(ARMISPF)' 'SSID (ssid)'`
- `EX 'clistLibName(ARMISPF)' 'APPLID (applid)'`
- `EX 'clistLibName(ARMISPF)' 'SSID (ssid) APPLID (applid)'`

To execute from an ISPF panel, add an option to the panel that executes ARMISPF. For example, if you use RM as the option name, enter one of the following lines in the panel definition, depending on whether you want to pass the SSID, the application ID (APPLID), or both:

- `RM, 'CMD(EX "clistLibName(ARMISPF)")'`
- `RM, 'CMD(EX "clistLibName(ARMISPF)" SSID(ssid))'`
- `RM, 'CMD(EX "clistLibName(ARMISPF)" APPLID(applid))'`
- `RM, 'CMD(EX "clistLibName(ARMISPF)" SSID(ssid) APPLID(applid))'`

### RECOVERY MANAGER online interface

The RECOVERY MANAGER online interface provides options that allow you to select objects, create object groups, prepare objects for backup or recovery, and maintain and recover system resources.

#### The Main Menu

To access the RECOVERY MANAGER Main Menu (see the following figure), use the method employed at your company.

If you want to return to the Main Menu while you are in RECOVERY MANAGER, press **F3** until the Main Menu appears. To access online help, press **F1**.

As you use RECOVERY MANAGER, you will encounter different types of panels depending on the task you are performing. Instructions for using each panel are shown on the panel and detailed help is available by pressing **F1**.

**Note**

Some online panels have additional options when used as part of the BMC Recovery Management for DB2 solution and may vary slightly from the samples shown in this manual.

**Figure 5: RECOVERY MANAGER Main Menu**

```
ARMPRIM ====== RECOVERY MANAGER FOR DB2 V11.2.00 - Main Menu =============
Command ===> _________________________________________________________________
```

Type selection. Then press Enter.
Figure 6 on page 86 shows the RECOVERY MANAGER menu when used as part of the Recovery Management solution. In addition to the standard options, you can also access the Log Master for DB2 online interface, the R+/CHANGE ACCUM online interface, and modify configuration options. For more information, see the Recovery Management for DB2 User Guide.

**Figure 6: RECOVERY MANAGER Main Menu (as part of the Recovery Management solution)**

ARMPRIM =========== RECOVERY MANAGEMENT FOR DB2 V11.2.00 - Main Menu =============
Command ===> _________________________________________________________________

Type selection. Then press Enter.

0. About - Get product and licensing information
1. Application groups - Backup, recover or maintain application groups
2. Appl. group definition - Select object(s) for an application group
3. System resources - Maintain or recover DB2* system resources
4. Subsystem options - Set subsystem default options
5. Product Option Sets - Set RECOVERY MANAGEMENT Product options
6. Subsystem recovery - Preparation and recovery of entire DB2 system
7. Recovery progress - Report recovery progress for objects or groups
8. LOG MASTER for DB2 - Invoke LOG MASTER for DB2 online interface
9. R+/CHANGE ACCUM for DB2 - Invoke R+/CHANGE ACCUM for DB2 online interface

Subsystem / group attach name. . . . DECI
Current SQLID. . . . . . . . . . . . RDAXXX_
Option Set . . . . . . . . . . . . . ARMP$OPTS
Site type . . . . . . . . . . . . . . 1 1. As running 2. Local 3. Recovery

**Subsystem group attach name**

RECOVERY MANAGER versions 9.2.00 and later use BMC DB2 Component Services (DBC) to show available SSIDs from the RECOVERY MANAGER Main Menu. Type a question mark (?) in the first position in the Subsystem / group attach name field to display the SSIDs.

To use this feature, you must have a DBC started task running. For more information about DBC, see the BMC Global Infrastructure Components Administration Guide.

**Current SQLID**

The Current SQLID is displayed on the Main Menu and defaults to your current user ID.
This ID becomes the high-level qualifier in the name of any group that you create in a RECOVERY MANAGER session. If you want to use one of your secondary IDs, change it on the Main Menu before proceeding to the next panel. If your primary user ID has SYSADM or system DBADM authority, you can change your SQL ID to any value.

**ISPF function keys**

In the RECOVERY MANAGER panels, you can display the active ISPF function keys by using the ISPF command PFSHOW ON or turn them off by using the PFSHOW OFF command.

The main keys that are used in RECOVERY MANAGER are

- **F3**—returns you to a prior panel
- **F4**—zooms in to display an entry that is too long to fit on the panel (such as objects with long names) or, on the Group Definition Display panel, the SELECT statement for groups defined with dynamic SQL
  To display the SELECT statement, position the cursor on the Type field of the DYN SQL line and press F4. You can also use zoom on the command line.
- **F1**—displays a help panel or additional information about a displayed message (see “Online help and messages” on page 100).
- **Enter**—confirms to RECOVERY MANAGER that your actions in the current panel are complete

**Online display of DB2 long names**

RECOVERY MANAGER supports objects having DB2 long names up to 128 bytes in length, including:

- Object creator name
- Index name
- STOGROUP name
- Collection name
- Package name

The names are truncated to fit on the panels but can be fully displayed using panel options. You can also control how the names are truncated by setting configuration options (see “Setting long name display options for a subsystem” on page 88).
Displaying full text of DB2 long names (Zoom)

Display the full text of a long name by using either the Zoom action code or the F4 key, as follows:

1. Enter Z in the Act field beside the object you want to view and press Enter or position the cursor anywhere in the field that you want to expand and press F4.

   A pop up panel displays the full text of both parts of the long name.

2. Press F3 to close the pop up panel.

Setting long name display options for a subsystem

Set truncation options that direct which portion of a long name is truncated when displayed on a panel and what characters are used as the substitution string for the truncated part of the object name.

To set truncation options for RECOVERY MANAGER

Perform the following tasks to set the truncation options for a subsystem.

1. From the Main Menu, select 5 Product Option Sets.

2. On the Product Option Sets panel, select the option set that you want to change. The default option set is ARM$OPTS.

3. On the panel that opens for the option set, select Subsystem Options, and then the subsystem ID for which you want to set truncation options.

4. Scroll through the options until you see the Truncation Position and Truncation Characters fields.

5. Set the Truncation Position (the default is E for End).

6. Set the Truncation Characters (the default is &gt &gt).
Example

The long name
RMD128CHARACTERCREATOR.IC15P21L128MAXIMUMCHARACTERIXN
will be truncated as follows: **Truncation Position: 3 (End), Truncation**
Characters: `>>`
RMD128 >>.IC15P21L128MAXIM >>
**Truncation Position: 2 (Middle), Truncation Characters: `>>/b>`
RMD>>TOR.IC15P21L>>ACTERIXN
**Truncation Position: 1 (Beginning), Truncation Characters: `!!`**
!!REATOR.!!IMUMCHARACTERIXN

Authorizations

This topic describes the authorizations required to access RECOVERY MANAGER, work with application groups, and execute backup or recovery JCL.

**Tip**
If you use the CA ACF2 security system and your shop is restricting TSO commands, add ARMUMAN, ARMUSEL, ARMOPTM, and DSNJU004 to the list of commands in the TSOCMDS module. If your site restricts the use of TSO commands through an option of the system security package (such as the IBM RACF component of the z/OS Security Server or ACF2) or an add-on product such as PCF, be sure the ARMUMAN, ARMUSEL, ARMOPTM, and DSNJU004 command names are added to the appropriate command table. Otherwise, the message IKJ56500I command COMMAND NOT FOUND is issued when attempting to invoke the RECOVERY MANAGER CLIST or when using the logging environment modeling tool.

RACF authorization

The RACF security administrator must define an Open Multiple Virtual Storage (OMVS) segment for each Recovery Management user.

The user ID assigned to the DBC started task must also have an OMVS segment defined.

The OMVS segment is required because DBC utilizes IBM z/OS UNIX System Services (USS) sockets for cross-address-space communication within an LPAR.
APF authorization

The RECOVERY MANAGER load library must be APF authorized. In addition, you must add SCCAUTH to the AUTHPGM NAMES section of member IKJTSOxx in SYS1.PARMLIB.

Note
SCCAUTH is a common authorization module used by multiple BMC products, including the components of the BMC Recovery Management for DB2 solution.

DB2 plan authorization

If the ssid.PUBLICPLAN configuration option is set to YES in the option set, RECOVERY MANAGER grants EXECUTE authority to PUBLIC the first time the product is run and then dynamically binds the plan. If the PUBLICPLAN option is set to NO, you must grant EXECUTE authority to users as needed.

For more information about configuration options, see “Option sets and configuration options” on page 733.

Group authorization

Each RECOVERY MANAGER application group has an owner (creator) who can give authority for that group to any number of users.

In addition to the creator, only authorized users can maintain and save groups. Any user who has SYSADM or system DBADM authority (or whose secondary IDs have SYSADM or system DBADM authority) are considered authorized users for all groups in the subsystem. All users can display a list of group names, but only the creator and authorized users can update or delete a group.

Note
External security, such as ACF2 and RACF, is supported for opening and saving groups.

Naming a new group

Each group name includes the authorization ID of its creator.

When you create a new group, RECOVERY MANAGER identifies it (until you save it under another name) as sqlID.UNNAMED_GROUP, where the variable sqlID is
the creator part of the name and is your current SQL ID (which defaults to your
primary logon user ID).

Note
You can change your SQL ID to one of your secondary user IDs on the Main Menu
before proceeding to create the new group.

If you save the group to the repository, the group is saved as sqlID.name, where the
variable name is a long ID string of your choice.

If you modify your SQL ID on the Main Menu, the change is saved and is shown the
next time you access RECOVERY MANAGER.

When you save a new group or save an existing group under a different name, you
can change the group name to one that uses one of your secondary user IDs. Users
who have SYSADM or system DBADM authority (or whose secondary IDs have
SYSADM or system DBADM authority), can specify any AUTHID as the creator of
the group.

Adding or revoking authorized users

You can add or revoke authorized users of a group through the Group Edit
Authorization panel. Although RECOVERY MANAGER tracks the ID of the grantor
of another user’s authorization, there is no cascading when revoking authorization.

For more information about setting group authorizations, see “Setting group
authorizations” on page 128.

Types of authorization

The types of user authorization for an existing group that are provided in
RECOVERY MANAGER are as follows:

- **TYPE A (ALL)** provides the authority to open a group, save a group with changes,
  and generate JCL.

- **TYPE O (OPEN)** provides the authority to open a group and generate JCL, but
does not allow the user to save any changes to the group back to the repository.

- **PUBLIC** allows any user to open or save object groups. PUBLIC may also be used
  as the creator part of any group name.
A user with SYSADM or system DBADM authority (or with secondary IDs with SYSADM or system DBADM authority) is considered to have both TYPE A and TYPE O authority on the group.

Some authorization scenarios

The following examples show how authorizations can be implemented to satisfy different requirements:

- A group that is used and maintained by an individual could use that user’s primary ID as the creator part of the name and selectively provide access to other users as needed.

- A group that is used and maintained by a group of people could use a secondary ID that represents that group of people. This is the most flexible scenario, because it gives access to all members with the secondary ID and can avoid duplicate security administration among RECOVERY MANAGER, DB2, and the security package.

- TYPE O authorization is provided to a user who generates backup or recovery JCL. The user is allowed to make changes to the "working" group but is not allowed to save it to the repository.

System resource authorization

RECOVERY MANAGER does not verify your authority to execute the following BMC, DB2, and operating system utilities which might be called in the generated JCL:

- NGT Copy
- CHECK PLUS
- NGT Recover
- DB2 Print Log Map (DSNJU004)
- DB2 Change Log Inventory (DSNJU003)
- Recovery Log Extractor (DSN1LOGP)
- DB2 RECOVER (DSNUTILB)
- DB2 COPY (DSNUTILB)
- DB2 CHECK (DSNUTILB)
- IEBGENER
- IDCAMS

Refer to the appropriate utility reference documentation for information about the authorizations required to use the listed utilities.
Disaster recovery authorizations

When you make disaster recovery preparations at the local site, you must have additional authorizations for the following activities:

- Creating offsite copies of archive logs by using the ARMBARC program (see “ARMBARC—Archive log data sets” on page 409)
- Creating system resource recovery JCL by using the ARMBSRR program (see “ARMBSRR—System resource recovery” on page 673)
- Analyzing the logs for subsystem recovery by using the ARMBCRC program (see “ARMBCRC—Conditional recovery to a timestamp” on page 425)

To run the system resource recovery JCL at the recovery site, you need authority (at the recovery site) to execute the DB2 and operating system utilities that are listed in “System resource authorization” on page 92. You also need the following authorizations:

- ALTER authority on the BSDS and active log data sets
- DB2 installation SYSADM or installation SYSOPR authority
- ALTER authority on the archive log data sets when you are performing subsystem recovery

Authorization to modify subsystem backup and recovery options

If you have authority to access RECOVERY MANAGER, you can also display, change, and add to any subsystem default backup and recovery options that are not already defined. However, to save any changes that you make to these options, you must have one of the following DB2 authorizations:

- Installation SYSADM
- SYSADM or system DBADM
- DBADM for the repository database

Authorization to use delete and redefine recovery options

To use the Delete STOGROUP objects and Redefine VCAT objects options on the Recovery Type Selection panel, you must have authority to issue DB2 STOP and DISPLAY commands on the objects that are being recovered.
You also need control authority on the physical data sets. See “General recovery options” on page 233 for information about the Delete and Redefine options.

RECOVERY MANAGER option sets and configuration options

You can browse and update RECOVERY MANAGER option sets and configuration options by selecting Product Option Sets on the Main Menu.

ARM$OPTS is the default option set. The option sets and configuration options includes the following items:

- BMC utilities that are available to RECOVERY MANAGER
- TSO, DB2, and user step libraries
- DB2 subsystem resources and JCL destination information
- Operating system resources and job card specification

Any changes that you make to the option set and configuration options are saved for new RECOVERY MANAGER sessions. For more information, see “Option sets and configuration options” on page 733.

WARNING
The configuration information is stored by subsystem ID. For data sharing systems, you may need to update the configuration information for each member. Each subsystem is listed under the option set that you select from Product Option Sets on the Main Menu.

RECOVERY MANAGER backup and recovery options

The utility options that you need to generate JCL for object backup or recovery are determined by your utility choices and your recovery strategy.

The utility options are also affected by whether you specify backup and recovery options for use globally, for use at the group level, or some combination of these. Because those options can be specified in different ways, RECOVERY MANAGER uses rules of precedence to select the appropriate option if conflicting instructions are provided during job preparation.
Option specifications can exist in three different ways. RECOVERY MANAGER provides configuration options which are established during installation that apply across all subsystems. Additionally, you can specify options by subsystem or data sharing group, and by group. This method allows up to three specifications for the same option to exist concurrently.

The rules (shown in “Important recommendation” on page 95) that determine which option specification is used for an object are as follows:

- When you specify one or more options for a group of objects, your choices apply to all objects in the group. If you do not specify a particular option for the group, the value of that option defaults to the value that applies to the subsystem.

- When you specify one or more options for the entire DB2 subsystem or data sharing group, your choices apply to all groups and all objects except where corresponding options have already been specified by group. If you do not specify an option at the subsystem or data sharing group level, the installed default applies (except in those cases already mentioned). See “Default backup and recovery options” on page 96 for more information.

Note
You can view the value in effect for a particular option at a particular level (that is, with any overrides from levels with a higher precedence already applied) only when you browse the options at the level of interest. When you select Update, you see only those values that have been set at that level.

Important recommendation

As a general rule, you should set backup and recovery options at the most general level and specify exceptions at the group level.

This provides ways to make changes to those options with minimum effort.
The rules of precedence that are used by RECOVERY MANAGER when conflicting values are specified for an option are shown in Figure 7 on page 96.

Figure 7: How option conflicts are resolved

For information about how to specify options, see the following topics:

- “Displaying, updating, and deleting backup options for standard copies” on page 158
- “Setting subsystem-level options” on page 98

Default backup and recovery options

For recovery options, RECOVERY MANAGER provides default values that are used across all DB2 subsystems unless you make changes at the individual subsystem or group levels.

These configuration options provide a set of values that will work for all objects across all subsystems and cannot be modified through the user interface. However, they are not optimal for all scenarios, and it is likely that you will want to specify
some options at the subsystem or group level to enhance backup or recovery job performance (for example, using NGT Recover instead of DB2 RECOVER).

\textbf{Note}

Some BMC utility options that may be used by RECOVERY MANAGER are neither accessible through the user interface nor provided as defaults by RECOVERY MANAGER.

For backup options, RECOVERY MANAGER uses values that are provided by the copy utility of choice. Except for identifying the default utility, RECOVERY MANAGER supplies no values directly.

For information about RECOVERY MANAGER option default values, see the following topics:

- “Displaying, updating, and deleting backup options for standard copies” on page 158
- “Working with recovery options” on page 229

\section*{Subsystem-level considerations}

You can browse, update, or delete subsystem backup and recovery options from the Main Menu.

Initially, no values are provided, and you must enter new values if you want them to be different from the defaults that are provided by RECOVERY MANAGER. Although you can display and change the current option settings, you cannot save these without the proper authorization. For more information, see “Authorization to modify subsystem backup and recovery options” on page 93.

The options that are available for update at the subsystem level are identical to those available at the group level. For more information, see the following topics:

- “Working with recovery options” on page 229
- “Displaying, updating, and deleting backup options for standard copies” on page 158

Use the procedure in “Setting subsystem-level options” on page 98 to update subsystem options. Only one set of subsystem-level options is stored in the repository; therefore, that set applies to all members within a data sharing group.
Tip

- Do not set the options for alternate resources and site type at the subsystem level. These options cause the recover utility to utilize specific recovery resources in a given order. Setting these values at the subsystem level results in all recoveries utilizing the same default recovery resources and site type, which may not be appropriate for all recovery scenarios.

- Establish backup and recovery option values for the subsystem before specifying values for groups in the subsystem.

- Keep the values of the default work unit and space allocations current, especially if you use RECOVERY MANAGER to prepare for disaster recovery.
  
  The system resource recovery job creation program, ARMBSRR, uses these values to dynamically allocate a significant number of data sets and to calculate catalog and directory space. Select Subsystem Options on the RECOVERY MANAGER Main Menu to verify the current values of the default work unit and space allocations.

- Specify a work file space equal to or larger than the space that is required by your largest directory space.

Setting subsystem-level options

Starting at the Main Menu, you can browse, update, or delete subsystem options for the entire DB2 subsystem.

You can select backup options, recovery options, work file options, and post-recovery copy options. If you are using the BMC Recovery Management for DB2 solution, you can also set your Service Level Agreement recovery time. If you have the proper authority, the changes take effect immediately.
Before you begin

You might need the following information to set subsystem options:

■ Recovery options:
  — The recover utility that you will use and the options that you will change
  — Work file option information

If you want post-recovery image copies created immediately after recovery, you need to know the following:
  — Which copy utility that you will use
  — Which image copy types that you will make
  — Which copy options that you will change

■ Backup options:
  — The backup utility and the backup options that you will change
  — Which image copy types that you will make

You will also need the following authorizations:

■ EXECUTE authority for RECOVERY MANAGER

■ SYSADM, DB2 installation SYSADM, system DBADM, DBADMAUTH, DBMAINTAUTH, or DBCNTLAUTH authority for the repository

You can also set up authorization verification mechanisms as follows:

**Note**

RECOVERY MANAGER performs external security checking if you select option 4 **Subsystem options** from the Main Menu. This check is not used in batch because subsystem options are only updated online.

■ If the DB2 DSNX@XAC authorization exit is available for your system, RECOVERY MANAGER uses this exit to verify authorization for external access. The exit is available from the following sources:

  — IBM provides a sample exit with DB2 for the IBM Resource Access Control Facility (RACF) component.

  — CA Technologies provides the DSNX@XAC exit with CA-ACF2 Security for DB2 and CA-Top Secret Security for DB2.

BMC recommends this mechanism for implementing external security. The access control authorization exit must be available in the STEPLIB, JOBLIB, linklist, or in the SYS3.DSN exit.
If the DSNX@XAC exit is not available, RECOVERY MANAGER uses the standard DB2 method to check security.

**To browse, update, or delete subsystem-level options**

Start this procedure at the RECOVERY MANAGER Main Menu.

1. In the RECOVERY MANAGER Main Menu, select **Subsystem options**.

2. In the Utility Options Specification panel, select **Browse**, **Update**, or **Delete** and **Recovery** or **Backup**, as required. You can also select **Process**, which enables you to update your service level agreement options for full subsystem recover.

   Follow the instructions that are displayed on this and subsequent panels to view or change options for this subsystem. For more information, see:

   - “Displaying, updating, and deleting backup options for standard copies” on page 158
   - “Displaying, updating, and deleting recovery options” on page 231
   - “Service level agreement options” on page 294

   **Note**

   If you confirm a deletion, the option defaults that are supplied with RECOVERY MANAGER will now apply for the subsystem instead of the deleted values. See “Default backup and recovery options” on page 96.

3. To return to the Main Menu, press **F3**.

---

### Online help and messages

RECOVERY MANAGER for DB2 provides online Help information as follows:

- Panel Help that provides assistance in using the current panel

- Message help which provides additional information about messages that you may receive while you are using RECOVERY MANAGER and which allows you to access the appropriate panel help

The following subsections provide more information about online presentations.
Panel help

To view Help information for a RECOVERY MANAGER panel, press **F1** while the panel is displayed.

Continue as follows in any topic Help panel:

- To continue to the next Help panel, press **Enter** or **F8**.
- To return to the previous Help panel on a continuation Help panel, press **F7**.
- To return to the RECOVERY MANAGER panel, press **F3**.

**Note**

The ISPF help commands S (SKIP), T (TOC), and I (INDEX) are not supported in RECOVERY MANAGER Help panels.

For information about the help available when working with option sets and configuration options in the DB2 Product Configuration technology, see “Option sets and configuration options” on page 733.

Messages and message help

RECOVERY MANAGER uses three classes of messages—informational, error, and warning.

<table>
<thead>
<tr>
<th>Message type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Informational messages</td>
<td>For information only and require no action on your part</td>
</tr>
<tr>
<td></td>
<td>For example, a GROUP SAVED message is issued when you successfully save a group to the repository. Informational messages have an <strong>I</strong> appended to the message number.</td>
</tr>
<tr>
<td>Warning messages</td>
<td>Generated when a situation occurs that could possibly have an unintended impact on your data</td>
</tr>
<tr>
<td></td>
<td>For example, the message NO MATCHES FOUND is issued when no DB2 objects can be found that match the given criteria. Warning messages have a <strong>W</strong> appended to the message number.</td>
</tr>
<tr>
<td>Error messages</td>
<td>Generated when RECOVERY MANAGER detects that you have provided incorrect input</td>
</tr>
<tr>
<td></td>
<td>For example, the short message INVALID ENTRY is issued when you type an incorrect action code in the Act column on an Object List panel. Error messages have an <strong>E</strong> appended to the message number.</td>
</tr>
</tbody>
</table>

RECOVERY MANAGER messages are available online. They are also available in the BMC Documentation Center.
In the online interface, RECOVERY MANAGER returns messages in several formats, as follows:

- A short, concise message may appear in the top right corner of the panel in which the error occurred. To obtain further information, press F1. A numbered long form of the message prefixed with BMC appears on line 3 of the current panel. Press F1 again to display the Error Message Long Text panel, which provides a full explanation and any actions that you should take.

When a short message is displayed, the message is removed when you do one of the following:

1. Press F3 to return to the prior panel.
2. Press F1 to access panel Help.
3. (if it is an informational message) Continue processing.
4. (if it is an error message) Correct the error and continue.

- A numbered long form message prefixed with BMC may appear on an Error Message Long Text panel. To display more information (explanation and user action), press F1 again.

- One or more numbered messages prefixed with BMC may appear on an Error Message List panel after pressing Enter to leave a task panel. To obtain an explanation and any user action, type S in the Sel (Select) column and then press Enter. The information appears on an Error Message Long Text panel.

Syntax rules

The following syntax rules apply to all batch programs of: RECOVERY MANAGER.

- Terminate commands with a semi-colon (;).
- You can code commands on multiple lines.
- You can code Multiple commands and execute each command after it passes its syntax check.
- RECOVERY MANAGER ignores columns 73-80.
- RECOVERY MANAGER ignores blank lines.
- You can specify comments in two ways, depending on whether you want the entire line to be considered a comment.
—Specify an asterisk (*) or double hyphen (--) starting in column 1 to cause the entire line to be considered as a comment.

—Specify a double hyphen (--), double slash (/ /), or slash asterisk (/ *) starting in column 2 through 72 to cause the remainder of the line to be considered a comment.

■ Wildcards are supported in group names as follows:

—The characters % (percent) and * (asterisk) can be used as wildcards to match any number of characters in a name.

—The character ? (question mark) can be used as a wildcard to match a single character in a name.
Syntax rules
Creating and working with groups

This chapter describes how to create and work with groups.

About RECOVERY MANAGER groups

A RECOVERY MANAGER group (or object set) is a set of table spaces and indexes (usually belonging to the same application) that are grouped together in order to perform common backup and recovery procedures.

Grouping simplifies the backup and recovery process because you can use the generated JCL to back up or recover the entire group at once. Grouping also simplifies the specification of backup and recovery options, because you can specify the options just once for the entire group.

You can build and process groups by using the online interface or by using the batch programs. Using the batch program is faster for creating groups that contain more than a few hundred objects.

Tip
Whenever possible, you should build your groups before a backup or recovery becomes necessary. In the case of a recovery to a prior point in time, you should ensure that indexes and objects that are related by referential integrity are included in the group.

Creating groups

You can create RECOVERY MANAGER groups either online or by using the ARMBGRP or ARMBGPS batch programs, as follows:

- The online interface provides the advantages of real-time information, validation, and flexibility while building the group. For more information, see “Using the online interface to build groups” on page 112.
The ARMBGRP program enables you to build very large groups quickly and copy or rename an existing group. For more information, see “ARMBGRP—Group creation and maintenance” on page 507.

The ARMBGPS program automatically creates balanced groups containing all objects in an entire DB2 subsystem or a subset of objects in the DB2 subsystem (based on the pattern that you specify). For more information, see “ARMBGPS—Subsystem group split” on page 473.

Using dynamic grouping

RECOVERY MANAGER versions 9.2.00 and later use dynamic grouping to obtain the objects in a group.

With dynamic grouping, when a group is referenced (created, opened, or updated), RECOVERY MANAGER returns all of the objects that meet the group definition at that moment in time. These groups include wildcards for pattern matching in their definitions. (For more information about wildcards, see “Set specification expansion” on page 110.)

---

**Note**

With dynamic grouping, RECOVERY MANAGER does not store the names of the objects in a group (as it did in RECOVERY MANAGER versions 9.1.00 and earlier) so there is no need to revalidate the contents of the group to keep that information up to date.

You can still create static groups by defining a group as a list of individual objects with no wildcards included. The group attempts to resolve to the same set of objects each time the group is opened. However, if one or more objects have been dropped, the dropped objects will not be present in the list of objects. Also, groups that you create based on exception statuses and volume serial numbers (VOLSERs) are created as static groups.

While the dynamic grouping capability is used at the time of JCL generation, once the JCL is generated, the list of objects in the JCL becomes a static list unless you use the OBJECTSET syntax.

With dynamic grouping, you can create an empty group (a group for which no objects are found that meet the definition). After you create the objects and then open the group, the objects are found and displayed in the group. This ability enables you to prepare to back up and recover objects that do not yet exist. For example, if you know that a new application is being added to your system, you can set up the group in advance using wildcards to define the group. When the objects are created and the group is opened, RECOVERY MANAGER automatically finds the objects for the group and adds them to the backup and recovery jobs.
Using the repository

RECOVERY MANAGER uses tables in the following repositories:

- The BMC Common DB2 repository
  RECOVERY MANAGER stores group information in this repository, so it is the primary repository for dynamic grouping. For more information about these tables, see “BMC Common DB2 repository tables” on page 767.

- The RECOVERY MANAGER repository
  RECOVERY MANAGER does not store group information in this repository, so it has no role in dynamic grouping but is needed for other RECOVERY MANAGER functions. For more information about these tables, see “RECOVERY MANAGER repository” on page 773.

Note
Throughout this book, the term "the repository" is used to refer to both of these repositories, the BMC Common DB2 repository and the RECOVERY MANAGER repository, because they are both used concurrently by RECOVERY MANAGER. Also, the term "object set" can be used interchangeably with the term "group." (The BMC DASD MANAGER PLUS product uses the object set nomenclature in their documentation.)

Specifying objects for a new group

You can specify objects for a new group in a number of ways, depending on your situation or application, as follows:
### Table 6: Group creation methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Table space and/or Owner</strong></td>
<td>Specify a table space name (<code>databaseName.tableSpaceName</code>) or pattern to create the group. This method is usually the fastest way to produce a group of table spaces and associated indexes. Optionally, you can specify the owner in the pattern. The default value is an * (asterisk). You can manipulate the list to exclude or add additional spaces. You can also include one or more partitions when working with partitioned table spaces. <strong>Note:</strong> You cannot include a specific partition of a given table space in the same list as the complete table space (the DSNUM 0 version). You can optionally include objects associated with the objects included in the group, as follows:  ■ Associated indexes  ■ Objects associated by referential integrity  ■ Objects associated by LOB relationship  ■ Objects associated by XML relationship  ■ Objects associated by a history (versioning) relationship</td>
</tr>
<tr>
<td><strong>Table name</strong></td>
<td>Specify a table name (<code>creator.tableName</code>) or pattern to create the group. You can manipulate the list to exclude or add additional tables. <strong>Note:</strong> You cannot list partitions with the list of tables. You can optionally include objects associated with the objects included in the group, as follows:  ■ Associated indexes  ■ Objects associated by referential integrity  ■ Objects associated by LOB relationship  ■ Objects associated by XML relationship  ■ Objects associated by a history (versioning) relationship</td>
</tr>
<tr>
<td><strong>Index</strong></td>
<td>Specify an index name (<code>authID.indexName</code>) or pattern to create the group. When you specify the indexes, you can also optionally list one or all partitions of the index.</td>
</tr>
<tr>
<td><strong>Indexspace</strong></td>
<td>Specify an index space name (<code>databaseName.indexSpaceName</code>) or pattern to create the group. When creating a group by index space, you can include any number of index space names or wildcard patterns.</td>
</tr>
<tr>
<td><strong>Volume</strong></td>
<td>Specify up to eight volumes concurrently by using the operating system volume names. You cannot use wildcards to specify a group of volumes. For more information, see “Creating volume-based groups” on page 118.</td>
</tr>
<tr>
<td><strong>STOGROUP</strong></td>
<td>Specify objects by storage group name or pattern to create the group.</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Plan</td>
<td>Specify objects by plan name or pattern to create the group. Note: Groups built by plan are obsolete in DB2 Version 10 due to an empty SYSIBM.SYSPLANDEP table. RECOVERY MANAGER handles this situation with the following restrictions: When running on DB2 Version 10 or later, RECOVERY MANAGER will include objects with package dependencies for groups built by plan name. These restrictions do not apply to Repository plan groups.</td>
</tr>
<tr>
<td>Package</td>
<td>Specify objects by package ((collectionID.package)) to create the group.</td>
</tr>
</tbody>
</table>
| Exception status | Specify objects by exception status to create the group. You can also choose to include all objects in one or more status types, all objects that are related by referential integrity, and all indexes on those objects. The status types that you can choose from are:  
  ■ Check pending (CHKP)  
  ■ Advisory reorg pending  
  ■ Copy pending (COPY)  
  ■ Advisory rebuild pending  
  ■ Informational copy pending  
  ■ Auxiliary check pending  
  ■ Logical page list (LPL)—the object has entries in the logical page list  
  ■ Auxiliary warning  
  ■ Recover pending (RECP)  
  ■ Error range detected  
  ■ Group recover pending (GRECP)  
  ■ Stopped error range (STOPE)—stopped by DB2 due to log RBA error  
  ■ Rebuild pending  
  ■ Reorg pending  
  ■ Persistent read only  
  ■ Read or replication only  
  Note: Recover pending is the default status when you choose to create a group by exception status. |
## Method Description

**Dynamic SQL**  
Specify objects by using a user-defined SQL SELECT statement, such as the following statement:

```
SELECT 'TS', DBNAME, NAME FROM SYSIBM.SYSTABLESPACE WHERE DBNAME = 'RMDDDB48'
```

or

```
SELECT 'SG', CREATOR, NAME, CREATOR, NAME FROM SYSIBM.SYSSTOGROUP WHERE NAME LIKE 'JTR%'
```

Specify TS, IX, or SG as the first variable. RECOVERY MANAGER checks to make sure that 'TS', 'IX', or 'SG' follows the SELECT statement. Only one SQL statement is accepted. The statement is limited to 16,000 characters. (See “To create a group using dynamic SQL” on page 114.)

**Recovery Groups**  
For ARMBGPS size balancing for a subset of the DB2 subsystem, specify the include and exclude patterns with the table space name or pattern to create the group. These groups are referred to as ARMBGPS application groups and simplify the method for generating multiple jobs for an application. (These groups are TYPE BA in the OBJSETS table.)

**Note:** ARMBGPS groups LOB, XML and History objects with their related base table space, not by using the explicit name of the LOB, XML, or History object.

**Repository plan**  
Specify a group of plan names corresponding to the repository plan names for RECOVERY MANAGER, NGT Recover, and Log Master.

**Group combination**  
Specify objects that are contained in other existing groups to create a new group. Groups created via this method simply copy the definitions of the specified existing groups and do not pull information directly from the catalog and directory. This method provides a way of create a new group using mixed criteria (for example, storage group name and table space pattern).

**Subsystem**  
Automatically create a set of balanced groups for an entire DB2 subsystem. For more information, see “Full subsystem recovery” on page 347.

### Set specification expansion

Wildcard processing uses the following characters and results in the use of a SQL LIKE operator for expansion:

- *(asterisk)* and % (percent sign): matches 0 to any number of characters in an object name string
? (question mark) : matches any single character in an object name string

If the qualifier does not contain any of the characters, RECOVERY MANAGER uses an equal operator (=) to locate a match. The qualifier terminates when the first blank encountered.

When the patterns are expanded, all exclude patterns are processed first to build an exclusion list that is then applied to the include patterns, which are processed in order of inclusion.

For more information, see “Using wildcards when building groups” on page 112.

SQL specification

If the pattern type is SQ, you enter a SQL statement. Whenever you enter or update the SQL statement, RECOVERY MANAGER performs a SQL PREPARE INTO to verify that the statement is valid. For more information, see “To create a group using dynamic SQL” on page 114.

Supporting Unicode

RECOVERY MANAGER supports DB2 objects with Unicode names (both those that can be translated to EBCDIC and those that cannot).

RECOVERY MANAGER online panels and reports make use of EBCDIC characters. Any Unicode characters that cannot be translated into EBCDIC are represented with hexadecimal substitution characters.

Supporting DB2 long names

RECOVERY MANAGER supports objects having DB2 long names up to 128 bytes in length, including

- Object creator name
- Index name
- STOGROUP name
- Collection name
- Package name

The names are truncated to fit on the panels but can be fully displayed using panel options. You can also control how the names are truncated by setting configuration options. See “Online display of DB2 long names” on page 87 for more information.
WARNING

Do not fall back to DB2 enable new function mode (ENFM) from new function mode (NFM) after creating objects with long names. RECOVERY MANAGER cannot process objects with long names in ENFM.

Supporting clone tables

RECOVERY MANAGER supports clone objects when running with DB2 Version 10 or later and not in compatibility mode.

Using wildcards when building groups

To specify a group of objects (except those based on volume, subsystem, or exception), you can include one or more wildcards in the name string; to specify a single object, you use an explicit name for the object.

When you specify a wildcard pattern to generate a list of objects, you can use additional wildcard patterns to add more objects to the list or replace the list entirely.

The characters % (percent) and * (asterisk) can be used as wildcards to match any number of characters in an object name string. The character ? (question mark) can be used as a wildcard in a name string to match a single character.

Name strings can be delimited by double quotation marks. However, characters that are contained in a delimited identifier are considered not to be wildcard characters. When a name string consists of multiple part names, the parts are separated by a period. Three identifier types are used as parts in specifying name strings:

- 8-character short IDs — can contain special characters when delimited
- 18-character long IDs — can contain special characters when delimited
- 8-character operating system short IDs — cannot contain special characters

Using the online interface to build groups

The online interface provides a great deal of flexibility and real-time information when building and maintaining groups.

It includes the following benefits:

- Add or merge lists of objects
- Delete objects either singly or by wildcard from the list
- Completely replace a list of objects
- Include related objects
- Verify that the objects in the group have acceptable status
- Revalidate the recoverability of a group
- Add or revoke individual user authorizations
- Specify backup and recovery options at the subsystem and group level

Note
For information about long name support, see “Supporting DB2 long names” on page 111 and “Online display of DB2 long names” on page 87.

Creating a new group

Use this online procedure to build a new group of objects for backup or recovery.

To start the build, select a method for object list generation, optionally edit the list, and then select and list those objects that are candidates for the backup or recovery.

To build a group starting with volume names, see “Building a volume group in batch mode” on page 121.

Before you begin

To perform this procedure, you need the following authorizations:

- EXECUTE authority for RECOVERY MANAGER
- DISPLAY DATABASE authority to list all objects in recover pending status when you attempt to build a group of such objects

To build a new group

Start this procedure at the RECOVERY MANAGER Main Menu.

1. In the RECOVERY MANAGER Main Menu, select Appl. group definition. The Object Selection panel is displayed.

2. In the Object Selection panel, select one of the available object types to use as a starting point to create the new group.

3. The Selection for Object List Generation panel is displayed. This panel varies depending on the object selection method that you chose. Enter a wildcard pattern or the object name, then other information as required. For additional information about the fields on this panel, press F1 for help.

Press Enter to continue processing.
4 When the Object List panel appears showing the objects in the group, you can choose to perform one or more of the following tasks:

- Add objects to the group (see “Adding objects to a group” on page 122)
- Exclude one or more objects from further processing
- Exit or save the group to the repository for later processing (see “Saving groups” on page 122)
- Select a recovery point for the group and continue with backup or recovery preparations (see “To browse or update group recovery options” on page 232 or “Generating recovery JCL in batch” on page 210 for recovery information; “Generating a backup job interactively” on page 163 for backup information)
- Change group options (see “Displaying, updating, and deleting backup options for standard copies” on page 158)
- Update group authorizations (see “Setting group authorizations” on page 128)
- Report the plans and packages impacted by the group (see “Using plan and package impact analysis and reporting” on page 139)

To create a group using dynamic SQL

Start this procedure at the RECOVERY MANAGER Main Menu.

1 On the RECOVERY MANAGER Main Menu, select Appl. group definition and press Enter.

The Object Selection panel appears (Figure 8 on page 114).

Figure 8: Object Selection panel

ARMUS001 == Object Selection ==
Command ===> __________________________________________________________
Choose object selection method, then press Enter.

1. Owner/DB/TS   - Select table spaces and indexes by Owner/DB/TS
2. Table         - Select table spaces and indexes by table pattern
3. Index         - Select indexes by index pattern
4. Volume        - Select table spaces and indexes by volume
5. STOGROUP      - Select table spaces and indexes by STOGROUP pattern
6. Plan          - Select table spaces and indexes by plan pattern
7. Package       - Select table spaces and indexes by package pattern
8. Exception Status - Select table spaces and indexes by exception status
9. SQL           - Select table spaces and indexes by user defined SQL
10. Recovery Groups - Select table spaces for ARMGPS size balancing
11. Indexspace    - Select indexes by indexspace pattern

F1=HELP     F2=SPLIT    F3=Exit     F4=RETURN   F5=RFIND    F6=RCHANGE
F7=Bkwd     F8=Fwd      F9=SWAP    F10=LEFT    F11=RIGHT   F12=PREV

2 Select 9. SQL, and press Enter.
The User Defined SQL Selection for Object List Generation panel appears (Figure 9 on page 115).

Figure 9: User Defined SQL Selection for Object List Generation panel

```
Command ==>
Type selection and press Enter.
Pre-populate the Dynamic SQL panel.  1
1. Do not pre-populate
2. TS DBNAME, NAME
3. TS DBNAME, NAME, PARTITION
4. IX CREATOR, NAME
5. IX IXCREATOR, IXNAME, PARTITION
6. SG CREATOR, NAME
7. TS without a full copy in the last 7 days
8. Objects without regular full copy after reorg or load
```

3 On the User Defined SQL Selection for Object List Generation panel, enter a valid SQL SELECT statement.

Blank lines are excluded. You may enter up to 16,000 characters. Only one SQL statement is accepted. Semicolons are not allowed.

If not specified, the partition number is set to 0.

The IX name length and number of partitions follow the rules of the DB2 Version that you are using.

The SQL statements must be exactly as shown in the examples in the following table and cannot deviate except in the WHERE clause.

Table 7: Examples of SQL statements allowed in dynamic SQL

<table>
<thead>
<tr>
<th>Object type</th>
<th>SQL statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>TS</td>
<td>SELECT 'TS', DBNAME, NAME FROM SYSIBM.SYSTABLESPACE WHERE DBNAME LIKE 'QZU%'</td>
</tr>
<tr>
<td></td>
<td>SELECT 'TS', DBNAME, TSNAME, PARTITION FROM SYSIBM.SYSTABLEPART WHERE DBNAME='QZUDPT22' AND TSNAME='QZUS0122' AND PARTITION IN (4090, 4092, 4094, 4096)</td>
</tr>
<tr>
<td>IX</td>
<td>SELECT 'IX', CREATOR, NAME, CREATOR, NAME FROM SYSIBM.SYSINDEXES WHERE DBNAME = 'R92DB59'</td>
</tr>
<tr>
<td></td>
<td>SELECT 'IX', IXCREATOR, IXNAME, IXCREATOR, IXNAME, PARTITION FROM SYSIBM.SYSINDEXPART WHERE IXCREATOR='QZU' AND PARTITION &gt; 100 AND PARTITION &lt;= 200</td>
</tr>
<tr>
<td>SG</td>
<td>SELECT 'SG', CREATOR, NAME, CREATOR, NAME FROM SYSIBM.SYSTOGROUPWHERE NAME LIKE 'JTR%'</td>
</tr>
</tbody>
</table>

4 Press Enter when you have entered your complete SQL SELECT statement.
RECOVERY MANAGER reads and verifies the statement by performing a SQL PREPARE. After the PREPARE, RECOVERY MANAGER executes the SQL and displays the Object List panel (ARMOB001) showing the resolution of the group definition.

5 Optionally, if you want to define additional SQL SELECT statements in the group definition, select option 5. Add Objects from the Object List panel.

You display the group definition for dynamic SQL by using option 2.3 from the Object List panel to display the GROUP Definition Display panel (Figure 10 on page 116).

**Figure 10: GROUP Definition Display panel**

```
ARMGF001  ================ GROUP Definition Display =========== Row 1 from 1
Command ====> ________________________________________________ Scroll ====> PAGE
Group Name   PUBLIC   Q1545928
Description
Type                      Name 1     Name 2    Name 3 Dsnum     I R By   L X H
Clone
EXCEPT CHECKPND            0        NNN      NNNN
EXCEPT COPYEND             0        NNN      NNNN
EXCEPT LPL                 0        NNN      NNNN
EXCEPT RECPEND             0        NNN      NNNN
EXCEPT STOPERR             0        NNN      NNNN
EXCEPT ERRRANGE            0        NNN      NNNN
EXCEPT ADVREORG             0        NNN      NNNN
EXCEPT ADVREBLD             0        NNN      NNNN
EXCEPT INFOCOPY             0        NNN      NNNN
EXCEPT AUXCHECK             0        NNN      NNNN
EXCEPT AUXWARN              0        NNN      NNNN
EXCEPT GRECP               0        NNN      NNNN
EXCEPT RebPend             0        NNN      NNNN
EXCEPT REORGpnd             0        NNN      NNNN
EXCEPT PRO                 0        NNN      NNNN
EXCEPT RREPL                0        NNN      NNNN
TS NAME  ARMDB01 TS01       *        O Y N N B O
N N
DYN SQL
SELECT 'TS', DBNAME, NAME FROM SYSIBM.SYSTABLESPACE WHERE DBNAME = 'RMDDB48'
******************************************************************************
```

Letters are used to represent related objects in the online Recovery Definition display and ARMBGRP Definition report. Table 8 on page 116 provides the letters used and their definitions. While BMC’s Recovery Management products can resolve all of these letters designations, RECOVERY MANAGER can create only \( Y \) (Yes) and \( N \) (No). Only DASD MANAGER PLUS version 11.1.00 and later can create \( B \) (Bases only) and \( O \) (Only this type), in addition to creating \( Y \) (Yes) and \( N \) (No).

**Table 8: Letters used to represent related objects**

<table>
<thead>
<tr>
<th>Letter designation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Y )</td>
<td>Return all related objects as API currently functions.</td>
</tr>
<tr>
<td>Letter designation</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>N</td>
<td>Do not return related objects as API currently functions</td>
</tr>
<tr>
<td>B</td>
<td>Return only base objects of a relationship and any other objects that do NOT have that relationship.</td>
</tr>
<tr>
<td>O</td>
<td>Return only objects of the type included.</td>
</tr>
</tbody>
</table>

To view the SQL SELECT statement, position the cursor on the **Type** field of the DYN SQL line and press **F4**.

For examples, see “Sample JCL and output” on page 573.

**To create a group by dynamic exception status**

1. On the RECOVERY MANAGER Main Menu, select **Appl. group definition** and press **Enter**.

   The Object Selection panel appears.

2. On the Object Selection panel, select **8. Exception Status** and press **Enter**.

   The Exception Status Specification for Object List Generation panel appears.

   ```plaintext
   ARMEX001 = Exception Status Specification for Object List Generation ========
   Command ===> Make selections where 1=Yes and 2=No , then press Enter.
   __Create Group as Dynamic.... 2  All Exception Statuses ... 2
   __Check Pending. ............ 2  Advisory Reorg Pending .. 2
   __Copy Pending ............... 2  Advisory Rebuild Pending .. 2
   __Informational Copy Pending .. 2  Auxiliary Check Pending.... 2
   __Logical Page List(LPL) .... 2  Auxiliary Warning...... 2
   __Recover Pending .......... 1  Error Range Detected .... 2
   __Group Recover Pending(GRECP) . 2  Stopped Error Range .... 2
   __Rebuild Pending .......... 2  Reorg Pending.... . . ... . 2
   __Persistent Read Only ...... 2  Read or Replication Only.. 2
   Include Related. . . . . . . 2 RI: 2 Cloned: 2 (1=Yes/2=No)
   LOB: 2 XML: 2 (1=Yes/2=No/3=Base/4=Only)
   List objects by part .... 2 (1=Yes/2=No)
   ```

3. In the **Create Group as Dynamic** field, specify 1 (for yes) and press **Enter**.

   Dynamic grouping is enabled.
Using a batch program to build groups

Use the RECOVERY MANAGER batch group creation and maintenance program, ARMBGRP to:

- Create groups of objects defined by volume, table space name, or table name pattern
- Change or define many of the backup and recovery options for a group
- Delete groups
- Copy groups
- Rename groups
- Report on group information

These actions are accomplished using the following parameters:

- GROUP
- VOLUMES
- REPOSITORY
- TABLE
- TABLESPACE
- INDEX
- INDEXSPACE
- EXCEPTION
- PACKAGE
- PLAN
- SQL
- STOGROUP

See “ARMBGRP—Group creation and maintenance” on page 507.

Use the RECOVERY MANAGER batch subsystem group creation program, ARMBGPS, to automatically create groups containing all objects in an entire DB2 subsystem. For more information, see “Full subsystem recovery” on page 347.

Creating volume-based groups

RECOVERY MANAGER allows you to create a static group of objects that reside on a specified volume.

Using the RECOVERY MANAGER user interface, you can generate volume groups in either foreground or background mode, as follows:
■ Foreground Generation

Searching the DB2 and operating system catalogs for the required objects in foreground mode is best used for creating volume groups after a volume failure. This method keeps RECOVERY MANAGER busy until the object list is generated and displayed and, on many systems, may take several minutes to complete. However, the volumes do not have to be available to RECOVERY MANAGER when you use this method.

■ Background Generation

Background generation automatically generates JCL that can be used to create volume groups. The ARMBGRP program supports two different methods of obtaining information for creating a volume group. You can create the group by performing catalog searches or by using the volume’s VSAM volume data set (VVDS).

Catalog search method

The catalog search method uses the DB2 and operating system catalogs to locate the DB2 objects for the desired subsystem or data sharing group. This method does not require the operating system volume to be online and so can be executed after a volume failure has occurred. This method is ideal for an ad hoc recovery after an unexpected volume failure.

VVDS method

The VVDS method uses the VSAM "table of contents" located on the volume and the DB2 catalog to identify objects belonging to the specified DB2 subsystem or data sharing group. This method requires the volume to be online and available at the time the job executes.

--- WARNING ---

Recovering with a group created by the VVDS method will not include objects created or moved to the volume after the group was generated.

Building a volume group interactively

You can build a group of table spaces and indexes interactively by specifying volume names.

You can choose whether to build the group in foreground mode or generate JCL that you can execute later to build the group in background mode. In background mode, you can also choose whether to search the DB2 and operating system catalogs for the required information or use the VVDS.
Before you begin

To perform this procedure, you need the following authorizations:

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- READ access to the VVDS for the volumes for the VVDS method
- TYPE A authority to execute the JCL and have the ARMBGRP module Authorized Program Facility (APF)-authorized (required when you use the background method and replace an existing group of the same name)
- The names of the volumes that are used for recovery

To build a volume group interactively

Start this procedure at the RECOVERY MANAGER Main Menu.

1. In the RECOVERY MANAGER Main Menu, select **Appl. group definition**. The Object Selection panel is displayed.

Enter the following information:

a  Up to eight operating system volume names (you cannot use wildcards in volume names).

Tip
BMC recommends that you specify as many volumes as possible at the same time. The time that is needed to search the operating system and DB2 catalogs is independent of the number of volumes.

b  Select one of the following methods to build a volume group:

- Foreground Execution—builds the group in the foreground by searching the DB2 and operating system catalogs.

- Background Execution, VVDS—builds the group in background mode using the VVDS. To use the VVDS, the volumes must be online and available to RECOVERY MANAGER.

- Background Execution, Catalog Search—builds the group in background mode by searching the DB2 and operating system catalogs. Provide a group name and description, and specify replace if the new group should replace an existing group of the same name.

c  In the Clones Only field, specify whether you want the group to contain clones located on the specified volumes. The default is No, which means that no clones are included in the group. Yes means that only clones are included. This option is displayed only when running on DB2 Version 10 or later and is not valid with compatibility mode.

Building a volume group in batch mode

Use the ARMBGRP program CREATE GROUP VIA VOLUMES command to create a group containing the objects on one or more volumes.

For more information, see “ARMBGRP—Group creation and maintenance” on page 507.
Saving groups

You can save groups to the repository for later retrieval and processing.

Any time you create or update a group, you are automatically given the chance to save. When you save a group, all of the options that you selected for the group are also saved. You can save the current group from the Object List panel by pressing F3 and selecting a save option. You can choose whether to save under the current name (Save) or under a new name (SaveAs). If you save the group under a new name, the name must be in the form creator.name. The first part of the name cannot exceed 8 characters; the second part cannot exceed 18 characters. The creator term can be PUBLIC.

Retrieving saved groups

This topic describes how to retrieve saved groups.

To retrieve a saved group

1 Select Application groups from the RECOVERY MANAGER Main Menu. Enter a group name or wildcard pattern to display one or more groups.

2 Type S by the group you want to select on the Group List panel and press Enter.

Updating groups

You can update the objects and utility options of groups that you have created.

The Group Edit panel allows you to add new objects, remove unnecessary objects, set utility options, set authorizations, and display the group definition.

Tip

For performance reasons, you should use the ARMBGRP batch program to create, update, or generate reports for groups containing more than a few hundred objects.

Adding objects to a group

Use this procedure to add objects to a group after it has been retrieved from the repository.
You can select those table spaces and indexes you want to add to the group. Full subsystem groups created by ARMBGPS cannot be updated using this method.

**Before you begin**

Before you begin

To perform this procedure, you need the following authorizations (if you are not the creator of the group):

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- TYPE O authority if you do not plan to save changes to the group
- TYPE A authority if you plan to save changes to the group
- DISPLAY DATABASE authority to list all objects in recover pending status when you attempt to add such objects to the target group

**To add objects to a group**

1. Display the group by performing the following steps:
   
   a. Select **Application groups** from the RECOVERY MANAGER Main Menu. Enter a group name or wildcard pattern to display one or more groups.

   b. Place **S** or **SA** by the group you want to select on the Group List panel.

2. In the Object List panel for the target group, select **5. Add objects** and then press Enter.

   **Note**

   You can display a CI column on the Object List panel. CI stands for Clone and Instance. This column is populated with N1 or N2 or Y1 or Y2. N indicates not cloned and Y indicates cloned. The number is the instance of the base. The CI column is for information only and does not affect the operation of this panel. You use the SHOWCI command to display this column. The default is to not display the CI column when entering the panel from a group. You enter the SHOWCI command on the Command line to display the CI column. Entering the SHOWCI command again toggles the CI column off.

   ![Figure 11: Object List panel (ARMOB0001)](image-url)
### T/X Name and DSNUM

<table>
<thead>
<tr>
<th>Act Status</th>
<th>T/X Name</th>
<th>DSNUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK</td>
<td>T ACPDBMRN TSMRNN1</td>
<td>0</td>
</tr>
<tr>
<td>OK</td>
<td>X ACP ICMRNN1</td>
<td>0</td>
</tr>
<tr>
<td>OK</td>
<td>X ACP INMRNN1</td>
<td>0</td>
</tr>
<tr>
<td>OK</td>
<td>T ACPDBMRN TSMRNN32</td>
<td>0</td>
</tr>
<tr>
<td>OK</td>
<td>X ACP ICMRNN32K</td>
<td>0</td>
</tr>
<tr>
<td>OK</td>
<td>X ACP INMRNN32K</td>
<td>0</td>
</tr>
<tr>
<td>OK</td>
<td>T ACPDBMRN TSMRNP1</td>
<td>0</td>
</tr>
<tr>
<td>OK</td>
<td>X ACP ICMRNP1</td>
<td>0</td>
</tr>
<tr>
<td>OK</td>
<td>X ACP INMRNP1</td>
<td>0</td>
</tr>
</tbody>
</table>

**Note**

If a table space has not changed, any associated COPY NO indexes are set to the status TS STAT and are not recovered. Indexes are only recovered for changed table spaces.

---

3 In the Object Selection panel, select a method of adding objects, then press **Enter**

to continue.

**Note**

*Volume* and *Exception status* are not valid choices when adding objects to an existing group

For *SQL*, select option 5. *Add objects* to define additional SQL SELECT statements in the group definition.

---

4 When the Selection for Object List Generation panel appears, perform the following substeps:

   a. Type a wildcard pattern or the name of the object to be added.

   b. Select **Replace** (the default).

   c. Press **Enter** to list the specified objects.

   d. Type the action code **S** by the objects to be added. Type **SA** to add all objects in the list.

   e. Press **Enter**.

5 When the List Generation Options panel appears, perform the following substeps:

   a. Choose to include or exclude objects and indexes related by referential integrity.

   b. Press **Enter** to return to the Object List panel for the target group.

   All selected objects are now included in the object list for the target group.

6 Press **F3** to exit or save the group to the repository. Select **Save**, **SaveAs**, **Exit**, or **Cancel**, as required.
Setting utility options

RECOVERY MANAGER enables you to specify backup and recovery utility options for groups.

You can specify these options for a single group using the online interface, or you can set options for multiple groups using the ARMBGRP batch program (see “ARMBGRP—Group creation and maintenance” on page 507).

To set or change utility options interactively

Start this procedure at the Group Edit panel, which you can reach by performing the following steps:

- Select option Application groups from the RECOVERY MANAGER Main Menu. Enter a group name or wildcard pattern to display one or more groups.

- On the Group List panel, type S by the group you want to select or SA to select all groups on the Group List panel.

  **Note**
  Selecting SA or multiple S entries merges the selected groups after the options are set (or even if no options changed). RECOVERY MANAGER goes to the Save panel when you exit.

- Select Option 2, Group Edit from the Object List panel.

  
  ARMGE001 ==================== Group Edit ===============================================
  Command ===> __________________________________________________________________

  Type selection, and then press Enter. You may modify the description.

  | Group name   | RDAXXX     | ALLOBJ01          |
  | Date created | 2012-06-13 | 13.39.59.601010   |
  | Date last updated | 2012-06-13 | 13.39.59.601010   |
  | Last updated by | RDASLJ     |                  |

  _ 1. Utility options - Display or modify backup & recovery options
  2. Authorizations - Display or modify authorizations
  3. Definition display - Display population definition
  4. Batch report - Batch group revalidation report
  5. Impact report - Report plans and packages impacted by the group

  Description. . . . . . GROUP ALL OBJS IN SUBSYS_

1 In the Group Edit panel, select Utility options, and then press Enter.

2 For information about backup utility options, see “Displaying, updating, and deleting backup options for standard copies” on page 158.

For information about recovery utility options, see “Displaying, updating, and deleting recovery options” on page 231.
Utility options can be set at the group and subsystem levels. RECOVERY MANAGER uses the following override rules when necessary:

- If you provide a value at the group level, that value overrides the corresponding subsystem value and is applied to all objects in the target group.

- If you do not provide option values by group, the subsystem values will be used when you are generating backup or recovery JCL.

- If you do not provide option values at the subsystem level, RECOVERY MANAGER default values are used for recoveries and the defaults of the selected backup utility are used for backups.

**Viewing group definitions**

When you save a group, you are actually storing the group definitions in the repository.

When you create or open the group, using dynamic grouping, RECOVERY MANAGER builds the object list that is resolved from the stored definitions.

**Definition types**

When you view definition types, the list that appears shows you how the group is constructed.

The list will contain one or more of the definition types shown in Table 9 on page 127.

RECOVERY MANAGER stores wildcard patterns only for table spaces and indexes—and then only if you typed SA (Select All) on the Object List Generation panel when you started the build; otherwise, individual space and index names are stored. RECOVERY MANAGER saves the selected names for storage group, plan, group, and package and saves patterns for these. RECOVERY MANAGER does not save volume names, but saves only the static list of objects from the volume. Exception groups are also static.

It will show a wildcard pattern if SA (Select All) is used. This way it picks up any new tables that match the pattern for Dynamic Groups.
Table 9: Definition types

<table>
<thead>
<tr>
<th>Definition Type</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARMBGPS</td>
<td>The group was built by the ARMBGPS full subsystem recovery program. The objects resulting from the include and exclude patterns are included in the group, called a ARMBGPS application group, and the group is size balanced. Note: ARMBGPS groups LOB, XML and History objects with their related base table space, not by using the explicit name of the LOB, XML, or History object.</td>
</tr>
<tr>
<td>EXCEPT</td>
<td>Groups include objects in specified exception status.</td>
</tr>
<tr>
<td>EXCL GP</td>
<td>Groups with the name or pattern that is shown were excluded from the group.</td>
</tr>
<tr>
<td>EXCL IS</td>
<td>Index spaces with the name or pattern that is shown were excluded from the group.</td>
</tr>
<tr>
<td>EXCL IX</td>
<td>Indexes with the name or pattern that is shown were excluded from the group.</td>
</tr>
<tr>
<td>EXCL PG</td>
<td>Packages with the name or pattern that is shown were excluded from the group.</td>
</tr>
<tr>
<td>EXCL PL</td>
<td>Plans with the name or pattern that is shown were excluded from the group.</td>
</tr>
<tr>
<td>EXCL SG</td>
<td>Storage groups with the name or pattern that is shown were excluded from the build.</td>
</tr>
<tr>
<td>EXCL TB</td>
<td>Tables with the name or pattern that is shown were excluded from the group.</td>
</tr>
<tr>
<td>EXCL TS</td>
<td>Spaces with the name or pattern that is shown were excluded from the group.</td>
</tr>
<tr>
<td>GROUP</td>
<td>The objects in the named group are included in this group.</td>
</tr>
<tr>
<td>IS NAME</td>
<td>The index space name or pattern that is shown are included in the group.</td>
</tr>
<tr>
<td>IX NAME</td>
<td>The index name or pattern that is shown are included in the group.</td>
</tr>
<tr>
<td>PKG DEP</td>
<td>The objects in the named package are included in this group.</td>
</tr>
<tr>
<td>PLAN DEP</td>
<td>The objects in the named plan are included in this group.</td>
</tr>
<tr>
<td>DYN SQL</td>
<td>The objects in the group are selected by user-defined SQL.</td>
</tr>
<tr>
<td>RELATED</td>
<td>The related items (XML, LOBs, or RI-objects) are included in the group.</td>
</tr>
<tr>
<td>STOGROUP</td>
<td>The objects in the named storage group are included in this group.</td>
</tr>
<tr>
<td>TS NAME</td>
<td>The table space name or pattern that is shown are included in the group.</td>
</tr>
<tr>
<td>TBN DEP</td>
<td>The table name that is shown was specified to include in the group. If you use SA (Select All), a wildcard pattern is shown. RECOVERY MANAGER uses the wildcard pattern to pick up any new tables that match the pattern for dynamic groups.</td>
</tr>
<tr>
<td>VOLUME</td>
<td>The volumes named are included in the group.</td>
</tr>
</tbody>
</table>

To view group definitions

1. Select **Application groups** from the RECOVERY MANAGER Main Menu. Enter a group name or wildcard pattern to display one or more groups.
2 Place S by the group you want to select or SA to select all groups on the Group List panel.

3 Select Group Edit from the Object List panel.

4 Select Definition Display from the Group Edit panel.

**Setting group authorizations**

You can grant or revoke authorization for the current group through the Group Edit panel.

The following types of authorization can be granted to users:

- **TYPE A (ALL)**
  This authorization provides a user with the authority to open and save a group.

- **TYPE O (OPEN)**
  This authorization provides a user with the authority to open a group but does not allow any changes to the group to be saved to the repository. However, with TYPE O authorization, a user can still proceed to generate JCL. A user with SYSADM or system DBADM authority (or with secondary IDs with SYSADM or system DBADM authority) is considered to have both TYPE A and TYPE O authority on the group.

  You can also grant PUBLIC authorization with either TYPE A or O authority. This authorization gives all users the same type of authority for a group.

  For more information, see “Authorizations” on page 89.

**Before you begin**

To perform this procedure, you need the following authorizations (if you are not the creator of the group):

- EXECUTE authority for the RECOVERY MANAGER DB2 plan

- TYPE A (ALL) authority
  You also need the following information that is specific to this task:

- The SQL ID or the user ID that you will add or delete from the existing list of authorized users (this ID can be PUBLIC)

- The type of authority (ALL or OPEN) that you will assign to an addition
  If you do not make an assignment, the type defaults to ALL.
To change group authorizations

Start this procedure at the Group Edit panel, which you can reach by performing the following steps:

- Select Application groups from the RECOVERY MANAGER Main Menu. Enter a group name or wildcard pattern to display one or more groups.
- Place S or SA by the group you want to select on the Group List panel.
- Select Group Edit from the Object List panel.

1 In the Group Edit panel, select Authorizations, and then press Enter. The Group Authorization panel is displayed.

```
ARMGA001 ==================== GROUP Authorization =============================
Command ===> ________________________________________________ Scroll ===> PAGE
Group Name   RDAXXX TEST2
Description
Type user id and type to add to the group. Then press Enter.
Add user id . . ________ Authorization type . . _ (1=All,2=Open)
Type Action to process selected value(s). Then press Enter.
R=Revoke
```

2 In the Group Authorization panel, you can perform the following tasks:

- To add a user, type the new user ID, select the All (the default) or Open authorization type and then press Enter. The ID is added to the list of authorizations.
- To revoke an existing group authorization, type R in the Act column beside the ID that you want to revoke. Then press Enter.
- To change the type of authorization for a user, first revoke the existing authorization, and then add a new one with the required type.
  Continue this process until all your updates are complete.

Renaming groups

You can change the name of a group interactively using the online interface.

You can also change the name of one or more groups at the same time by using the ARMBGRP RENAME GROUP functionality. See “RENAME GROUP syntax and option descriptions” on page 560.
Before you begin

You need the following authorizations (if you are not the creator of the group):

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- TYPE A (ALL) or SYSADM or system DBADM authority on your primary or secondary user ID

To rename a group interactively

Start this procedure at the Group List panel which is displayed after you select Application groups on the Main Menu.

1. In the Group List panel, display a group or groups by entering a group name or pattern.

2. Enter C (Change name) beside the group that you want to change, and then press Enter.

3. In the Change Name Confirmation panel, enter the new name in the New group name field, select Change group name, and then press Enter. To exit without changing the name, select Cancel change and press Enter.

Note
You can modify the description of the group on the Group Edit panel. Display the Group Edit panel by selecting a group (S), then choosing Group edit on the Object List panel. Changing the name of the group does not update the UPDATED_BY column in the GROUPS TABLE.

Validating the objects in a group

The RECOVERY MANAGER for DB2 product performs object validation when you build an object group and prepare to generate backup or recovery JCL.

The type of validation depends largely on the method that you use to build the group and the type of processing that you select.

Note
RECOVERY MANAGER includes only objects with an acceptable status in the generated JCL. For more information on object statuses, see “RECOVERY MANAGER object exception status” on page 839.
The types of validations that are performed by RECOVERY MANAGER on objects in a group are as follows:

- Verification that an object is available to DB2
- Verification that an object is eligible for backup or recovery
- Verification that a suitable image copy (or DFSMS concurrent copy) exists for the object

All object validation is performed in the context of building a group or revalidating recoverability, regardless of whether the task is performed interactively or in batch mode.

### Verifying object availability to DB2

Verifying the availability of an object to DB2 detects such things as non-DB2 objects, misspelled names, and incorrect partition numbers.

When the objects are specified by one or more wildcard patterns, verifying object availability is inherent in the pattern expansion process. Similarly, when a table space, table, index, plan, package, or storage group is named explicitly and the disposition is REPLACE, the name is verified on the Object List Generation panel.

Object availability to DB2 is also verified by the revalidation process whenever the group is revalidated. It is possible that an object which was available when the group was built no longer exists. This situation is detected by the revalidation process.

### Verifying eligibility for backup or recovery

Verifying that an object is eligible for inclusion in the generated backup or recovery JCL involves detecting such objects as catalog and directory spaces that are not owned by the current DB2 subsystem. This verification is first performed when the object list is generated. If you use the group again, eligibility is verified again.

Catalog and directory spaces are marked with a SYS OBJ status and are excluded from the backup or recovery JCL. These spaces are not backed up or recovered as part of the group and must be processed using System resources on the Main Menu.

Work file database objects are marked with a WORKFILE status and are excluded from the backup or recovery JCL. These spaces must be processed using the System resources option on the Main Menu.
Verifying object recoverability

RECOVERY MANAGER verifies the recoverability of each object for the recovery point that you selected. The results of this process are shown in the Status column when the Object List panel is displayed. An OK status indicates that the object is recoverable to the recovery point selected; such objects are included in the generated JCL.

Responding to unsatisfactory object status

An object must have a status of OK for RECOVERY MANAGER to include it in recovery JCL. For backup JCL, objects with certain non-OK statuses are included in addition to those with an OK status.

For information about object statuses, see “RECOVERY MANAGER object exception status” on page 839.

Processing object lists

After you have displayed a list of objects to be included in the group, you can replace it with a new list, merge it with another list, or exclude one or more objects from the list.

---

**Note**

When you start the build by using a table space wildcard pattern or name, you can specify individual partitions or data sets. (This does not apply when you build your group with table names or patterns.) However, when you start with an index pattern, you can specify only individual partitions.

---

You can also request the following during object list generation:

- **Partition expansion**
  A partitioned object can be expanded into its constituent partitions. This technique is useful if you are planning to:
  - Selectively back up or recover partitions
  - Perform a recovery using DB2 RECOVER and the existing image copies were made by partition

- **Index inclusion**
  Any indexes on the table spaces in the list can be included in the list. If you are building a group primarily for recovering to the current time, indexes are usually not an issue. However, if you are building a group for a recovery to a prior point
and indexes were not included, the indexes may not be synchronized with the data when the space is recovered. You have two opportunities to include indexes—during list generation and again at recover time if point-in-time recovery is selected.

**WARNING**

Do not choose this option if you are using `INDEX ALL` when generating recovery JCL (for more information, see “General recovery options” on page 233).

- **RI inclusion**
  Objects that are related by referential integrity can be included in the list during list generation. If you are building a group primarily for recovering to the current time, referential integrity is usually not an issue. However, if you are building a group for a recovery to a prior point and objects related by referential integrity were not included, the entire RI set may not be in a consistent state after a recovery.

- **LOB Inclusion**
  Any objects related by LOB structure to an object in the list can be included in the list during list generation. Because of their unique structure, LOB-related objects must always be recovered to the same point.

- **XML Inclusion**
  Any objects related by XML structure to an object in the list can be included in the list during list generation. Because of their unique structure, XML-related objects must always be recovered to the same point.

- **History (Versioning) Inclusion**
  Any objects related by a history (versioning) relationship to an object in the list can be included in the list during list generation.

---

**Group recovery revalidation and reporting**

Group revalidation allows you to revalidate the recoverability of the group.

You can run a revalidation job on a regular basis to ensure that the object in the group are recoverable.

You can revalidate the recoverability of a RECOVERY MANAGER group in batch mode using the ARMBGPV program (see “ARMBGPV—Group recovery revalidation” on page 489). RECOVERY MANAGER also provides online support for using ARMBGPV.
Be aware of the following information:

- For performance reasons, you should use the batch programs to revalidate or generate JCL for groups containing more than a few hundred objects.

- For information about revalidating groups created by ARMBGPS, see “Full subsystem recovery” on page 347.

### Revalidating and reporting on groups in batch

Use the ARMBGPV program to revalidate the recoverability of a group in batch.

You can optionally request detailed group-specific printed reports, as follows:

- Object recoverability
- Recovery resources that are required by the group
- Tape volumes that are needed for recovery
- Archived data sets that are required for recovery

You can also perform the following tasks:

- Specify whether the reports are for the local site or a recovery site
- Specify the recovery point for analysis

When you use the batch method, you can routinely revalidate several groups concurrently or just one group.

You can generate printed batch group revalidation reports by using the ARMBGPV program in the following ways:

- Use the **R** (report) or the **RA** (report all) option in the Group List panel when you want to generate JCL for the revalidation reports. **R** provides a report for the selected groups only; **RA** provides a report for all groups that are listed on the panel.

- Select **Batch report** on the Group Edit panel to automatically generate the JCL for the revalidation reports.

When selecting to run batch group revalidation, you must supply the data set names in which the output JCL is to be placed and job card information for that JCL. Both items can include symbolic variables. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.
Generating JCL for batch revalidation and reporting

You can automatically generate revalidation JCL for one or more selected groups from either the Group List panel or from the Group Edit panel.

Before you begin

If you are not the group creator of the group, you need the following authorizations:

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- TYPE A (ALL) authority to save changes to the group
- TYPE O (OPEN) authority if you do not intend to save changes
  You also need TYPE O authority for any dependent groups that may be opened for resource, data set recall, or pick list reports

To generate batch revalidation JCL for one or more groups

Start this procedure at the RECOVERY MANAGER Main Menu.

1. In the RECOVERY MANAGER Main Menu, select Application groups. Press Enter to display the Group List panel.

2. In the Group List panel, type a wildcard pattern or the name of a single group in the format creator.name and press Enter. The group or groups that you specified are displayed.

3. To select a group, perform one of the following steps:
   - To report on selected groups only, type R beside one or more individual groups. Type RA in any space in the Act column to report on all groups. Then press Enter.
   - Type S in the Act column beside a group. The Object List panel is displayed. Select Group Edit and press Enter. Then, in the Group Edit panel, select Batch report and press Enter. This method produces reports for the current group only.

```plaintext
ARMVD001 == Batch Group Report Options
Command ===> _________________________________________________________________
Type information below. Then press Enter.
Report type . . . . . . . 1 1. Revalidation  2. Impact analysis
  Site type . . . . . . . . . 1 1. Local  2. Recovery
  JCL type . . . . . . . . . 1 1. Local  2. DR
  Recovery Resources . . . 1 1. Yes  2. No
  Recoverability . . . . . 1 1. Yes  2. No
  Tape pick list . . . . . 1 1. Yes  2. No
```
In the Batch Group Report Options panel, select **Revalidation** at the **Report type** prompt. Accept the defaults or provide data as needed on the panel and then press **Enter**.

For option descriptions, see “Batch group report option descriptions” on page 136.

When the second Batch Group Report Options panel is displayed, specify the recovery parameters that you want the revalidation to include and press **Enter**.

When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:

- The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.
- The job statement must contain a symbolic variable (‘&’#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

Save the JCL data set or submit the job as required.

### Batch group report option descriptions

This section describes the fields and the options available on the Batch Group Report Options panel for generating batch group revalidation reports.
Table 10: Batch group report option fields

<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| Archive copy   | None                     | Generates a report for a copy of the archive log or the offsite log, as follows: With Site type = Recovery,  
|                |                          |  ■ Archive 1—report on ARCHLOG1.  
|                |                          |  ■ Archive 2—report on ARCHLOG2.  
|                |                          |  ■ Offsite Log—report on offsite logs created by PACLOG or ARMBARC (the default). With Site type = Local, report on ARCHLOG1.                                                                                     |
| Backout        | RECOVERY MANAGER only: No | Validates the availability of resources required for a backout recovery.  
|                | Recovery Management: Auto| This option is available when using NGT Recover or DB2 RECOVER (DSNUTILB) as the recover utility.  
|                |                          | **Backout Auto** is only available when using the Recovery Management solution password.  
<p>|                |                          | <strong>Note:</strong> You must select Quiesce or Specific Logpoint as the recovery point to specify BACKOUT YES or BACKOUT AUTO. If you accept the default recovery point of Current, RECOVERY MANAGER overrides BACKOUT AUTO and converts the selection to BACKOUT NO. |
| Full copy only | No                       | Specifies whether to use only full copies for recovery. If you select this option, RECOVERY MANAGER ignores all incremental copies and recovers to the full copy that you specify in the Copy or quiesce field. |
| Copy or quiesce| 0                        | Identifies an image copy or quiesce by specifying a number from 0 to 99, where 0 represents the most recent image copy or quiesce.                                                                                |
| Data Set Recall List | Yes           | Produces a report of the archived data sets that are needed for recovery.                                                                                                                                  |
| JCL type       | Local                   | Specifies whether the JCL is for local recovery or disaster recovery. To invoke mirroring, the JCL type must be DR.                                                                                          |
| Mirror System  | No                       | Verifies and reports the mirroring status of the subsystem based on the subsystem mirroring strategy. This option is only available if you are using the RECOVERY MANAGER solution. |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| Report clones only    | No                       | Generates the CLONES ONLY option in the validation JCL syntax  
The CLONES ONLY option causes clone objects to be included in the validation report and non-cloned objects and clone bases to be excluded. This option is displayed only when running on DB2 Version 10 or later and is not valid with compatibility mode. |
| Recover to            | Current                  | Revalidates the group for recovery to one of the following points: Current, Image Copy, Quiesce, Specific Logpoint, Restart LRSN/RBA, Timestamp, or Logmark  
Timestamp and Logmark recovery are available with the RECOVERY MANAGER Solution only. |
| Recoverability        | Yes                      | Validates the recoverability of each object in the group to the selected recovery point  
This option enables you to verify recoverability by identifying recoverability exceptions; that is, data sets for which no copies were found. |
| Recovery Resources    | Yes                      | Reports the resources that are needed for the recovery of a group  
This option includes image copy data sets, log data sets, and R+/CHANGE ACCUM files. Use this report as a checklist of data sets that must be taken offsite in preparation for disaster recovery.  
If the current subsystem is a member of a data sharing group, the resources report is for the group. Otherwise, it is for the current subsystem. |
| Report type           | Revalidation             | Specifies either a revalidation or an impact analysis report. (This field is displayed only when reached from the group list panel.) |
| Resolve Inflights     | No                       | Resolves all inflight units of work for a recovery to a specific log point. This option is available only with the Recovery Management for DB2 solution. |
| Site type             | Local                    | Specifies whether the selected reports are for the local site or a recovery site |
| Tape Pick List        | Yes                      | Generates a report of the tape volumes that are needed for recovery  
This report enables tape operators to get the tapes ready prior to a recovery. |
| To Logpoint           | None                     | Specifies the RBA or LRSN for revalidation |
| To Timestamp          | None                     | Specifies the timestamp for revalidation. This option is not available for non-data sharing systems. |
| To Logmark            | None                     | Specifies the log mark name for revalidation |
Using plan and package impact analysis and reporting

RECOVERY MANAGER allows you to determine what plans or packages (and therefore what applications) are potentially unavailable during a planned backup and recovery.

You can obtain a list either online or from a batch report using the ARMBGIM program (see “ARMBGIM—Impact analysis” on page 465).

RECOVERY MANAGER determines the impact information by analyzing the SYSIBM.SYSPLANDEP and SYSIBM.SYSPACKDEP tables for plans and packages respectively.

You can request a plan or package impact report for the current group, which may be either a new (unsaved) group that you have just created or one that you have retrieved from the repository.

You make your request in the Group Edit panel or you can generate JCL to run a report by using the R or RA options on the Group List panel.

**Note**

If you want to perform an analysis on multiple groups, you can select all groups of interest in the Group List panel and proceed to display a combined list of objects in the Object List panel. Then make the request in the Group Edit panel.

Generating impact reports in foreground or background mode

You can generate a list of plans and packages that would be impacted should you recover the current group or groups.

You can generate this report online (foreground mode) or create JCL to submit offline (background mode).
Before you begin

You need the following authorizations to perform this analysis:

- EXECUTE authority on the RECOVERY MANAGER DB2 plan
- READ authority (TYPE O) on the group

**Note**

DB2 must be active when you execute the JCL.

To perform impact analysis for the current group

Start this procedure at the RECOVERY MANAGER Main Menu.

1. In the RECOVERY MANAGER Main Menu, select **Application groups**.

2. In the Group List panel, type a wildcard pattern or the name of a single group in the format `.creator.name` and press **Enter**. The group or groups that you specified are displayed.

3. To generate an impact analysis report, perform either step A or step B.

   a. To report on selected groups, type **R** beside one or more individual groups. To report on all groups, type **RA** in any space in the **Act** column. Press **Enter**. The Batch Group Report Options panel is displayed.

   ```
   ARMV0001 == Batch Group Report Options                               =========
   Command ===> _________________________________________________________________
   Type information below. Then press Enter.
   Report type . . . . . . . 1 1. Revalidation  2. Impact analysis
   Site type . . . . . . . 1 1. Local  2. Recovery
   JCL type . . . . . . . 1 1. Local  2. DR
   Recovery Resources . . . 1 1. Yes  2. No
   Recoverability. . . . . . 1 1. Yes  2. No
   Tape pick list . . . . . 1 1. Yes  2. No
   Data set recall list . . . . . . 1 1. Yes  2. No
   Archive copy . . . . . . _ 1. Archive 1  2. Archive 2  3. Offsite Log
   ```

   In the Batch Group Report Options panel, select **Impact analysis** and specify any of the other options as necessary. For descriptions of the options, see “Batch group report option descriptions” on page 136. Press **Enter**.

   A second Batch Group Report options panel appears.
Impact analysis options:
Analyze plans . . . . . 1 1. Yes 2. No
Analyze packages . . . . 1 1. Yes 2. No
Report routing . . . . . 1 1. SYSOUT 2. Data set
If data set, enter complete data definition below:
   //ARMPRINT DD___________________________________________________________
   ________________________________________________________________________
WARNING - must include member name if the data set is partitioned.

In the second Batch Group Report Options panel, perform the following substeps:

- **At Analyze plans**, select **Yes** to obtain a plan impact report.
- **At Analyze packages**, select **Yes** to obtain a package impact report.
- **At Report routing**, if you do not want to use the default SYSOUT, you must also supply a complete data definition for the report output. If the data set that you specify is a partitioned data set, you must provide the member name.

The JCL Specification panel is displayed (go to Step 5 on page 142).

b This method allows you to generate the selected reports in foreground mode. It can only be used for the current group.

Type S in the Act column beside a group. The Object List panel is displayed. Select **Group Edit** and press **Enter**. The Group Edit panel is displayed.

On the Group Edit panel, select option **5. Impact report** and press **Enter**. The Plan/Package Impact Report panel is displayed.
For background execution report routing... 1. SYSOUT 2. Data set
if data set, enter complete data definition below:
.. //ARMPRINT DD___________________________________________________________
.. _______________________________________________________________________
WARNING - must include member name if the data set is partitioned.

4 In the Plan and Package Impact Report Specification Options panel, perform the following substeps:

a. At Analyze plans, select Yes to obtain a plan impact report.

b. At Analyze packages, select Yes to obtain a package impact report.

c. Select Foreground or Background mode for this report.

--- Note ---
If you select Background and do not want to use the default SYSOUT, you must also supply a complete data definition for the report output. If the data set that you specify is a partitioned data set, you must provide the member name.

Depending on your selections, one of the following panels appears:

- If you selected Foreground, a list of the packages and/or plans involved in the recovery of the current group is displayed in the Plan and Package Impact Report panel. Press F3 to exit the panel.

<table>
<thead>
<tr>
<th>Plan</th>
<th>Collection ID</th>
<th>Package</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACPSUP39</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACPSUP40</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACPSUP47</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACPSUP48</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACPSUP49</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- If you selected Background, the JCL Specification panel appears. Continue with Step 5 on page 142.

5 When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:

- The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.

- The job statement must contain a symbolic variable (&#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.
6 Save the JCL data set or submit the job as required.

Using RECOVERY MANAGER object sets for backup and recovery setup

RECOVERY MANAGER creates object sets and stores them in the BMC Common DB2 repository on each DB2 subsystem. An object set is a collection of DB2 table spaces, indexes, or both, as determined by the object set definitions.

Note

- The terms group and object set are synonymous.
- RECOVERY MANAGER does not support delimited object set names. Delimited names are those enclosed by double quotation marks and can contain special characters (for example, "creator"."name").

When accessing an object set, RECOVERY MANAGER returns objects based on the object set definition:

- If the definitions are explicit (no wildcards), RECOVERY MANAGER returns only the explicitly specified objects.
- If the definition contains wildcards, RECOVERY MANAGER returns all objects matching the wildcard pattern or patterns. For example, if a definition is DB40.*, RECOVERY MANAGER returns all table spaces in database DB40. Also, if you subsequently add a table space to that database, RECOVERY MANAGER dynamically recognizes the new table space.

For information about OBJECTSET, see “OBJECTSET” on page 452.

Programs for managing object sets

With RECOVERY MANAGER, you can use the ARMBGRP, ARMBGPS, and ARMBSET programs to create and manage object sets:

- ARMBGRP is a batch interface for creating and managing object sets. You can use a variety of definitions to create the object sets, such as by:
  - Table space
  - Table
  - Index
Index space
Plan
Package
Stogroup
User-specified SQL
Object status

You can use a mixture of definitions to create your preferred collection of DB2 objects. For more information, see “ARMBGRP—Group creation and maintenance” on page 507.

■ ARMBGPS creates a collection of object sets (up to 99) as specified by MAXGROUPS nn syntax. These object sets contain only table spaces. ARMBGPS automatically distributes the objects into multiple size-balanced object sets, keeping related object types (such as LOB, XML, temporal, and archive enabled) together in the same object set. RECOVERY MANAGER can then treat these object sets as independent units for backup and recovery. Because these object sets are balanced by size, the backup and recovery jobs that use them should run for approximately the same length of time.

Note
You should use the same object sets for recovery that you use for backup. If using tape, attempting to recover with different object sets in multiple jobs can cause contention between jobs that need the same tape. To maintain parallelism, you can use multiple recover jobs per object set. However, each job will need to read the DB2 log. So, optimally, you should use only one or two concurrent jobs per LPAR for data sharing, and a maximum of four for non-data sharing. Contention might occur on the log (even if it is on disk) if too many jobs are running in parallel and attempt to read the log simultaneously. This situation can degrade performance.

For more information, see “ARMBGPS—Subsystem group split” on page 473.

■ ARMBSET is an object set utility. You can use ARMBSET to perform the following actions for objects in a specified object set:

■ CHECK
■ REPAIR
■ QUIESCE
■ RUNSTATS
■ START
■ STOP

For more information about ARMBSET, see “ARMBSET—OBJECTSET processing” on page 659.
JCL examples for backing up and recovering object sets

The following JCL examples show how to back up and recover with object sets using the RECOVERY MANAGER for DB2, BMC Next Generation Technology Copy for DB2 for z/OS, and BMC Next Generation Technology Recover for DB2 for z/OS products.

You create this JCL only once; you do not need to regenerate it each time. However, the JCL requires that the copies be created by object set when tape is involved to avoid contention for volumes across the recovery jobs.

RECOVERY MANAGER SAMPLIB member ARMBSET contains the following jobs:

- **JOB00011**: (Figure 12 on page 145) creates a big object set by table space without indexes, and then uses ARMBGPS to create four size-balanced object sets from it.

- **JOB00021**: (Figure 13 on page 146) through **JOB00024**: (Figure 16 on page 149) make image copies for each object set.

- **JOB00031**: (Figure 17 on page 150) through **JOB00034**: (Figure 20 on page 153) recover each object set.

- **JOB00041**: (Figure 21 on page 154) creates a big object set by table space with indexes, and then uses ARMBSET to start spaces in RW.

**Figure 12: JOB00011—CREATE INITIAL OBJECTSET/CREATE BALANCED OBJECTSETS**

```plaintext
//JOB00011 JOB PACP,NAME,CLASS=A,MSGCLASS=X,
          NOTIFY=RDAWXP,REGION=0M
/*
  ****************************************************************************
  //* CREATE INITIAL OBJECTSET
  ****************************************************************************
  //CREATEG1  EXEC PGM=ARMBGRP,
            PARM='DIZ,ARMOPTS=ARM$OPTS'
  //STEPLIB      DD DISP=SHR,DSN=RMD.INST1110.BMCLINK
                DD DISP=SHR,DSN=SYS3.DIZ.DSNEXIT
                DD DISP=SHR,DSN=CSGI.DB2V10M.DSNLOAD
  //ARMMSGS    DD DISP=SHR,DSN=RMD.INST1110.BMCCNTL(ARMMSGS)
  //ARMERROR  DD SYSOUT=*  
  //ARMPRINT    DD SYSOUT=* 
  //ARMIN           DD *
CREATE GROUP RDAWXP.BIGGROUP
  DESCRIPTION "ALL OBJECTS"
  REPLACE YES
  VIA TABLESPACE RDAWXP.*
  INCLUDEIX NO
  */
/*
  ****************************************************************************
  //* CREATE BALANCED OBJECTSETS
  ****************************************************************************
  //SPLITGP EXEC PGM=ARMBGPS,
            PARM='DIZ,ARMOPTS=ARM$OPTS',
            REGION=4M
  //STEPLIB      DD DISP=SHR,DSN=RMD.INST1110.BMCLINK
                DD DISP=SHR,DSN=SYS3.DIZ.DSNEXIT
                DD DISP=SHR,DSN=CSGI.DB2V10M.DSNLOAD
```

Chapter 3 Creating and working with groups 145
Figure 13: JOB00021—COPY EACH OBJECTSET

//JOB00021 JOB PACP.NAME,CLASS=A,MSGCLASS=X, NOTIFY=RDAWXP,REGION=0M
//**************COPY OBJECTSET
//NOTES:
//** - Indexes larger than IXSIZE will be copied.
//** - Table spaces larger than OUTSIZE will be copied to BIGDDN
//** - If a copy fails for any reason, and you wish to re-run the job
//** from the beginning, change the restart parm from NEW/RESTART to
//** NEW, and Copy+ will automatically TERM the old utilid.
//**
//**************COPY EACH RECOVERY OBJECTSET
//**
//BMCCOPY1 EXEC PGM=ACPMAIN,REGION=0M,
//         PARM='DIZ,,NEW/RESTART,MSGLEVEL(1)'
//STEPLIB DD DSN=RMD.INST1110.BMCLINK,DISP=SHR
//         DD DSN=SYS3.DIZ.DSNEXIT,DISP=SHR
//         DD DSN=CSGI.DB2V10M.DSNLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*  
//SYSDUMP DD SYSOUT=*   
//ACPERROR DD SYSOUT=*  
//SYSIN DD *       
//OPTIONS
MAXTASKS 8
OUTSIZE 1500 M
IXSIZE 1500 M
XBMID XBMDB2

-- BIG COPIES ON THEIR OWN TAPE
OUTPUT BIGTAPE UNIT CARTVTS STACK NO
   DSNAME RDAWXP2.&DB.&TS.&LPART.&UNIQ

-- SMALLER COPIES ON STACKED TAPE
OUTPUT TAPE UNIT CARTVTS STACK YES
   DSNAME RDAWXP2.&DB.&TS.&LPART.&UNIQ

-- CABINET COPIES
OUTPUT CABINET UNIT CARTVTS STACK CABINET
   DSNAME RDAWXP2.MYGRP01.CAB.&UNIQ

-- SNAP COPIES (REQUIRES DSNUM DATASET)
OUTPUT SNAP UNIT SYSDA
   STORCLAS DEVSMS
   DSNAME RDAWXP2.&DB.&TS.D&DSNUM.&UNIQ
   DSSNAP AUTO

Using RECOVERY MANAGER object sets for backup and recovery setup
COPY TABLESPACE OBJECTSET RDAWXP.MYGRP00
COPY TABLESPACE OBJECTSET RDAWXP.MYGRP01
EXCLUDE RDAWXP02.MDSNP01, RDAWXP02.MDSNP02
INDEXES YES
COPYDDN(TAPE)
BIGDDN(BIGTAPE)
SHRLEVEL CHANGE
GROUP NO

/*

// Figure 14: JOB00022—COPY EACH OBJECTSET

//JOB00022 JOB PACP.NAME,CLASS=A,MSGCLASS=X,
//         NOTIFY=RDAWXP,REGION=0M
//*
//*****************************************************************************
//* COPY OBJECTSET
//* -----------------------------------------------------------------
//* NOTES:
//* - Indexes larger than IXSIZE will be copied.
//* - Table spaces larger than OUTSIZE will be copied to BIGDDN
//* - If a copy fails for any reason, and you wish to re-run the job
//*   from the beginning, change the restart parm from NEW/RESTART to
//*   NEW, and Copy+ will automatically TERM the old utilid.
//******************************************************************************
//*****************************************************************************
//*****************************************************************************
//* COPY EACH RECOVERY OBJECTSET
//*****************************************************************************
/BMCCOPY1 EXEC PGM=ACPMAIN,REGION=0M,
   PARM='DIZ,,NEW/RESTART,MSGLEVEL(1)'
/STEPLIB DD DSN=RMD.INST1110.BMCLINK,DISP=SHR
   DD DSN=SYS3.DIZ.DSNEXIT,DISP=SHR
   DD DSN=CSGI.DB2V10M.DSNLOAD,DISP=SHR
/SYSPRINT DD SYSOUT=* 
/SYSUDUMP DD SYSOUT=* 
/ACPERROR DD SYSOUT=* 
/SYSIN DD *
OPTIONS
   MAXTASKS 8
   OUTSIZE 1500 M
   IXSIZE 1500 M
   XBMID XBMDB2

-- BIG COPIES ON THEIR OWN TAPE
OUTPUT BIGTAPE UNIT CARTVTS STACK NO
   DSNAME RDAWXP2.&DB.&TS.P&LPART.&UNIQ

-- SMALLER COPIES ON STACKED TAPE
OUTPUT TAPE UNIT CARTVTS STACK YES
   DSNAME RDAWXP2.&DB.&TS.P&LPART.&UNIQ

-- CABINET COPIES
OUTPUT CABINET UNIT CARTVTS STACK CABINET
   DSNAME RDAWXP2.MYGRP01.CAB.&UNIQ

-- SNAP COPIES (REQUIRES DSNUM DATASET)
OUTPUT SNAP UNIT SYSDA
   STORCLAS DEVSMS
   DSNAME RDAWXP2.&DB.&TS.D&DSNUM.&UNIQ
   DSSNAP AUTO
COPY TABLESPACE OBJECTSET RDAWXP.MYGRP02
   EXCLUDE RDAWXP02.MDSNP01, RDAWXP02.MDSNP02
   INDEXES YES
   COPYDDN(TAPE)
   BIGDDN(BIGTAPE)
   SHRLEVEL CHANGE
   GROUP NO
/*
*/
Figure 15: JOB00023—COPY EACH OBJECTSET

//JOB0023 JOB PACP,NAME,CLASS=A,MSGCLASS=X,
   NOTIFY=RDAWXP,REGION=0M
/*
*/
/*COPY OBJECTSET
*-----------------------------------------------------*/
/* NOTES:
*/
/* - Indexes larger than IXSIZE will be copied.
/* - Table spaces larger than OUTSIZE will be copied to BIGDDN
/* - If a copy fails for any reason, and you wish to re-run the job
/* - from the beginning, change the restart parm from NEW/RESTART to
/* - NEW, and Copy+ will automatically TERM the old utilid.
*/
/*-----------------------------------------------------*/
/*COPY EACH RECOVERY OBJECTSET
*-----------------------------------------------------*/
//BMCCOPY1 EXEC PGM=ACPMAIN,REGION=0M,
   PARM='DIZ,,NEW/RESTART,MSGLEVEL(1)'
//STEPLIB DD DSN=RMD.INST1110.BMCLINK,DISP=SHR
   DD DSN=SYS3.DIZ.DSNEXIT,DISP=SHR
   DD DSN=CSGI.DB2V10M.DSNLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*  
//SYSDUMP DD SYSOUT=*  
//ACPERROR DD SYSOUT=*  
//SYSIN DD *
OPTIONS  
   MAXTASKS 8  
   OUTSIZE 1500 M  
   IXSIZE 1500 M  
   XBMID XBMDB2  
   -- BIG COPIES ON THEIR OWN TAPE
   OUTPUT BIGTAPE UNIT CARTVTS STACK NO
      DSNAME RDAWXP2.&DB.&TS.P&LPART.&UNIQ
   -- SMALLER COPIES ON STACKED TAPE
   OUTPUT TAPE UNIT CARTVTS STACK YES
      DSNAME RDAWXP2.&DB.&TS.P&LPART.&UNIQ
   -- CABINET COPIES
   OUTPUT CABINET UNIT CARTVTS STACK CABINET
      DSNAME RDAWXP2.MYGRP01.CAB.&UNIQ
   -- SNAP COPIES (REQUIRES DSNUM DATASET)
   OUTPUT SNAP UNIT SYSDA
      STORCLAS DEVSMS
      DSNAME RDAWXP2.&DB.&TS.D&DNUM.&UNIQ
      DSSNAP AUTO
COPY TABLESPACE OBJECTSET RDAWXP.MYGRP03
   EXCLUDE RDAWXP02.MDSNP01, RDAWXP02.MDSNP02
Figure 16: JOB00024—COPY EACH OBJECTSET

```
//JOB0024 JOB PACP,NAME,CCLASS=A,MSGCLASS=X,
   NOTIFY=RDAXXK,REGION=OM
//*
ịnh/ispensal
* COPY OBJECTSET
/*/  
/** NOTES: */
/* */  
/* - Indexes larger than IXSIZE will be copied. */
/* - Table spaces larger than OUTSIZE will be copied to BIGDDN */
/* - If a copy fails for any reason, and you wish to re-run the job */
/* from the beginning, change the restart parm from NEW/RESTART to */
/* NEW, and Copy+ will automatically TERM the old utilid. */
/* */  
/*ATICLAR: */
/* */  
/*COPY EACH RECOVERY OBJECTSET */
/*ATICLAR: */
//BMCCOPY1 EXEC PGM=ACPMAIN,REGION=OM,
   PARM='DIZ,,NEW/RESTART,MSGLEVEL(1)'  
//STEPLIB DD DSN=RMD.INST1110.BMCLINK,DISP=SHR  
   DD DSN=SYS3.DIZ.DSNEXIT,DISP=SHR  
   DD DSN=CSG1.DB2VI0M.DSNLOAD,DISP=SHR
//SYSPRINT DD SYSOUT=*  
//SYSUDUMP DD SYSOUT=*  
//SYSPRINT DD SYSOUT=*  
//ACPERRO $ DD *  
//SYSPRINT DD SYSOUT=*  
//ACPERRO $ DD *  
//SYSPRINT DD SYSOUT=*  
//ACPERRO $ DD *
//OPTIONS
   MAXTASKS 8
   OUTSIZE 1500 M
   IXSIZE 1500 M
   XBMID XBMDB2

-- BIG COPIES ON THEIR OWN TAPE
OUTPUT BIGTAPE UNIT CARTVTS STACK NO
   DSNAME RDAWXP2.&DB.&TS.P&LPART.&UNIQ

-- SMALLER COPIES ON STACKED TAPE
OUTPUT TAPE UNIT CARTVTS STACK YES
   DSNAME RDAWXP2.&DB.&TS.P&LPART.&UNIQ

-- CABINET COPIES
OUTPUT CABINET UNIT CARTVTS STACK CABINET
   DSNAME RDAWXP2.MYGRP01.CAB.&UNIQ

-- SNAP COPIES (REQUIRES DNSNUM DATASET)
OUTPUT SNAP UNIT SYSDA
   STORCLAS DEVSMS
   DSNAME RDAWXP2.&DB.&TS.D&DNSUM.&UNIQ
   DSSNAP AUTO
COPY TABLESPACE OBJECTSET RDAWXP.MYGRP04
   EXCLUDE RDAWXP2.MDNSNP01, RDAWXP2.MDNSNP02
   INDEXES YES
   COPYDDN(TAPE)
```
**Figure 17: JOB00031—RECOVER EACH OBJECTSET**

```
//JOB00031 JOB PACP,NAME,CLASS=A,MSGCLASS=X,
NOTIFY=RDAXXP,REGION=OM
//*
/***************************************************************************/
//* RECOVER OBJECTSET
//** INSTRUCTIONS:
//** 1. Set the RECOVERYPOINT timestamp in each job,
//** or remove the RECOVERYPOINT syntax for recovery to current.
//** 2. For best results, submit the jobs on separate LPARs.
//** For recovery at a DR site:
//** - Set BACKOUT NO
//** - Set REDEFINE NOSCRATCH to eliminate mount messages for
//** primary site volumes that do not exist at the recovery site.
//** ADDITIONAL NOTES:
//** - RESOURCE SELECTION is used to spread log reads across the
//** active and archive logs.
//** - With BACKOUT AUTO, BACKOUT is automatically set to NO when you
//** are recovering to current.
//** - With INDEXLOG AUTO, if an image copy of an index is available,
//** an index will be recovered from the copy. If no copy is available
//** the index will be rebuilt.
//***************************************************************************/
//ARM0001 EXEC PGM=AFRMAIN,
// PARM=(DIZ,,
// 'NEW/RESTART','MSGLEVEL(1)..'RDB2STAT(YES)',
// AFROPTS(AFROPTS))
//STEPLIB DD DISP=SHR,DSN=RMD.INST1110.BMCLINK
// DD DISP=SHR,DSN=SYS3.DIZ.DSNEXIT
// DD DISP=SHR,DSN=CSGI.DB2V10M.DSNLOAD
//SYSUDUMP DD SYSOUT=*
//SYSPICK DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSERR DD SYSOUT=*
//SYSIN DD *
OPTIONS
INDEXLOG AUTO
BACKOUT AUTO
ON ERROR CONTINUE 10
RECOVERYPOINT
TIMESTAMP 2014-07-17-11.00.00.000000
URIDDDN(URID0)
RESOURCE SELECTION
LOGS(ACT1,ACT2,ARC1,ARC2)
OUTPUT URIDO UNIT SYSDA
DSNAME &UID.&DATE.T&TIME.URIDS.&UNIQ
RECOVER TABLESPACE OBJECTSET RDAXXP.MYGRP00 -- DELTA GROUP (NEW OBJ)
INDEXES YES
EXCLUDE (RDAXXP02.MDSNP01, RDAXXP02.MDSNP02)
REDEFINE NO
RECOVER TABLESPACE OBJECTSET RDAXXP.MYGRP01
INDEXES YES
```
Figure 18: JOB00032—RECOVER EACH OBJECTSET

```sql
//JOB00032 JOB PACP,NAME,CLASS=A,MSGCLASS=X,
// NOTIFY=RDAXP,REGION=0M
//*
//*************************************************************
//* RECOVER OBJECTSET
//*************************************************************
//* INSTRUCTIONS:
//* *
//* 1. Set the RECOVERYPOINT timestamp in each job,
//* or remove the RECOVERYPOINT syntax for recovery to current.
//* 2. For best results, submit the jobs on separate LPARs.
//* *
//* For recovery at a DR site:
//* - Set BACKOUT NO
//* - Set REDEFINE NOSCRATCH to eliminate mount messages for
//*   primary site volumes that do not exist at the recovery site.
//* *
//* ADDITIONAL NOTES:
//* *
//* - RESOURCE SELECTION is used to spread log reads across the
//*   active and archive logs.
//* - With BACKOUT AUTO, BACKOUT is automatically set to NO when you
//*   are recovering to current.
//* - With INDEXLOG AUTO, if an image copy of an index is available,
//*   an index will be recovered from the copy. If no copy is available
//*   the index will be rebuilt.
//* *
//*************************************************************
//*************************************************************
//*************************************************************
//* RECOVER OBJECTSET
//*************************************************************
//ARM0001 EXEC PGM=AFRMAIN,
// PARM=(DIZ,,
// 'NEW/RESTART','MSGLEVEL(1)',,'RDB2STAT(YES)',
// AFROPTS(AFROPTS))
//STELIB DD DISP=SHR,DSN=RMD.INST1110.BMCLINK
// DD DISP=SHR,DSN=SYS3.DIZ.DSNEXIT
// DD DISP=SHR,DSN=CSGI.DB2V10M.DSNLOAD
//SYSUDUMP DD SYSOUT=*  
//SYSPICK DD SYSOUT=*  
//SYSOUT DD SYSOUT=*  
//SYSSERR DD SYSOUT=*  
//SYSIN DD *
// OPTIONS
// INDEXLOG AUTO
// BACKOUT AUTO
// ON ERROR CONTINUE 10
// RECOVERYPOINT
// TIMESTAMP 2014-07-17-11.00.00.000000
// URIDDN(URID0)
// RESOURCE SELECTION
// LOGS(ACT2,ARC1,ARC2,ACT1)
// OUTPUT URIDO UNIT SYSDA
// DSNNAME &UID.D&DATETTIME.URIDS.&UNIQ
// RECOVER TABLESPACE OBJECTSET RDAXP.MYGRP02
// INDEXES YES
// EXCLUDE (RDAXP02.MDSNP01, RDAXP02.MDSNP02)
```
REDEFINE NO

Figure 19: JOB00033—RECOVER EACH OBJECTSET

Figure 19: JOB00033—RECOVER EACH OBJECTSET

```c
/*
   //
   // Figure 19: JOB00033—RECOVER EACH OBJECTSET
   //*/
   // JOB00033 JOB PACP,NAME,CLASS=A,MSGCLASS=X,
   // NOTIFY=RDAWXP,REGION=0M
   //
   //*****************************************************************************
   // * RECOVER OBJECTSET
   // * INSTRUCTIONS:
   // * 1. Set the RECOVERYPOINT timestamp in each job,
   // *    or remove the RECOVERYPOINT syntax for recovery to current.
   // * 2. For best results, submit the jobs on separate LPARs.
   // * For recovery at a DR site:
   // * - Set BACKOUT NO
   // * - Set REDEFINE NOSCRATCH to eliminate mount messages for
   // *    primary site volumes that do not exist at the recovery site.
   // *
   //*****************************************************************************
   // * ADDITIONAL NOTES:
   // * - RESOURCE SELECTION is used to spread log reads across the
   // *    active and archive logs.
   // * - With BACKOUT AUTO, BACKOUT is automatically set to NO when you
   // *    are recovering to current.
   // * - With INDEXLOG AUTO, if an image copy of an index is available,
   // *    an index will be recovered from the copy. If no copy is available
   // *    the index will be rebuilt.
   // *
   //*****************************************************************************
   //*****************************************************************************
   // * RECOVER OBJECTSET
   //*****************************************************************************
   // ARM0001 EXEC PGM=AFRMAIN,
   // PARM=(DIZ,,
   // 'NEW/RESTART','MSGLEVEL(1)',,'RDB2STAT(YES),
   // AFROPTS(AFR$OPTS))
   // STEPLIB   DD DISP=SHR,DSN=RMD.INST1110.BMCLINK
   //           DD DISP=SHR,DSN=SYS3.DIZ.DSNEXIT
   //           DD DISP=SHR,DSN=CSGI.DB2V10M.DSNLOAD
   // SYSUDUMP  DD SYSOUT=* 
   // SYSPICK   DD SYSOUT=* 
   // SYSOUT    DD SYSOUT=* 
   // SYSERR    DD SYSOUT=* 
   // SYSIN     DD * 
   // OPTIONS
   // INDEXLOG AUTO
   // BACKOUT AUTO
   // ON ERROR CONTINUE 10
   // RECOVERYPOINT
   // TIMESTAMP 2014-07-17-11.00.00.000000
   // URIDDNN(URID0)
   // RESOURCE SELECTION
   // LOGS(ARC1,ARC2,ACT1,ACT2)
   // OUTPUT URID0 UNIT SYSDA
   // DSNNAME &UID.D&DATE.T&TIME.URID0.&UNIQ
   // RECOVER TABLESPACE OBJECTSET RDAWXP.MYGRP03
   // INDEXES YES
   // EXCLUDE (RDAXWP02.MDSNP01, RDAWXP02.MDSNP02)
   // REDEFINE NO
```
Figure 20: JOB00034—RECOVER EACH OBJECTSET

/*

Figure 20: JOB00034—RECOVER EACH OBJECTSET

//JOB00034 JOB PACP,NAME=CLASS=A,MSGCLASS=X,
//     NOTIFY=RDAXXP,REGION=OM
//*
//*****************************************************
//* RECOVER OBJECTSET
//*****************************************************
//* INSTRUCTIONS:
//*
//* 1. Set the RECOVERYPOINT timestamp in each job,
//*    or remove the RECOVERYPOINT syntax for recovery to current.
//* 2. For best results, submit the jobs on separate LPARs.
//*
//* For recovery at a DR site:
//** - Set BACKOUT NO
//** - Set REDEFINE NOSCRATCH to eliminate mount messages for
//**     primary site volumes that do not exist at the recovery site.
//**
//*****************************************************
//* ADDITIONAL NOTES:
//*
//** - RESOURCE SELECTION is used to spread log reads across the
//**   active and archive logs.
//** - With BACKOUT AUTO, BACKOUT is automatically set to NO when you
//**   are recovering to current.
//** - With INDEXLOG AUTO, if an image copy of an index is available,
//**   an index will be recovered from the copy. If no copy is available
//**     the index will be rebuilt.
//**
//*****************************************************
//* RECOVER OBJECTSET
//*****************************************************
//ARM0001 EXEC PGM=AFRMAIN,
// PARM=(DIZ,,
// 'NEW/RESTART','MSGLEVEL(1)',,'RDB2STAT(YES),
// AFR$OPTS)
//STEPLIB   DD DISP=SHR,DSN=RMD.INST1110.BMCLINK
//                   DD DISP=SHR,DSN=SYS3.DIZ.DSNEXIT
//                   DD DISP=SHR,DSN=CSGI.DB2V10M.DSNLOAD
//SYSUDUMP  DD SYSOUT=*  
//SYSPICK   DD SYSOUT=*  
//SYSOUT    DD SYSOUT=*  
//SYSERR    DD SYSOUT=*  
//SYSIN     DD *
//OPTIONS
//    INDEXLOG AUTO
//    BACKOUT AUTO
//    ON ERROR CONTINUE 10
//    RECOVERYPOINT
//    TIMESTAMP 2014-07-17-11.00.00.000000
//    URIDDDN(URID0)
//    RESOURCE SELECTION
//    LOGS(ARC2,ACT1,ACT2,ARC1)
//    OUTPUT URID UNIT SYSDA
//    DSNNAME &UID.&DATE.T&TIME.URIDS.&UNIQ
//    RECOVER TABLESPACE OBJECTSET RDAXXP.MYGRP04
//    INDEXES YES
//    EXCLUDE (RDAXXP02.MDSNP01, RDAXXP02.MDSNP02)
//    REDEFINE NO

Chapter 3  Creating and working with groups  153
Figure 21: JOB00041—CREATE AN OBJECTSET THAT INCLUDES TS & IX/ARMSET TO START SPACES

//JOB00041 JOB PACP,NAME=CLASS=A,MSGCLASS=X,
//NOTIFY=RDAXXP,REGION=OM
/*

******************************************************************************
// CREATE AN OBJECTSET THAT INCLUDES OUR TS & IX
******************************************************************************
CREATEGP EXEC PGM=ARMBGRP,
PARM='DIZ,ARMOPTS=ARM$OPTS'
//STEPLIB DD DISP=SHR,DSN=RMD.INST1110.BMCLINK
//                  DD DISP=SHR,DSN=SYS3.DIZ.DSNEXIT
//                  DD DISP=SHR,DSN=CSGI.DB2V10M.DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=RMD.INST1110.BMCCNTL(ARMMSGS)
/*RMTRACE DD SYSOUT=*
/*ARMERROR DD SYSOUT=*
/*ARMPRINT DD SYSOUT=*
/*ARMIN DD *
-- CREATE AN OBJECTSET THAT INCLUDES INDEXES (FOR CONVENIENCE)
CREATE GROUP RDAWXP.BIGGROUP
DESCRIPTION "ALL OBJECTS"
REPLACE YES
VIA TABLESPACE RDAWXP.*
INCLUDEIX YES
;
/*
******************************************************************************
// RECOVERY MANAGER ARMSET TO START SPACES IF NECESSARY
******************************************************************************
ARM00001 EXEC PGM=ARMSET,
PARM='DIZ,ARMOPTS=ARM$OPTS'
//STEPLIB DD DISP=SHR,DSN=RMD.INST1110.BMCLINK
//                  DD DISP=SHR,DSN=SYS3.DIZ.DSNEXIT
//                  DD DISP=SHR,DSN=CSGI.DB2V10M.DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=RMD.INST1110.BMCCNTL(ARMMSGS)
/*ARMPRINT DD SYSOUT=*
/*ARMERROR DD SYSOUT=*
/*RMTRACE DD SYSOUT=*
/*ARMIN DD *

SET CURRENT SQLID = RDAWXP

START RW
OBJECTSET RDAWXP.BIGGROUP
;
/*
******************************************************************************
// ARMSET READS GROUPS (OBJECTSET) FROM THE BMC COMMON DB2
// REPOSITORY AND PROCESSES ALL OBJECTS IN THE OBJECTSET
// ACCORDING THE COMMAND.
******************************************************************************
/*
/*
/* >>-SET CURRENT SQLID = <sqlid>--------------------------------------------->
/*
/*
/* |-------UNSET-----|--------<object def>---------|--------<sqlid>------|--------
/* |--------NONE-----|--------<object def>---------|--------<sqlid>------|--------
/* |--------CHECK----|--------<object def>---------|--------<sqlid>------|--------
/* |--------REPAIR---|--------<object def>---------|--------<sqlid>------|--------
/* /* >>-CHECK_PEND_ACTION--
/* |--------LEVELID------|--------<object def>---------|--------<sqlid>------|--------
/* |--------<objpatt def>------|--------<sqlid>------|--------
/* /* >>-QUIESCE------------------------------------------<object def>------
/*
Using RECOVERY MANAGER object sets for backup and recovery backup and recovery setup

---

Chapter 3  Creating and working with groups    155

---

```
/*
**  >>-RUNSTATS----------------------------------<objset def>----->
**  >>-RESET_GRECP_LPL------------------------------------>>
**  >>-START--------|---RW----------|------------<objset def>---->>
**                  |---RO----------|
**                  |---UT----------|
**                  |---FORCE------|
**  >>-STOP--------------------------------------<objset def>----->
**  >>-WAIT---------|  1 to 86400   |  ( number of seconds )
**  <options>
**  >>-RECOVERTYPE--|---TOCURRENT---|-------->>
**                  |---TOCOPY------|
**                  |---TOQUIESCE---|
**                  |---TOCOMMONPT--|
**                  |---TOLOGPOINT--|
**                  |---TOTIMESTAMP-|
**                  |---TOLOGMARK---|
**  >>-BACKOUT------|---YES---------|-------->>
**                  |---NO----------|
**                  |---AUTO--------|
**  >>-INDEX ------|---REBUILD ----|-------->>
**                  |---RECOVER-----|
**  >>-CLONE---------------------------------->
**  >>-REPAIRTS-----|---YES---------|-------->>
**                  |---NO----------|
**  >>-WRITE--------|---YES---------|-------->>
**                  |---NO----------|
**  >>-REPORT-------|---YES---------|-------->>
**                  |---NO----------|
**  >>-UPDATE-------|---ALL---------|-------->>
**                  |---ACCESSPATH-|
**                  |---SPACE------|
**                  |---NONE-------|
**  >>-SHRLEVEL-----|---REFERENCE---|-------->>
**                  |---CHANGE-----|
**  <objset def>
**  >>-OBJECTSET----ObjectSetName------------>>
**  <objpatt def>
**  >>-OBJECT_PATTERN----dbname.tsname------>>
**       ... wildcards allowed
```
This chapter describes how to back up groups.

About backup options

RECOVERY MANAGER enables you to set backup options at the subsystem and group levels.

BMC recommends that you establish backup option values for the subsystem before you specify backup option values for a group. If you choose not to set any backup options, RECOVERY MANAGER uses default options, most of which are the default values of the specified backup utility. For more information, see “RECOVERY MANAGER backup and recovery options” on page 94.

**Tip**

For performance reasons, you should use the ARMBGRP batch program to set options for groups containing more than a few hundred objects. For more information about ARMBGRP, see “ARMBGRP—Group creation and maintenance” on page 507.

RECOVERY MANAGER supports the backup utilities in Table 11 on page 157.

<table>
<thead>
<tr>
<th>Backup utility</th>
<th>Type of copies available</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC Next Generation Technology Copy for DB2 for z/OS</td>
<td>Full or incremental image copies of both table spaces and indexes—NGT Copy can automatically decide which based on a threshold. (NGT Copy and NGT Recover versions 8.1 or later are required for incremental index copies) Instant Snapshots (with SUF or XBM) Encrypted copies (BMC Recovery Management for DB2 solution version 7.3 or later required)</td>
</tr>
<tr>
<td>DSNUTILB COPY</td>
<td>Full or incremental image copies DFSMS copies</td>
</tr>
</tbody>
</table>
### Backup utility

<table>
<thead>
<tr>
<th>Backup utility</th>
<th>Type of copies available</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC Next Generation Technology Recover for DB2 for z/OS (OUTCOPY)</td>
<td>Copies to a prior point in time</td>
</tr>
<tr>
<td>Online Consistent Copy</td>
<td>Online consistent copies (BMC Recovery Management for DB2 solution required)</td>
</tr>
</tbody>
</table>

**Note**

RECOVERY MANAGER will not display an option on a panel if your version of the selected utility does not support it. RECOVERY MANAGER checks the xxxVRSN load module from the STEPLIB. If the xxxVRSN module is not found for a product, RECOVERY MANAGER does not produce informational message. If the xxxVRSN is found, RECOVERY MANAGER produces an informational message if the version is not supported.

---

**Displaying, updating, and deleting backup options for standard copies**

To set backup options, you need the following authorizations (if you are not the creator of the group):

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- TYPE A authority if you intend to save the group
- TYPE O authority if you do not intend to save the group

**To set backup options at the group level**

1. Display a group (see “Retrieving saved groups” on page 122).
2. In the Object List Panel, select 2. Group Edit.
4. In the Utility Options Specification panel, select Update as the Action, then Backup as the Utility Type.
5. Browse or update the options in Table 12 on page 159. From any option panel, press **Enter** to save your changes or press **F3** to cancel any changes without saving.
Table 12: Options to browse or update

<table>
<thead>
<tr>
<th>Options</th>
<th>More information</th>
</tr>
</thead>
<tbody>
<tr>
<td>General backup options</td>
<td>“General backup options” on page 167</td>
</tr>
<tr>
<td>NGT Copy options</td>
<td>“NGT Copy-specific options” on page 169</td>
</tr>
<tr>
<td>DB2 Copy (DSNUTILB) options</td>
<td>“DB2 COPY (DSNUTILB) options” on page 180</td>
</tr>
<tr>
<td>NGT Recover (OUTCOPY) options</td>
<td>“NGT Recover (OUTCOPY) options” on page 183</td>
</tr>
<tr>
<td>Output data set options</td>
<td>“Output copy data set options” on page 186</td>
</tr>
<tr>
<td>FULLDDN output data set options</td>
<td>“Output copy data set options” on page 186</td>
</tr>
<tr>
<td>BIGDDN output data set options</td>
<td>“Output copy data set options” on page 186</td>
</tr>
</tbody>
</table>

**Note**

The options that are displayed reflect the options in effect for the group, including options set at the group level, as well as options that defaulted from the system or RECOVERY MANAGER default levels. To update an entry, type over the existing field.

To delete group backup options

1. Display a group (see “Retrieving saved groups” on page 122).

2. In the Object List panel, select **Group Edit**, and then press **Enter**.

3. Select **Utility options** and then press **Enter**.

4. In the Utility Options Specification panel, select **Delete** and then press **Enter**.

After you confirm the deletion, the appropriate backup option values from other levels will be in effect for this group. For more information, see “RECOVERY MANAGER backup and recovery options” on page 94.

### Setting backup options for Instant Snapshots

A hardware-based Instant Snapshot can be made with RECOVERY MANAGER by using NGT Copy in conjunction with XBM or SUF.

RECOVERY MANAGER can then use these copies to recover DB2 spaces in conjunction with NGT Copy.
For more information about Instant Snapshots, see “About Instant Snapshots” on page 193 or see the EXTENDED BUFFER MANAGER and SNAPSHOT UPGRADE FEATURE User Guide.

**To set options for Instant Snapshots**

1. Display a group (see “Retrieving saved groups” on page 122).
2. In the Object List panel, select **Group Edit**.
3. Select **Utility options**.
4. In the Utility Options Specification panel, select **Update** and **2. Backup**.
5. Select **General backup options**, then specify the following options:
   a. Select NGT Copy as the **Copy utility**.
   b. Specify the **Output types** you want to make and press **Enter**.

   The Update Backup Utility Options Specification panel is displayed again.

   **Note**

   Instant Snapshots make only the LP copy. However, you can optionally make additional standard copies from Instant Snapshot copies for the LB, RP, and RB output types. NGT Copy selects the primary Instant Snapshot copy to use as the source for COPY IMAGECOPY based on the values you specify in the NGT Copy options panel.

   For more information about the **General backup options**, see “General backup options” on page 167.

6. Select **NGT Copy specific options**, then specify the following options:
   a. On the first NGT Copy Options panel, specify the following options:
      - **Full Copy Yes**
      - **DSSNAP Yes**
      Press **Enter**. The second NGT Copy Options panel is displayed.
   b. On the second NGT Copy Options panel, specify the following options:
      - Select a **Shrlevel** type.
      For a description of each type, see “SHRLEVEL and Instant Snapshots” on page 197.
Setting backup options for online consistent copies

Online consistent copies require the BMC Recovery Management for DB2 solution and you must be running RECOVERY MANAGER with the solution password.
For more information about online consistent copies, see the *Recovery Management for DB2 User Guide*.

**To make online consistent copies**

1. Display a group (see “Retrieving saved groups” on page 122).

2. In the Object List panel, select **Group Edit**.

3. Select **Utility options**.

4. In the Utility Options Specification panel, select **Update** and **2. Backup**.

5. On the General Backup Options panel, specify **1. NGT Copy** as the Copy Utility, and optionally specify additional Output types.

   **Note**

   Online Consistent Copy can make only LP copies, but you can use Copy Imagecopy to create LB, RP, and RB copies.

6. On the NGT Copy Specific Options, specify Shrlevel **7. Change Consistent**.

7. If you specified additional output types on the General Backup Options panel, go to the NGT Copy Specific Options panel, press **Enter** to display the second panel, and enter the output types that you want to create using Copy Imagecopy.

---

**Setting backup options for cabinet copies**

The cabinet copy feature of the BMC Recovery Management for DB2 solution enables you to copy all the spaces in a group into a single data set called a cabinet file.

The cabinet file is allocated and deallocated only once, regardless of the number of objects that are copied to or recovered from the cabinet file, which can greatly improve performance. Cabinet copies require the BMC Recovery Management for DB2 solution and you must be running RECOVERY MANAGER with the solution password. For more information about cabinet copies, see the *Recovery Management for DB2 User Guide*.

**To make cabinet copies**

1. Display a group (see “Retrieving saved groups” on page 122).

2. In the Object List panel, select **Group Edit**.
3 Select Utility options.

4 In the Utility Options Specification panel, select Update and 2. Backup.

5 On the Output Data Set Options panel, specify Stack: Cabinet. You can also set this option for BIGDDN and FULLDDN output copies.

Note
Cabinet copies are incompatible with compressed indexes.

Generating a backup job interactively

RECOVERY MANAGER allows you to generate a backup job interactively for the current group.

You can submit the JCL for execution or save it for later scheduling.

Before you begin

To perform this procedure, you need the following authorizations (if you are not the creator of the group):

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- TYPE A authority if you intend to save changes to the group
- TYPE O authority if you do not intend to save changes to the group
- Authority to update the output data set for the JCL

To generate a backup job

You start this procedure at the Object List panel that appears after you have created or retrieved a group.

1 Select Gen Backup JCL foreground to generate JCL for all objects that have an acceptable status.

Note
Objects that do not have an acceptable status are not included in the backup JCL. For more information, see “RECOVERY MANAGER object exception status” on page 839.

2 When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:
- The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.

- The job statement must contain a symbolic variable (&#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

3. Save the JCL data set or submit the job as required.

Generating a backup job in batch

RECOVERY MANAGER allows you to generate backup JCL for one or more groups by using the ARMBGEN batch program.

All of the options available in the ARMBGRP UPDATE command are also available in ARMBGEN. The updated values are set for the duration of the ARMBGEN job and are not saved in the RECOVERY MANAGER repository. For more information, see “Copy and recover utility options” on page 847.

Generating a batch ARMBGEN job interactively

RECOVERY MANAGER provides online support for creating ARMBGEN batch jobs.

You specify batch JCL generation from the Group List panel or Object List panel, and then proceed to specify the recovery point and job information.

Before you begin

The following authorizations are required:

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- TYPE O (Open) access for the target group or groups

DB2 must be active when you execute the generated JCL.
To generate a backup job using ARMBGEN

Start this procedure at either the Group List panel or the Object List panel. If you start at the Group List panel, start at Step 1 on page 165. If you start at the Object List panel, start at Step 2 on page 165.

Note
To generate JCL for multiple groups, make your selections in the Group List panel. To generate JCL for a single group, make your selections in the Object List panel. You can also generate JCL for a single group from the Group List panel.

1 In the Group List panel, display the list of groups you want to back up. Then follow these substeps:
   a In the Act (action) column, type B (generate JCL for selected groups) or BA (generate JCL for all groups) to specify one or more groups for processing.
   b Press Enter. When the Batch Group JCL Backup Generation Options panel appears, continue with Step 3 on page 165.

2 In the object list panel, select the Gen backup JCL background processing option and then press Enter.

The Batch Group JCL Generation Options panel appears.

Note
If this is a new group or one that contains unsaved changes, the Group Save Confirmation panel appears. Follow the instructions that are displayed until the Batch Group Generation panel is displayed.

3 In the Batch Group JCL Generation panel, enter the SYSUT2 DD statement for batch JCL output (including //).

The optional SYSUT4 DD statement may also be available if you are using mirroring as part of your recovery strategy. It is only available when all of the following criteria are true:

- You are using the Recovery Management for DB2 solution
- The recovery point is TORESTARTRBA
- JCLTYPE is DR
- Mirroring is selected for the group

4 When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:
The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.

The job statement must contain a symbolic variable (&#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

5 Save the JCL data set or submit the job as required.

Restarting a failed backup job

If a job fails during execution, you can restart it at the failed job step.

To restart a failed backup job

1 Add a RESTART=stepname option to the RECOVERY MANAGER job statement, where stepname is the name of the job step that failed during the prior execution.

2 Continue as follows:

   a If the failed job step is a DSNUTILB step,

      ■ Issue a DB2 DISPLAY UTILITY(*) command to determine whether DB2 considers it necessary to restart the utility.

      ■ If necessary, add RESTART as the third parameter on the job step EXEC statement. For example, PGM=DSNUTILB, PARM='ssid,,RESTART',

   b If the failed job step is a BMC utility, no modification to the restart parameter is necessary. RECOVERY MANAGER uses NEW/RESTART for NGT Copy, and NEW/RESTART(PHASE) for NGT Recover (OUTCOPY).

No other modifications are necessary to restart NGT Copy. However, restarting NGT Recover (OUTCOPY) and DSNUTILB may require modifications to DD statements in those job steps.

Note

RECOVERY MANAGER uses the default utility ID for each DB2 utility and each BMC utility.

Refer to the appropriate reference manual for more information about restarting an IBM or BMC utility.
Backup option descriptions

You can set the following backup options using the RECOVERY MANAGER online interface.

For information about setting the options in the JCL, see “Copy and recover utility options” on page 847.

General backup options

You can set options on the General Backup Options panel that apply to all supported backup utilities.

Those options are listed alphabetically in the following table along with the corresponding RECOVERY MANAGER defaults.

Table 13: General backup options

<table>
<thead>
<tr>
<th>Option</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copy All Indexes</td>
<td>NO</td>
<td>Copies all indexes for the table spaces that are included in the group using the NGT Copy INDEXES(YES) syntax. This option is ignored if the index is ineligible for backup or the required version of NGT Copy is not available. Note: RECOVERY MANAGER does not generate the JCL for INDEXES(YES) if you specify the table spaces in your group by DSNUM.</td>
</tr>
</tbody>
</table>
| Copy Index Spaces    | NO                         | Yes—back up eligible index spaces. No—do not back up index spaces. Auto—back up eligible indexes as large or larger than the size specified by the Index size threshold option. (COPY PLUS version 6.3.00 or later is required to use AUTO.) Backs up eligible index spaces, as follows: Index spaces are eligible for backup if made using one of the following utilities:  
  - NGT Copy with FULL YES  
  - NGT Recover - OUTCOPY  
  - DB2 COPY with FULL YES when the space is COPY YES |
<table>
<thead>
<tr>
<th>Option</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| Copy utility        | RECOVERY MANAGER only: DSNUTILB (DB2 COPY) Recovery Management: NGT Copy | Specifies the backup utility to be used at the subsystem or group level, as follows:  
- NGT Copy  
- NGT Recover - OUTCOPY  
- DB2 Copy (DSNUTILB)  
- Online Consistent Copy (available with the BMC Recovery Management for DB2 solution only) |
| Index size threshold| None                     | Specifies the size threshold at which indexes are backed up rather than rebuilt. This option is used in conjunction with Copy Index Space Auto. Enter the size as follows:  
- M—Megabytes (the default). Valid range is 0-4194303.  
- K—Kilobytes. Valid range is 0-4294967295.  
- G—Gigabytes. Valid range is 0-4095. |
| Output types:       | YES NO NO NO             | Specifies the types of image copies to be made for the spaces in the group, as follows:  
- Local site primary (LP)  
- Local site backup (LB)  
- Recovery site primary (RP)  
- Recovery site backup (RB)  
You cannot make a backup copy unless you make a corresponding primary copy.  
**Note:** To use COPY IMAGECOPY to make output copies, specify the copies on this panel and provide the appropriate response at the Use COPY IMAGECOPY prompt on the NGT Copy Options panel |
| Quiesce After       | NO                       | Establishes a quiesce point for each space in the group immediately after the copy process completes.                                                                                                      |
| Quiesce Before      | NO                       | Establishes a quiesce point for each table space in the group before the copy process starts.                                                                                                              |
| Quiesce Group       | NO                       | Requests a common quiesce point for all table spaces within the current group.  
**Note:** This option is ignored if both Quiesce Before and Quiesce After are set to No.                                                                 |
| QUIESCE WRITE       | YES                      | Instructs DB2 to finish writing any pending transactions for the target spaces before applying the quiesce.  
**Note:** This option is ignored if both Quiesce Before and Quiesce After are set to No. QUIESCE WRITE NO is ignored for objects having the attribute NOT LOGGED. |
<table>
<thead>
<tr>
<th>Option</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region Size</td>
<td>4 MB</td>
<td>Specifies the amount of virtual storage used by the copy utility. The default value is 4 MB. The valid range is -1 through 2047 MB. A value of -1 specifies that RECOVERY MANAGER will not generate region size at the step level. Note: For best performance, BMC recommends a region size of 0 MB, in which case the amount of virtual storage needed to run the job is automatically made available when the recover utility runs. Some data centers do not allow a region size of 0 MB, so 4 MB usually ensures adequate storage.</td>
</tr>
<tr>
<td>Scope</td>
<td>None</td>
<td>Indicates the scope of the copy operation for the specified objects. This field is only displayed when working with DB2 Version 10 or later.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ALL - indicates that all specified objects should be copied. This is the default.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PENDING - indicates that only objects in DB2 status of COPY-pending (COPY) or information COPY-pending (ICOPY) will be copied.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note: PENDING is only valid when you use DSNUTILB as the backup utility.</td>
</tr>
</tbody>
</table>

**NGT Copy-specific options**

You can set backup options that are specific to NGT Copy on the NGT Copy Options panel for the named subsystem or group.

**OBJECTSET support**

RECOVERY MANAGER supports the OBJECTSET option of NGT Copy when you specify NGT Copy or Online Consistent Copy as the backup utility.

This option enables NGT Copy to use the RECOVERY MANAGER repository to identify the objects in the group, which means that you do not need to regenerate the backup JCL when objects in the group change.
OBJECTSET is a replacement for RMGROUP, which was available in RECOVERY MANAGER version 9.1.00.

RECOVERY MANAGER generates the TABLESPACE OBJECTSET option (formerly RMGROUPPTS) and the INDEX OBJECTSET option (formerly RMGROUPPIX) in the backup JCL. The jobs use the following values:

- The RESETMOD option defaults to the NGT Copy configuration option default value.
- NGT Copy uses the GROUP YES option to ensure that all objects in the group are recovered to the same point.
- If one of the repository tables (BMCSYNC or BMCUTIL) is included in the group, the entire group is copied with SHRLEVEL CHANGE.

**NGT Copy backup options**

The options are listed alphabetically in the following table along with the defaults which, in this instance, are almost all NGT Copy defaults.

*Note* If configuration option value is indicated as the default value, RECOVERY MANAGER does not generate the keyword in the JCL. This enables the NGT Copy configuration option value to be used.

For more detailed information about the NGT Copy utility options that you can use with RECOVERY MANAGER, see the *BMC Next Generation Technology Copy for DB2 for z/OS Reference Manual*.

**Table 14: NGT Copy backup options**

<table>
<thead>
<tr>
<th>Option/NGT Copy keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACPGDG data set</td>
<td>Null</td>
<td>Specifies the name of a data set that is used to define a GDG base when no data set is named on the output copy data set options panel. The data set must contain the control cards that are necessary to perform an IDCAMS DEFINE as well as the symbolic variable &amp;BASE, which NGT Copy replaces with the GDG base name.</td>
</tr>
<tr>
<td>Option/NGT Copy keyword</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| After Init Phase        | CONTINUE      | Causes NGT Copy to pause or continue after completing XBM registration of SHRLEVEL CONCURRENT copies. The job halts after all INIT processing for a group of table spaces completes. At that time, all of the affected table spaces are under control of XBM. For more information, see the *BMC Next Generation Technology Copy for DB2 for z/OS Reference Manual*.  
**Note:** Pause is valid only when you specify Shrlevel Concurrent Required and Group Yes. |
| Auto read percent:      | Configuration option value | specifies the percentage of changed pages that must occur to allow escalation from random I/O to full table space scan.  
**Note:** This option is valid only when you specify Readtype and either Full No or Full Auto. |
| Checkerror              | Configuration option value | Specifies an integral number from 1 through 254 to use as a condition code when encountering page checking errors. A code greater than 4 causes job termination at the point of error. |
| Checktslevel            | Configuration option value | Specifies the level of table space checking to identify damaged pages during copying, as follows:  
0—standard minimal checking  
1— intrapage integrity checks for all pages  
2— both intrapage and interpage checks for all pages |
| Cumulative              | YES           | Merges the requested incremental RESETMOD NO copy with the most recent prior incremental RESETMOD NO copy  
**Note:** This option is valid only when you specify Full No or Full Auto. |
| Day of the week         | Null          | Specifies a day of the week on which a full copy will be made. Valid values are as follows:  
1—Monday  
2—Tuesday  
3—Wednesday  
4—Thursday  
5—Friday  
6—Saturday  
7—Sunday  
**Note:** This option overrides all other Full Auto options and is valid only when you specify Full Auto. |
<table>
<thead>
<tr>
<th>Option/NGT Copy keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
</table>
| DSSNAP                  | NO            | Specifies making a hardware-based Instant Snapshot copy of DB2 data, as follows:  
Yes—make an Instant Snapshot copy  
Auto—made a hardware copy if possible, but fall back to a standard copy if the hardware copy fails (for example if XBM, SUF, or the required hardware is not in place)  
No—make a standard copy  
**Note:** The BMC NGT Copy and XBM or SUF products are required to use this option. DSSNAP is incompatible with the ENCIPHER option. DSSNAP is incompatible with groups containing compressed indexes. DSSNAP requires RESETMOD NO. |
| Empty                   | YES           | Avoids making a copy when no pages changed since the last incremental copy was made  
**Note:** This option is valid only when you specify **Full No** or **Full Auto**. |
<table>
<thead>
<tr>
<th>Option/NGT Copy keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full copy</td>
<td>YES</td>
<td>Specifies whether to make full or incremental image copies</td>
</tr>
<tr>
<td>Keyword: FULL</td>
<td></td>
<td>■ Yes—full image copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ No—incremental copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Auto—automatically escalates to a full copy when it encounters one of the following conditions:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— An entry in SYSCOPY prohibits an incremental copy.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— The target object or partition is in copy pending status.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— The target is a special case catalog or directory table space.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— A specified number of incremental copies is reached.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— A specified percentage of changed pages is reached.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— A specified day of the week occurs.</td>
</tr>
</tbody>
</table>

**Note:**

■ Special case table spaces are certain spaces in DSNDB01 and DSNB06. For more information, see the BMC Next Generation Technology Copy for DB2 for z/OS Reference Manual.

■ RECOVERY MANAGER supports making incremental index copies if you have NGT Copy and NGT Recover. For more information, see the BMC Next Generation Technology Copy for DB2 for z/OS Reference Manual.
<table>
<thead>
<tr>
<th>Option/NGT Copy keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full percent</td>
<td></td>
<td>Use the Full percent and Incremental percent options together to determine whether no copy, an incremental copy, or a full copy is made.</td>
</tr>
<tr>
<td>Incremental percent</td>
<td></td>
<td>Enter the incremental and full percentage values in either decimal or integer format. Valid integer values range from 0 to 100. Valid decimal values are 0.0 to 99.9. A decimal value can only be specified to the tenth’s place (1/10 of a percent). To create an incremental copy when changed pages are between two limits, provide the lower limit at Incremental percent and the upper limit at Full percent. No copy is made if changed pages are less than or equal to Incremental percent. To create a full copy when changed pages are equal to or more than a specific limit, provide that limit at Full percent. <strong>Note:</strong> This option is valid only when you specify Full Auto.</td>
</tr>
<tr>
<td>Keyword: FULLPCT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Incremental percent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Keyword: INCRPCT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Configuration option value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Configuration option value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Group</td>
<td>NO</td>
<td>Specifies that all spaces in the RECOVERY MANAGER group share a common consistent point.</td>
</tr>
<tr>
<td>Keyword: GROUP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Keep previous</td>
<td>YES</td>
<td>Retains the entry for the most recent prior (merged) incremental copy in the SYSCOPY table.</td>
</tr>
<tr>
<td>Keyword: KEEP</td>
<td></td>
<td><strong>Note:</strong> This option is valid only when you specify Full No or Full Auto.</td>
</tr>
<tr>
<td>Maximum incrementals</td>
<td></td>
<td>Escalates to a full image copy when the number of incremental copies that are registered in SYSCOPY since the last full copy reaches a number that you specify (1 through 100). <strong>Note:</strong> This option is valid only when you specify Full Auto.</td>
</tr>
<tr>
<td>Keyword: MAXINCRS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Configuration option value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Configuration option value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximum tasks</td>
<td></td>
<td>Specifies the maximum number of tape tasks and the maximum number of tasks that are used by NGT Copy when making image copies. <strong>Note:</strong> You must specify Group Yes to be able to use MAXTASKS for multitasking.</td>
</tr>
<tr>
<td>Keyword: MAXTASKS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>configuration option value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>configuration option value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Minimum pages</td>
<td></td>
<td>Specifies the minimum number of pages that must exist in a space or partition before an incremental copy can be made. Valid values are 1 through 999999. If the number of pages is less than this value, a full copy is made. <strong>Note:</strong> This option is valid only when you specify Full Auto.</td>
</tr>
<tr>
<td>Keyword: MINPAGES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>configuration option value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>configuration option value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Option/NGT Copy keyword</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>NACTIVE</td>
<td>NO</td>
<td>Specifies whether to update the NACTIVE statistic in SYSIBM.SYSTABLESPACE. RECOVERY MANAGER uses the NACTIVE statistic in SYSIBM.SYSTABLESPACE (which shows the number of active pages in the table space) for sizing and group split optimization. <strong>Note:</strong> This option is valid only when you specify Full Yes or Full Auto.</td>
</tr>
<tr>
<td>Number of read/write buffers</td>
<td>configuration option value</td>
<td>Specifies the number of read/write buffers to use. Valid values are integers from 2 through 16.</td>
</tr>
</tbody>
</table>
| ON DUPLICATEDS          | Configuration option value | Specifies what action to take if NGT Copy encounters a copy data set that is already registered in SYSCOPY or BMCXCOPY, as follows:  
  - ERROR—terminate processing if the data set is already registered  
  - DELETE—continue processing when a data set is already registered. When NGT Copy registers the new copy, it deletes the row containing the duplicate data set in SYSCOPY or BMCXCOPY. Only rows with the same DSNAME, TSNAME (IXNAME), or DSNUM as the new copy are deleted. |
| ON ERROR BADSTATUS      | configuration option value | Specifies how to proceed when NGT Copy encounters a table space or partition that has an unacceptable status or has a BMC or DB2 utility running against it, as follows:  
  - END—terminate processing with a RC=12  
  - SKIP—issue a message, skip over the space, and continue processing |
| ON ERROR ICEXISTS       | Configuration option value | Specifies how to proceed if NGT Copy encounters a table space or partition for which an image copy already exists, as follows:  
  - END—terminate processing with a RC=12  
  - SKIP—issue a message, skip over the space, and continue processing other spaces as specified.  
  This option is effective for the COPY IMAGECOPY command. |
<table>
<thead>
<tr>
<th>Option/NGT Copy keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
</table>
| **ON ERROR NOTSUPPORTED**<br>Keyword: ON ERROR NOTSUPPORTED | Configuration option value | Specifies how to proceed if NGT Copy encounters a table space or partition that is of a type that is not supported by NGT Copy, as follows:  
- END—terminate processing with a RC=12  
- SKIP—issue a message, skip over the space, and continue processing other spaces as specified.  
This option is effective for both the COPY and COPY IMAGECOPY commands. |
| **Outsize threshold for BIGDDN**<br>Keyword: OUTSIZE | configuration option value | Specifies a size threshold for making copies to an alternate DD or output descriptor  
This option can be used to escalate output to tape rather than DASD. OUTSIZE is specified as number of pages from 0 through 999999. The default is 0, which means this option has no effect.  
OUTSIZE can be used with any FULL option. The size of the copy is based on the size of a full copy. |
| **Readtype**<br>Keyword: READTYPE | RANDOM | Specifies the method that is used to make incremental copies  
Random—use the conventional (random I/O) method of making incremental copies  
Fullscan—use full table space scan techniques to identify changed pages  
Auto—determine the I/O method based on the number of changed pages  
**Note:** This option is valid only when you specify Full No or Full Auto. |
| **Report statistics**<br>Keyword: REPORT | NO | Reports statistics via SYSPRINT  
This option is used in conjunction with the RUNSTATS option. |
| **Resetmod**<br>Keyword: RESETMOD | Configuration option value | Specifies whether to reset the modified page indicators in the table spaces and space maps after you make an image copy. |
| **Resync**<br>Keyword: RESYNC | YES | Instructs XBM or SUF to resynchronize hardware mirroring activities after an Instant Snapshot copy. You can specify No to leave the mirrors unsynchronized during subsequent user processing. If you choose No, you must manually reestablish the mirroring synchronization using XBM or SUF.  
**Note:** This option is ignored if the Instant Snapshot copy is made without hardware mirroring in place. |
<table>
<thead>
<tr>
<th>Option/NGT Copy keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RUNSTATS</td>
<td>NO</td>
<td>Updates the DB2 catalog or the BMCSTATS table with statistics that are collected concurrently with the creation of full image copies. If you collect statistics, you can also choose to report statistics (Report statistics option) and specify how the DB2 catalog tables or the BMCSTATS table should be updated. <strong>Note:</strong> To use this option, you must specify Yes for at least one of the following two suboptions: ■ Update BMCSTATS ■ Update DB2 catalog</td>
</tr>
<tr>
<td>Separate by partition</td>
<td>NO</td>
<td>Specifies making copies by partition when backing up partitioned table spaces. <strong>Note:</strong> ■ This option is valid when the objects have not been listed by part ■ This option is invalid with OBJECTSET syntax</td>
</tr>
<tr>
<td>Option/NGT Copy keyword</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Shrlevel</td>
<td>Reference</td>
<td>Specifies the level of access to the target spaces that NGT Copy permits during the copy process, as follows:</td>
</tr>
<tr>
<td>Keyword: SHRLEVEL</td>
<td></td>
<td>- Reference—allows only read-only access by other programs during the copy process.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Change—maintains the initial level of access of the spaces.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Any—uses Shrlevel Change unless NGT Copy encounters any conditions that require more restrictive access.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- None—stops all access by other programs to the target spaces during the copy process.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Concurrent—utilizes the SUF product to make consistent copies of the table spaces while updates to those table spaces are in progress. You can specify a value of required or preferred for this option.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Concurrent required—terminates the copy with a return code of 12 if a consistent point cannot be obtained or maintained.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Concurrent preferred—uses Shrlevel Change when a consistent copy cannot be obtained or maintained or if initialization of the SUF or XBM fails.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> Shrlevel Concurrent is not valid with compressed indexes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Change Consistent Yes—use for Online Consistent Copy. This option requires a solution password and must be set at the group level.</td>
</tr>
<tr>
<td>Squeeze</td>
<td>Configuration option value</td>
<td>Specifies whether to consolidate the rows on each target table space page so that all free space is contiguous. This option is obsolete for NGT Copy version 6.3 and later.</td>
</tr>
<tr>
<td>Keyword: SQUEEZE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Option/NGT Copy keyword</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Start message</td>
<td>Null</td>
<td>Writes a string of not more than 50 characters to the operating system system log. This option is valid only with Group Yes and when using SUF to make Sh level Concurrent copies. <strong>Note:</strong> Do not use quotation marks in the Start message field. Doing so will result in an error message at run time.</td>
</tr>
<tr>
<td>Support Full Copy DDs (FULLDDN)</td>
<td>Null</td>
<td>Instructs NGT Copy to use an alternative descriptor for COPYDDN when the FULL AUTO FULLPCT settings cause a full copy to be made. If you specify Yes and a full copy is made, the image copy output goes to an alternate set of DDs that are specified with the following keywords:</td>
</tr>
<tr>
<td>Support OUTSIZE option (BIGDDN)</td>
<td>NO</td>
<td>Instructs NGT Copy to use an alternative descriptor for COPYDDN when the number of pages to copy is equal to or greater than the outsize threshold value. If you specify Yes and the threshold condition is met or exceeded, the image copy output goes to an alternate set of DDs that are specified with the following keywords:</td>
</tr>
<tr>
<td>■ BIGDDN</td>
<td></td>
<td>■ BIGDSN</td>
</tr>
<tr>
<td>■ BIGRECDDN</td>
<td></td>
<td>■ BIGRECDSDN</td>
</tr>
<tr>
<td>■ BIGRECDSN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unit Count</td>
<td>Null</td>
<td>Specifies the number of units to be allocated for the output image copy data sets. Valid values are 0 to 59. Zero (0) enables you to control the unit count with SMS if required. The default is no unit count at all. If you are using DSNUTILB COPY, you can cause RECOVERY MANAGER to calculate the unit count by setting a value in Max primary allocation.</td>
</tr>
<tr>
<td>Update BMCSTATS table</td>
<td>NO</td>
<td>Collects statistics and update the BMCSTATS table. This option is used in conjunction with the RUNSTATS option.</td>
</tr>
</tbody>
</table>
DB2 COPY (DSNUTILB) options

You can set backup options that are specific to the IBM DB2 COPY utility on the DSNUTILB Copy Options panel.

The options are listed alphabetically in Table 15 on page 181 along with the DSNUTILB defaults used by RECOVERY MANAGER. For some options, availability depends on the version of DB2 that is installed in your system.

For more detailed information about the DSNUTILB COPY utility options that you can use with RECOVERY MANAGER, see the IBM DB2 command and utility reference.
Table 15: DSNUTILB backup options and defaults

<table>
<thead>
<tr>
<th>Option</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANGELIMIT</td>
<td>Null</td>
<td>Specifies whether to make incremental or full image copies based on a specified percentage of changed pages.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Use this option in conjunction with Full copy Auto, and provide a percentage at the Incremental percent and Full percent prompts, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ To create an incremental copy when changed pages are more than zero but less than a percentage, type the percentage at Incremental percent.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ To create an incremental copy when changed pages are between two percentages, type the smaller percentage at Incremental percent and the larger percentage at Full percent. No copy is made if changed pages are less than or equal to the smaller percent.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ To create a full copy when changed pages are equal to or more than a percentage, type it at Incremental percent or at Full percent.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ To create a full copy when no pages have changed, type 0 at either prompt. Otherwise, no copy is made when no pages have changed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> You may enter the incremental and full percentage values in either decimal or integer format. Valid integer values range from 0 to 100. Valid decimal values are 00.0 to 99.9. A decimal value can only be specified to the tenth’s place (1/10 of a percent).</td>
</tr>
<tr>
<td>Concurrent/DFSMS</td>
<td>NO</td>
<td>Makes concurrent copies using the DFSMS Concurrent Copy utility. This option is valid only when you specify a full image copy. SHRLEVEL REFERENCE is required for objects with a page size greater than 4K.</td>
</tr>
<tr>
<td>Copy Objects as a Group</td>
<td>NO</td>
<td>Specifies that all target objects share a common point of consistency. This option is valid for table spaces, index spaces, and indexes. Only full image copies are supported for indexes.</td>
</tr>
<tr>
<td>Option</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Copy Objects in Parallel</td>
<td>NO</td>
<td>Copies objects in parallel Be aware of the following information:  ■ Concurrent/DFSMS and Copy Objects in Parallel are mutually exclusive.  ■ You must specify Yes to Copy Objects as a Group to be able to copy objects in parallel.  ■ Copy Objects in Parallel is not available when copying objects to tape.</td>
</tr>
<tr>
<td>Full copy</td>
<td>YES</td>
<td>Specifies whether a full or incremental copy should be made, as follows:  ■ Yes—full image copy  ■ No—incremental image copy  ■ Auto—use the CHANGELIMIT option to escalate to a full image copy request when the following conditions exist:  — The most recent full image copy is a DFSMS concurrent copy.  — No full image copy exists for the spaces that are being copied  — This is the first image copy request after a REORG or LOAD  — This is the first copy since a partial recovery was performed  — The most recent copy job for this space was terminated  — The space is a &quot;special case&quot; table space.</td>
</tr>
<tr>
<td>Max parallel objects</td>
<td>0</td>
<td>Specifies the maximum number of objects that should be processed in parallel</td>
</tr>
<tr>
<td>Option</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Shrlevel</td>
<td>REFERENCE</td>
<td>Specifies the level of access that is allowed to DB2 applications and utilities that are executing concurrently during the copy process, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Shrlevel Reference—allows read-only access by other programs to the spaces in the group during the copy process.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Shrlevel Change—allows other programs to write to the spaces in the group. This option is not valid for table spaces having a page size of 32 kilobytes (KB) when you specify Concurrent/DFSMS Yes.</td>
</tr>
</tbody>
</table>

**NGT Recover (OUTCOPY) options**

You can use NGT Recover to make updated image copies from typical recovery resources such as prior image copies, change accumulation files, and DB2 logs. During an OUTCOPY ONLY operation, NGT Recover writes the output to a sequential image copy data set instead of a DB2 space, which enables you to make copies without accessing the DB2 space or interfering with normal DB2 access in any way. If you elect to end the process at any of the following points, the new copy, if registered, is registered as a SHRLEVEL REFERENCE full image copy:

■ The log point of a SHRLEVEL REFERENCE incremental
■ The log point of a quiesce point
■ The log point of the last -ARCHIVE MODE (QUIESCE) command
■ The last successful subsystem shutdown (in a non-data-sharing environment)

*Note*

OUTCOPY cannot be used for compressed indexes.

For more detailed information about the NGT Recover OUTCOPY option, see the *BMC Next Generation Technology Recover for DB2 for z/OS User Guide*.

The NGT Recover OUTCOPY options are listed alphabetically in Table 16 on page 184 along with the defaults that apply in this case.
### Table 16: NGT Recover OUTCOPY backup options

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
</table>
| ANALYZE    | YES              | Prints information about objects that are targeted for recovery when you execute the recovery JCL, as follows:  
  - Yes—prints a recovery plan before performing the recovery.  
  - No—prints only a small subset of the recovery plan information.  
  - Only—terminates execution of the recovery JCL after the plan is printed.                                                                                                                                                                                                         |
| AUTOSIZE   | YES              | Turns dynamic sizing for output image copies or change accumulation output files on or off:  
  - Yes—specifies dynamic sizing for output image copies or change accumulation output files allocated to DASD.  
  - No—specifies that output image copies or change accumulation output files are allocated to DASD using the primary and secondary quantities that are specified in the R+/CHANGE ACCUM repository.                                                                 |
| EARLYCAT   | YES              | Verifies that data sets that are marked as cataloged during the ANALYZE phase actually exist in the operating system catalog. Specify No to delay verification until the data sets are allocated.                                                                                                                                                       |
| EARLYRECALL| YES              | Retrieves archived image copies and log data sets during the ANALYZE phase  
  Specify No to delay recall until the data sets are accessed or allocated.                                                                                                                                                                                                                                                                  |
| MAXDRIVES  | NGT Recover configuration option | Specifies the maximum number of tape drives to be used during backup. The default is the value of the NGT Recover MAXDRIVES configuration option.                                                                                                                                                                                          |
| MAXPRIM    | 0                | Sets a maximum amount of disk space (in the units specified by SPACE) to allocate as primary space. Valid values are 0 through 65535. A nonzero value establishes an upper limit for primary space allocation; 0 specifies no limit.                                                                                                                                        |
### Option descriptions

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SORTDEVT</td>
<td>NGT Recover</td>
<td>Specifies the device type for temporary work data sets that are required for log sorts</td>
</tr>
<tr>
<td></td>
<td>SORTDEVT</td>
<td>Note: If no configuration option value is specified, this option defaults to the system sort routine installation default.</td>
</tr>
<tr>
<td>TOLOGPOINT</td>
<td>CURRENT</td>
<td>Specifies the point in the DB2 log to which the backup copies should be made, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Current—make backup copies of all recoverable objects in the group to the current time.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- LASTQUIESCE—make a backup of each recoverable object to the last quiesce that is registered in SYSCOPY for that object.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Specific RBA—make backup copies of all objects in the group to the same RBA. Specify that RBA at the Specific RBA prompt.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- LASTARCHQUIESCE—make backup copies of each object in the group to the point in the log that is established by the most recent ARCHIVE LOG MODE(QUIESCE) for the DB2 subsystem.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- LASTSHUTDOWN—make backup copies of all objects in the group to the point in the log that is established by the most recent STOP DB2 command.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- LASTCOMMONQUIESCE—make backup copies of all objects in the group to the point in the log that is established by the most recent common point.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- TIMESTAMP—make backup copies of all objects in the group to the RBA associated with the timestamp in the format YYYY-MM-DD-HH.MM.SS.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- LOGMARK—make backup copies of all objects in the group to the RBA associated with the logmark.</td>
</tr>
</tbody>
</table>

### Alternate resources

When you use the OUTCOPY ONLY feature of NGT Recover to make backups, you can indicate which resources (image copies, logs, and change accum files) should be used as a basis for the new copies.
The order in which you rank these alternate resources determines which resource is used. If the first choice is not available, NGT Recover falls back to the second choice. If you set a choice to zero, NGT Recover does not use the resource. If you do not specify any order for the alternate resources, RECOVERY MANAGER uses the NGT Recover default values. The alternate resources panel (ARMRO05C) is displayed only if the value for Alternate resources on panel ARMROR41 Update NGT Recover Options is set to Yes or Auto.

For more information, see the BMC Next Generation Technology Recover for DB2 for z/OS Reference Manual.

WARNING
BMC recommends that you do not set the options for alternate resources at the subsystem level. Setting alternate resources allows the recover utility to utilize specific recovery resources in a given order. Setting this value at the subsystem level results in all recoveries utilizing the recovery resources as defined in the alternate resource selection.

Output copy data set options

You can set a wide range of output copy data set parameters that apply to all of the supported backup utilities.

The options are defined separately for each type of output copy (local site primary, local site backup, recovery site primary, recovery site backup) and are available at the subsystem and group levels. The options are listed alphabetically in Table 17 on page 187 along with the defaults that, in this instance, are all RECOVERY MANAGER defaults.

Note
You specify the types copies that you want to make in the General Backup Options panel (described in “General backup options” on page 167). The default specification is to make only a local primary copy.

You can specify separate output data set options when NGT Copy is the backup utility, as follows:

- FULLDDN—use FULLDD output data sets when a NGT Copy FULL AUTO incremental copy is escalated to a full image copy. RECOVERY MANAGER generates FULLDDN syntax when you have set the Support Full Copy DDs (FULLDDN) NGT Copy option to Yes.

- BIGDDN—use the BIGDD output data set options when you want full copies that exceed a specified number of pages to be allocated to a different output descriptor
or data set name. RECOVERY MANAGER generates BIGDDN syntax when you have set the appropriate NGT Copy options. The Support Outsize Option (BIGDDN) option must be set to Yes and the Outsize Threshold for BIGDDN must be set to a value other than 0.

For more information on NGT Copy backup options, see “NGT Copy backup options” on page 170.

**Note**

One options panel is displayed for each of the following copy types:
- Local site primary
- Local site backup
- Recovery site primary
- Recovery site backup

### Table 17: Output data set option defaults and validity

<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACPGDG</td>
<td>Null</td>
<td>Specifies a fully qualified data set name to define a GDG base for this copy. The named data set must contain the control cards that are needed to perform an IDCAMS DEFINE as well as the symbolic variable &amp;BASE, which replaces the GDG base name. The variable 'xx' is LP, LB, RP, or RB, as appropriate.</td>
</tr>
<tr>
<td>Option</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DSSNAP</td>
<td>NO</td>
<td>Specifies making a hardware-based Instant Snapshot copy of DB2 data, as follows:</td>
</tr>
<tr>
<td>Keyword:</td>
<td></td>
<td>Yes—make a hardware-based Instant Snapshot copy</td>
</tr>
<tr>
<td>DSSNAP</td>
<td></td>
<td>Auto—made a hardware copy if possible, but fall back to a standard copy if</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the hardware copy fails (for example if XBM, SUF, or the required hardware is</td>
</tr>
<tr>
<td></td>
<td></td>
<td>not in place)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>No—make a standard copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note: The BMC NGT Copy and EXTENDED BUFFER MANAGER or SNAPSHOT UPGRADE FEATURE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>products are required to use this option. The following restrictions apply:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ DSSNAP is not allowed with Compressed Index.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ DSSNAP is not allowed with Copy Type INCR or AUTO.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ DSSNAP is not allowed with Encipher.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ DSSNAP is not allowed with Resetmod YES or DEFLT.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ DSSNAP is not allowed with Online Consistent Copy.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ FULLDD is only valid with Copy Type Full Auto.</td>
</tr>
<tr>
<td>Allocation</td>
<td>Cylinder</td>
<td>Specifies whether the primary and secondary allocations quantities are</td>
</tr>
<tr>
<td>type</td>
<td></td>
<td>expressed in cylinders or tracks for disk units</td>
</tr>
<tr>
<td>Keyword:</td>
<td></td>
<td>■ Primary allocation—specifies the primary allocation for disk units. The</td>
</tr>
<tr>
<td>SPACE</td>
<td></td>
<td>default is 10. Use this option only when RECOVERY MANAGER is unable to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>estimate the quantity.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Secondary allocation—specifies the secondary allocation quantity for disk</td>
</tr>
<tr>
<td></td>
<td></td>
<td>units. The default is 20. Use this option only when RECOVERY MANAGER is</td>
</tr>
<tr>
<td></td>
<td></td>
<td>unable to estimate the quantity.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note: Primary and secondary allocation values are ignored if the selected</td>
</tr>
<tr>
<td></td>
<td></td>
<td>backup utility is NGT Copy. NGT Copy calculates the correct allocations.</td>
</tr>
<tr>
<td>Catalog</td>
<td>Yes</td>
<td>Catalogs the data sets in the operating system catalog</td>
</tr>
<tr>
<td>Keyword:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CATALOG</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Option</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Data set name</td>
<td>&amp;USERID.&amp;DB.&amp;TS.&amp;TYPE&amp;DATETIME</td>
<td>Specifies the name of the disk or tape data set for the current copy type. The RECOVERY MANAGER default is &amp;USERID.&amp;DB.&amp;TS.&amp;TYPE&amp;DATETIME. Note: You can use symbolic variables to construct this name (see “Subsystem-level considerations” on page 97).</td>
</tr>
</tbody>
</table>
| EATTR for EAV           | Not specified    | Specifies whether a data set supports extended attributes or not. Specifying no value for EATTR allows the value for EATTR to be set by an SMS DATACLAS. Valid values are:  
  - OPT—specifies that extended attributes are optional for the data set. You must set OPT to allocate an extended format sequential data set. By using OPT, NGT Copy supports sequential data sets in the cylinder-managed portion of EAVs. Extended format sequential data sets must be allocated on SMS-managed volumes and the size of the data set must be greater than the EAV break point, which is typically 10 cylinders.  
  - NO—specifies that the data set cannot have extended attributes. |
<p>| Encrypt the copy        | NO               | Specifies making encrypted image copies to prevent unauthorized access to sensitive company information. (This option is only displayed when you are using RECOVERY MANAGER as part of the BMC Recovery Management for DB2 solution and are using the solution password.) Copy encryption is incompatible with DSSNAP YES or AUTO. Copy encryption is incompatible with compressed indexes. DSSNAP is set in the NGT Copy Specific Options panel. |
| Expiration date         | 1999/000         | Specifies the expiration date for a tape copy data set, in the format yyyy/ddd. The RECOVERY MANAGER default is 1999/000, which indicates no expiration. Expiration date and Retention period are mutually exclusive. |</p>
<table>
<thead>
<tr>
<th>Option</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Migrate copy data sets</td>
<td>NO</td>
<td>Specifies Hierarchical Storage Management (HSM) migration of copy data sets when NGT Copy is finished with them. Available only with COPY PLUS version 6.4 or later. Valid values are as follows:</td>
</tr>
<tr>
<td>Keyword: MIGRATE</td>
<td></td>
<td>■ No — suppresses migration</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ HSM — specifies migration to compressed disk. Ensure that enough space is available on the disk when using this value.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ HSM ML2 — causes immediate migration to a migration level 2 (MIGRATIONLEVEL2) volume.</td>
</tr>
<tr>
<td>Model data set name</td>
<td>Null</td>
<td>Specifies the name of a cataloged data set used to define the model DCB. The data set name must be fully qualified.</td>
</tr>
<tr>
<td>(DCB) Keyword: MODELDCB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Retention period</td>
<td>None</td>
<td>Specifies the tape copy data set retention period in days (1 through 999). Retention period and Expiration date are mutually exclusive.</td>
</tr>
<tr>
<td>Keyword: RETPD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMS data class</td>
<td>Null</td>
<td>Specifies a valid SMS data class name for disk data sets. The name must not exceed 8 characters. RECOVERY MANAGER forces Catalog Yes when you specify this option.</td>
</tr>
<tr>
<td>Keyword: DATACLAS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMS management class</td>
<td>Null</td>
<td>Specifies a valid SMS management class name for disk data sets. The name must not exceed 8 characters. RECOVERY MANAGER forces Catalog Yes when you specify this option.</td>
</tr>
<tr>
<td>Keyword: MGMTCLAS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SMS storage class</td>
<td>Null</td>
<td>Specifies a valid SMS storage class name for disk data sets. The name must not exceed 8 characters. RECOVERY MANAGER forces Catalog Yes when you specify this option.</td>
</tr>
<tr>
<td>Keyword: STORCLAS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Option</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>Stack (copies on tape)</td>
<td>NO</td>
<td>Specifies whether to stack image copies of the same type contiguously on the same tape. Always respond No to Stack if you specified a disk unit for the image copy data sets; otherwise you will receive an INVALID COMBINATION message. You will also receive that message if you respond Yes to Stack and leave Tape blank. For Recovery Management solution only - you can specify Cabinet to create cabinet copies. Cabinet copies can be made to either disk or tape. Stack is not valid with compressed indexes. For more information, see the <em>Recovery Management for DB2 User Guide</em>.</td>
</tr>
<tr>
<td>Tape</td>
<td>NO</td>
<td>Specifies whether the unit is a tape or disk, as follows: Yes—tape (you must also provide the name of a tape unit at the Unit prompt). No—disk. <strong>Note:</strong> If you provide a unit name and leave Tape blank, you will receive an INVALID COMBINATION message.</td>
</tr>
<tr>
<td>Unit</td>
<td>SYSALLDA</td>
<td>Specifies the name of the disk or tape unit to which the image copy data sets are written.</td>
</tr>
<tr>
<td>Volume count</td>
<td>0</td>
<td>Specifies the largest number of tape volumes (1 through 255) that you expect to create. This option applies only to tape data sets. You can leave this field blank if you expect no more than five tape volumes to be created.</td>
</tr>
</tbody>
</table>

### Symbolic variables in image copy data set names

With RECOVERY MANAGER, you can request that image copies of the objects that you recover be made when the recovery completes.

If you supply data set names instead of accepting RECOVERY MANAGER defaults, you can construct those names by using the symbolic variables that are shown in Table 18 on page 191.

**Table 18: Symbolic variables for copy data set names**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;DB</td>
<td>The name of the database containing the space being copied</td>
</tr>
<tr>
<td>&amp;TS</td>
<td>The name of the object being copied</td>
</tr>
<tr>
<td>&amp;ICTYPE</td>
<td>Type of image copy</td>
</tr>
</tbody>
</table>
### Symbol & Result

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;DSNUM, &amp;PART</td>
<td>The number of the data set or partition being copied</td>
</tr>
<tr>
<td>&amp;PART5</td>
<td>The partition number of the object (always a 5-digit number, with leading zeros)</td>
</tr>
<tr>
<td>&amp;USERID, &amp;USER, &amp;UID</td>
<td>Your TSO user ID</td>
</tr>
<tr>
<td>&amp;SSID</td>
<td>The ID of this DB2 subsystem</td>
</tr>
<tr>
<td>&amp;ATTACH</td>
<td>The data sharing group name</td>
</tr>
<tr>
<td>&amp;DATE</td>
<td>The current date (in yymmdd format)</td>
</tr>
<tr>
<td>&amp;JDATE</td>
<td>The current date (in yyddd Julian date format)</td>
</tr>
<tr>
<td>&amp;TIME</td>
<td>The current time (in hhmmss format)</td>
</tr>
<tr>
<td>&amp;LDSNUM; &amp;LPART</td>
<td>The number of the data set or partition being copied (long name support)</td>
</tr>
<tr>
<td>&amp;YEAR</td>
<td>The current year (in yy format)</td>
</tr>
<tr>
<td>&amp;MONTH</td>
<td>The current month (in MM format)</td>
</tr>
<tr>
<td>&amp;DAY</td>
<td>The current day (in DD format)</td>
</tr>
<tr>
<td>&amp;JDAY</td>
<td>The current day (in DDD Julian format)</td>
</tr>
<tr>
<td>&amp;HOUR</td>
<td>The current hour (in HH format)</td>
</tr>
<tr>
<td>&amp;MINUTE</td>
<td>The current minute (in MM format)</td>
</tr>
<tr>
<td>&amp;SECOND</td>
<td>The current second (in SS format)</td>
</tr>
<tr>
<td>&amp;TYPE</td>
<td>The type of output being produced</td>
</tr>
<tr>
<td></td>
<td>- LP for a local site primary copy</td>
</tr>
<tr>
<td></td>
<td>- LB for a local site backup copy</td>
</tr>
<tr>
<td></td>
<td>- RP for a remote site primary copy</td>
</tr>
<tr>
<td></td>
<td>- RB for a remote site backup copy</td>
</tr>
<tr>
<td>&amp;UNIQ or &amp;UQ</td>
<td>1- to 8-character value, based on the system clock, that is used to generate unique copy data set names</td>
</tr>
</tbody>
</table>

- The maximum total length that is allowed for a data set name is 44 bytes.
- Symbols with a numeric result must be prefixed by one or more alphabetic characters.
About incremental index copies

RECOVERY MANAGER supports incremental copies for indexes if you have BMC-supported versions of both NGT Copy and NGT Recover.

Having the ability to produce incremental (FULL NO) index space copies reduces the need for excessive DB2 log application for an index space recovery. In addition, incremental index space copies usually require much less disk space than full copies.

The implementation of the incremental index space copy process parallels the incremental process for table spaces. When you specify a copy with INDEXES YES, the FULL NO or FULL AUTO options are implemented for both the indexes and table spaces. This includes the FULL NO/AUTO parameters such as FULLPCT, MAXINCRS, FULLDAY, MINPAGES, EMPTY, CUMULATIVE, KEEP, READTYPE, and READPCT.

If you have NGT Recover, RECOVERY MANAGER will recognize incremental index copies as recovery resources. NGT Recover is required to recover from incremental index copies.

For more information, see the BMC Next Generation Technology Copy for DB2 for z/OS Reference Manual and the BMC Next Generation Technology Recover for DB2 for z/OS Reference Manual.

About Instant Snapshots

A hardware-based Instant Snapshot can be made with RECOVERY MANAGER by using NGT Copy in conjunction with EXTENDED BUFFER MANAGER (XBM) or SNAPSHOT UPGRADE FEATURE (SUF).

RECOVERY MANAGER can then use these copies to recover DB2 spaces in conjunction with NGT Recover.

Instant Snapshots are always full copies. XBM or SUF provides the facility to identify the existence of specialized hardware and use the hardware and its control unit to make Instant Snapshots at the data set level. Instant Snapshots do not require the I/O that is needed to make a standard copy. For more information about Instant Snapshots, see the EXTENDED BUFFER MANAGER and SNAPSHOT UPGRADE FEATURE User Guide.

Note

Instant Snapshots are different from cache-based Snapshot Copies and from the hardware Snapshot Copies that utilize mirroring, which are both specified in NGT Copy by using SHRLEVEL CONCURRENT DSSNAP NO.
Instant Snapshots can be used to recover DB2 spaces only by using NGT Recover as the recover utility. For more information about recovering Instant Snapshots, see the *BMC Next Generation Technology Recover for DB2 for z/OS Reference Manual*.

For a table space or an index, you can specify a mixture of Instant Snapshots and standard copies. For example, the local primary can be a Instant Snapshot copy and the remote primary can be a standard tape copy.

You specify Instant Snapshots for an application group or a single object by using the DSSNAP option in the NGT Copy Options panel, which has the following valid values:

- **DSSNAP NO** (the default) makes standard copies—not Instant Snapshots
- **DSSNAP YES** makes Instant Snapshots.
- **DSSNAP AUTO** makes Instant Snapshots if possible, but falls back to a standard copy if necessary (for example, if XBM, SUF, or the required hardware is not in place).

The *Use COPY IMAGECOPY for this output type* option on the NGT Copy options panel lets you make additional standard copies from Instant Snapshot copies. NGT Copy selects the primary Instant Snapshot copy to use as the source for COPY IMAGECOPY based on the value specified in the RECOVERY MANAGER NGT Copy options panel.

**Note**
The COPY IMAGECOPY command ignores the DSSNAP option because it cannot make Instant Snapshots. COPY IMAGECOPY is incompatible with compressed indexes.

So in addition to making quick copies for local recoveries, Instant Snapshots enable you to make standard copies for disaster recovery or migration.

Note that NGT Copy turns off copy pending status if only an Instant Snapshot is made, even though DB2 is not aware of the copy.

## Allocation of Instant Snapshots

Instant Snapshots require the use of dynamic allocation and output copies to DASD.

NGT Copy output data sets cannot be GDG data sets. However, output data set names can be constructed using symbolic variables.
The data sets created by Instant Snapshots are always cataloged in the ICF catalog. These data sets are VSAM linear data sets and are physical copies of the original DB2 data set.

**Note**

For Instant Snapshot copies, the **Data set name** that you specify on the Output options panel is the VSAM cluster name. The data component is named by the hardware implementation, which differs from vendor to vendor. Therefore, the maximum length of the data set name for Instant Snapshots is 39 characters. If you request DSSNAP and the data set name is longer than 39 characters, RECOVERY MANAGER ignores the DSSNAP option and inserts a comment in the JCL indicating the data set name is too long.

The target allocation information must meet the requirements of the associated hardware in order to use the copy facilities. The source data set must also meet hardware requirements to utilize hardware copy features. For details regarding supported hardware devices and their requirements for source and target devices, see the _EXTENDED BUFFER MANAGER and SNAPSHOT UPGRADE FEATURE User Guide_.

### Registration of Instant Snapshots

Instant Snapshots are registered in the BMC BMCXCOPY table with an STYPE of V.

They are not registered in SYSIBM.SYSCOPY because they are not in the standard format for copies and cannot be used by standard DB2 utilities.

If a standard backup copy (LB or RB) is produced and its associated primary copy (LP or RP) is an Instant Snapshot and as such is registered in BMCXCOPY, the backup is also registered in BMCXCOPY, even if it is a standard copy.

If either a local primary (LP) or a local backup (LB) is requested with DSSNAP AUTO and the Instant Snapshot fails (and standard copies are made), all of the copies made are registered in SYSCOPY.

When NGT Copy registers an Instant Snapshot in BMCXCOPY, it also inserts a row with ICTYPE T in SYSIBM.SYSCOPY to prevent incremental copies from being made. (This prevents integrity problems that could arise if a DSNUTILB incremental copy were attempted following an Instant Snapshot.)

For Instant Snapshots, the DSNAME column of BMCXCOPY will contain the VSAM data component data set name (not the VSAM cluster name) for ease of use by subsequent processes.
If COPY IMAGECOPY makes a backup copy of a primary copy that is an Instant Snapshot, the backup copy is registered in BMCXCOPY. If COPY IMAGECOPY is used to make a RP or LP copy of an Instant Snapshot, that copy is registered in SYSCOPY.

Command option restrictions for Instant Snapshots

When you are specifying Instant Snapshots, in addition to other requirements mentioned in the section, the following restrictions apply:

- Instant Snapshots require output data sets to DASD.
- Because Instant Snapshots are always full copies, FULL YES is required with DSSNAP YES or DSSNAP AUTO.
- RESETMOD NO is required for Instant Snapshots.
- XBMID (set at the subsystem or group level) is required for DSSNAP YES or DSSNAP AUTO. This value is used to specify XBMID in the COPY command and applies to SHRLEVEL CONCURRENT copies.

Note

If the XBMID is not specified at the group level, a comment stating that DSSNAP requires that this value is inserted in the JCL. However, RECOVERY MANAGER assumes that the installation value for the XBMID is to be used and issues no error message.

- DSNUM ALL is not allowed for Instant Snapshots. See “DSNUM and Instant Snapshots” on page 197 for restrictions for different values of DSNUM with DSSNAP YES or DSSNAP AUTO.
- Compressed indexes are incompatible with Instant Snapshots.

OUTPUT command options applied to Instant Snapshots

The following output options apply to Instant Snapshot copies:

- Output descriptor name
- DSNNAME
- DATACLAS
- MGMTCLAS
- STORCLAS
- VOLUMES
COPY command options ignored for Instant Snapshots

Some COPY command options are ignored if only an Instant Snapshot is made (because NGT Copy does not actually read or write DB2 pages for an Instant Snapshot). The following options do not apply to Instant Snapshot copies and are ignored:
- CHECKTSLEVEL
- CHECKERROR
- RUNSTATS and its suboptions
- SQUEEZE
- NACTIVE

DSNUM and Instant Snapshots

RECOVERY MANAGER sets DSNUM=ALL in all cases except the following:
- If you specify DSSNAP YES or DSSNAP AUTO, RECOVERY MANAGER sets DSNUM=DATASET.
- If you are using NGT Copy and you are copying a table space, RECOVERY MANAGER sets DSNUM=DATASET.
- If you are making copies of table spaces by partition and the number of partitions is greater than zero, RECOVERY MANAGER sets DSNUM=DATASET.
- If you specify DSNUM=n, where n is any number other than zero. If you specify a value, RECOVERY MANAGER uses that value.

**Note**

If you specify DSNUM ALL for a table space, this option does not appear in the generated JCL because it is the default. Also, DSNUM DATASET is only used for indexes when DSSNAP YES or DSSNAP AUTO is specified. If you specify DSNUM DATASET for an index space, this option does not appear in the generated JCL.

SHRLEVEL and Instant Snapshots

Any value of SHRLEVEL--CHANGE, CONCURRENT, REFERENCE, ANY, and NONE—is allowed for Instant Snapshots. Thus, Instant Snapshots are useful for either consistent copies or "fuzzy" copies. By using SHRLEVEL CHANGE, you can take advantage of the quick Instant Snapshot for backup and recovery without the need of a QUIESCE.
If a standard Snapshot Copy is also being produced, the copy proceeds like any other SHRLEVEL CONCURRENT copy. Therefore, the copies can use the XBMRRSTRT=YES option.

When SHRLEVEL CONCURRENT PREFERRED and DSSNAP YES are specified, if the Instant Snapshot fails, the copy does not continue as a SHRLEVEL CHANGE copy. The copy fails. However, when SHRLEVEL CONCURRENT PREFERRED and DSSNAP AUTO are specified, if the Instant Snapshot fails, NGT Copy attempts a standard Snapshot Copy. Then, if the Snapshot Copy fails, NGT Copy continues the copy as a SHRLEVEL CHANGE copy.

**Instant Snapshot use by other BMC utilities**

Instant Snapshots are recognized and used by other BMC products that access the BMCXCOPY table in which these copies are registered.

- NGT Recover uses these copies for recovery.
- The MODIFY command in NGT Copy deletes the Instant Snapshots from BMCXCOPY and the ICF catalog through its standard functions. All commands operate on the Instant Snapshots registered in BMCXCOPY as well as their associated standard backup copies. However, template-generated copies are standard copies only—not Instant Snapshot copies.
- UNLOAD PLUS unloads data from these copies.
Recovering groups

This chapter discusses how to recover groups.

Preparing recovery jobs

To prepare a recovery job for one or more groups, you must supply a recovery point and values for some related options.

You can usually accept the defaults for most of these variables. You must also supply the names of the data set where the output JCL is to be placed and the job card information for that JCL (see “Output data sets, job cards, and symbolic variables” on page 51).

**Note**
RECOVERY MANAGER does not currently support objects having the following DB2 Version 8 statuses:
- Versioned table spaces (ALTSCHEMA status)
- Table spaces defined with table-based partitioning (TBLPART status)

Message BMC80539 will be generated when generating backup or recovery JCL for groups that contain objects with these statuses.

You can generate the recovery JCL entirely interactively or by using the batch generation program, ARMBGEN. Which method you use depends on such things as whether you want to change group and object recovery options and the reason for generating the JCL. For more discussion about the criteria you can use to decide how to create the recovery job, see “Choosing interactive or batch generation” on page 208.

Whichever method you use, you can select the type of recovery for the group or groups that you specify. The types of recovery available when you use RECOVERY MANAGER to generate a recovery job for a group are:

- Recovery to the current time
- Recovery to a specified image copy
- Recovery to a specified quiesce point
- Recovery to a selected common point
- Recovery to a restart RBA (not available interactively)
- Recovery to a specified LRSN (with or without inflight resolution)
- Recovery to a timestamp (BMC Recovery Management for DB2 solution only)
- Recovery to a Log Master for DB2 log mark (BMC Recovery Management for DB2 solution only)

You can also optionally specify the following:

- A recovery strategy that eliminates unchanged objects from the recovery
- A recovery that uses only the DB2 log (see “BACKOUT recovery” on page 201)
- A printed report of the resources that are required prior to a recovery
  You can request this report either without performing the recovery or in addition to performing the recovery.
- A simulation of recovery for the application spaces (see “Recovery simulation for application spaces” on page 206)

This is a feature of the Recovery Management solution.

For more information about recovery type specifications, see “Options for recovery JCL generation” on page 213.

**Note**

Unless you plan to accept the current recovery option values for the group, whatever they may be, you must change them before you generate the recovery JCL. Recovery options are accessible from the Object List panel (for individual objects) and from the Group Edit panel (by group). For detailed information, see “To set or change utility options interactively” on page 125.

---

**Supporting OBJECTSET**

RECOVERY MANAGER supports the OBJECTSET option of NGT Recover when you specify NGT Recover as the recovery utility. This option enables NGT Recover to use the repository to identify the objects in the group, which means that you do not need to regenerate the recovery JCL when objects in the group change.

RECOVERY MANAGER generates the RECOVER OBJECTSET creator.groupName syntax in the recovery JCL.

The ARMBSET program (“ARMBSET— OBJECTSET processing” on page 659) is generated after a NGT Recover step to issue a CHECK or REPAIR for each object in
an OBJECTSET. With the implementation of OBJECTSET for NGT Recover, ARMBSET is needed for the following reasons:

- The **Check Pend Action** option ("General recovery options" on page 233 and "General recovery options" on page 847) does not generate a step for CHECK or REPAIR when OBJECTSET syntax is generated.

- You cannot use classic CHECK or REPAIR JCL, which would represent a static set of objects, because it may not match the contents of the OBJECTSET, which is dynamic.

## BACKOUT recovery

A backout recovery backs out log records to undo or redo the changes that occurred between the selected point in time and the current point.

This method returns the spaces and indexes to the required state without the overhead of restoring image copies, or rebuilding or restoring indexes. A backout recovery does not require image copies to perform a point-in-time recovery. In most cases, the BACKOUT recovery strategy is dramatically faster than traditional forward recovery. For more information about the BACKOUT option, see the *BMC Next Generation Technology Recover for DB2 for z/OS Reference Manual*.

BACKOUT recovery requires that spaces be undamaged and not be in RECP, RECP*, RBDP, RBDP*, PSRCP, PSRBD, GRECP, WEPR, or STOPE status or have an LPL range. BACKOUT also cannot be used for the following spaces:

- LOB spaces
- NOT LOGGED spaces

**Note**

With a BMC Recovery for DB2 solution password, you can use the group or an object pattern to display backup and recover time estimates and to save time estimates in a data set.

You can also use BACKOUT when you choose DB2 RECOVER (DSNUTILB) as the recovery utility. The default value is NO. BACKOUT with DSNUTILB has the same restrictions as BACKOUT with NGT Recover.

If DSNUTILB is selected as the recovery utility and the DB2 version is less than Version 10, RECOVERY MANAGER changes BACKOUT to NO and continues.

When you specify the backout to forward recovery strategy by using BACKOUT AUTO, RECOVERY MANAGER generates JCL that causes NGT Recover to perform back out recoveries for eligible objects and forward recoveries for any objects that
cannot be backed out. This strategy provides the fastest possible recovery of your objects. The BACKOUT AUTO strategy is a feature of and is the default value for the BMC Recovery for DB2 solution and requires a valid BMC Recovery Management for DB2 solution password. For more information, see the Recovery Management for DB2 User Guide.

LOB and XML object recovery

Because of their unique structure, LOBs and XML objects have different recovery requirements than ordinary table spaces.

The following LOB-related objects must always be recovered to the same point:

- Base table space—contains the LOB base table, where the large object column is stored
- LOB table space—contains the LOB auxiliary table, where the data is physically stored
- Index on the auxiliary table

The following XML-related objects must always be recovered to the same point:

- The following XML-related objects must always be recovered to the same point:
  - base XML table space — contains the XML base table, where the logical XML column is stored
- DocID index on the DOCID column in the base table
- XML table space — contains the XML auxiliary table, where the data is physically stored
- NodeID index on the XML table space
- XML index on the XML table space

RECOVERY MANAGER performs a number of checks and special processing to ensure that LOBs and XML objects are backed up and recovered correctly, as follows:

- Enables you to automatically include all LOB-related or XML-related spaces in the application groups that you create
- Supports backup and recovery of LOB spaces and XML spaces using BMC or IBM utilities
- Issues warnings if you attempt to recover an object without its LOB or XML-related spaces
- Optionally generates CHECK or REPAIR steps after recovery to remove CHECK-pending, REBUILD-pending, or AUXW statuses

**Note**

RECOVERY MANAGER can generate backout recoveries on the base table spaces and indexes, but not on LOB or XML table spaces. If you specify Backout Auto, RECOVERY MANAGER automatically passes the LOB or XML table spaces to the forward recovery step. If you specify Backout Yes, RECOVERY MANAGER issues an error message.

Table 19 on page 203 shows the status in which DB2 places LOB or XML-related objects after different types of recoveries. RECOVERY MANAGER generates JCL to remove the objects from pending status when possible.

**Table 19: LOB and XML object status after being recovered**

<table>
<thead>
<tr>
<th>Object</th>
<th>Recovery type</th>
<th>Base table space status</th>
<th>Index on auxiliary table status (ROWID, NodeID, or XML values)</th>
<th>LOB or XML table space status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base table space</td>
<td>Current RBA or LRSN</td>
<td>None</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Base table space</td>
<td>Point-in-time</td>
<td>CHECK-pending</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Index on the auxiliary table (ROWID, node ID, or XML)</td>
<td>Current RBA or LRSN</td>
<td>None</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>Index on the auxiliary table (ROWID, node ID, or XML)</td>
<td>Point-in-time</td>
<td>None</td>
<td>CHECK-pending</td>
<td>None</td>
</tr>
<tr>
<td>LOB or XML table space</td>
<td>Current RBA or LRSN, LOB or XML with LOG(YES)</td>
<td>None</td>
<td>None</td>
<td>None</td>
</tr>
<tr>
<td>LOB or XML table space</td>
<td>Current RBA or LRSN, LOB or XML with LOG(NO)</td>
<td>None</td>
<td>None</td>
<td>Auxiliary warning</td>
</tr>
<tr>
<td>LOB or XML table space</td>
<td>TOCOPY copy was SHRLEVEL REFERENCE</td>
<td>CHECK-pending</td>
<td>REBUILD-pending</td>
<td>None</td>
</tr>
<tr>
<td>Object</td>
<td>Recovery type</td>
<td>Base table space status</td>
<td>Index on auxiliary table status (ROWID, NodeID, or XML values)</td>
<td>LOB or XML table space status</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>--------------------------------</td>
<td>-------------------------</td>
<td>---------------------------------------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>LOB or XML table space</td>
<td>TOCOPY copy was SHRLEVEL CHANGE</td>
<td>CHECK-pending</td>
<td>REBUILD-pending</td>
<td>CHECK-pending or auxiliary warning</td>
</tr>
<tr>
<td>LOB or XML table space</td>
<td>TORBA or TOLOGPOINT (not a quiesce point)</td>
<td>CHECK-pending</td>
<td>REBUILD-pending</td>
<td>CHECK-pending or auxiliary warning</td>
</tr>
<tr>
<td>LOB or XML table space</td>
<td>TORBA or TOLOGPOINT (at a quiesce point)</td>
<td>CHECK-pending</td>
<td>REBUILD-pending</td>
<td>None</td>
</tr>
</tbody>
</table>

a  Dependent table spaces that are related by informational referential constraints are not put into CHECK-pending status.

b  RECOVERY MANAGER does not generate REPAIR JCL for a LOB or XML table space defined as LOG(NO) even when you set the Check Action to Repair because doing so would remove the exception status.

c  If a log record is applied to a LOB or XML table space, and the LOB or XML is marked invalid, the LOB or XML table space is set to auxiliary warning status.

d  RECOVERY MANAGER generates JCL to remove the index from REBUILD-pending status if the index is in the same group as the LOB or XML table space.

e  RECOVERY MANAGER generates JCL to remove the LOB or XML table space from CHECK-pending status if it is in the same group as the LOB or XML base table space. If the table space was defined as LOG(NO), recovered, and updated since the last image copy, it is placed in Auxiliary Warning status rather than CHECK-pending. In this case, RECOVERY MANAGER does not generate REPAIR JCL. Specify CHECK PEND REPAIR if you want the AUXW status repaired.

**NOT LOGGED object recovery**

RECOVERY MANAGER supports recovery of NOT LOGGED objects.

Table spaces or index spaces that have a logging attribute of NOT LOGGED can only be recovered to one of three types of recoverable points, as follows:

- Creation point of the NOT LOGGED object
- Alter point at which the object was altered from LOGGED to NOT LOGGED

---

204  *RECOVERY MANAGER for DB2 User Guide*
Image copy of the NOT LOGGED object

RECOVERY MANAGER performs a number of checks and special processing to ensure that NOT LOGGED objects are backed up and recovered correctly, as follows:

— Supports backup and recovery of NOT LOGGED spaces using BMC or IBM utilities

— Issues warnings if you attempt to copy an index without its NOT LOGGED table space

— Issues an error message if you attempt to use LOG ONLY recovery

   Recovery using LOG ONLY is not valid for NOT LOGGED objects for any point between the last copy and the recovery point because UNDO and REDO entries do not exist for those objects.

— Handles requests for BACKOUT recovery

   BACKOUT recovery is not valid for NOT LOGGED objects because UNDO and REDO entries do not exist for those objects. If you specify BACKOUT AUTO, RECOVERY MANAGER automatically passes the NOT LOGGED spaces to the forward recovery step. If you specify BACKOUT YES, RECOVERY MANAGER issues an error message.

— Issues a warning if you attempt to recover an object to current because changes might have occurred since the last recoverable point

   If a change has occurred to the NOT LOGGED object since the last recoverable point, a recovery to the current time could result in data loss.

— Ignores a request for QUIESCE WRITE NO and continues processing and issues an informational message

   Invoking QUIESCE WRITE NO does not create a recoverable point for NOT LOGGED objects.

### Timestamp recovery

RECOVERY MANAGER supports inflight resolution technology and the timestamp recovery feature of the BMC Recovery Management for DB2 solution.

This feature completely eliminates the need to perform application quiesces by resolving inflight transactions when performing a recovery to any user-specified timestamp or log point. A valid BMC Recovery Management for DB2 password is required to use this feature. For more information, see the Recovery Management for DB2 User Guide.
Log mark recovery

RECOVERY MANAGER supports recovery to log marks defined in Log Master for DB2 and registered in the Log Master ALPMARK table.

The ability to create a log mark is a unique feature of Log Master. Log marks enable you to associate a name with a point on the log. When you specify a log mark, you can later refer to that point by the log mark name. You do not have to know the actual RBA/LRSN.

This feature requires the use of:

- NGT Recover version 9.1.00 or later as the recovery utility
- A valid BMC Recovery Management for DB2 password

For more information about log marks, see the Log Master for DB2 Reference Manual and the Log Master for DB2 User Guide. For more information about recovery using log marks, see the Recovery Management for DB2 User Guide.

Recovery simulation for application spaces

The recovery simulation feature simulates all aspects of a recovery of the spaces in a group up to, but not including, the actual I/O. You might find simulation useful in reducing your disaster recovery or local recovery testing costs. Simulation is a feature of the BMC Recovery Management for DB2 solution and requires the solution password. For more information, see the Recovery Management for DB2 User Guide.

Optimized recovery JCL

RECOVERY MANAGER optimizes the performance of your JCL based on the recover utility you select and the maximum number of jobs you use.

For the best results, you should:

- Select NGT Recover as the recover utility
- Provide a number for the maximum number of jobs that you want to use (based on your processing environment)
RECOVERY MANAGER creates multiple recovery jobs that run concurrently whenever possible.

For more information, see “Multiple job optimization” on page 71 for more information.

**Note**

For some scenarios, you can choose to generate the multiple recovery jobs into separate members to provide more control over job submission. For more information, see “Separating jobs from a multi-job batch job stream” on page 77.

RECOVERY MANAGER streamlines tape unit use by analyzing stacked input image copies and generating the DD statements.

RECOVERY MANAGER specifies REGION=0M for NGT Recover job steps. If you code a value for REGION in your job card, that value will override the RECOVERY MANAGER specification on all job steps.

When you are recovering indexes, using a check utility, or making output copies after a recovery, RECOVERY MANAGER automatically sizes the data sets, based on the data set sizing value that is set in the utility options. Data sets in groups that are created by using the ARMBGPS do not require automatic sizing based on the data set sizing value.

**Tip**

To enable RECOVERY MANAGER to provide highly accurate estimates of data set sizes and minimize the time that is required to provide those estimates, use the RUNSTATS utility regularly to record up-to-date statistics in the DB2 catalog.

RECOVERY MANAGER groups indexes to minimize table space scanning. It also groups spaces to minimize log scanning when applying log records to the spaces that are being recovered.

If symbolic names are specified for output image copies, RECOVERY MANAGER expands them and passes the results to NGT Recover or DB2 COPY when those utilities are utilized. If NGT Copy is the utility of choice, the symbolic names are passed to NGT Copy.

If you use a check utility, the work space that is required is limited by using one job step per object. The amount of work space is limited in the same way when indexes are recovered by DB2 RECOVER.
Choosing interactive or batch generation

You can choose to generate recovery JCL interactively by using the RECOVERY MANAGER online interface or in batch by using the RECOVERY MANAGER program ARMBGEN:

- The interactive method allows you to specify recovery points, specify recovery options at the group level and revalidate the recoverability of the current group. RECOVERY MANAGER generates the JCL based on your choices. Apart from providing a job statement for the generated recovery job, you are not required to code any JCL. This method might require a lengthy TSO session. For more information, see “Generating recovery JCL interactively” on page 208.

- The batch generation method enables you to use the online interface to create JCL, which you can execute to produce a recovery job. This approach requires only a short TSO session, but you cannot change the values of recovery options, which are those in effect for the target group when the ARMBGEN recovery job executes. For more information, see “Generating recovery JCL in batch” on page 210.

Generating recovery JCL interactively

You can generate JCL for a recovery of one or more selected groups interactively.

You can specify different recovery points, recover utilities, or exclusion for individual objects, as required. You can also specify a recovery to a specified copy, a specified quiesce point, a common point, a specified RBA or LRSN, a specified timestamp, or a specified log mark. For these types of recovery points, RECOVERY MANAGER validates the recoverability of each object in the group and displays a list of related objects that can not be included in the recovery.

Before you begin

To generate recovery JCL, you need the following authorizations if you are not the creator of the group:

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- TYPE A authority if you intend to save changes to the group
- TYPE O authority if you do not intend to save changes to the group
- Authority to update the output data set for the JCL

To generate a recovery job interactively

Start this procedure at the Object List panel, which appears after you have created or retrieved a group.
1 Select option **1 Recovery point** and press **Enter**. The Recovery Type Selection panel is displayed.

2 Select a recovery type, then specify the other options.

   a For option descriptions, see “Options for recovery JCL generation” on page 213.

   b If you selected **Current**, proceed to Step 7 on page 209. Otherwise continue with the next step. The Partial Recovery Verification panel is displayed.

3 To display a list of objects related to the objects included in your group, specify **1** (Yes) beside one or more of the options shown. Then press **Enter**.

4 The Error Message List panel is displayed if any objects are found.

5 If you selected a recovery to a common point, the Recovery Point List panel is displayed. Otherwise, the Object List panel is displayed (Step 7 on page 209).

6 In the Recovery Point List panel, process the objects that are listed as follows:

   ■ **H**—lists all objects in the group for which this is a valid recovery point. These objects will be included in the recovery if you select this point as the recovery point.

   ■ **M**—lists all objects in the group that do not have a valid recovery point at this RBA or LRSN. These objects will be marked with the status NOTAVAIL if you select this point as the recovery point.

   ■ **S** or **/**—selects a point as the recovery point for all objects in the group that have a valid recovery point. Objects that do not have a valid recovery point will be marked with a status of NOTAVAIL and will not be included in the recovery.

When you have completed your processing, the Object List panel appears.

7 If necessary, process individual objects. For more information about process options, see “Actions on objects after recovery point selection” on page 220.

8 Select Option 3, **Gen Recover JCL foreground** to generate JCL for all objects that have an **OK** status. *Any object with any other status is not included in the recovery JCL.*

   **Note**

   Depending on the number of objects to be recovered, generating the JCL may take a noticeable length of time.

9 When the JCL Specification panel appears, enter a fully qualified output data set name.
Generating recovery JCL in batch

RECOVERY MANAGER enables you to use the ARMBGEN batch program to generate recovery JCL for one or more groups.

Using the ARMBGEN program provides:

- Support for coordinated recoveries
- Better automation for point-in-time recovery after an application failure
- Increased automation for disaster-recovery planning
- Reduced demands on TSO sessions by performing recovery analysis offline

When you use ARMBGEN, you can specify the recovery point for one or more groups. The type of recovery points available are as follows:

- Recovery to the current time
- Recovery to a specified image copy
- Recovery to a specified quiesce point
- Recovery to a specified common point
- Recovery to a specified RBA or LRSN (with or without inflight resolution)
- Recovery to a restart RBA
- Recovery to a timestamp with inflight resolution (BMC Recovery Management for DB2 solution only)
- Recovery to a log mark (BMC Recovery Management for DB2 solution only)

If you recover to a previous point in time, ARMBGEN can optionally check for related objects that are not in the group. (If the group is defined via SAP, the check is unnecessary and so is not performed. Also, if INDEX ALL is specified in the utility options, ARMBGEN does not check for related indexes in the group because their inclusion is implied.)

You can also specify whether you are creating the recovery JCL to run at the local site (local recovery) or at a recovery site (disaster recovery) and whether you want to simulate the recovery (requires the Recovery Management solution password).
Note
By default, ARMBGEN does not perform the check for UNCHANGED status. The XUNCHANGED option can be used to perform this check when used with SITETYPE LOCAL and a recovery point other than TO CURRENT. For more information on UNCHANGED status, see “Object status after recovery point selection” on page 221. For more information about ARMBGEN, see “ARMBGEN—Backup and recovery JCL” on page 431.

All groups that are specified in the same recovery must have the same type of recovery point.

Using ARMBGEN in disaster recovery planning

You can use ARMBGEN to provide more automation for the recovery of your applications in a disaster recovery situation.

ARMBGEN uses the end RBA that is stored in the archive history file to generate ready-to-run application recovery jobs that you can transport to the recovery site. At the recovery site, after the system resource recovery jobs have been run, you can execute the application recovery jobs.

For more information about disaster recovery, see “Recovering from a DB2 system disaster” on page 291.

Generating batch recovery JCL interactively

RECOVERY MANAGER provides online support for creating ARMBGEN jobs to perform recovery in batch mode.

You specify batch JCL generation from the Group List panel or Object List panel, and then proceed to specify the recovery point and job information. If you are using the BMC Recovery Management for DB2 solution password, you can also generate JCL to simulate a recovery of the selected groups or objects. For more information about simulation, see the Recovery Management for DB2 User Guide.

Note
JCL generation for application recovery considers BACKUP SYSTEM full volume backups as a valid backup for DSNUTILB recovery. NGT Recover does not support full volume backups so BACKUP SYSTEM backups are ignored if the recover utility is AFRMAIN.
**Before you begin**

The following authorizations are required:

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- TYPE O (Open) access for the target group or groups
  
  DB2 must be active when you execute the generated JCL.

**To generate a batch recovery job**

Start this procedure at either the Group List panel or the Object List panel.

1. Select a group or groups for processing, as follows.
   
   a. If you begin in the Group List panel, type a wildcard pattern or the name of a single group in the format `creator.name`, and then press **Enter**. In the Act (action) column, type **J** (generate JCL for selected groups) or **JA** (generate JCL for All groups) to specify one or more groups for processing.
   
   The Batch Group JCL Generation Options panel is displayed.
   
   b. If you begin in the Object List panel, select the **Gen recover JCL background** processing option and then press **Enter**.
   
   **Note**
   
   To generate JCL for multiple groups, make your selections in the Group List panel.
   
   The Batch Group Recovery Point Specification and Batch Group JCL Generation Options panels appears.

---

**Batch Group Recovery Point Specification**

```
Command ===> _________________________________________________________________

Recover to . . . . . 1 1. Current             2. Image copy
               3. Quiesce             4. Common recovery point
               5. Specific Logpoint   6. Restart RBA
               7. Timestamp           8. Logmark

Full copy only . . 2 1. Yes      2. No  (for option 2)
Relative point . . 0_   0=Last to 99=99th previous (for options 3,4 or 8)
To Timestamp . . . . . . . . . . . . . . . . (option 2)
To Logpoint . . . . . . . . . . . . . . . . (option 5)
To Logmark . . . . . . . . . . . . . . . . (option 8)
Resolve Inflights . . . 2 1. Yes      2. No  (for options 5 and 8)
```

**Batch Group JCL Generation Options**

```
Command ===> _________________________________________________________________

Log Only . . . . . 2 1. Yes      2. No
Sitetype . . . . . 1 1. LOCAL    2. RECOVERY
JCL type . . . . . 1 1. LOCAL    2. DR
```
In the Batch Group Recovery Point Specification panel, select a recovery type, then specify the other options as described in “Options for recovery JCL generation” on page 213.

In the Batch Group JCL Generation Options panel, enter the SYSUT2 DD statement for batch JCL output (including //).

**Note**

The optional SYSUT4 DD statement may also be available if you are using mirroring as part of your recovery strategy. It is only available when all of the following criteria are true:

- You are using the BMC Recovery Management for DB2 solution
- The recovery point is TORESTARTRBA
- JCLTYPE is DR
- Mirroring is selected for the group

When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:

- The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.

- The job statement must contain a symbolic variable (&#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

Save the JCL data set or submit the job as required.

## Options for recovery JCL generation

This section describes the recovery fields available on the RECOVERY MANAGER panels, including those for establishing a recovery point and for generating background JCL.
Table 20: Recovery JCL generation fields

<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recover to</td>
<td>Current</td>
<td>Specifies the recovery point, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Current—recovers to the current time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If you do not specify a recovery strategy, RECOVERY MANAGER uses the most recent full and incremental copies to recover the spaces, then applies log records to make the spaces current. If you are using NGT Recover, you can specify Log Sort to merge the image copy records with the log records before they are applied to the spaces.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Image Copy—recovers to a specified full or incremental image copy that is registered in SYSIBM.SYSCOPY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This option is used in conjunction with the <strong>Full copy only</strong> and the <strong>Copy or quiesce</strong> fields.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note</strong>: DFSMS concurrent copies that are registered in SYSIBM.SYSCOPY are counted as an image copy.)</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Recover to</td>
<td></td>
<td>■ <strong>Quiesce</strong>—recovers to a specified quiesce point that is registered in SYSIBM.SYSCOPY by the DB2 QUIESCE command. This option is used in conjunction with the Copy or quiesce field. <strong>Note:</strong> The RBA for the quiesce point is not the same for all objects unless DB2 performed a quiesce on all of the objects in the group at the same time.</td>
</tr>
</tbody>
</table>
|                        |                          | ■ **Common recovery point**— recovers to a valid previous recovery point for a number of objects in the current group. RECOVERY MANAGER analyzes the log to locate valid recovery points for objects in the group within a specified date and time range, then displays the recovery points as a list. From the list, you can perform the following tasks:  
  — Select a recovery point. Only objects for which the point is valid are recovered.  
  — Obtain a list of objects for which the point is valid (hits).  
  — Obtain a list of objects for which the point is not valid (misses).  
  **Note:** If you are generating JCL in batch mode, the ARMBGEN program considers only common recovery points that have no misses. (A miss indicates that one or more objects in the group do not have a share level reference image copy, a quiesce, or are within a quiet range at a given point.) In contrast, online JCL generation gives you the opportunity to select a common point that has one or more misses. |
<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recover to</td>
<td></td>
<td>■ Specific LOGPOINT—rescopes all recoverable objects in the group to a specified RBA or LRSN. You must supply the appropriate 12-digit hexadecimal value in the To LOGPOINT field.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Restart RBA—recovers to the RBA of the last disaster recovery point.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Timestamp—rescopes to a user-specified timestamp and resolves inflight units of work. (BMC Recovery Management for DB2 solution only)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Logmark—rescopes to a user-specified log mark and requires the use of NGT Recover version 9.1.00 or later as the recovery utility. (BMC Recovery Management for DB2 solution only)</td>
</tr>
<tr>
<td>Full copy only</td>
<td>No</td>
<td>Specifies whether to use only full copies for recovery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If you select this option, RECOVERY MANAGER ignores all incremental copies and recovers to the full copy that you specify in the Copy or quiesce field.</td>
</tr>
<tr>
<td>Relative point</td>
<td>0</td>
<td>Specifies the copy, quiesce point, common point, or log mark to which to recover</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide a value from 0 to 99, where 0 represents the most recent copy, quiesce, common point, or log mark. Used in conjunction with Current, Image Copy, Quiesce, or Logmark.</td>
</tr>
<tr>
<td>To Logpoint</td>
<td></td>
<td>Specifies the RBA or LRSN to which to recover</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide a 12-digit hexadecimal value of an RBA or LRSN. Used in conjunction with Specific LOGPOINT.</td>
</tr>
<tr>
<td>Resolve Inflights</td>
<td>No</td>
<td>Specifies whether to resolve inflight units of work when recovering to a log point or a log mark.</td>
</tr>
<tr>
<td>To Timestamp</td>
<td></td>
<td>Specifies the timestamp for the recover</td>
</tr>
<tr>
<td>To Logmark</td>
<td>No</td>
<td>Specifies the name of the log mark for recover</td>
</tr>
<tr>
<td>Sitetype</td>
<td>LOCAL</td>
<td>Specifies whether to use local or recovery-site resources for the recovery</td>
</tr>
<tr>
<td>JCL type</td>
<td>LOCAL</td>
<td>Specifies whether the JCL is to be used for local recovery or disaster recovery</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> This option is used for mirroring support only. To invoke mirroring for a group, the JCL type must be DR.</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Check unchanged</td>
<td>NO</td>
<td>Analyzes SYSLGRNX to determine which objects in the selected groups have not changed since the last backup and then exclude those objects from the recovery. Excluding unchanged objects can significantly reduce the amount of time required for recovery. This option is ignored for recover to current and for sitetype recovery. <strong>Note:</strong> Organizations with heavy update activity may experience contention on SYSLGRNX when this option is used.</td>
</tr>
<tr>
<td>Simulate recovery</td>
<td>NO</td>
<td>Simulates recovery for the objects in the group or groups (requires the BMC Recovery Management for DB2 solution password). The recovery simulation feature simulates all aspects of recovery up to, but not including, the actual I/O. For more information about recovery simulation, see the <em>Recovery Management for DB2 User Guide</em>.</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Backout</td>
<td>RECOVERY MANAGER only:</td>
<td>Invokes the BACKOUT strategy for NGT Recover or DB2 RECOVER (DSNUTILB) (depending on which recovery utility you selected) for point-in-time recovery</td>
</tr>
<tr>
<td></td>
<td><strong>No</strong></td>
<td>Specify one of the following choices:</td>
</tr>
<tr>
<td></td>
<td><strong>Auto</strong></td>
<td>- <strong>Yes</strong>—backout recovery for all objects in the group</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>No</strong>—do not perform backout recovery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>Auto</strong>—backout-to-forward recovery strategy (requires the BMC Recovery Management for DB2 solution password. For point-in-time recoveries when you use Recovery Management, see “BACKOUT recovery” on page 201.) BACKOUT AUTO is the default value. <strong>Backout Auto</strong> is not valid with DSNUTILB.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Be aware of the following issues:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- To use <strong>Backout Auto</strong> or <strong>Backout Yes</strong>, you must specify <strong>Quiesce</strong>, <strong>Common recovery point</strong>, or <strong>Specific LOGPOINT, or Timestamp</strong> as the recovery point. If you specify <strong>Current</strong> (the default), RECOVERY MANAGER overrides <strong>Backout Auto</strong> and converts the option to <strong>Backout No</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>Backout Auto</strong> overrides the <strong>Outcopy by Recover</strong> option and uses the copy utility that you specified in the general backup options panel. (If you selected NGT Recover - OUTCOPY as the copy utility on the general backup options panel, the product uses DSNUTILB.)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>Backout Auto</strong> overrides <strong>Unloadkeys</strong> and proceeds with the backout. If you specify <strong>Backout Yes</strong> with <strong>Unloadkeys</strong>, an error message is issued and you must change one option or the other to continue.</td>
</tr>
</tbody>
</table>
### Options for recovery JCL generation

<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Backout (continued)</td>
<td></td>
<td>- Backout Auto and Backout Yes overrides Rebuild all indexes and changes it to Rebuild No.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Logscan Yes overrides Backout Auto and changes it to Backout No. RECOVERY MANAGER issues a warning message.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Backout recoveries cannot be performed on the following objects:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- LOB spaces</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- NOT LOGGED spaces</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If you specify Backout Auto, these spaces will be recovered by the forward recovery method. If you specify Backout Yes, RECOVERY MANAGER issues an error message.</td>
</tr>
<tr>
<td>Log Only</td>
<td>No</td>
<td>Applies log records to a data set that has already been restored to a prior point in time. You can apply the records to one of the following points:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Backout Auto and Backout Yes overrides Rebuild all indexes and changes it to Rebuild No.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The current time</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- A specified RBA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- A specified LRSN</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> You cannot use the Log Only option when you perform a recovery to a specified copy.</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Analyze</td>
<td>Yes</td>
<td>Enables you to print a recovery plan before executing the recovery, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ <strong>Yes</strong>—prints a recovery plan and generates JCL for recovery. Information includes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Names of any required image copy data sets</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Names of any required log data sets</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Log ranges, if any</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Phases that will occur during execution</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Number of log pages to be read</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Record sizes for index sort work data sets</td>
</tr>
<tr>
<td></td>
<td></td>
<td>— Steps to occur within each phase</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ <strong>No</strong>—prints only the information in the first four bullets and generates JCL for recovery.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ <strong>Only</strong>—execution terminates after the information is printed.</td>
</tr>
<tr>
<td>Recover clones only</td>
<td>No</td>
<td>Generates the CLONES ONLY option in the recovery JCL syntax</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The CLONES ONLY option causes clone objects to be included in the recovery and non-cloned objects and clone bases to be excluded. This option is displayed only when running on DB2 Version 10 or later and is not valid with compatibility mode.</td>
</tr>
</tbody>
</table>

**Actions on objects after recovery point selection**

After you have selected a recovery point (current or partial) for the current group, you can perform certain actions on individual objects currently included in the group.

In the Object List panel you can do the following tasks:

- Use action code **X** to exclude an object from further processing.
- Use action code **O** to display and change the recovery options for an object.
- Use action code **Z** to display the full text of a DB2 long name. (You can also position the cursor in the field containing the long name and press **F4** to display...
the long name.) See “Online display of DB2 long names” on page 87 for more information.

- If a row exists, use action code D to display SYSCOPY row information for the selected recovery point for an object. In general, this action is useful only for Image Copy and Quiesce recoveries.

- Use action code L to display all available recovery points for an object. These recovery points include full and incremental image copies and quiesce points back to the most recent LOAD REPLACE LOG point. This option allows you to select an alternate recovery point for an object with a NOTAVAIL or BAD TYPE status.

  Note
  You cannot select an alternate recovery point if the recovery point is a result of a REORG LOG NO, LOAD LOG NO, or LOAD REPLACE LOG NO operation.

Object status after recovery point selection

This topic describes the object status after recovery point selection.

Unrecoverable object statuses that can be returned by RECOVERY MANAGER are shown in “RECOVERY MANAGER object exception status” on page 839. In some cases, you can recover such objects in a separate recovery. In other cases you can recover such objects by using an alternate recovery point or by regenerating the object list. However, it is also possible that there is no valid recovery point for the object. Additionally, for point in time recoveries, objects may have the status of UNCHANGED. This status indicates that the object has not been updated between the selected recovery point and the current time. Because recovery of such objects to the selected recovery point would result in no change, they will be excluded from the recovery, thus eliminating unnecessary processing.

Submitting your recovery job

Before submitting a recovery job that is generated by RECOVERY MANAGER, verify the following:

- Are you fully authorized to use all of the utilities that are needed for recovery?

- If you are recovering from a volume failure, have you initialized the new volumes with the same name as the failed volumes? If you do not do so, you must modify the JCL to reflect the different names.
Restarting failed recovery jobs

This section describes the actions that you should take to restart or rerun the recovery JCL generated by RECOVERY MANAGER.

■ To restart a single recovery job, see “To restart a single failed recovery job” on page 222

■ To restart jobs created using the Multiple Job Optimization feature, see “Restarting a recovery for a set of concurrent jobs” on page 223.

Note
You cannot restart a recovery simulation job. You must resubmit the JCL.

To restart a single failed recovery job

If a single job fails during execution, you can restart it at the failed job step, as follows:

1 Make a backup copy of the recovery JCL before you begin to edit it for restart.

2 Add a RESTART= stepName option to the RECOVERY MANAGER job statement, where stepName is the name of the job step that failed during the prior execution.

3 Continue as follows:

a If the failed job step is a DSNUTILB step,

■ Issue a DB2 -DISPLAY UTILITY(*) command to determine whether DB2 considers it necessary to restart the utility.

■ If the recover utility appeared in the previous DISPLAY UTILITY(*), add RESTART as the third parameter on the recovery job step EXEC statement.

For example, if the subsystem ID is DBDF, the parameter list should look like the following:

```
EXEC PGM=DSNUTILB,PARM='DBDF,,RESTART'
```

b If the failed job step is a BMC utility, no modification to the restart parameter is necessary. RECOVERY MANAGER uses NEW/RESTART for NGT Copy, NEW/RESTART(PHASE) for NGT Recover, and NEW for CHECK PLUS.

c If the failed job step is the execution of the ARMBSTP program preceding the IDCAMS DELETE step required to support the Delete STOGROUP objects and Redefine VCAT object options, do one of the following:
■ Wait for the objects to stop and restart the job at the next step.

■ Submit the job again.

If you want to change the number of times that ARMBSTP attempts to stop the objects or the wait period between attempts, you must change the values in the ARMBSTP job step. The default values are 30 tries and 2 seconds wait time.

No other modifications are necessary to restart NGT Copy and CHECK PLUS. However, restarting NGT Recover and DSNUTILB may require modifications to DD statements in those job steps.

---

**Note**

RECOVERY MANAGER uses the default utility ID for each DB2 utility and each BMC utility.

Refer to the appropriate reference manual for more information about restarting an IBM or BMC utility.

4 Submit the edited JCL.

---

## Restarting a recovery for a set of concurrent jobs

The following topics describe the procedures for restarting a recovery for a set of jobs.

A recovery for the set of jobs consists of all of the jobs into which RECOVERY MANAGER splits a recovery for purposes of optimization, including any required synchronization and clean-up jobs. See “Optimized recovery job processing” on page 74.

RECOVERY MANAGER has the following paths for restarting a recovery for a set of jobs:

■ For jobs generated online and by ARMBGEN for application data, RECOVERY MANAGER uses ARMBMJO and the JOB_RESTART table (“Job history: JOB_RESTART table” on page 806) to restart failed jobs. For more information, see “Restarting jobs that recover application data” on page 224.

■ For ARMBSRR jobs for system resource recovery, RECOVERY MANAGER uses a synchronization file to restart failed jobs. For more information, see “Restarting system resource recovery (ARMBSRR) jobs” on page 226.
Restarting jobs that recover application data

Restarting application data recovery jobs created online or by ARMBGEN uses the ARMBMJO program and the RECOVERY MANAGER JOB_RESTART table.

For the RECOVERY MANAGER JOB_RESTART table, see “Job history: JOB_RESTART table” on page 806).

If any job or jobs in a multi-job set that recovers application data fails, you restart the job or jobs in one of the following ways:

**Note**
Do not code a RESTART parm on the JOB card.

- Fix the problem and resubmit only the failing job or jobs. The ARMBMJO steps determine what jobs and steps need to be run.

- If no jobs in the set are still executing, fix the problem and resubmit the entire JCL set. The ARMBMJO steps determine what jobs and steps need to be run.

**Note**
If a failing job is executing NGT Recover using the UNLOADKEYS/BUILDINDEX strategy, refer to the *BMC Next Generation Technology Recover for DB2 for z/OS Reference Manual* for more information.

Rerun or restart?

If any of the generated jobs fail, you must first determine what caused the failure and correct the situation. Then you should decide whether to rerun the entire job stream (by resubmitting the generated JCL) or restart the jobs at the point of failure.

You change the value of the PARM parameter in the ARMBMJO syntax to determine whether the jobs are to rerun or restart. The following options are available:

- To run the entire set of jobs for the first time, set PARM NEW/RESTART.
  
  A row is inserted in the JOB_RESTART table for the set of jobs, for each job and for each step. All steps in all jobs will run.

- To run the entire set of jobs after a failure without a restart, set PARM NEW.
  
  Edit the ARMBMJO syntax in the first job of the set changing NEW/RESTART to NEW. All steps in all jobs will run.
To run the entire set of jobs after a failure with restart, set PARM NEW/RESTART.
Submit the original set of JCL with no change. Only the steps that did not complete successfully will run.

To run an individual job from a set after a failure with restart, set PARM NEW/RESTART.
Submit only the failed job. Only the steps that did not complete successfully for that job will run.

To run an individual job from a set after a failure without restart, set PARM NEW.
Edit the ARMBMJO syntax in the failed job and change NEW/RESTART to NEW.
Submit only the failed job. All steps in the job will run.

ARMBMJO CLEAR_TABLE

The ARMBMJO$ SAMPLIB contains the CLEAR_TABLE member. Run this member to delete all rows in the JOB_RESTART table. IDENTIFIER is optional, and if specified, deletes only rows for the named identifier.

```
CLEAR_TABLE
  SET_IDENTIFIER identifierName
```

This syntax is not generated by RECOVERY MANAGER.

ARMBMJO report

The ARMBMJO$ SAMPLIB contains the following REPORT member:

```
REPORT
  SET_IDENTIFIER identifierName
```

Run this member to generate a report that selects all rows for the specified identifier and writes this information to a report similar to the one in Figure 22 on page 226. IDENTIFIER is optional and if it is not included, the report includes all rows in the JOB_RESTART table.

Notice that the report includes information at the following levels:

- Set (set of jobs)—where the row in the report does not include information for either the JOBNAME or STEPNAME

- Job—where the row in the report includes information for the JOBNAME but not the STEPNAME
### Step

where the row in the report includes information for both the JOBNAME and the STEPNAME

---

**Figure 22: Example ARMBMJO report**

<p>| IDENTIFIER: RDAJLW3.DEDL.D120325.T130826 |
|-----------------|-----------------|---------|-----------------|-----------------|-----------------|</p>
<table>
<thead>
<tr>
<th>JOBNAME</th>
<th>STEPNAME</th>
<th>STATUS</th>
<th>EVENT</th>
<th>RC</th>
<th>START_TIME</th>
<th>END_TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJECTSET</td>
<td>RESTART</td>
<td>COMPLETE</td>
<td></td>
<td>0</td>
<td>2012-03-26-07.24.16</td>
<td>2012-03-26-07.25.30</td>
</tr>
<tr>
<td>PUBLIC.O1642823_PART</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JLW0001</td>
<td>ARM0002</td>
<td>COMPLETE</td>
<td>DEL/DEF</td>
<td>0</td>
<td>2012-03-25-15.21.31</td>
<td>RDAJLW3.JLW0001</td>
</tr>
<tr>
<td>2012-03-25-15.23.14</td>
<td>ARM0010</td>
<td>COMPLETE</td>
<td>RECOVER</td>
<td>0</td>
<td>2012-03-25-15.23.06</td>
<td>RDAJLW3.JLW0001</td>
</tr>
<tr>
<td>JLW0001</td>
<td>ARM0014</td>
<td>COMPLETE</td>
<td>REBUILDX</td>
<td>0</td>
<td>2012-03-25-15.23.18</td>
<td>RDAJLW3.JLW0001</td>
</tr>
<tr>
<td>2012-03-25-15.23.27</td>
<td>Arm0019</td>
<td>COMPLETE</td>
<td>REPAIR</td>
<td>0</td>
<td>2012-03-26-07.25.26</td>
<td>RDAJLW3.JLW0001</td>
</tr>
<tr>
<td>2012-03-26-07.25.27</td>
<td>Arm0012</td>
<td>COMPLETE</td>
<td>RECOVERY</td>
<td>0</td>
<td>2012-03-26-07.24.16</td>
<td>RDAJLW3.JLW0002</td>
</tr>
<tr>
<td>2012-03-26-07.24.18</td>
<td>Arm0003</td>
<td>COMPLETE</td>
<td>RECOVER</td>
<td>0</td>
<td>2012-03-26-07.24.21</td>
<td>RDAJLW3.JLW0002</td>
</tr>
<tr>
<td>2012-03-26-07.24.27</td>
<td>Arm0007</td>
<td>COMPLETE</td>
<td>REBUILDX</td>
<td>0</td>
<td>2012-03-26-07.24.28</td>
<td>RDAJLW3.JLW0002</td>
</tr>
<tr>
<td>2012-03-26-07.24.35</td>
<td>Arm0012</td>
<td>COMPLETE</td>
<td>REPAIR</td>
<td>0</td>
<td>2012-03-26-07.24.38</td>
<td>RDAJLW3.JLW0002</td>
</tr>
<tr>
<td>2012-03-26-07.24.40</td>
<td>Arm0003</td>
<td>COMPLETE</td>
<td>RECOVER</td>
<td>0</td>
<td>2012-03-26-07.24.18</td>
<td>RDAJLW3.JLW0002</td>
</tr>
<tr>
<td>2012-03-26-07.24.28</td>
<td>Arm0007</td>
<td>COMPLETE</td>
<td>REBUILDX</td>
<td>0</td>
<td>2012-03-26-07.24.30</td>
<td>RDAJLW3.JLW0003</td>
</tr>
<tr>
<td>2012-03-26-07.24.36</td>
<td>Arm0012</td>
<td>COMPLETE</td>
<td>REPAIR</td>
<td>0</td>
<td>2012-03-26-07.24.39</td>
<td>RDAJLW3.JLW0003</td>
</tr>
<tr>
<td>2012-03-26-07.24.40</td>
<td>Arm0004</td>
<td>COMPLETE</td>
<td>RECOVER</td>
<td>0</td>
<td>2012-03-26-07.24.18</td>
<td>RDAJLW3.JLW0003</td>
</tr>
<tr>
<td>2012-03-26-07.24.25</td>
<td>Arm0007</td>
<td>COMPLETE</td>
<td>REBUILDX</td>
<td>0</td>
<td>2012-03-26-07.24.22</td>
<td>RDAJLW3.JLW0004</td>
</tr>
<tr>
<td>2012-03-26-07.24.29</td>
<td>Arm0012</td>
<td>COMPLETE</td>
<td>REPAIR</td>
<td>0</td>
<td>2012-03-26-07.24.32</td>
<td>RDAJLW3.JLW0004</td>
</tr>
<tr>
<td>2012-03-26-07.24.33</td>
<td>Arm0004</td>
<td>COMPLETE</td>
<td>RECOVER</td>
<td>0</td>
<td>2012-03-26-07.24.18</td>
<td>RDAJLW3.JLW0004</td>
</tr>
<tr>
<td>2012-03-25-15.23.44</td>
<td>Arm0005</td>
<td>COMPLETE</td>
<td>CHECKLPL</td>
<td>0</td>
<td>2012-03-26-07.25.24</td>
<td>RDAJLW3.JLW0005</td>
</tr>
<tr>
<td>2012-03-26-07.25.26</td>
<td>Arm0012</td>
<td>COMPLETE</td>
<td>REBUILDX</td>
<td>0</td>
<td>2012-03-26-07.25.28</td>
<td>RDAJLW3.JLW0005</td>
</tr>
<tr>
<td>2012-03-26-07.25.30</td>
<td>Arm0004</td>
<td>COMPLETE</td>
<td>RECOVER</td>
<td>0</td>
<td>2012-03-26-07.25.30</td>
<td>RDAJLW3.JLW0005</td>
</tr>
</tbody>
</table>

---

### Restoring system resource recovery (ARMBSRR) jobs

Restarting system resource recovery (ARMBSRR) jobs uses a synchronization file built by RECOVERY MANAGER.

**Overview of recovery for a set of ARMBSRR jobs**

The job cards for the jobs in the set must contain a symbolic variable that allows RECOVERY MANAGER to number the jobs. RECOVERY MANAGER uses numbers 0 through n, where n is the maximum number of jobs into which RECOVERY
MANAGER is able to split the recovery. RECOVERY MANAGER also imbeds synchronization steps within the JCL. These steps execute the ARMBSYN program, which updates and monitors the job synchronization file. The first job of the job set is Job 0, which allocates the synchronization file and then submits recover Jobs 1 - n to the internal reader. The first recover job (Job 1) submits an additional cleanup job that waits on all of the recover jobs to complete. If all jobs complete successfully, the synchronization file is deleted by this cleanup job.

**Rerun or restart?**

If any of the generated jobs fail, you must first determine what caused the failure and correct the situation. Then you should decide whether to rerun the entire job stream (by resubmitting the generated JCL) or restart the jobs at the point of failure. RECOVERY MANAGER provides an EDIT macro called ARMSBGEN to assist in restarting the failed jobs (see “Restarting synchronized jobs” on page 227). Sometimes it is quicker to resubmit the generated JCL than to identify step restarts for each recovery job.

**Restarting synchronized jobs**

To assist in restarting the jobs, an EDIT macro named ARMSBGEN has been supplied in the RMGR.DBCLIB library.

The macro separates the original JCL member into n + 1 members using the default prefix of JOB for the member names. Thus, JOB000 is the restart job used to reset the synchronization file and monitor job completion, and JOB001 - JOB00n are the separate recovery jobs generated by RECOVERY MANAGER.

---

**Note**

The RECOVERY MANAGER-generated JCL must reside in a partitioned data set (PDS) in order for you to use the ARMSBGEN macro.

---

If you choose to restart the jobs, copy the ARMSBGEN member from the .DBCLIB library to a library in your SYSPROC concatenation, then perform the following steps:

**To restart a synchronized job**

1. For each recover job (1 - n), locate the job output and note the step name in which the job got the first return code greater than 4. Some jobs may have completed successfully and will not need to be rerun.

2. If the original JCL is not in a PDS, copy it to a PDS and execute ARMSBGEN to separate the JCL into its component jobs.
- Edit the RECOVERY MANAGER-generated member using ISPF EDIT.

- Type ARMSBGEN prefix on the command line, where prefix is an optional 1 to 5 character prefix to use for the member name. If prefix is not specified, ARMSBGEN creates members with a prefix of JOB.

**Note**

After ARMSBGEN is complete, it CANCELS out of the ISPF EDIT, leaving the RECOVERY MANAGER-generated JCL intact.

3 Edit member JOB000 (or prefix000). ARMSBGEN has added the following card to the JCL directly after the job card:

```
// RESTART=ARM0002.ARMSYNC
```

Perform the following substeps:

a Place a comma at the end of your job card and verify the restart card is immediately after your job card. Do not change the step name on the RESTART card. The job is now ready to prepare the synchronization file for restart and clean up again at the end as needed.

b Submit this job before you submit other recover jobs. It waits on the other jobs to complete before it completes.

4 Edit member JOB001. ARMSBGEN has added the following card to the JCL directly after the job card:

```
// RESTART=ARM?????
```

Perform the following substeps:

a Place a comma at the end of your job card and ensure the restart card is located immediately after your job card.

b Change the ARM????? on the restart card to the step name of the first step that received a return code greater than 4, if any, in the original execution.

c If the failing step was DSNUTILB, you may need to add RESTART to the parameter list on the DSNUTILB execution.

For example, if the subsystem ID is DBDF, the parameter list should look like this:

```
EXEC PGM=DSNUTILB,PARM='DBDF,,RESTART'
```

**Note**

Adding the RESTART parameter is required only if DSNUTILB was started and placed in a STOPPED status at the time of the failure.
5 Repeat Step 4 on page 228 for each remaining job (JOB002 - JOB00n).

6 Submit member JOB000 if you have not already done so (see Step 3 on page 228).

7 Submit members JOB001 - JOB00 n that require restart due to previous failures.

—— Note ——
If for any reason you need to restart again, change the restart parameters as required in JOB001 - JOB00 n and resubmit JOB000 - JOB00 n.

Working with recovery options

RMGR enables you to select and change recovery options while preparing to recover the current group.

You can set recovery options (including post-recovery copy options) at the subsystem and group levels. Option validity varies according to your choice of recover utility and its version.

—— Tip ——
For performance reasons, you should use the ARMBGRP batch program to set options for groups containing more than a few hundred objects.

RECOVERY MANAGER provides a set of default recovery option values that apply to all groups, spaces, and indexes for all of the DB2 subsystems that are specified at installation. After installing RECOVERY MANAGER, you can change any of the recovery values for a particular subsystem through Subsystem options on the Main Menu. If you do not change values by group, the subsystem values will be used in the recovery JCL.

—— Note ——
RECOVERY MANAGER maintains a single set of subsystem options; therefore, the subsystem options apply to all members of a data sharing group.

If you change a value at the group level, that change overrides the corresponding subsystem value and is applied to all objects in the target group. See “RECOVERY MANAGER backup and recovery options” on page 94 for more information.

The Lvl column indicates the level at which a utility option is set for the particular group or object that you are viewing. Options can be set at the group level (G), system level (S) (recommended), or RECOVERY MANAGER default level (blank).
When you change an option on this panel, the level for that option changes to **S** or **G** (depending on whether you are modifying options for the system or a group).

You can remove an option that is set at the level that you are viewing. For example, if you are viewing a group, you can remove G-level options. To remove an option, enter a blank space in the option field. The option then reverts to its most recent setting. For example, removing a G-level setting causes the option to revert to either the system-level or RECOVERY MANAGER-level setting, depending on which was used most recently for that group.

**Maximizing concurrency of key sorts**

Two recovery options are available that can increase the speed of sorting keys during index rebuilds.

The **Unloadkeys/buildindex** option distributes the index keys for all indexes being rebuilt over number of sorting jobs that you specify and these jobs can then run in parallel.

---

**Note**

Unloadkeys/buildindex is not valid when running a simulated recovery (SIMULATE YES).

---

The **Maximum Key Sorts** (MAXKSORT) option (available with NGT Recover) distributes the index keys for all indexes being rebuilt over the number of sorts that you specify and these sorts can then run in parallel.

---

**Note**

If you specify Dynamic Sortworks=NO, RECOVERY MANAGER generates MAXKSORT=1, which essentially turns off the parallel processing of index rebuilds.

---

When choosing whether to use **Unloadkeys/buildindex** or **Maximum Key Sorts**, consider the following:

- Number of concurrent sorts
  
  In most instances, you should choose the option that provides the greater number of concurrent sorts.
  
  The concurrency of **Unloadkeys/buildindex** is limited by the number of available initiators in your system. This value is specified in the **Max concurrent jobs** option.
  
  The concurrency of **Maximum Key Sorts** is limited by the amount of memory available below the line for BMCSORT processing. In most environments, this creates a practical limit of 12 to 16 sorts.
Disparity of key length
If the key lengths of the indexes vary widely in size, **Maximum Key Sorts** can be more efficient than **Unloadkeys/buildindex**. **Unloadkeys/buildindex** expands the amount of memory allocated for all keys to the size of the largest key, whereas **Maximum Key Sorts** is able to allocate only what is needed.

Partitioned indexes
If the rebuild includes both partitioned and nonpartitioned indexes, **Maximum Key Sorts** can be more efficient than **Unloadkeys/Buildindex** because, if set to a value of 3 or greater, it can sort the partitioned indexes separately from the nonpartitioned.

Amount of key data
If you have more than 50G of key data, **Unloadkeys/Buildindex** can be more efficient than **Maximum Key Sorts**.

Restartability issues
Restart using **Maximum Keysorts** can cause keys already extracted and sorted to be extracted and sorted again, but the restart process is relatively straightforward. Restart using **Unloadkeys/Buildindex** avoids resorting previously extracted and sorted keys, but the restart process can be complex.

Displaying, updating, and deleting recovery options
You can browse, update, or delete the recovery options for the current group by selecting the **Group edit** option in the Object List panel for that group.

**Tip**
BMC recommends that you establish recovery option values for the subsystem before you specify recovery option values for a group.

Before you begin
To perform this procedure, you need the following authorizations (if you are not the creator of the group):

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- TYPE A authority if you intend to save the group
- TYPE O authority if you do not intend to save the group
To browse or update group recovery options

Start this procedure at the Object List panel, which is displayed after you have created or retrieved a group.

1 In the Object List panel, select Group Edit, and then press Enter.

2 Select Utility options and then press Enter.

3 In the Utility Options Specification panel, select Browse or Update and Recover, and then press Enter.

4 In the Recover Options Specification panel, select one of the following choices and press Enter to display those options:
   - General recovery options (see “General recovery options” on page 233 for option descriptions)
   - NGT Recover options (For option descriptions, see “NGT Recover options” on page 238)
   - DB2 Recover (DSNUTILB) options (for option descriptions, see “DB2 RECOVER options” on page 249)
   - Work File options (for option descriptions, see “Symbolic variables in post-recovery image copy data set names” on page 259)
   - Output data set options (for option descriptions, see “Output data set option descriptions” on page 254)

   **Note**
The options that are displayed reflect the options in effect for the group, including options set at the group level, as well as options that defaulted from the system or RECOVERY MANAGER default levels. To update an entry, type over the existing field.

5 Make changes as needed, then perform one of the following tasks:
   - To cancel your changes and return to the Recover Options Specification panel, press F3.
   - To accept your changes and return to the Recover Options Specification panel, press Enter.

To delete group recovery options

1 In the Object List panel, select Group Edit, and then press Enter.
2 Select **Utility options** and then press **Enter**.

3 In the Utility Options Specification panel, select **Delete** and then **Recover**, and then press **Enter**.

4 To delete all options set at the group level, select **Delete**. To exit without deleting options, select **Cancel delete**.

**Note**

After deletion, the appropriate recovery option values from other levels will be in effect for this group. For more information, see “**RECOVERY MANAGER backup and recovery options**” on page 94.

5 To return to the Group Edit panel, press **F3**.

---

## General recovery options

Fields on the General Recovery Options panel apply to all supported recovery utilities.

Those fields are listed alphabetically in Table 21 on page 233 along with the corresponding RECOVERY MANAGER defaults.

For additional information about any of the utility options, see the BMC **BMC Next Generation Technology Recover for DB2 for z/OS Reference Manual**, **BMC Next Generation Technology Copy for DB2 for z/OS Reference Manual**, **CHECK PLUS for DB2 Reference Manual**, or the IBM DB2 command and utility reference as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| Always rebuild indexes  | No                        | Specifies whether rebuild indexes from table data or to recover them from image copies and log data, as follows:  
**Yes**—rebuilds all indexes from table data, even if an image copy and log data are available.  
**No**—attempts to recover indexes from image copies and log data when possible. Any index that cannot be recovered is automatically rebuilt.  
**Note:** When used with BACKOUT YES or AUTO, RECOVERY MANAGER automatically resets this option to No and issues a warning message. |
<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allocate in KILOBYTES</td>
<td>No</td>
<td>Converts cylinder or track specifications to kilobytes when you redefine VCAT objects. Used in conjunction with <strong>Redefine VCAT objects</strong>.</td>
</tr>
<tr>
<td>Check pend action</td>
<td>None</td>
<td>Specifies the action to correct check pending status on recovered spaces after a point-in-time recovery, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ <strong>Check</strong>—run the specified check utility to correct check pending and auxiliary warning (AUXW) status</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ <strong>Repair</strong>—run the REPAIR utility to turn off check pending and AUXW status</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ <strong>None</strong>—no action</td>
</tr>
<tr>
<td>Check utility</td>
<td>DB2 CHECK</td>
<td>Specifies either the BMC CHECK PLUS utility or the IBM DB2 CHECK (DSNUTILB) utility for performing integrity checks on the current group or object</td>
</tr>
<tr>
<td>Copy after (post-recovery)</td>
<td>No (all types)</td>
<td>Specifies a copy type to make after the recovery is complete, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ <strong>LP</strong>—makes a primary image copy for the local site after the group or object has been recovered.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ <strong>LB</strong>—makes a backup image copy for the local site after a group or object has been recovered. If you select this option, you must also specify an <strong>LP</strong> copy.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ <strong>RP</strong>—makes a primary image copy for the recovery site after the group or object has been recovered.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ <strong>RB</strong>—makes a backup image copy for the recovery site after the group or object has been recovered. If you select this option, you must also specify an <strong>RP</strong> copy.</td>
</tr>
<tr>
<td>Copy utility</td>
<td>DB2 COPY (DSNUTILB)</td>
<td>Specifies either the BMC NGT Copy utility or the IBM DB2 COPY (DSNUTILB) utility for making post-recovery image copies immediately after the current group or object has been recovered</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>--------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Data set sizing</td>
<td>Catalog</td>
<td>Sizes objects when generating JCL, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>Catalog</strong>—use the DB2 and ICF catalog information for sizing purposes at the time of JCL generation. This method requires sizing calculations at the time of JCL generation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>Defaults</strong>—use existing default sizing information from the <strong>Work File</strong> options established in the <strong>Recovery</strong> options.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>BMCSTATS</strong>—use statistics from the BMCSTATS tables. Statistics in the BMCSTATS table are collected by DASD MANAGER PLUS and optionally by NGT Copy.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For more information on data set sizing see “Data set sizing” on page 80.</td>
</tr>
<tr>
<td>Delete STOGROUP objects</td>
<td>No</td>
<td>Deletes STOGROUP spaces before the recover utility executes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>To execute recovery JCL that includes this step, you must have DB2 STOP and DISPLAY authority and control authority on the physical data sets. Be aware of the following information:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- If you specify SITETYPE RECOVERY, the product forces the option <strong>Delete STOGROUP objects</strong> to <strong>Yes</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Specifying <strong>Delete STOGROUP objects</strong> to <strong>Yes</strong> causes the data set to be deleted if the object is STOGROUP-defined, regardless of the setting of the REUSE parameter.</td>
</tr>
<tr>
<td>Limit SYSCOPY search</td>
<td>0</td>
<td>Limits the number of days of records to search in the SYSIBM.SYSCOPY table when looking for a requested copy or quiesce point</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Limiting SYSIBM.SYSCOPY searches avoids unneeded I/O operations, excessive memory use, and improves performance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0—search all rows in the SYSCOPY catalog table</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 through 99—number of days of SYSCOPY entries to include in the search.</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>--------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Max concurrent jobs| 1                        | Specifies the maximum number of concurrent jobs to process the current group.  
**WARNING:** Do not specify a value that is higher than the available number of initiators. Doing so could cause an unending wait situation. |
| Mirroring          | No                       | Specifies that the objects in the group are mirrored and that the mirrors are considered to be valid backups at the disaster recovery site.  
In order to use this option, you must have set at least mirroring Level 2 (BSDS, active logs, and catalog and directory) in the subsystem-level General Recovery Options.  
This option is only available if you are setting recovery options for a group and using the BMC Recovery Management for DB2 solution. |
| Recover utility    | RECOVERY MANAGER only:   | Specifies either NGT Recover or DB2 RECOVER (DSNUTILB) to recover the current group or object.  
**Note:** You must select NGT Recover to use recovery simulation. |
<p>|                    | DSNUTILB                 | |
|                    | Recovery Management:     | |
|                    | NGT Recover              | |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| Redefine VCAT objects       | No                       | Includes IDCAMS delete and define steps for VCAT spaces before the recover utility executes  
Be aware of the following information:  
- To use this option, you must have DB2 STOP and DISPLAY authority and control authority on the physical data sets.  
- If you specify SITETYPE RECOVERY, the product forces the option Redefine VCAT objects to Yes.  
- If you specify RSITEDELDEF=NO in the option set and SITETYPE = RECOVERY and JCLTYPE = LOCAL in the options statement, the product does not generate the IDCAMS delete and define steps for VCAT or STOGROUP objects. For more information, see “Option sets and configuration options” on page 733.  
- RECOVERY MANAGER cannot perform delete and define steps for VCAT-defined spaces that do not have ICF catalog data (for example, objects that have been deleted or migrated). For those objects, RECOVERY MANAGER performs one of the following actions:  
  - For user-defined VCAT objects, ARMBGEN issues a warning message and generates JCL with the delete and define steps commented out. You can manually retrieve the object, then alter the JCL to include the delete and define statements.  
  - For system VCAT objects (such as the DB2 catalog and directory, the BMC Common DB2 repository, the RECOVERY MANAGER repository, and the CHANGE ACCUM repository), ARMBSRR issues an error message and fails. |
| Region Size                 | 0 MB                     | Specifies the amount of virtual storage used by the recover utility. The default value is 0 MB. The valid range is -1 through 2047 MB.  
A value of -1 specifies that RECOVERY MANAGER will not generate region size at the step level.  
**Note:** For best performance, BMC recommends a region size of 0 MB, in which case the amount of virtual storage needed to run the job is automatically made available when the recover utility runs. Some data centers do not allow a region size of 0 MB. A typical NGT Recover step requires between 5 MB and 8 MB of virtual storage for code, control blocks, and I/O buffers. |
NGT Recover options

You can set recover options that are specific to NGT Recover on the NGT Recover Options panel.

The options are listed alphabetically in Table 22 on page 239 along with the RECOVERY MANAGER defaults. For more detailed information about the NGT Recover utility options that you can use with RECOVERY MANAGER, see the RECOVERY MANAGER for DB2.
If configuration option value is indicated as the default value, RECOVERY MANAGER does not generate the keyword in the JCL. This enables the NGT Recover configuration option value to be used.

Table 22: NGT Recover recovery option fields

<table>
<thead>
<tr>
<th>Field/NGT Recover keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alternate resources</td>
<td>No</td>
<td>Enables you to specify the resources and the order in which they will be used in a recovery, as follows: Yes —select from alternate resources, such as recovery site image copies, local backup image copies, or DSNUTILB FlashCopy image copies. No —use only the normally available recovery resources. Auto — job optimization uses both archive copies one and two. This option enables NGT Recover jobs to run two at a time. You can specify which copies of the following to use: ▪ Image copies ▪ Logs ▪ Copies of the change accumulation files. For more information, see “Alternate recovery resource options” on page 248.</td>
</tr>
<tr>
<td>AUTOSIZE</td>
<td>YES</td>
<td>Turns dynamic sizing for output image copies or change accumulation output files on or off: Yes —specifies dynamic sizing for output image copies or change accumulation output files allocated to DASD. No —specifies that output image copies or change accumulation output files are allocated to DASD using the primary and secondary quantities that are specified in the R+/CHANGE ACCUM repository.</td>
</tr>
<tr>
<td>Field/NGT Recover keyword</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| CHECKPOINT               | Configuration option value | Controls the overhead that is associated with taking checkpoints, as follows:  
  - **No**—causes no checkpoints to be taken, except those necessary to synchronize NGT Recover execution with the execution of other BMC utilities and the MERGE checkpoints that are necessary to guarantee the integrity of output copy registration. This option is recommended for short NGT Recover jobs in which you do not want to incur checkpoint overhead and which you do not mind rerunning if necessary.  
  - **Sync**—causes a checkpoint to be taken at the end of each processing phase and also at the completion of each log data set in the LOG APPLY phase. This allows either phase restart or sync restart in the LOG APPLY phase. Specify this option for recoveries that require the reading of many log data sets using the RESTORE/LOGAPPLY, LOGAPPLY ONLY, or LOGONLY strategies.  
  - **Phase**—causes a checkpoint to be taken at the end of each processing phase if a set amount of time has passed. Choose this option for longer jobs when it would be costly to rerun the entire job. |
| Diagnostic messages      | NO            | Provides diagnostic messages regarding the sort functions |
|                         |               | Specifies the type of sort work allocation, as follows:  
  - **Yes**—dynamically allocate sort works  
  - **No**—allocate sort works using DD statements  
  You can limit the allocation of the sort work file by specifying the **Max primary allocation** (see “Output data set option descriptions” on page 254). |
<p>| EARLYCAT                 | YES           | Verifies (during the ANALYZE phase) that all cataloged data sets that are required for recovery exist in the operating system catalog |</p>
<table>
<thead>
<tr>
<th>Field/NGT Recover keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
</table>
| EARLYRECALL               | YES           | Specifies the early retrieval (during the ANALYZE phase) of any archived image copies and log data sets that are required during recovery.  
**Note:** If you are using the Recovery Management solution and generating estimation JCL (ESTIMATE YES), RECOVERY MANAGER always converts the value of EARLYRECALL to NO. |
| KSORTSHARE                | YES           | Specifies if key sorts are shared among NGT Recover table space recoveries (MERGE phases) running in parallel.  
- **YES**—uses up to the value specified for MAXKSORT active key sorts at any given time. If sufficient key sorts are not available when a table space recovery begins execution, keys will be obtained later by an UNLOAD phase.  
- **NO**—each MERGE phase has its own set of key sorts and up to MAXKSORT * MAXLSORT key sorts can be active at any given time. Since the number of sorts that can be active in a system is fairly small - usually no more than 30 - a value of NO for this option may severely limit the number of recovery operations that NGT Recover can perform in parallel when index rebuilds are also requested. |
<table>
<thead>
<tr>
<th>Field/NGT Recover keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
</table>
| LOGSCAN                   | No            | Scans the log and provides a report on the number and size of log records required for recovery. This option includes only the NGT Recover LOGSCAN JCL in the job. No recovery JCL is generated, and no STOP, DELETE/DEFINE, START, or REPAIR steps are included in the JCL. The generated jobs contains a comment message stating that the LOGSCAN option was selected and no other job steps were created. Be aware of the following information:  
  - **LOGSCAN Yes** and the recovery simulation feature are mutually exclusive. Objects with **LOGSCAN Yes** are excluded from simulation.  
  - A recovery to a copy overrides the LOGSCAN option and creates a normal recover job.  
  - If you specify **Yes**, you cannot perform a Backout Auto recovery for the group. RECOVERY MANAGER changes **Backout Auto** to **Backout No** and issues a warning message. |
### Maximum Key Sorts

#### Keyword: MAXKSORT

<table>
<thead>
<tr>
<th>Field/NGT Recover keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum Key Sorts</td>
<td>Configuration option value</td>
<td>Specifies the maximum number of index key sorts that can be run concurrently. Valid values are from 1 to 999. The default is blank, which causes RECOVERY MANAGER to use the value set in the NGT Recover options. The NGT Recover default is two times the number of CPUs. For each table space, index keys for all indexes being rebuilt are distributed over the number of sorts that you specify for this option and these sorts can then run in parallel. For a partitioned table space, if the partitioning index is being rebuilt, the rebuild of each partition is done at the completion of the MERGE or UNLOAD for each partition of the table space. (The rebuild can run concurrently with the MERGE or UNLOAD for the next partition if the MAXKSORT number is not exceeded). Running concurrent index key sorts can increase the speed of the recovery. See “Maximizing concurrency of key sorts” on page 230 for more information. Be aware of the following information:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Maximum Key Sorts overrides the WORKDDN value that is specified in the Work File Options panel.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Maximum Key Sorts and Unloadkeys/Buildindex are mutually exclusive.</td>
</tr>
</tbody>
</table>

### MAXLOGS

#### Keyword: MAXLOGS

<table>
<thead>
<tr>
<th>Field/NGT Recover keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAXLOGS</td>
<td>Configuration option value</td>
<td>Specifies the maximum number of log files that RECOVERY MANAGER allocates at the same time during a log input phase MAXLOGS provides a way to limit the number of tape log files that are read at the same time. The RECOVERY MANAGER default is 0 (zero), which causes the option to default to the NGT Recover configuration option value.</td>
</tr>
<tr>
<td>Field/NGT Recover keyword</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>MAXLSORT</strong></td>
<td>0</td>
<td>Specifies the maximum number of log sorts that can run concurrently and also determines the number of MERGE/RESTORE/SNAP phases that can run in parallel, whether or not log records are processed. You use MAXLSORT to improve recovery performance. Valid values are 0 to 999. When MAXLSORT=0, the default value is determined by NGT Recover. BMC suggests values from 10 to 12 for MAXLSORT. When you specify MAXLSORT, the following files are dynamically allocated if you do not code them in JCL: LOGOU <em>nnn</em>: sort message files <em>nnn</em> is the number of the log sort and is a number between 1 and the value that is specified for MAXLSORT. L <em>xxx</em>WK <em>nn</em>: sort work files <em>xxx</em> is the number of the log sort and is a number between 1 and the value that is specified for MAXLSORT. <em>nn</em> is the number of the work data set. For example, if MAXLSORT=3 and two sort work files are required for each sort, the DDs would be specified as follows: L001WK01 DD... L001WK02 DD... L002WK01 DD... L002WK02 DD... L003WK01 DD... L003WK02 DD... When you use dynamic allocation for these files, NGT Recover determines the optimal number of files to use.</td>
</tr>
<tr>
<td><strong>MAXPRIM</strong></td>
<td>0</td>
<td>Sets a maximum amount of disk space (in the units specified by SPACE) to allocate as primary space. Valid values are 0 through 65535. A nonzero value establishes an upper limit for primary space allocation; 0 specifies no limit.</td>
</tr>
</tbody>
</table>
MSGLEVEL
Keyword: MSGLEVEL

<table>
<thead>
<tr>
<th>Field/NGT Recover keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSGLEVEL</td>
<td>1</td>
<td>Specifies the output files and messages NGT Recover returns, as follows:</td>
</tr>
</tbody>
</table>

- **2—Plan Summary**
  - AFRPRINT - execution messages
  - AFRSUMRY - maintenance applied, phases completed, utility return codes
  - AFRSTMT - input statements and options as specified in SYSIN, configuration option values, and log file resources
  - AFRTIME - reports the ten table spaces that took the longest amount of elapsed time to recover (available only with the BMC Recovery Management for DB2 solution)
  - AFROSUM - object summary for objects being recovered
  - AFRPLAN - execution plan

- **1—Object Summary**
  - AFRPRINT
  - AFRSUMRY
  - AFRSTMT
  - AFRTIME
  - AFROSUM

- **0—Standard**
  - AFRPRINT
  - AFRSUMRY
  - AFRSTMT
  - AFRTIME
## ON ERROR CONTINUE

**Keyword:**

**ON ERROR CONTINUE**

<table>
<thead>
<tr>
<th>Field/NGT Recover keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ON ERROR CONTINUE</td>
<td>10</td>
<td>Use this option to determine how NGT Recover is to proceed when errors are encountered. The default value is 10. The valid range is 0 to 2,147,483,646. ON ERROR CONTINUE nnnnnnnnnnn allows nnnnnnnnnnn + 1 errors before NGT Recover terminates. If nnnnnnnnnnn is 0, NGT Recover stops processing immediately when the first recognized severe error occurs. If you specify ON ERROR CONTINUE 0, the subtask to preallocate VSAM data sets is disabled, which could increase the execution time by several seconds for each object recovered. <strong>Note:</strong> If you are using the Recovery Management solution and specify BACKOUT AUTO, the number of errors allowed is not limited.</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>Field/NGT Recover keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OUTCOPY by Recover Keyword: OUTCOPY</td>
<td>Configuration option value</td>
<td>Uses the NGT Recover OUTCOPY function to make updated image copies from existing recovery resources such as prior image copies, change accumulation files, and DB2 logs. OUTCOPY writes the output to a sequential image copy data set instead of a DB2 space, which enables you to make copies without accessing the DB2 space or interfering with normal DB2 access in any way. Specify how to make the output copies for partitioned table spaces after a successful recovery, as follows:</td>
</tr>
<tr>
<td>■ ASCODED—makes copies with the same DSNUM designation that is used for the recovery. For example, if DSNUM ALL is used for recovery, the copies are made by table space. If DSNUM n is used for recovery (n&gt;0), the copy is for partition n.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ BYPART—makes all copies of partitioned table spaces by partition whether or not the recovery is by table space (DSNUM ALL).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ NO—uses the specified copy utility instead of OUTCOPY. Copies are made with the same DSNUM designation as is used for recovery.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUTCOPY is invalid in the following situations:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ If you specify LOGSORT No (see “Preparing recovery jobs” on page 199)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ If you specify BACKOUT AUTO (see “BACKOUT recovery” on page 201)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ If you specify SIMULATE YES (“Recovery simulation for application spaces” on page 206). The syntax is generated for OUTCOPY, but the steps are bypassed and no copies are made.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>■ If you use compressed indexes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Unloadkeys/Buildindex

**Keyword:** UNLOADKEYS BUILDINDEX

- **Default value:** No
- **Description:** Facilitates the rebuild of large nonpartitioned indexes on partitioned table spaces by providing concurrency in the extraction of keys from multiple partitions. Using this option can dramatically reduce the elapsed time required to rebuild a nonpartitioned index.

Be aware of the following information:

- **Unloadkeys/Buildindex** should be used in conjunction with **Max concurrent jobs**, which should have a setting greater than 1. If you choose INDEX ALL in the general recovery options, then RECOVERY MANAGER does not generate the Unloadkeys/Buildindex syntax.

- **Unloadkeys/Buildindex Yes** and the recovery simulation feature are mutually exclusive. Objects using Unloadkeys/Buildindex Yes are excluded from simulation.

- **Unloadkeys/Buildindex** and **Maximum Keys Sorts** (MAXKSORT) option are mutually exclusive. MAXKSORT is available with NGT Recover version 4.1 or later and enables multiple key sorts to run concurrently. For more information, see “Maximizing concurrency of key sorts” on page 230 for more information.

- Unloadkeys/Buildindex cannot be used with compressed indexes.

## XBMID

**Keyword:** XBMID

- **Configuration option value:**
- **Description:** Specifies the 1-8 character ID of the EXTENDED BUFFER MANAGER (XBM) subsystem that is required for use with Instant Snapshot copies.

---

### Alternate recovery resource options

When you use the OUTCOPY YES feature of RECOVERY MANAGER to make backups after a recovery, you can select which resources (image copies, logs, and change accumulation files) should be used as a basis for the new copies.

The order in which you rank these alternate resources determines which resource is used. If the first choice is not available, RECOVERY MANAGER falls back to the
second choice. If you set a choice to zero, RECOVERY MANAGER does not use the resource.

**Note**

If you do not specify any order for the alternate resources, RECOVERY MANAGER uses the RECOVERY MANAGER default values. For more information, see the *RECOVERY MANAGER for DB2 User Guide* and the *R+/CHANGE ACCUM for DB2 User Guide*.

The Alternate Resource Selection panel lets you specify the recovery resources to be used in the recovery and the order in which they should be used.

For example, if your practices include taking a local site backup image copy and a secondary copy of the archive log to a recovery site (instead of offsite copies), you can select those resources when you use this procedure at the recovery site.

**Note**

If you select the RP or RB copy as your first choice for the image copy, RECOVERY MANAGER considers the site type to be RECOVER. This value overrides any other site type setting you make, including the site type specified in ARMBGEN and ARMBGPV syntax.

---

**DB2 RECOVER options**

You can set recover options that are specific to DB2 RECOVER (DSNUTILB) on the DSNUTILB Options panel.
The fields are listed alphabetically in the following table along with the RECOVERY MANAGER defaults.

Table 23: DB2 RECOVER (DSNUTILB) recovery option fields

<table>
<thead>
<tr>
<th>Field/DSNUTILB keyword</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSNUTILB sitetype</td>
<td>Local</td>
<td>Specifies whether to recover the local site image copy (DSNUTILB keyword LOCALSITE) or the remote site image copy (DSNUTILB keyword RECOVERYSITE). <strong>Note</strong>: BMC recommends that you not set this option unless you always intend to use the same copy type. An entry in this field overrides any other site type setting you make, including any in the ARMBGEN and ARMBGPV syntax. BMC recommends instead that you use the Site type field on the Main Menu for online generation or the OPTIONS SITETYPE command when using the batch process.</td>
</tr>
<tr>
<td>SORTKEYS</td>
<td>No</td>
<td>Specifies that index keys are sorted in parallel with the reload and build phases to improve performance. BMC recommends using this option if you need to create more than one index. Any WORKDDN specifications are ignored when you specify <strong>Yes</strong>.</td>
</tr>
<tr>
<td>STATISTICS</td>
<td>No</td>
<td>Gathers index statistics from the DB2 catalog. If STATISTICS is set to <strong>No</strong>, the values in REPORT, UPDATE, and KEYCARD are not generated in the JCL.</td>
</tr>
<tr>
<td>REPORT</td>
<td>Yes</td>
<td>Prints the statistics collected (the ACCESSPATH and SPACE statistics reports). Used in conjunction with <strong>STATISTICS Yes</strong>.</td>
</tr>
<tr>
<td>UPDATE</td>
<td>NONE</td>
<td>Updates the DB2 catalog tables with the collected statistics, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>NONE</strong>—makes no update to catalog tables. This is valid only when you specify <strong>REPORT Yes</strong>.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>ALL</strong>—inserts all collected statistics in the DB2 catalog tables.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>ACCESSPATH</strong>—updates only those columns used for access path selection.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>SPACE</strong>—updates only those columns that provide statistics about the status of the target indexes.</td>
</tr>
</tbody>
</table>
Work file option descriptions

You can specify sort and work file options that might be required when extracted keys are sorted during index rebuilds or when using the CHECK DATA and CHECK INDEX utilities.

This information is used for the following purposes:

- As the data set sizing default (primarily allocation values) when RECOVERY MANAGER cannot determine data set sizes using information from the DB2 or ICF catalog
- When generating the Log Master (ALPMAIN) step during a disaster recovery (only available with the BMC Recovery Management for DB2 solution)

**Note**

If you are using the BMC Recovery Management for DB2 solution, the Log Master step generates the file that contains the SQL statements that you use to return the data collection information from the recovery site to the local site. (For more information, see the *Recovery Management for DB2 User Guide*.)

The work file fields are listed alphabetically in Table 24 on page 251 along with the corresponding RECOVERY MANAGER defaults.

**Table 24: Work file field descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allocation type</td>
<td>Cylinder</td>
<td>Specifies whether the primary and secondary allocation quantities are expressed in cylinders or tracks</td>
</tr>
<tr>
<td>Field</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Max primary allocation</td>
<td>0</td>
<td>Limits the amount of primary allocation space to be used for sort work space. Valid values are 0 - 9999, where 0 indicates no limit.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>WARNING:</strong> Ensure that you use a maximum primary value that fits on your DASD devices. If the maximum primary value exceeds the capacity of a volume, the job will fail. See Table 25 on page 253 for capacity guidelines.</td>
</tr>
<tr>
<td>Max primary allocation affects the following values:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>▪ Primary Allocation, Dynamic Sortworks, and WORKDDN Sizing</td>
<td></td>
<td>RECOVERY MANAGER compares the maximum primary allocation to the calculated primary allocation value and selects the smaller of the two. If the maximum primary value is selected, the primary space allocation is set to that value, and the secondary space allocation is set to 1/15 of that value. The same calculations are used for the sort work files.</td>
</tr>
<tr>
<td>▪ Unit Count (for DSNUTILB COPY)</td>
<td></td>
<td>The unit count for DSNUTILB COPY is based on the maximum primary value. If the space required for the copy is greater than the maximum primary allocation, RECOVERY MANAGER calculates the required unit count. The unit parameter is UNIT=(work_unit,n) where n is the calculated number of units up to a maximum of 59. If the maximum primary value is 0, no unit count is calculated. (See Table 14 on page 170.)</td>
</tr>
<tr>
<td>Primary allocation</td>
<td>10</td>
<td>Specifies the primary allocation quantity when RECOVERY MANAGER is unable to estimate the quantity due to problems or when the Data Set Sizing option in the Recovery Options Specification panel is set to Default. You can limit the primary allocation value by specifying the Max primary allocation.</td>
</tr>
<tr>
<td>Secondary allocation</td>
<td>20</td>
<td>Specifies the secondary allocation quantity when RECOVERY MANAGER is unable to estimate the quantity due to problems or when the Data Set Sizing option in the Recovery Options Specification panel is set to Default.</td>
</tr>
<tr>
<td>Field</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Work prefix</td>
<td>Not specified</td>
<td>Specifies the high-level name to use when allocating sort and work files. &lt;ul&gt;&lt;li&gt;If you specify a work prefix in the group options, RECOVERY MANAGER uses that work prefix.&lt;/li&gt;&lt;li&gt;If you do not specify work prefix in the group options, RECOVERY MANAGER uses the value of Work File Prefix in ARMSOPTS (see page 755).&lt;/li&gt;&lt;li&gt;If WORKPREFIX has no value in ARMSOPTS, RECOVERY MANAGER uses the user ID.&lt;/li&gt;&lt;/ul&gt;</td>
</tr>
<tr>
<td>Work unit</td>
<td>SYSALLDA</td>
<td>Specifies a disk file for use when dynamically allocating work files. &lt;p&gt;&lt;strong&gt;Tip:&lt;/strong&gt; To determine the size of the work file, run DSN1LOGP with SUMMARY(ONLY) and allocate the output to a disk file.&lt;/p&gt;</td>
</tr>
<tr>
<td>WORKDDN</td>
<td>NO</td>
<td>Directs the sorting of extracted index keys in index recoveries. &lt;ul&gt;&lt;li&gt;NO (the default) sorts the keys without writing them to SYSUT1 and omits that file from the recovery JCL.&lt;/li&gt;&lt;li&gt;YES writes extracted index keys to a work file (SYSUT1) before sorting.&lt;/li&gt;&lt;li&gt;You can limit the amount of space allocated for the work file by specifying the Max primary allocation. RECOVERY MANAGER ignores this option if you specify a value greater than 1 for Maximum Key Sorts (MAXKSORT).&lt;/li&gt;&lt;/ul&gt;</td>
</tr>
</tbody>
</table>

You must ensure that the value that you specify for Maximum primary value fits on your DASD devices. If the maximum primary value exceeds the capacity of a volume, the job will fail. Table 25 on page 253 gives examples of the capacity of some typical DASD devices:

**Table 25: Capacity of typical DASD devices**

<table>
<thead>
<tr>
<th>Physical data for 3380 (per device)</th>
<th>Physical data for 3390 (per device)</th>
<th>Physical data for 9345 (per device)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Density (Models D &amp; J)</td>
<td>Tracks: 3,275 Cyls: 885</td>
<td>Model 1 (Single) Tracks: 16,695 Cyls: 1,113</td>
</tr>
<tr>
<td>Double Density (Model E)</td>
<td>Tracks: 26,550 Cyls: 1,770</td>
<td>Model 2 (Double) Tracks: 33,390 Cyls: 2,226</td>
</tr>
<tr>
<td></td>
<td>Model 3</td>
<td>Model 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tracks: 21,600 Cyls: 1,440</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tracks: 32,340 Cyls: 2,156</td>
</tr>
</tbody>
</table>
### Physical data for 3380 (per device)

- **Triple Density (Model K)**
  - Tracks: 39,825
  - Cyls: 2,655

- **Not applicable**

### Physical data for 3390 (per device)

- **Triple**
  - Tracks: 50,085
  - Cyls: 3,339

- **Not applicable**

### Physical data for 9345 (per device)

- **Not applicable**

### Output data set option descriptions

You can specify the parameters required for making image copies immediately after a group has been recovered.

The choices are presented once for each copy type (local site primary, local site backup, recovery site primary, recovery site backup).

The fields are listed alphabetically in Table 24 on page 251 along with the corresponding RECOVERY MANAGER defaults.

#### Table 26: Output data set option fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allocation type</td>
<td>Cylinder</td>
<td>Specifies whether the primary and secondary allocations quantities are expressed in cylinders or tracks. This option does not apply to tape units.</td>
</tr>
<tr>
<td>Catalog</td>
<td>Yes</td>
<td>Catalogs the data sets in the operating system catalog</td>
</tr>
<tr>
<td>Field</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>----------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Data set name                 | &USERID.&DB.&TS.&TYPE&DATE.T&TIME &USERID.&DB.&TS.&TYPE&DATE | Specifies the name of the disk or tape data set for the current copy type. You can use symbolic variables to construct this name (see “System resources” on page 265). Generation data groups are not allowed. The data set specification is in two parts. The first part is applicable to all recover utilities. The second part is used only when NGT Recover is the recovery utility. The specified data set name is used as a prefix to which is appended the partition number in the form Ann (the number at the end of the data set name in the VSAM catalog).  
**Note:** The data set specified in the NGT Recover by part field is only used when making copies by partition after a recovery using the NGT Recover OUTCOPY feature when the group is defined as DSNUM=0. All other output is sent to the data set specified in the Data set name field. |
| EATTR for EAV                  | Not specified                                | Specifies whether a data set supports extended attributes or not. Specifying no value for EATTR allows the value for EATTR to be set by an SMS DATACLAS. Valid values are:  
- **OPT**—specifies that extended attributes are optional for the data set.  
You must set OPT to allocate an extended format sequential data set. By using OPT, NGT Recover supports sequential data sets in the cylinder-managed portion of EAVs. Extended format sequential data sets must be allocated on SMS-managed volumes and the size of the data set must be greater than the EAV break point, which is typically 10 cylinders.  
- **NO**—specifies that the data set cannot have extended attributes. |
<table>
<thead>
<tr>
<th>Field</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expiration date</td>
<td>1999/000</td>
<td>Specifies the expiration date for a tape copy data set. The date must be in the format <code>yyyy/ddd</code>. The RECOVERY MANAGER default is 1999/000, which indicates no expiration.</td>
</tr>
<tr>
<td>Max primary allocation</td>
<td>0</td>
<td>Limits the amount of primary allocation space to be used for the output copy data sets made to DASD using DSNUTILB COPY or NGT Copy. Valid values are 0 - 9999, where 0 indicates no limit. RECOVERY MANAGER compares the maximum primary allocation to the calculated primary allocation value and selects the smaller of the two. If the maximum primary value is selected, the primary space allocation is set to that value and the secondary space allocation is set to 1/15 of that value.</td>
</tr>
<tr>
<td>Field</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>---------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Max primary allocation (continued)</td>
<td></td>
<td>Max primary allocation affects the following values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Primary Allocation, Dynamic Sortworks, and WORKDDN Sizing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NGT Recover compares the maximum primary allocation to the calculated primary allocation value and selects the smaller of the two. If the maximum primary value is selected, the primary space allocation is set to that value, and the secondary space allocation is set to 1/15 of that value. The same calculations are used for the sort work files.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Unit Count (for DSNUTILB COPY)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The unit count for DSNUTILB COPY is based on the maximum primary value. If the space required for the copy is greater than the maximum primary allocation, RECOVERY MANAGER calculates the required unit count. The unit parameter is UNIT=(workUnit,n) where n is the calculated number of units up to a maximum of 59. If the maximum primary value is 0, no unit count is calculated. (See “NGT Copy backup options” on page 170.)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You must ensure that you use a maximum primary value that fits on your DASD devices. If the maximum primary value exceeds the capacity of a volume, the job will fail. See “Work file option descriptions” on page 251 for the capacity of typical DASD devices.</td>
</tr>
<tr>
<td>Model data set name (DCB)</td>
<td>None</td>
<td>Specifies the fully qualified name of a cataloged data set to define the model data control block (DCB)</td>
</tr>
<tr>
<td>Field</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Primary allocation    | 10      | Specifies the primary allocation quantity (disk only)  
Use this option only when RECOVERY MANAGER is unable to estimate the quantity or when the **Data Set Sizing** option is set to **Default**.  
**Note**: This value is used when you make copies using DSNUTILB COPY or when you use NGT Copy to make CABINET copies to disk. It is ignored when you use NGT Copy to make any other type of copy because NGT Copy performs its own data set sizing. |
| Retention period      | 0       | Specifies the tape copy data set retention period in days  
The valid range is 1 through 999. **Retention period** and **Expiration date** are mutually exclusive.                                                                                                                                                                                                                                                                                      |
| Secondary allocation  | 20      | Specifies the secondary allocation quantity (disk only)  
Use this option only when RECOVERY MANAGER is unable to estimate the quantity or when the **Data Set Sizing** option is set to **Default**.  
**Note**: This value is used when you make copies using DSNUTILB COPY or when you use NGT Copy to make CABINET copies to disk. It is ignored when you use NGT Copy to make any other type of copy because NGT Copy performs its own data set sizing. |
| SMS data class        | Null    | Specifies a valid SMS data class name for disk data sets  
The name must not exceed 8 characters. **RECOVERY MANAGER** forces **Catalog Yes** when this option is specified.                                                                                                                                                                                                                                                                                     |
| SMS management class  | Null    | Specifies a valid SMS management class name for disk data sets  
The name must not exceed 8 characters. **RECOVERY MANAGER** forces **Catalog Yes** when this option is specified.                                                                                                                                                                                                                                                                                     |
| SMS storage class     | Null    | Specifies a valid SMS storage class name for disk data sets  
The name must not exceed 8 characters. **RECOVERY MANAGER** forces **Catalog Yes** when this option is specified.                                                                                                                                                                                                                                                                                     |
### Stack (copies on tape)

- **Default**: No
- **Description**: Specifies whether to stack image copies of the same type contiguously on the same tape. Used in conjunction with the `Tape` option. For Recovery Management solution only - you can specify `Cabinet` to create cabinet copies. Cabinet copies can be made to either disk or tape. For more information, see the *Recovery Management for DB2 User Guide*.

### Tape

- **Default**: No
- **Description**: Specifies that the unit is a tape when you respond `Yes` and when you also provide the name of the tape unit at the `Unit` prompt. Responding `No` specifies a disk. **Note**: If you provide a unit name and leave `Tape` blank, you will receive an INVALID COMBINATION message.

### Unit

- **Default**: SYSALLDA
- **Description**: Specifies the name of the disk or tape unit to which the image copy data sets will be written.

### Vol count

- **Default**: 0
- **Description**: Specifies the largest number of tape volumes that are expected to be created. The valid range is 1 through 255. This option applies only to tape data sets. You can leave this field blank if you expect not more than five tape volumes will be created.

---

### Symbolic variables in post-recovery image copy data set names

With RECOVERY MANAGER, you can request that image copies of the table spaces that you recover be made immediately after the recovery completes.

If you supply data set names instead of accepting RECOVERY MANAGER defaults, you can construct those names using the symbolic variables that are shown in Table 24 on page 251.

**Table 27: Symbolic variables for post-recovery copy data set names**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;ATTACH</td>
<td>Data sharing group name</td>
</tr>
<tr>
<td>Symbol</td>
<td>Result</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>&amp;DATE</td>
<td>Current date (in ( yyddd ) format)</td>
</tr>
<tr>
<td>&amp;DAY</td>
<td>Current day (in ( dd ) format)</td>
</tr>
<tr>
<td>&amp;DB</td>
<td>Name of the database containing the space that is being copied</td>
</tr>
<tr>
<td>&amp;DSNUM, &amp;PART</td>
<td>Number of the data set or partition being copied</td>
</tr>
<tr>
<td>&amp;HOUR</td>
<td>Current hour (in ( hh ) format)</td>
</tr>
<tr>
<td>&amp;ICTYPE</td>
<td>Type of image copy</td>
</tr>
<tr>
<td>&amp;JDATE</td>
<td>Current date (in ( yyddd ) Julian date format)</td>
</tr>
<tr>
<td>&amp;JDAY</td>
<td>Current day (in ( ddd ) Julian format)</td>
</tr>
<tr>
<td>&amp;MINUTE or &amp;MIN</td>
<td>Current minute (in ( mm ) format)</td>
</tr>
<tr>
<td>&amp;MONTH</td>
<td>Current month (in ( mm ) format)</td>
</tr>
<tr>
<td>&amp;SECOND or &amp;SEC</td>
<td>Current second (in ( ss ) format)</td>
</tr>
<tr>
<td>&amp;SEQ</td>
<td>(NGT Copy only) The sequence number that increments with each reference. The sequence number restarts at 1 for each job step and is used to provide unique output data set names.</td>
</tr>
<tr>
<td>&amp;SSID</td>
<td>ID of this DB2 subsystem</td>
</tr>
<tr>
<td>&amp;TIME</td>
<td>Current time (in ( hhmmss ) format)</td>
</tr>
<tr>
<td>&amp;TS</td>
<td>Name of the table space that is being copied</td>
</tr>
<tr>
<td>&amp;TYPE</td>
<td>Type of output that is being produced</td>
</tr>
<tr>
<td></td>
<td>■ LP—Local site primary copy</td>
</tr>
<tr>
<td></td>
<td>■ LB—Local site backup copy</td>
</tr>
<tr>
<td></td>
<td>■ RP—Remote site primary copy</td>
</tr>
<tr>
<td></td>
<td>■ RB—Remote site backup copy</td>
</tr>
<tr>
<td>&amp;UNIQ or &amp;UQ</td>
<td>1- to 8-character value, based on the system clock, that is used to generate unique copy data set names</td>
</tr>
<tr>
<td></td>
<td>The first character is always an uppercase letter. Each remaining character is either an uppercase letter or a numeral from 0 through 9.</td>
</tr>
<tr>
<td>&amp;USERID, &amp;USER, &amp;UID</td>
<td>TSO user ID</td>
</tr>
<tr>
<td>&amp;YEAR</td>
<td>Current year (in ( yy ) format)</td>
</tr>
</tbody>
</table>

a The maximum total length that is allowed for a data set name is 44 bytes.

b Symbols with a numeric result must be prefixed by one or more alphabetic characters.
Exporting and migrating groups

RECOVERY MANAGER enables you to convert group definitions into CREATE GROUP syntax and write them to a specified data set. You can then either keep the data set as a backup or use it to migrate the converted definitions to another DB2 subsystem.

Exporting and migrating group definitions involves the following steps:

1. Use the online interface to prepare ARMBGRP syntax and JCL. For more information, see “Task title” on page 261.

2. Execute the ARMBGRP batch program. For more information, see “ARMBGRP—Group creation and maintenance” on page 507.

Task title

Use the following procedure and the online interface to export and migrate groups.

Before you begin

The following software must be installed on all target subsystems:

- RECOVERY MANAGER
- A version of DB2 that RECOVERY MANAGER supports

To export and migrate groups by using the online interface

1. From the RECOVERY MANAGER Main Menu, select 1. Application groups, and enter a group name or wildcard pattern to display one or more groups.

2. On the Group List panel, select the groups whose definitions you want to export, and press Enter:

   - To select a specific group or groups, type X next to each one.
To export all groups to the export file, type \texttt{XA}.

\textbf{Figure 23: Group List panel}

\begin{verbatim}
ARMGP001 ================ Group List ===================== Row 1 of 5
Command ===> ______________________________________ Scroll ===> PAGE

Type CREATOR.GROUP name or pattern. Then press Enter.
Name or pattern: . . . . PUBLIC.Q187*___________________

GROUP list disposition . . 1 1. Replace 2. Add 3. Exclude
Type Action to process selected value. Then press Enter.
A=About S=Select B=Backup J=Recover R=Report
D=Delete SA=Select all BA=Backup all JA=Recover all RA=Report all
C=Change Name F=Definitions G=ARMBGRP JCL GA=ARMBGRP all H=Authorizations
O=Util Options P=Progress Rpt X=Export XA=Export all

Number  Act Prv Creator  Group  Description  Updated  Obj

X       PUBLIC   Q1876526  test for jlw  2015-05-08-13.40  34
X       PUBLIC   Q1878348  2015-04-20-11.49  1750
__      PUBLIC   Q1878802  test for jlw  2015-05-06-14.48  5
__      PUBLIC   Q1878802_IX test for jlw  2015-05-07-13.14  10
__      PUBLIC   Q1878802_IXONLY test for jlw  2015-05-07-13.59  17

RECOVERY MANAGER generates ARMBGRP EXPORT JCL for the specified groups.

3 On the Batch Group Export/Migrate JCL Generation panel, select one or more target SSIDs for the export:

\begin{itemize}
  \item Select one of the displayed options to export to all SSIDs on the LPAR, the currently active SSIDs, or specific SSIDs that you will select.
  \item If you chose to select specific SSIDs (option 3), select them from the displayed list, and specify the DD card to which RECOVERY MANAGER should write the exported group definitions.
\end{itemize}

\textbf{Note}
The specified SSIDs must be active. Do not specify an SSID that is the same as the SSID for the currently connected subsystem. RECOVERY MANAGER eliminates duplicate SSIDs.

\textbf{Figure 24: ARMGJ003—Batch Group Export/Migrate JCL Generation panel}

\begin{verbatim}
ARMGJ003 ========= Batch Group Export/Migrate JCL Generation ============
Command ===> _____________________________________________________

Type information below. Then press Enter.
Type ARMEXPRT DD statement for ARMBGRP JCL output (including //).
//ARMEXPRT DD ISP=SHR,_____________________________________________
//          DSN=RDAJLW.ARMBGRP.EXPORT(ARMEXPRT)______________________
_____________________________________________________________________

NOTE: This must be a complete data definition, including member name if the data set is partitioned. The LRECL must be 80.
\end{verbatim}
Migrate group(s) to ssid (optional)
- 1. All SSIDs defined on this LPAR
- 2. All currently active SSIDs on this LPAR
- 3. Select SSIDs from a list

NOTE: Recovery Manager must be installed on all target SSIDs. SSIDs with a DB2 version that is not supported will be bypassed.

Figure 25: ARMGJ004—Batch Group Export/Migrate JCL Generation panel

<table>
<thead>
<tr>
<th>Option</th>
<th>SSID</th>
<th>DB2 Version</th>
<th>Status</th>
<th>DB2 Mode</th>
<th>Data Sharing</th>
<th>RMGR Installed</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DEBH</td>
<td>0710</td>
<td>ACTIVE</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td></td>
<td>DEBN</td>
<td>0810</td>
<td>ACTIVE</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td></td>
<td>DECB</td>
<td>0810</td>
<td>ACTIVE</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>
Managing DB2 system resources

This chapter describes how to manage DB2 system resources.

System resources

You can use the RECOVERY MANAGER for DB2 product to automatically generate JCL to perform recovery or maintenance on various DB2 system resources.

You can generate JCL to perform the tasks in Table 28 on page 265.

Table 28: Tasks performed by RECOVERY MANAGER

<table>
<thead>
<tr>
<th>Task</th>
<th>More information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Back up and recover the DB2 catalog and directory</td>
<td>“DB2 catalog and directory” on page 266</td>
</tr>
<tr>
<td>Recover log data sets, both active and archive</td>
<td>■ “DB2 active logs” on page 269</td>
</tr>
<tr>
<td></td>
<td>■ “DB2 archive logs” on page 273</td>
</tr>
<tr>
<td>Recover bootstrap data sets (BSDSs)</td>
<td>“DB2 BSDS recovery and maintenance” on page 277</td>
</tr>
<tr>
<td>Perform any of the BSDS maintenance</td>
<td>“DB2 BSDS recovery and maintenance” on page 277</td>
</tr>
<tr>
<td>Reallocate data sets in the temporary work file database</td>
<td>“Work file database” on page 285</td>
</tr>
<tr>
<td>List the attributes of a specified system data set or non-system data set (that is, generate an IDCAMS LISTCAT job)</td>
<td>“Physical data set attributes” on page 286</td>
</tr>
<tr>
<td>Back up and recover the BMC Common DB2 repository and the RECOVERY MANAGER repository as well as the R+/CHANGE ACCUM and Log Master repositories (if installed)</td>
<td>“The repository” on page 288</td>
</tr>
<tr>
<td>Specify options for disaster recovery preparation</td>
<td>“Recovering from a DB2 system disaster” on page 291</td>
</tr>
<tr>
<td>Examine or model the DB2 logging environment</td>
<td>“Modeling the DB2 logging environment” on page 369</td>
</tr>
</tbody>
</table>
All system backup, recovery, and maintenance options are accessed through the System Resources option on the Main Menu.

**DB2 subsystem status**

You can use the subsystem recovery and maintenance options of RMGR whether or not DB2 is active. In general, no access is required to the DB2 catalog or to the repository. (Only repository backup and recovery requires access to the catalog.)

When DB2 is not available for BSDS or log recovery, RECOVERY MANAGER issues a message that reports that the information could not be obtained from DB2. You can continue with recovery JCL generation, but you will have to supply some values; RECOVERY MANAGER will take other values from the RECOVERY MANAGER option set (ARM$OPTS).

**Authorizations required to access system resources**

If you have authority to access RECOVERY MANAGER, you can also access system resource recovery and maintenance features and generate the associated JCL.

RECOVERY MANAGER does not verify authority to execute the IBM Print Log Map (DSNJU004) or Change Log Inventory (DSNJU003) utilities that are utilized in the JCL. For information about authorizations that are required by those utilities, see the IBM DB2 command and utilities reference.

**DB2 catalog and directory**

You can generate backup and recovery JCL for any of the catalog and directory table spaces and indexes in DSNDB01 and DSNDB06 in the DB2 subsystem.

The contents of the DB2 catalog and directory vary by version and release of DB2. If DB2 is active, RECOVERY MANAGER automatically obtains the version of DB2 that is installed. If DB2 is not active, RECOVERY MANAGER obtains the version from the RECOVERY MANAGER option set (ARMSOPTS). You should verify that the option set contains the correct version of DB2.

When you use the catalog and directory recovery feature, RECOVERY MANAGER handles backup and recovery options as follows:

- For backups, the utility options in effect for the current subsystem are used; no group level options are available. Additionally, the following restrictions apply:
— You cannot use NGT Recover as the backup utility.

— You cannot use the SNAPSHOT UPGRADE FEATURE (SUF) for SYSCOPY, SYSUTILX, or DSNDDB01 spaces.

— You cannot use the COPY IMAGECOPY command for SYSCOPY, SYSUTILX, or DSNDDB01 spaces.

  If the subsystem options in effect violate any of the listed restrictions, RECOVERY MANAGER uses DSNUTILB (DB2 COPY) as the backup utility.

  ■ For recoveries, you cannot change either the recovery options or save the group to the repository. RECOVERY MANAGER uses only the DSNUTILB utility (DB2 RECOVER); you cannot use NGT Recover.

**Note**

If you are recovering a volume group that contains catalog and directory objects, you should recover the catalog and directory objects before you recover any applications on the volume.

Use the procedure “Backing up and recovering the DB2 catalog and directory” on page 267 to generate the JCL that is required for catalog and directory backup or recovery.

### Backing up and recovering the DB2 catalog and directory

This procedure generates backup or recovery JCL for the DB2 catalog and directory for the specified DB2 subsystem. You can exclude any spaces or indexes that you do not want to include in the generated JCL.

**Before you begin**

To perform this procedure, you need the following items:

■ EXECUTE authority for RECOVERY MANAGER

■ DB2 SYSADM authority to execute the recovery JCL

■ Authority to execute backup JCL (for further information, see the IBM DB2 utility reference or the BMC BMC Next Generation Technology Copy for DB2 for z/OS Reference Manual)

■ The following information that is specific to catalog and directory backup or recovery:
  — Any spaces or indexes that you do not want to back up or recover
The following work unit information for index recovery:

- The work unit (disk file) name
- Primary and secondary allocations (if the sizing is not available)
- Whether the allocations are for cylinders or tracks
- Whether you want extracted index keys to be written to a work file before sorting

To generate JCL for DB2 catalog and directory recovery or backup

Start this procedure at the RECOVERY MANAGER Main Menu.

1. In the RECOVERY MANAGER Main Menu, select System resources.

2. In the System Resource Recovery and Maintenance panel, select Catalog and Directory to list all of the spaces and indexes in the catalog and directory.

3. Generate JCL as follows:

   **Note**
   To exclude one or more objects from the backup or recovery JCL, type X in the Act column by each such object and then press Enter.

   - To generate recovery JCL, select Gen recover JCL in foreground and press Enter.
   - To generate backup JCL, select Gen backup JCL and press Enter.

4. If you selected Gen backup JCL, continue with Step 6 on page 268. If you selected Gen recover JCL in foreground, the Work File Options panel is displayed.

5. Enter information as required and press Enter. For option descriptions, see “Work file option descriptions” on page 251.

6. When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:

   - The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.
   - The job statement must contain a symbolic variable (&#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

7. Save the JCL data set or submit the job as required.
DB2 active logs

You can use RECOVERY MANAGER to generate JCL to recover from the more common active log failures, as follows:

- A read or write error occurs on one active log in a subsystem where dual logs are employed. RECOVERY MANAGER generates JCL to recover the failed copy from the good copy.

- A read or write error occurs on an active log in a subsystem where only a single log is employed. RECOVERY MANAGER generates JCL to recover the failed log from the most recent archived log.

- A read or write error occurs on both active logs in a subsystem where dual logs are employed. RECOVERY MANAGER generates JCL to recover both logs from the most recent archived log.

- A loss of synchronization occurs between logs in a subsystem where dual logs are employed. RECOVERY MANAGER generates JCL to reproduce one log from the other. Logs can be out of synchronization if, for example, a volume was recovered outside of DB2 that contained one or more active data sets.

See the IBM DB2 administration guide for a discussion of input/output (I/O) and other errors that can occur with DB2 logs.

If DB2 is active, RECOVERY MANAGER ascertains from DB2 the archive log data set prefixes and whether single or dual archive logs are in use. If DB2 is not active, the information comes from the RECOVERY MANAGER option set (ARMSOPTS). You should, therefore, always verify the information in the option set before starting to generate active log recovery JCL.

Use the procedure described in “Generating JCL to recover a DB2 active log” on page 270 to generate recovery JCL.

Tip
Before using RECOVERY MANAGER to recover active logs, you can, if necessary, use the IDCAMS LISTCAT option on the System Resource Recovery and Maintenance panel to get physical information about the active log. If you need to recover the active log from the archive log, you can use the Print Log Map utility to get information about the most recent archive log. To recover active logs, you need archive log information only if the active log cannot be recovered from a second copy of the active log.
Generating JCL to recover a DB2 active log

This procedure generates recovery JCL for the most common DB2 active log errors.

You select the error type, provide the required data, and then generate recovery JCL. The JCL must be submitted when DB2 is not active.

Note
BMC recommends that you put the recovered active log on a volume that is different from the failed log.

Before you begin

To perform this procedure, you need the following authorizations:

- EXECUTE authority for RECOVERY MANAGER
- Data set authority to run the JCL

Start this procedure at the RECOVERY MANAGER Main Menu.

To generate JCL to recover a DB2 active log

1. Select System resources to display the System Resource Recovery and Maintenance panel.

2. Select Active Logs. The Active Log Recovery panel is displayed.

3. In the Active Log Recovery panel, select the appropriate error condition.

4. A second panel appears, allowing you to specify options about the error condition. Enter the information as required and press Enter. For more information about the error conditions, see “Log error conditions” on page 271.

5. When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:
   - The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.
   - The job statement must contain a symbolic variable (&##) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

6. Save the JCL data set or submit the job as required.
Note

DB2 must be inactive when you submit the recovery JCL.

Log error conditions

The following table describes the log error conditions you might need to specify.

Table 29: Log error conditions

<table>
<thead>
<tr>
<th>Error condition</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>I/O error with dual logs</td>
<td>■ Failed data set disposition—delete or rename the failed active log</td>
</tr>
<tr>
<td></td>
<td>■ Rename VSAM cluster—rename the failed data set, specifying its new VSAM cluster name in 42 characters or less</td>
</tr>
<tr>
<td></td>
<td>■ Failed copy number—specify which copy of the active log failed</td>
</tr>
<tr>
<td></td>
<td>■ Failed log number—specify the number of the failed active log</td>
</tr>
<tr>
<td></td>
<td>■ New volume—specify the name of the volume where the recovered active log data set will reside</td>
</tr>
</tbody>
</table>

Note: BMC recommends that you put the recovered active log on a volume that is different from the volume where the good copy of the active log resides.
<table>
<thead>
<tr>
<th>Error condition</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>I/O error with single log</td>
<td>- <strong>Rename VSAM cluster</strong>—specify a new VSAM cluster name in 42 characters or less to rename the failed log data set</td>
</tr>
<tr>
<td></td>
<td>- <strong>Failed active log number</strong>—specify the number of the failed active log</td>
</tr>
<tr>
<td></td>
<td>- <strong>Recovery archive log number</strong>—specify the number of the archive log to use in recovery</td>
</tr>
<tr>
<td></td>
<td>- <strong>Recovery archive log timestamp</strong>—specify the prefix for the archive log to be used in recovery if the archive log data set is named by DB2 using timestamps</td>
</tr>
<tr>
<td></td>
<td>- <strong>Recovery archive log volume</strong>—specify the volume number of the archive log to be used in recovery if the archive log data set is not cataloged</td>
</tr>
<tr>
<td></td>
<td>- <strong>Recovery archive unit</strong>—specify the unit name of the device where the archive resides if the archive log data set is not cataloged</td>
</tr>
<tr>
<td></td>
<td>- <strong>New active log volume</strong>—specify the name of the volume where the new active log will reside (optional)</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>If you leave this field blank, the active log will be stored on the volume where it is currently defined.</td>
</tr>
<tr>
<td>Error condition</td>
<td>Options</td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>I/O error with both dual logs</td>
<td>■ <strong>Rename VSAM cluster copy 1</strong>—specify a new VSAM cluster name in 42</td>
</tr>
<tr>
<td></td>
<td>characters or less to rename the first failed log data set</td>
</tr>
<tr>
<td></td>
<td>■ <strong>Rename VSAM cluster copy 2</strong>—specify a new VSAM cluster name in 42</td>
</tr>
<tr>
<td></td>
<td>characters or less to rename the second failed log data set</td>
</tr>
<tr>
<td></td>
<td>■ <strong>Failed active log number</strong>—specify the number of the failed active</td>
</tr>
<tr>
<td></td>
<td>log</td>
</tr>
<tr>
<td></td>
<td>■ <strong>Recovery archive log number</strong>—specify the number of the archive log</td>
</tr>
<tr>
<td></td>
<td>to use in recovery</td>
</tr>
<tr>
<td></td>
<td>■ <strong>Recovery archive log prefix</strong>—specify the prefix of the archive log</td>
</tr>
<tr>
<td></td>
<td>data set to use in recovery if the archive log data set name is</td>
</tr>
<tr>
<td></td>
<td>created by DB2 using a timestamp</td>
</tr>
<tr>
<td></td>
<td>■ <strong>Recovery archive log volume</strong>—specify the volume number of the</td>
</tr>
<tr>
<td></td>
<td>archive log to use in recovery if the archive log data set is not</td>
</tr>
<tr>
<td></td>
<td>cataloged</td>
</tr>
<tr>
<td></td>
<td>■ <strong>Recovery archive unit</strong>—specify the unit name of the device on which</td>
</tr>
<tr>
<td></td>
<td>the archive log resides if the archive log data set is not cataloged</td>
</tr>
<tr>
<td></td>
<td>■ <strong>New active log volume copy 1</strong>—specify the volume where Copy 1 of</td>
</tr>
<tr>
<td></td>
<td>the new active log will reside. If you leave this field blank, the</td>
</tr>
<tr>
<td></td>
<td>recovered active log is put on the volume where it resided before the</td>
</tr>
<tr>
<td></td>
<td>failure.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>New active log volume copy 2</strong>—specify the volume where copy 2 of</td>
</tr>
<tr>
<td></td>
<td>the new active log will reside. If you leave this field blank, the</td>
</tr>
<tr>
<td></td>
<td>recovered active log is put on the volume where it resided before the</td>
</tr>
<tr>
<td></td>
<td>failure.</td>
</tr>
<tr>
<td>Out of synchronization</td>
<td>■ <strong>Copy requiring recovery</strong>—specify which copy of the active log will</td>
</tr>
<tr>
<td></td>
<td>be recovered</td>
</tr>
<tr>
<td></td>
<td>■ <strong>Active log number</strong>—specify the number of the active log that will</td>
</tr>
<tr>
<td></td>
<td>be recovered</td>
</tr>
</tbody>
</table>

**DB2 archive logs**

You can use RECOVERY MANAGER to generate JCL to recover a failed archive log from a second copy of the log.
Although DB2 does automatically use the alternate copy if it is available when the first one fails, it is good practice to make another copy to protect against the loss of both copies. If this loss occurs, the log may be unrecoverable.

The log may also be unrecoverable when there is only a single archive log and it fails. As a safeguard, make a second copy of the log while it is still in good condition.

If DB2 is active, RECOVERY MANAGER obtains the following information from DB2:

- Whether single or dual archive logs are used
- The archive log data set prefixes
- The block size, allocation type, and allocation quantities
- The archive log retention period

If DB2 is not active, the information for the first two items comes from the RECOVERY MANAGER option set (ARM$OPTS). The device type is always obtained from the option set. The information for the last two items is shown as defaults on the data entry panel that appears when you select Archive Logs in the System Resource Recovery and Maintenance panel. Verify that this information is current if DB2 is not active.

If the archive log is not cataloged, a second procedure must be performed. When this procedure is necessary, RECOVERY MANAGER adds JCL that calls the DSNJU004 (Print Log Map) utility. You can use the output from the utility as input to BSDS maintenance procedures that delete information about the failed data set and add the information about the newly recovered data set. See “BSDS maintenance” on page 279 for more information.

### Generating JCL to recover a DB2 archive log

This procedure generates JCL to recover a failed DB2 archive log.

You provide the required data and generate the recovery JCL. If the archive log is not cataloged, additional BSDS maintenance steps are necessary to generate recovery JCL.

**Before you begin**

To perform this procedure, you need the following authorizations:

- EXECUTE authority for RECOVERY MANAGER
- Authority to use the Print Log Map utility
Data set authority to run the recovery JCL

Start this procedure at the RECOVERY MANAGER Main Menu.

To generate JCL to recover a DB2 archive log

1 Select System resources to display the System Resource Recovery and Maintenance panel.

2 Select Archive Logs to display the Archive Log Recovery panel.

3 In the Archive Log Recovery panel, specify information as required and press Enter. For more information about the fields, see “Archive log recovery options” on page 275.

WARNING

If you select Delete for the disposition of a failed tape data set, it is uncataloged and you will not be able to rename the tape.

4 When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:

- The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.

- The job statement must contain a symbolic variable (&#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

5 Save the JCL data set or submit the job as required.

6 If the failed archive log data set is cataloged, no more steps are required. If the archive log is uncataloged, the JCL includes code to print the log map, and you must perform Step 7 on page 275.

7 Execute the JCL produced in Step 4 on page 275 and review the output from the Print Log Map utility to determine what BSDS maintenance must be performed to complete the recovery. Follow the steps in the IBM DB2 administration guide using RECOVERY MANAGER to generate the maintenance jobs as described on “BSDS maintenance” on page 279.

Archive log recovery options

The following table describes the archive log recovery fields you might need to complete.
### Table 30: Archive log recovery fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failed data set disposition</td>
<td>Deletes or renames the failed archive log if it is stored on disk. This option is not valid if the failed archive log is on tape. You cannot rename an archive log stored on tape and attempting to delete it either has no effect or uncatalogs the data set if it was cataloged.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If RECOVERY MANAGER issues an error stating that the log is on tape when it is really on DASD or vice versa, verify the ARCONTAPE configuration option value in the option set (ARMSOPTS).</td>
</tr>
<tr>
<td>Rename archive data set</td>
<td>Specifies the new name of the archive log data set. The new name must be 44 characters or less.</td>
</tr>
<tr>
<td>Rename BSDS copy data set</td>
<td>Specifies the BSDS copy data set name. Specify a new BSDS name if you lost a copy of the BSDS (which is likely if the lost archive log was on tape).</td>
</tr>
<tr>
<td>Failed copy</td>
<td>Specifies which copy of the archive log failed.</td>
</tr>
<tr>
<td>Failed archive log number</td>
<td>Specifies the number of the failed active log.</td>
</tr>
<tr>
<td>Archive data set timestamp</td>
<td>Specifies the timestamp used for the failed archive log if the archive logs are made by DB2 using timestamps. Specify this value in the format <em>Dyyddd.Thhmmsst</em>. If you are DB2’s EXT option, specify the timestamp in the format <em>Dyyyyddd.Thhmmsst</em>.</td>
</tr>
<tr>
<td>Good archive log volume</td>
<td>Specifies the volume name if the volume containing the name of the good log is not cataloged.</td>
</tr>
<tr>
<td>Archive log 1 unit</td>
<td>Specifies the unit name of the device on which the first copy of the archive log resides.</td>
</tr>
<tr>
<td>Archive log 2 unit</td>
<td>Specifies the unit name of the device on which the second copy of the archive log resides.</td>
</tr>
<tr>
<td>Archive retention period</td>
<td>Specifies the time in days to retain the log (for a new archive log on tape only).</td>
</tr>
<tr>
<td>Archive log allocation type</td>
<td>Specifies the allocation type in cylinders, tracks, or records.</td>
</tr>
<tr>
<td>Archive primary allocation</td>
<td>Specifies the primary allocation quantity (for a new archive log on disk only).</td>
</tr>
<tr>
<td>Archive secondary allocation</td>
<td>Specifies the secondary allocation quantity (for a new archive log on disk only).</td>
</tr>
<tr>
<td>Archive block size</td>
<td>Specifies the block size of the new archive log data set. This value must be a multiple of 4096 and between 4096 and 28672 inclusive.</td>
</tr>
</tbody>
</table>
DB2 BSDS recovery and maintenance

RECOVERY MANAGER allows you to generate JCL to perform BSDS recovery and maintenance as follows:

■ Recover one or both BSDSs in several different error situations
■ Reallocate one or both BSDSs
■ Modify the contents of the BSDS
■ Deactivate/Destroy/Restore for data sharing members

BSDS recovery

The BSDS can be recovered from a second copy or from the archive log.

If it is recovered from the second copy, it is current and needs no further action. If it is recovered from the last archive log, it does not contain the last archive log, log information, or BSDS maintenance that was performed after the archive log was written. Changes to the BSDS are not logged, so updating cannot be automatically applied. You must perform BSDS maintenance to update the archive copy.

You can use RECOVERY MANAGER to generate JCL to recover one or both BSDSs in the following situations:

■ You are using dual BSDSs, and one of them has failed.
■ You are using dual BSDSs, and one of them has failed causing DB2 to fail during restart.
■ You are using dual BSDSs, and both have failed causing DB2 to fail.

When you are generating BSDS recovery JCL and DB2 is active, RECOVERY MANAGER obtains the following information from DB2:

■ Whether single or dual BSDSs are in use
■ The BSDS data set prefixes

If DB2 is not active, the information comes from the RECOVERY MANAGER option set (ARM$OPTS). Always verify that this information is current in this situation.

Use the procedure described in “Generating JCL to recover or reallocate the BSDSs” on page 278 to generate BSDS recovery JCL.

See the IBM DB2 administration guide for a detailed discussion of BSDS recovery.
**BSDS reallocation**

You can also use RMGR to generate JCL to reallocate one or both BSDSs when DB2 is inactive in order to change the size of the BSDS or the volumes where it is stored.

Use the procedure described in “Generating JCL to recover or reallocate the BSDSs” on page 278 to generate JCL to reallocate the BSDS.

**Generating JCL to recover or reallocate the BSDSs**

This procedure generates recovery JCL for error conditions that can occur with DB2 bootstrap data sets.

You select the error situation that you want to correct, provide the required data, and then generate the recovery JCL. In two of the scenarios, additional maintenance is required to complete the recovery. You can also use this procedure to generate JCL to reallocate one or both BSDSs.

**Before you begin**

To perform this procedure, you need the following authorizations:

- EXECUTE authority for RECOVERY MANAGER
- Authority to use the Print Log Map utility
- Data set authority to run the recovery JCL

**To generate JCL for recovering or reallocating DB2 BSDSs**

**Note**

If both BSDSs failed in a dual BSDS system, perform all steps that are described in this section. If only one BSDS failed (or you are reallocating the BSDSs), perform only Step 1 on page 278 through Step 5 on page 279.

1. In the RECOVERY MANAGER Main Menu, select System resources and press **Enter** to display the System Resource Recovery and Maintenance panel.

2. Select BSDS, and then press **Enter**.

3. In the BSDS Recovery and Maintenance menu, select BSDS recovery.

4. Select the appropriate error condition and press **Enter**. When a second panel appears, allowing you to specify options about the error condition, press **Enter**.
**WARNING**

If you select Delete for the disposition of a failed tape data set, it is uncataloged and you will not be able to rename the tape.

5 When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:

- The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.

- The job statement must contain a symbolic variable (&amp;) for the job number. See “Output data sets, job cards, and symbolic variables” on page 51 for more information.

6 Save the JCL data set or submit the job as required.

If the JCL is for a dual BSDS failure, it calls the Print Log Map utility. Continue with Step 7 on page 279. Otherwise, no more steps are required.

**Note**

Ensure that this recovery JCL is submitted as follows:

- When DB2 is active for a single BSDS failure on a dual system
- When DB2 is inactive for a restart failure or a dual BSDS failure

7 Review the output from the Print Log Map utility to determine the BSDS maintenance that must be performed. Follow the steps in the IBM DB2 administration guide and use RECOVERY MANAGER to generate the BSDS maintenance jobs (see “BSDS maintenance” on page 279).

8 Execute the maintenance JCL that was generated in the previous step.

This step completes the recovery of BSDS01 and BSDS02.

**Note**

Ensure this recovery JCL is submitted when DB2 is down.

---

**BSDS maintenance**

DB2 automatically updates the BSDS with records of log events.

However, you may want to modify the BSDS when you:
Add more active logs
Copy active logs to newly allocated data sets
Move log data sets to other devices
Discard obsolete archive log data sets
Create or cancel conditional restart control records
Add or change a distributed data facility communication record
Update the checkpoint queue
Change or add passwords for access to the DB2 catalog and directory
Change the system VSAM catalog name
Change the highest written or off-loaded RBA
Change or add passwords for access to the archive log data set

See the IBM DB2 administration guide for more detailed information about changing the BSDS log inventory.

You can use RECOVERY MANAGER to generate JCL for Change Log Inventory jobs as follows:

**Modify active log records**
This option allows you to delete an existing active log data set or specify a new active log to replace one that received an I/O error.
You can validate the modification requests before generating the JCL. You can also accumulate the requests before generating the JCL so that all requests are included in the same JCL. The accumulate feature allows you to combine delete and add actions in the same JCL.

**Modify archive log records**
This option allows you to delete an existing archive log data set or specify a new archive log data set.
As with the active logs, you can validate the modification requests before generating the JCL, and you can accumulate the requests before generating the JCL.

**Modify checkpoint records**
This option allows you to update the checkpoint queue with the checkpoint records for the starting RBA and ending RBA.

**Modify conditional restart control records**
This option allows you to either create a new conditional restart control record (CRCR) to control the next restart of DB2 or to cancel the current active CRCR.

**Modify system data set password**
This option allows you to specify passwords for the DSNDB01 database (the DB2 directory) and for the DSNDB06 database (the DB2 catalog).
Modify system VSAM catalog name
This option allows you to change the VSAM catalog name in the BSDS.

Modify distributed data facility record
This option allows you to add or change a distributed data facility (DDF) record or remove a DDF password.

Modify highest written or off-loaded RBA
This option allows you to update the highest log RBA in the active log data set or to modify the highest off-loaded RBA in the archive log data set.

**WARNING**
Modifying the highest log RBA can affect data consistency.

Modify archive log data set password
This option allows you to delete password protection for archives that are created after an archive operation. It also allows you to provide a password for all archives that are created after an archive operation.

Print the log map via the DB2 DSNJU004 utility
This option allows you to print the log map in order to determine the BSDS maintenance that must be performed.

Use the procedure described in “Creating JCL to make changes to the BSDS log inventory” on page 281 to generate JCL to perform BSDS maintenance.

---

**Creating JCL to make changes to the BSDS log inventory**

This procedure shows you how to generate JCL to change information in the DB2 BSDSs.

**Before you begin**

To perform this procedure, you need the following authorizations:

- EXECUTE authority for RECOVERY MANAGER
- Authority to use the Print Log Map utility

**To generate JCL for making changes to the BSDS**

1. In the RECOVERY MANAGER Main Menu, select **System resources**, then press **Enter** to display the System Resource Recovery and Maintenance panel.
2 Select **BSDS** and press **Enter**. Select the item on which maintenance is to be performed.

3 In the BSDS Maintenance panel, provide the requested data. If you are *not* performing log maintenance, skip to Step 3.c on page 282. If you *are* performing active or archive log maintenance, perform the following substeps:

   a To verify the completed request, select **Validate**.
   
   b Select **Save request** to save and accumulate the request.

   **Note**
   
   When you accumulate requests, the maintenance JCL is for all validated requests.
   
   c Type the data for the next request and return to Step 3.a on page 282. When you have completed all your requests, continue with Step 3.d on page 282.
   
   d To generate JCL, select **Generate JCL**.

4 When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:

   - The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.
   
   - The job statement must contain a symbolic variable (**&##** for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

5 Submit the JCL to complete the specified BSDS maintenance.

---

**Deactivate/Destroy/Restore for data sharing members**

You can also use RECOVERY MANAGER to generate JCL to deactivate, destroy, or restore data sharing members.

Use the procedure described in “Creating JCL to Deactivate/Destroy/Restore data sharing members” on page 283 to generate the JCL for this task.

**Note**

Support for this feature requires DB2 Version 10 new-function mode (NFM) or higher. (IBM APAR PM42528 introduces this function.)
Creating JCL to Deactivate/Destroy/Restore data sharing members

This procedure shows you how to generate JCL to deactivate, destroy, or restore data sharing members.

To generate JCL to deactivate, destroy, or restore data sharing members

1. In the RECOVERY MANAGER Main Menu, select System resources, then press Enter to display the System Resource Recovery and Maintenance panel.

2. Select BSDS and press Enter to display the BSDS Recovery and Maintenance panel.

3. Select DS member to display the Deactivate/Destroy/Restore for data sharing members panel shown in Figure 26 on page 283.

   **Figure 26: GROUP Definition Display panel**

   ARMSRBS ========= Deactivate/Destroy/Restore data sharing member ========= Row 1 from 5
   Command ===> Scroll ===> PAGE

   Enter the action for one member and press Enter to generate JCL.
   Action: S=Deactivate  D=Destroy  R=Restore

   | Action | Subsystem Id | Member Name | Member Id | Status
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>_</td>
<td>IW</td>
<td>DIW1</td>
<td>1</td>
<td>ACTIVE</td>
</tr>
<tr>
<td>_</td>
<td>DIW3</td>
<td>DIW3</td>
<td>2</td>
<td>ACTIVE</td>
</tr>
<tr>
<td>_</td>
<td>DIW4</td>
<td>DOW4</td>
<td>4</td>
<td>DEACT</td>
</tr>
<tr>
<td>_</td>
<td>DIW5</td>
<td>DIW5</td>
<td>5</td>
<td>DEACT</td>
</tr>
<tr>
<td>_</td>
<td>DIW6</td>
<td>DIW6</td>
<td>6</td>
<td>ACTIVE</td>
</tr>
</tbody>
</table>

   ******************************* Bottom of data ********************************

4. Make your entries on this panel and press Enter to generate the JCL.

Print Log Map utility

You can use RECOVERY MANAGER to generate JCL that invokes the IBM Print Log Map (DSNUJ004) utility.

This utility prints the contents of the BSDS, including those items that you can modify using JCL generated by RECOVERY MANAGER.

The Print Log Map utility provides the following information:

- The log data set names for both copies of all active and archive logs
- The RBAs for both copies of all active and archive log data sets
- Any existing passwords for all active and archive log data sets
- The active log data sets that are available for new log data
- The status of all conditional restart control records in the BSDS
- The contents of the checkpoint record queue in the BSDS
- Any distributed data facility communication record for the BSDS
- The contents of the quiesce history record
- The system and utility time stamps
- The contents of the checkpoint queue

For detailed information about the Print Log Map utility, see the IBM DB2 command and utility reference and the IBM DB2 administration guide.

Use the procedure described in “Using the DSNJU004 utility to print the log map” on page 284 to generate JCL to print a log map.

**Using the DSNJU004 utility to print the log map**

This procedure shows you how to generate JCL to print the log map using the DB2 DSNJU004 utility.

**Before you begin**

To perform this procedure, you need the following authorizations:

- EXECUTE authority for RECOVERY MANAGER
- Authority to use the Print Log Map utility

In a a data sharing environment, you also need to know whether the log map is for the entire data sharing group or just one member of the group.

**To print the log map**

1. In the RECOVERY MANAGER Main Menu, select **System resources**, then press **Enter** to display the System Resource Recovery and Maintenance panel.

2. Select **BSDS** and then press **Enter**.

3. Select **Print Log Map**, and then press **Enter**.
4 When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:

- The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.

- The job statement must contain a symbolic variable (&#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

5 Submit the JCL to print the log map.

Work file database

DB2 work file databases are reserved as temporary space for certain SQL operations and cannot be recovered by using a recover utility.

When DB2 is started, situations might occur that prevent data sets in a work file database from being allocated. When this situation happens, you must allocate the database again. When the problem occurs on a volume in a user-defined work file database data set, you can use RECOVERY MANAGER to generate JCL to perform the required reallocation.

**Note**

If DB2 is not available, RECOVERY MANAGER redefines the work file with a CSIZE of 4K.

For more information about work file database problems, see the IBM DB2 administration guide.

Use the procedure described in “Generating JCL to reallocate a work file database” on page 285 data set.

Generating JCL to reallocate a work file database

This procedure shows you how to generate JCL for allocating a temporary data set in a work file database.

**WARNING**

The generated job does not perform alter operations for STOGROUP-managed spaces.
Before you begin

To perform this procedure, you need these authorizations:

- EXECUTE authority for RECOVERY MANAGER
- Authority to use the Print Log Map utility

To generate JCL to reallocate a work file database data set

Start this procedure at the RECOVERY MANAGER Main Menu.

1. Select **System resources** and press **Enter** to display the System Resource Recovery and Maintenance panel.

2. Select **Work File Database**, enter the fields as required, and then press **Enter**.

3. When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:
   - The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.
   - The job statement must contain a symbolic variable (&#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

4. Submit the JCL to reallocate the data set.

Physical data set attributes

You can use RECOVERY MANAGER to generate JCL for an IDCAMS LISTCAT job that reports on the attributes of the data sets that you specify.

You can specify one or more data sets for the current DB2 subsystem in a single job, as follows:

- **system** data sets
  - Select one or more of the following data sets:
    - The BSDSs
    - The active log data sets
    - The archive log data sets
non-system data sets

Supply the VCAT name, the database name, and the space name.

To generate JCL for additional non-system data sets, you must repeat the procedure for each data set. Use the procedure “Generating an IDCAMS LISTCAT job” on page 287 to view physical attributes for the data set.

Generating an IDCAMS LISTCAT job

This procedure shows you how to generate IDCAMS LISTCAT JCL to view physical data set attributes.

Before you begin

To perform this procedure, you need these authorizations:

- EXECUTE authority for RECOVERY MANAGER
- Authority to use the Print Log Map utility

Start this procedure at the RECOVERY MANAGER Main Menu.

To generate an IDCAMS LISTCAT job

1. Select System resources, then press Enter to display the System Resource Recovery and Maintenance panel.

2. Select IDCAMS LISTCAT, enter information as required, and then press Enter.

3. When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:

   - The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.

   - The job statement must contain a symbolic variable (&#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

4. Submit the JCL to obtain the target data set attributes.
The repository

The repository stores all information relating to the application groups that you have created, including the RECOVERY MANAGER options.

In the event of a failure that involves the repository, you can continue to use RECOVERY MANAGER to create groups and generate backup and recovery JCL online; however, you cannot save group information or retrieve information that is already saved. You should, therefore, be in position to recover the repository quickly whenever the need arises and you should back up the repository table spaces and indexes on a regular basis.

**Note**

You must recover the repository at the disaster recovery site to use the data collection feature of the Recovery Management for DB2 solution.

You can group the repository objects by selecting the **Repository** item in the System Resource Recovery and Maintenance menu. This method automatically lists all of the repository objects for you. If installed, the R+/CHANGE ACCUM repository objects are also included. You can then proceed to change backup or recovery options, select a recovery point, add objects to the group, exclude objects from the group, or generate backup or recovery JCL in the same way as you would any other object group.

DB2 must be active to perform this task.

The tables in the repository contain the following information:

- Object set definition
- Object set SQL definition
- Group options
- Product registration
- Group authorizations
- Subsystem backup options
- Group backup options
- BMCUTIL and BMCSYNC tables
- Coordinated disaster recovery points
- Transaction recovery information
- Data collection information for actual, simulated, and estimated disaster recoveries

Additionally, BMC common tables that are used by RECOVERY MANAGER and the R+/CHANGE ACCUM repository objects (if installed) are treated as part of the repository for backup and recovery purposes.
Use the procedure described in “Creating a repository group for backup or recovery” on page 289 to create a group of repository table spaces and indexes.

Creating a repository group for backup or recovery

This procedure shows you how to create a group comprising the table spaces and indexes in the repository and then generate backup or recovery JCL.

Before you begin

To perform this procedure, you need the following:

- EXECUTE authority for RECOVERY MANAGER
- DB2 to be active
- Authority to execute backup or recovery JCL

Start this procedure at the RECOVERY MANAGER Main Menu.

To create a repository group

1. Select System resources then press Enter.

2. In the System Resource Recovery and Maintenance panel, select Repository to see a list of all table spaces and indexes in the repository.

3. When the Object List panel appears, you can choose from the following tasks:

   - Exclude one or more objects from further processing
     Type X in the Act (action) column by the object and then press Enter.

   - Exit or save the group to the repository for later processing
     Press F3 to display the Group Save Confirmation panel. Follow the instructions on the panel, and select Save, SaveAs, Exit, or Cancel as required.

   __Note__
   If you save the group under a new name, the name must be in the form creator.name. The first part of the name cannot exceed 8 characters; the second part cannot exceed 18 characters. The creator term can be PUBLIC.
Logging environment modeling tool

The logging environment modeling tool allows you to examine and view different logging scenarios for a selected DB2 subsystem in order to optimize the amount of DASD space required by the archive logs.

In addition, you can display active log information, archive log information, and view output from the DSNJU004 utility (the print log map). By making use of the logging environment modeling tool’s optimization capabilities, you can dramatically reduce the amount of DASD space required by your logging environment.

For detailed instructions on using the logging environment modeling tool, see “Modeling the DB2 logging environment” on page 369.
Recovering from a DB2 system disaster

This chapter describes how to recover from a DB2 system disaster.

Basic information

When a disaster recovery is necessary, you can use RECOVERY MANAGER to prepare for and perform a disaster recovery using image copies and DB2 archive logs to build and restore DB2 system resources at the recovery site.

RECOVERY MANAGER enables you to automatically back up and verify recoverability of all recovery resources, and then generates the most efficient jobs possible to make recovery at the recovery site straightforward and fast. To recover successfully, you must regularly back up your DB2 objects and the DB2 logs. Only local site data that is already dispatched to the recovery site can be used in the recovery.

If you are mirroring some or all of your resources, RECOVERY MANAGER enables you to perform recoveries that automatically use the mirrored resources to the best advantage. For more information, see “Hardware mirroring support” on page 675. If you are using RECOVERY MANAGER as part of the BMC Recovery Management for DB2 solution, see the Recovery Management for DB2 User Guide.

If you are using RECOVERY MANAGER as part of the BMC Recovery Management for DB2 solution, you can also use simulation and estimation for disaster recovery scenarios for both the system resources and your applications. The recovery simulation feature simulates all aspects of recovery up to, but not including, the actual I/O. Disaster recovery simulation can be useful in reducing your disaster recovery testing costs. See “Recovery simulation” on page 294. The estimation feature provides estimates of recovery time and can track previous run times and other history information from past simulations, estimates, and actual recoveries. You can then use this information to work on improving recovery times. For more information, see the Recovery Management for DB2 User Guide.
The recovery point

If you are not mirroring system resources, the DB2 subsystems and applications can be recovered to the end point of the latest archive log data set available at the recovery site. If you are mirroring the BSDS and DB2 logs, you can recover almost to the point of the disaster.

RECOVERY MANAGER also supports disaster recovery scenarios where the target application objects are copied by methods other than DB2, such as full volume dumps or XRC, and archive logs are shipped to the disaster recovery site after the RECOVERY MANAGER ARMBSRR program has run at the local site. A recovery of this type requires recognition of the additional logs and a modification to the BSDS and the conditional restart point.

Note

If you used PACLOG with the COMPRESS YES option to create the offsite archive logs, you must start the PACLOG compression started tasks (BMCP, BMCBCSS) at the recovery site before the archive logs can be read.

System resource recovery

Before applications can be recovered at a recovery site, all DB2 system resources (catalog and directory, BSDS, active and archive log data sets, and temporary work file database) must be restored at that site.

Also, special conditions (inflight utilities and INDOUBT transactions) must be properly handled. RECOVERY MANAGER generates all JCL required to restore the DB2 subsystem.

If you have indicated that you are mirroring your system resources, RECOVERY MANAGER bypasses those recovery steps made unnecessary by the hardware mirroring. For more information, see “Hardware mirroring support” on page 675.

Application recovery

After restoring system resources, you can use RECOVERY MANAGER to recover your applications at the recovery site to the point in time that is dictated by the last archive log data set that is taken to the recovery site.

If you have indicated that you are mirroring your application groups, RECOVERY MANAGER bypasses those recovery steps made unnecessary by the hardware mirroring. For more information, see “Hardware mirroring support” on page 675.
Data sharing considerations

If you are making contingency plans for the disaster recovery of a DB2 data sharing group, you must set up and install a data sharing group at the recovery site that is identical to the local site by using the same subsystem IDs and member names.

Each member must have all system resources restored before application recovery can begin.

--- Note ---

- RECOVERY MANAGER enables you to generate JCL for the situation in which all members exist on a single LPAR at the remote site even if they are on multiple LPARs at the local site.

- ARMBSRR supports data sharing when the data sharing group includes members with different IBM DB2 version levels. In these cases, you should execute ARMBSRR to connect to the member with the higher version level. For example, if a data sharing group has both DB2 Version 11 members and DB2 Version 12 members, you should execute ARMBSRR to connect to a DB2 Version 12 member.

BMC recommends that you read the following information in the appropriate DB2 planning and administration guide:

- Discussions of the prerequisites for disaster recovery

- How to avoid using group naming conventions that conflict with the coupling facility (XCF) group names for disaster recovery

Permanently quiesced subsystems

It is possible for your data sharing system to have one or more permanently quiesced members that are no longer in use and do not need to be recovered even in the event of a system-wide disaster.

The ARMBSRR program allows you to enter the system IDs of such quiesced subsystems in order to exclude them from a disaster recovery.

In the event that you need to start DB2 with a permanently quiesced member, you may receive the following error message:

**DSNR020I csect-name START MEMBER member, OR REPLY 'NO' OR 'QUIESCED'**

Respond QUIESCED to tell DB2 that the member is quiesced.
A second message might appear:

```plaintext
DSNR030I  csect-name WILL CONTINUE WITHOUT THE
          member MEMBER'S LOG, REPLY 'YES' OR 'NO'
```

Respond YES to continue without the quiesced member’s log.

**WARNING**

You must ensure that none of the logs of the quiesced members are needed for recovery.

---

### Service level agreement options

If you are using the BMC Recovery Management for DB2 solution, you can record the maximum amount of time in which you should be able to recover the entire subsystem, as outlined in your service level agreement.

This figure is used in the System Recovery Reports for comparison purposes with the actual, simulated, or estimated recovery time. For more information, see the *Recovery Management for DB2 User Guide*.

---

### Recovery simulation

Recovery simulation is a feature of the BMC Recovery Management for DB2 solution and requires both the RECOVERY MANAGER component and the NGT Recover component, as well as the BMC Recovery Management for DB2 solution password.

Recovery simulation enables you to go a step beyond previewing recovery activity to actually testing it. This nondestructive option performs most of the work of a normal recovery. After reading and merging image copies, log files, and change accumulation files, it discards the output and leaves the underlying table space unaffected. Using this option, you can see exactly how a recovery will run without sacrificing data availability.

Recovery simulation provides a way for you to validate that you can recover your application data. You can use recovery simulation to verify that needed recovery resources are valid and available and that log apply can be done.

For information about the recovery simulation feature, see the *Recovery Management for DB2 User Guide*.
Recovery estimation

Recovery estimation is a feature of the BMC BMC Recovery Management for DB2 solution and requires both the RECOVERY MANAGER component and the NGT Recover component.

The recovery estimation feature can provide an estimate in hours, minutes, and seconds for the amount of time required to perform a complete disaster recovery, including both system and application resources. It also produces a list of the ten objects that would take the longest amount of time to recover.

For more information about the recovery estimation feature, see the Recovery Management for DB2 User Guide.

Extending the recovery point

RECOVERY MANAGER supports disaster recovery scenarios where the target application objects have been copied by methods other than DB2 (such as full volume dumps or XRC) and log only recovery is desired.

In these scenarios, you continue to ship archive logs to the disaster recovery site after running ARMBSRR at the local site. This type of recovery recognizes the additional logs and modifies the BSDS and the conditional restart point. To extend the recovery point at the disaster recover site, use Extend Recovery Point at DR on the Disaster Recovery - System Resources Recovery panel.

For information about ARMBSSR and extending the recovery point, see “Extending the recovery point at the disaster recovery site” on page 676.

Using RECOVERY MANAGER at the local site

This topic describes the local preparation that is needed to implement a disaster recovery plan.

The following steps are required to prepare your local site data for transportation to the recovery site.

Tip

All necessary steps are included in the following material. The order of these steps is extremely important in ensuring the success of your disaster recovery plan. For a summary of the required steps, see “Overview of local site procedures” on page 296.
Before you begin

To prepare for disaster recovery, you must perform the following steps:

- Determine the time of day that copies are sent offsite.

- Determine the process that causes copies to be sent offsite.

- Decide how many hours (or days) of data loss that you are willing to accept in the case of a disaster. (This value is important in scheduling the RECOVERY MANAGER disaster recovery preparation programs in that it establishes the maximum time between their executions. You should also consider the time that is required to transport the assets to your recovery site.)

- Determine the method you will use to create offsite copies of archive logs.

- Perform the following basic RECOVERY MANAGER setup procedures:
  
  — Verify that the option set is accurate. (See “RECOVERY MANAGER option sets and configuration options” on page 94 for more information.)
  
  — Set up a data set that contains the job card to use with ARMBSRR (the systems resource recovery batch program). The job name must include the variable &##. The user ID on the job card must have installation SYSADM or installation SYSOPR authority when DB2 is brought up at the remote site.
  
  — Set up a GDG base to hold JCL that is generated by ARMBSRR. If you are using SIMULATE YES, set up a GDG base that has a final node ending in SIM or .SIM to ensure that the JCL is not mistaken for actual disaster recovery JCL. ARMBSRR issues an error message if you specify a data set without the final node ending in SIM or .SIM. For example, ARMGDG.BASE.SIM or ARMGDG.BASESIM. If you are using ESTIMATE YES, set up a GDG base that has a final node ending in EST or .EST to ensure that the JCL is not mistaken for actual disaster recovery JCL.

  — Set up a GDG base to hold JCL that is generated by ARMBGEN.

Overview of local site procedures

The following figure shows the tasks required at the local site to prepare for disaster recovery.
Task 1: Creating copies of business applications

These copies will be used to recover your business applications at the recovery site.

See “Generating a backup job interactively” on page 163 for more information on creating copies of business applications.

Before you begin

Before you create copies of your business applications, perform the following steps:
Create and revalidate recovery site groups

Consider creating a set of RECOVERY MANAGER groups specifically for use in disaster recovery. You may want to name these groups to reflect the sequence in which you want them to be recovered at the recovery site. For example:

- DRGROUP1 — accounts receivable
- DRGROUP2 — inventory
- DRGROUP3 — order entry

For more information on creating groups, see “Creating and working with groups” on page 105.

When creating these groups, if possible, limit the number of objects in any one group to no more than a few hundred, including both table spaces and indexes. One large group may require significantly more time for JCL generation than is required for the same set of objects when divided into smaller groups.

Note

BMC recommends that you create application groups for all BMC tables that will be needed at the disaster recovery site (for example, NGT Copy, NGT Recover, and so on).

Audit recoverability

After you have created a set of disaster recovery groups, you should run the RECOVERY MANAGER program ARMBGPV to audit their recoverability by using the SITETYPE RECOVERY option. You may also want to save the picklist report for future reference.

Execute batch revalidation procedures regularly

Schedule batch revalidation jobs for your disaster recovery groups. It is important to verify the recoverability of groups regularly. How often you should schedule these jobs depends on the frequency of object changes and scheduling of your offsite backups. See “Revalidating and reporting on groups in batch” on page 134 for more information.

Regenerate the backup JCL after revalidation

For complete information on backup options, see “Backing up groups” on page 157.

To create copies

1. Verify that the copies are for the recovery site. (Although not required, using the recovery site designation is highly recommended.)

2. If you are taking SHRLEVEL CHANGE copies, ensure that you are taking enough archive logs offsite to complete recovery.

3. Make copies and send the output media to an offsite location.
Task 2: Creating copies of repositories

These copies will be used to recover the RECOVERY MANAGER application and the R+/CHANGE ACCUM application (if installed).

It will also recover the BMCSYNC, BMCUTIL, and BMCXCOPY spaces that are needed by BMC utilities. You must recover the repository to use the data collection feature of the BMC Recovery Management for DB2 solution.

Before you begin

Before you create copies of the repository and table spaces, perform the following steps:

- Create a group for the repository, then generate backup JCL (see “Creating a repository group for backup or recovery” on page 289). Recovery of the repository will be necessary if you want to generate JCL through RECOVERY MANAGER at the disaster site.

- If necessary, create a group for Log Master, and then generate backup JCL.

Note

Recovering the repository includes the recovery of the BMCSYNC, BMCUTIL, and BMCXCOPY spaces that are required for NGT Recover. It also includes the R+/CHANGE ACCUM and Log Master repositories, if these products are installed.

To create copies

1. Verify that the copies are for the recovery site. (Although not required, using the recovery site designation is highly recommended.)

2. Make copies and send the output media to an offsite location. For information about generating the backup JCL for the repository, see “Generating a backup job interactively” on page 163.

Task 3: Creating copies of required libraries

These copies will be used to restore the load libraries and files that are required to run DB2 and BMC products.
To create copies

1 Assemble a ZPARMS module for use at the disaster recovery site. Change DSN6SPRM from RESTART, ALL to DEFER, ALL, and change SITETYP = LOCALSITE to SITETYP = RECOVERYSITE (recommended).

2 Make the following recovery site copies:
   ■ DB2 LOAD and EXIT libraries
   ■ BMC LOAD libraries
   ■ RECOVERY MANAGER libraries
   ■ Any other product or application libraries

   **Note**
   Remember the ARM$OPTS file, which is stored in the BMC DB2 Component Services (DBC). The ARM$OPTS file stores the option set with the configuration options and their values.

3 Send the output media to an offsite location.

Task 4: Creating copies of the DB2 catalog and directory

These copies will be used to recover the DB2 subsystems.

**Note**
You should not make catalog and directory copies by part.

Before you begin

Generate backup JCL for the catalog and directory (see “Back up and recovering the DB2 catalog and directory” on page 267).

To create copies

1 Verify that the copies are for the recovery site. (Although not required, this step is highly recommended.)

2 Make copies and send the output media to an offsite location. For information about generating the backup JCL for the catalog and directory, see “Back up and recovering the DB2 catalog and directory” on page 267.
Note

SHRLEVEL CHANGE copies are acceptable because the archive logs will be taken offsite.
If you are using a data sharing subsystem, run this procedure for only one of the members. Consider copying the large spaces to a separate tape to allow greater use of concurrent processing at recovery time.

Task 5: Preparing system resources

This topic describes the steps that are necessary to prepare to recover system resources for a disaster recovery.

You can perform these steps either manually or online (recommended). If you use the online method, you can generate all necessary JCL either separately for each step or together in a single job stream.

Before you begin

Determine the method you want to use to establish the recovery point.

To prepare system resources

The following steps are necessary when you prepare the system resources for a disaster recovery:

1. Set the options required to establish a recovery point (see “Establishing a recovery point and creating an archive log” on page 311).

2. Set the options required to create copies of the archive log (see “Creating an archive log copy job” on page 315).

   Note

   Even if you do not take archive log copies that are generated by RECOVERY MANAGER offsite, this step is necessary to obtain information that enables RECOVERY MANAGER to analyze stacked tapes correctly for system resources recovery.

3. Set the options required to recover system resources (see “Creating a system resource recovery job (ARMBSRR)” on page 320).

4. Generate JCL for the preceding steps, either as individual jobs or as a single job stream.
5 You can optionally generate JCL for a simulated or estimated recovery at this time. You must have both RECOVERY MANAGER and NGT Recover to generate simulation JCL. You must have the BMC Recovery Management for DB2 solution to generate estimation JCL. For more information about estimation, see the Recovery Management for DB2 User Guide.

6 Run the JCL that you generated.

**Task 6: Revalidating and auditing application groups**

This procedure ensures that all objects within the scope of the group definition are recoverable and that they are included in the recovery.

---

**Note**

You should run the recoverability report as part of your routine disaster recovery preparations.

---

**Before you begin**

Generate JCL for ARMBGPV for each application group (see “Validating the objects in a group” on page 130). Use the following options:

- Use TORESTARTRBA as the recovery point for analysis.

- Use RECOVERABILITY YES and SITETYPE RECOVERY for recovery analysis. Optionally use the LOGCOPY keyword to indicate which copy of the log you are using at the recovery site.

**To revalidate and audit application groups**

1 Wait for the ARMBSRR job (or step) to finish.

   For more information about ARMBSRR JCL, see “ARMBSRR—System resource recovery” on page 673.

2 Run the JCL that you generated for ARMBGPV.

   Jobs may be run in parallel.

**Task 7: Generating application recovery JCL (ARMBGEN)**

This procedure creates JCL that you can use to recover application resources at the recovery site.
Before you begin

Perform the following tasks:

- Using the recovery groups established in Task 1 (see “Task 1: Creating copies of business applications” on page 297), ensure the group recovery options are set as follows:
  
  — If the volumes at the recovery site are not the same device type as those at the local site, select *Allocate in kilobytes*.
  
  — Set the value of **Max concurrent jobs** to a value not greater than the number of initiators you want to use for their recovery at your recovery site.

    ___**Note**___
    
    If you are using SIMULATE YES, you cannot set a value greater than 1.

    For more information, see “General recovery options” on page 233.

  — If possible, limit the SYSCOPY search.

- Generate JCL for ARMBGEN. Use the following options:
  
  — Use a recovery point of TORESTARTRBA.
  
  — Use SITETYPE to indicate whether local or recovery site copies will be used.

    ___**Note**___
    
    If SITETYPE RECOVERY is selected, ARMBGEN will automatically set *Delete STOGROUP Objects* to **Yes** and set *Redefine VCAT Objects* to **Yes**.

    For more information on ARMBGEN, see “Generating recovery JCL in batch” on page 210.

To create application recovery JCL

1. Wait for the ARMBSRR job (or step) and optional ARMBGPV job to finish. (For more information, see “Task 6: Revalidating and auditing application groups” on page 302.)

2. Run the JCL that you generated for ARMBGEN. Jobs can be run in parallel.
Task 8: Create a tape management catalog backup

To ensure successful access to your tape recovery resources, you need to copy the catalog of the tape management system.

Note
This task is usually the responsibility of the operating system programmers. It is shown here to clarify the required order of backup.

Before you begin

Generate JCL to back up your tape management system’s catalog.

To create a tape management catalog backup

1. Ensure that all previous tasks have been completed.
2. Submit the backup job.

Task 9: Creating operating system catalog backups

To ensure successful access to all operating system data sets, the master catalog and all user catalogs should be copied.

Note
This task is usually the responsibility of the operating system programmers. It is shown here to clarify the required order of backup.

Before you begin

Create JCL to back up all systems catalogs.

To create backups of the operating system catalog

1. Ensure that the backup of the tape management catalog is complete.
2. Submit the backup job.
Task 10: Transport copies and JCL to the recovery site

To ensure successful disaster recovery, all copies and generated JCL must be transported to the recovery site.

Using RECOVERY MANAGER at the recovery site

This subsection describes using RECOVERY MANAGER to assist you in performing system resource and application recovery at a recovery site.

RECOVERY MANAGER performs the following functions:

- Recovers DB2 system resources by using the recovery site copies of the archive log data sets, the system resource recovery jobs that were created at the local site, and the appropriate recovery site image copies
- Optionally recovers the BMC tables required to run RECOVERY MANAGER and NGT Recover
- Manages the recovery of business application data from the recovery site image copies of DB2 table spaces, the recovery site log, and recovery site copies of the R+/CHANGE ACCUM files

Tip

All necessary tasks are included in the following material. The order of these tasks is extremely important in ensuring the success of your disaster recovery plan.

Note

ARMBSRR creates one data set that contains multiple jobs that will initialize your DB2 subsystem and recover the DB2 catalog and directory. The Phase 1 job or jobs initialize DB2 system resources. The Phase 2 job or jobs recover the catalog and directory of the DB2 subsystem. If you have the BMC Recovery Management for DB2 solution, a third job is generated to process data collection information. If more than one job is created within a phase, RECOVERY MANAGER includes job steps that synchronize job execution without requiring manual intervention. The Phase 2 job or jobs are placed in the Job Entry Subsystem (JES) hold queue to be released at successful completion of Phase 1.
Overview of recovery site procedures

The following figure shows the tasks that are required at the recovery site to perform a disaster recovery.

Figure 28: Overview of disaster recovery preparation at the recovery site

Task 1: Initializing DB2 resources for recovery (Phase 1)

Before you can recover your DB2 subsystem, you must restore the operating system and tape catalogs and define and initialize DB2 subsystem data sets. The Phase 1 job set creates the necessary log data sets, BSDSs, and VSAM files for the catalog and directory and establishes the conditional restart point.

Before you begin

Before you initialize the DB2 resources, perform the following tasks:

- Restore operating system catalogs and aliases and the tape management catalog. (This responsibility usually belongs to the operating system programmers.)
- Restore DB2, BMC, and other application load libraries and data sets.
- Unload the system resource recovery jobs that were created by ARMBSRR.
If you are recovering a data sharing group, clean out old information from the coupling facility structures. See the IBM DB2 administration guide for additional information on this procedure.

Bring up BMC DB2 Component Services (DBC).

**To initialize DB2 resources**

1. Verify and, if necessary, edit the JCL that was created by ARMBSRR. For each DB2 subsystem, you may need to perform the following tasks:

   - Edit the JCL and search for ACTION(ABORT) to locate any INDOUBT transactions. Determine whether to modify the JCL from ACTION(ABORT) to ACTION(COMMIT) in order to allow those transactions to be committed. Otherwise INDOUBT transactions are aborted.

   - If you do not use the same volume names at the recovery site as you do at the local site, you must modify the JCL to reflect the recovery site volume names.

   - If your data sharing members are not on the same operating systems as the local site, you must change the ROUTE cards.

2. Submit the JCL that was created by ARMBSRR. DB2 must not be started at this time.

   The Phase 1 job or jobs begin executing immediately, whereas the Phase 2 job or jobs are held for execution. If you used MAXLOGJOBS > 1, the Phase 1 jobs will submit additional log copy jobs. For more information about ARMBSRR and the number of jobs in each phase, see “ARMBSRR—System resource recovery” on page 673.

   **Note**
   The ARMBSRR program performs stacked tape analysis prior to creating the Phase 1 JCL. The number of log copy jobs may vary based on this analysis and may possibly be less than you requested with the MAXLOGJOBS option.

   If multiple jobs were created for Phase 1, they will be submitted by this JCL when it executes. See “About Phase 2” on page 339 for a diagram illustrating Phase 1 multiple jobs.

3. Upon completion of the Phase 1 job or jobs, check the job output for warning or error messages. Correct any error situations to complete Phase 1 successfully.

   If you are using the BMC Recovery Management for DB2 solution, check the job output from the ARMD* jobs to verify that data collection from the recovery processed correctly. For actual recoveries, data collection errors do not affect the
job return code. This feature ensures that problems with data collection do not impact the disaster recovery.

Tip
If you need to restart the job or jobs during Phase 1, BMC recommends that you cancel the Phase 2 job that is on hold, then resubmit the JCL to execute from the beginning. For information on restarting Phase 1 or Phase 2 jobs, see “Running and restarting DB2 conditional restart recovery jobs” on page 337.

Task 2: Recovering the DB2 catalog and directory (Phase 2)

Recover the DB2 catalog and directory by releasing the Phase 2 job or jobs.

The Phase 2 job or jobs recover the catalog and directory, initialize the work file databases, and optionally recover the BMC Common DB2, RECOVERY MANAGER, R+/CHANGE ACCUM, and Log Master repositories. If you are using the BMC Recovery Management for DB2 solution, the jobs also collect data about the recoveries.

Before you begin

- Start DB2 in maintenance mode with the ZPARM assembled for your recovery site (see “Task 3: Creating copies of required libraries” on page 299) or by typing the following command:
  -START DB2,PARM(zparmName),ACCESS(MAINT)
  (where the variable zparmName is the ZPARM module that you assembled for your recovery site.)

- Reply Y to all conditional restart (DSNJ245I) messages.

Note
The Phase 2 jobs must use the installation SYSADM ID on the job cards. This is a DB2 requirement for some of the recovery steps.

To recover the DB2 catalog and directory

1 Release the Phase 2 job from the JES hold queue.

   If multiple jobs were created for Phase 2, they will be submitted by this job when it executes.

2 Upon completion of the Phase 2 job or jobs, check the job output for any warning or error messages. Correct any error situations to complete Phase 2 successfully.
Task 3: Recovering DB2 applications

After the DB2 catalog and directory are recovered and workfile databases are initialized, you can recover your DB2 applications.

Ready-to-run recovery JCL that was created by ARMBGEN speeds application recovery.

Before you begin

Before you recover your DB2 applications, perform the following tasks:

- Start DB2. The last step of the Phase 2 job or jobs stopped DB2. Restart it now. Use your recovery site ZPARM, but do not select maintenance mode.
  
  -START DB2,PARM(zparmName)
  
  (where zparmName is the ZPARM module you assembled for your recovery site.)

- Restore the jobs that were created by ARMBGEN (see “Task 7: Generating application recovery JCL (ARMBGEN)” on page 302).

To recover DB2 applications

1. Submit the offsite application recovery jobs that were created in “Task 7: Generating application recovery JCL (ARMBGEN)” on page 302.

2. If you want to recover applications other than those that were recovered in the previous step, use RECOVERY MANAGER to revalidate the recoverability of the desired application groups, confirm recovery resources, and generate the recovery JCL.

3. If the volumes at the recovery site are not the same device type as those at the local site, select Allocate in kilobytes.

Note

If SITETYPE RECOVERY is selected, ARMBGEN will automatically set Delete STOGROUP Objects to Yes and set Redefine VCAT Objects to Yes.
Task 4: Resuming normal DB2 operation

When you are satisfied that all required business applications have been recovered, you are ready to restart the system for business usage.

Before you begin

Verify that you have recovered all required applications. You can use the RECOVERY MANAGER Appl. Group Definition option to identify all spaces in exception status. If you find required table spaces that have not been recovered, use the instructions in “Task 3: Recovering DB2 applications” on page 309 to generate JCL for these objects.

To resume normal DB2 operation

1. Stop DB2 and restart for LOCALSITE operations by using your normal production ZPARM, as follows:

   -START DB2

2. Run backups for the DB2 catalog and directory.

Task 5: Returning data collection file to local site

If you are using RECOVERY MANAGER as a component of the BMC Recovery Management for DB2 solution, data about the disaster recovery is saved to a flat file.

You should take this file back to the local site so that the information can be added to the RECOVERY MANAGER data collection tables.

   Note

By default, the data collection file created at the recovery site is called userId.BMCARMDC.Ddate.Ttime.MIGSQL. It contains SQL INSERT statements that you can use to populate the data collection tables at your local site.

For more information, see the Recovery Management for DB2 User Guide.

Preparing system resources for recovery

System resource recovery preparation includes the following steps:

- Establish a coordinated recovery point for all members
- Cause DB2 to create an archive log for all members
- Create copies of the archive log
- Generate subsystem recovery JCL

You can perform these steps by entering the options you want using the RMGR online interface. RMGR then generates the JCL to perform the required functions either separately or as a single multi-job job stream.

**Note**
You can specify the names of up to two subsystems (of 1-8 characters) that have been permanently quiesced and thus never need to be recovered in the event of a disaster. RMGR will not generate disaster recovery JCL to recover these quiesced subsystems. For more information, see “Permanently quiesced subsystems” on page 293.

The following sections describe the steps that are necessary when you prepare the system resources for a disaster recovery.

**Establishing a recovery point and creating an archive log**

This procedure creates log analysis JCL that you can use to establish a recovery point in order to perform a disaster recovery.

It also causes the creation of an archive log for all DB2 members.

**Note**
If used in a data sharing environment, RECOVERY MANAGER generates the necessary JCL to establish a coordinated recovery point for all data sharing members.
About the archive log command

You can use the archive log creation program, ARMBLOG, to issue the appropriate archive log command and wait for the offload to finish.

For more information about ARMBLOG, see “ARMBLOG—Archive log creation” on page 603.

You can also issue the -ARCHIVE LOG command manually. If you issue the command manually, select the appropriate environment type in the following table and perform the associated step.

<table>
<thead>
<tr>
<th>Environment</th>
<th>Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-data-sharing</td>
<td>Issue the -ARCHIVE LOG command</td>
</tr>
<tr>
<td>Data sharing</td>
<td>Issue an -ARCHIVE LOG,SCOPE(GROUP) or -ARCHIVE LOG,MODE(QUIESCE) command on any one of the data sharing members</td>
</tr>
</tbody>
</table>

Before you begin

To establish a recovery point, you need the following authorizations:

- To create the JCL: EXECUTE authority for RECOVERY MANAGER
- To execute the JCL:
  - APF authority for ARMBCRC
  - READ authority for BSDS data sets
  - ALTER authority for the active log data sets
To establish a recovery point and create an archive log

Start this procedure at the RECOVERY MANAGER Main Menu.

1. In the RECOVERY MANAGER Main Menu select System Resources.

2. In the System Recovery and JCL Generation panel, select Disaster Recovery Prep.

3. In the Disaster Recovery panel, select Establish a Recovery Point.

4. In the Establish Recovery Point panel, choose the subsystem recovery point. For more information, see Table 31 on page 314.

5. Specify the ARMBLOG options as follows and press Enter:

   ■ Number of times to check for offload complete (defaults to 5)
   ■ Number of seconds to wait before re-checking for offload complete (defaults to 180 seconds)

   The System Resource Recovery and Maintenance panel is displayed again.

6. At this point, you can generate the JCL to establish the recovery point, or you can wait until you have also set options for preparing the archive logs and subsystem recovery and then generate JCL for all three steps into a single job.

   a. To generate the job now, select Establish a Recovery Point (ARMBLOG or ARMBCRC), and then press Enter.

   b. To generate a single job for all three disaster recovery steps, set all options, then select Option 9, Generate ALL Disaster Recovery Jobs, and then press Enter.

7. On the JCL Specification panel, enter a fully qualified output data set name. Be aware of the following information:

   ■ The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.

   ■ The job statement must contain a symbolic variable (&#) for the job number. See “Output data sets, job cards, and symbolic variables” on page 51 for more information.

8. Save the JCL data set or submit the job as required.
### Table 31: Recovery point selection

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>End of the archive log created using the ARMBLOG program</td>
<td>Generates JCL to run the ARMBLOG program, which performs an ARCHIVE LOG command. The timestamp of the ARCHIVE LOG command then becomes the point of recovery.</td>
</tr>
<tr>
<td>User specified timestamp, using the ARMBCRC program</td>
<td>Specifies a timestamp to be used as the point of recovery. <strong>Note:</strong> The timestamp recorded in the CRRDRPT table must be equal to or less than the current time. If you enter a timestamp greater than the current time (that is, a time in the future), ARMBCRC bypasses the entry and issues an informational message. When you recover to a user-specified timestamp, RECOVERY MANAGER generates multiple steps, as follows:</td>
</tr>
<tr>
<td></td>
<td>■ The ARMBTSI program inserts the timestamp into the RECOVERY MANAGER coordinated disaster recovery (CRRDRPT) table.</td>
</tr>
<tr>
<td></td>
<td>■ The ARMBCRC program converts this timestamp into a valid RBA or LRSN. If data sharing, multiple ARMBCRC steps are generated and are synchronized to begin after the ARMBTSI step completes.</td>
</tr>
<tr>
<td></td>
<td>■ The ARMBLOG program issues an ARCHIVE LOG command. If data sharing, this step is synchronized to begin after all ARMBCRC steps have completed. <strong>Note:</strong> For data sharing systems, you can choose to convert the timestamps for all members of the data sharing system in a single job, which simplifies scheduling and monitoring the timestamp conversion process. Otherwise, RECOVERY MANAGER generates a separate job for each data sharing member. Each job runs on the LPAR on which the member resides.</td>
</tr>
<tr>
<td>Last recovery point registered in the Coordinated Disaster recovery table (CRRDRPT)</td>
<td>Uses the most recent timestamp in the CRRDRPT table. This option generates multiple steps, as follows:</td>
</tr>
<tr>
<td></td>
<td>■ The ARMBCRC program converts the timestamp into a valid RBA. If data sharing, multiple ARMBCRC steps are generated.</td>
</tr>
<tr>
<td></td>
<td>■ The ARMBLOG program issues an ARCHIVE LOG command. If data sharing, this step is synchronized to begin after all ARMBCRC steps have completed. <strong>Note:</strong> You must have previously used the ARMBTSI program to place a timestamp in this table (see “ARMBTSI—Time stamp insertion” on page 719).</td>
</tr>
</tbody>
</table>
Creating an archive log copy job

This procedure generates JCL that you can use to create up to two additional copies of the archive log data sets for use at a recovery site.

*Note*

If used in a data sharing environment, RECOVERY MANAGER generates ARMBARC jobs for each data sharing member.

---

About archive log copies

Creating one or two copies of the archive log allows you to send those copies (ARCHLOG3 and ARCHLOG4) offsite, while keeping both copies of your archive logs onsite for use in the event of an input/output (I/O) error on a recovery.

(Some organizations send the second archive log copy offsite.) If you do not want to create extra copies of the archive log to send offsite, you should still run the ARMBARC program to register the copies of DBD01, SYSDBDXA, SYSCOPY, and SYSUTILX in the history file so RECOVERY MANAGER can perform stacked tape analysis to optimize the recovery of the catalog and directory.

*Note*

This step can also be performed by the PACLOG product.

---

If you are not creating copies of the archive logs but want to update the history file with the information for the image copies of the catalog and directory spaces, you only need to run the JCL on one of the members. However, it must be the member that made and registered the catalog and directory backups. You may want to schedule ARMBARC to run more frequently so that it does not have to process as
many logs during this step. RECOVERY MANAGER generates JCL to delete and create the history file whether or not you are creating copies of the archive logs.

Tip
BMC strongly recommends that you regenerate the JCL for ARMBARC after you migrate to a newer version of RECOVERY MANAGER. This ensures that your JCL is able to take advantage of new features.

Before you begin

To perform this procedure, you need the following items:

- If you use the ARMBLOG program to issue the -ARCHIVE LOG command, check for a return code of zero (RC=0). Otherwise, wait for the DSNJ139I message from the -ARCHIVE LOG command, which indicates that the archive is complete. If you are using a data sharing environment, wait for this message from each member.

- EXECUTE authority for RECOVERY MANAGER to create the JCL:

- The following authorizations to execute the JCL and make copies of the archive logs:
  - APF authority for ARMBARC
  - READ authority for archive log data sets
  - READ authority for BSDS data sets
  - ALTER authority for the new archive log data sets to be created, if any
  - ALTER authority for the archive history file

- Archive log data sets that are cataloged

- The following information that is specific to archive log creation:
  - How many copies of the archive log data sets that you want (if any)
  - Whether you will use disk or tape for the copies of the log data sets
  - The disk or tape options and other information for the output copies

To create recovery site copies of archive logs

Start this procedure at the RECOVERY MANAGER Main Menu.

1 Access the Disaster Recovery - Archive Log Copy Options panel, as follows:
   a Select System Resources
   b Select Disaster Recovery Prep
c Select Copy Archive Logs

2 Enter information about the number and types of copies that you need on the following series of panels. If you chose to create more than one copy, an Offsite Copy Options panel is displayed for each copy. For descriptions of the available options, see “Field definitions—archive log copy options” on page 317.

3 Generate the JCL to copy the archive logs as follows:

a To generate the job now, select option 5, Copy the Archive Logs (ARMBARC).

b To generate a single job for all three disaster recovery steps, set all options, then select option 9, Generate ALL Disaster Recovery Jobs, and then press Enter.

4 When the JCL Specification panel appears, enter a fully qualified output data set name, and either save the JCL data set or submit the job. Be aware of the following information:

- The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.
- The job statement must contain a symbolic variable (&#) for the job number. See “Output data sets, job cards, and symbolic variables” on page 51.

Field definitions—archive log copy options

The following table describes fields found on the series of panels for the archive log copy options.

Table 32: Archive log copy fields

<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of copies</td>
<td>None</td>
<td>Specifies the number of recovery site archive log copies to make as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0 — Update the history file, but make no copies.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 — One copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 — Two copies</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Limit logs copied by</td>
<td>None</td>
<td>Specifies the method (LOGS, HOURS, or RBARANGE) to be used to limit the logs that are being copied. If you choose LOGS or HOURS, enter a number in the Limit Value field. If you choose RBARANGE, enter a start and end RBA. <strong>Note:</strong> BMC recommends that you use RBARANGE to ensure that all logs are copied. If you do not use RBARANGE and the ARMBARC job is delayed, it is possible that some logs could be missed.</td>
</tr>
<tr>
<td>Limit value</td>
<td>None</td>
<td>Specifies the maximum number of logs or maximum number of hours of log data that you wish to copy (used in conjunction with Limit logs copied by)</td>
</tr>
<tr>
<td>Start RBA</td>
<td>None</td>
<td>Specifies the start RBA value that is used to limit the number of logs that are copied (used in conjunction with RBARANGE). Once established, this value never needs to change. The archive history file records the logs that are already processed.</td>
</tr>
<tr>
<td>End RBA</td>
<td>FFFFFFFFFFFFFF</td>
<td>Specifies the end RBA value that is used to limit the number of logs that are copied (used in conjunction with RBARANGE). <strong>Tip:</strong> BMC recommends that you use FFFFFFFFFFFFFF to ensure that you always process all logs that have been created since ARMBARC was last executed.</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Data set prefix</td>
<td>None</td>
<td>Specifies the data set prefix for each offsite archive log copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Because the archive log number must be appended to the data set name, the name specified cannot exceed 35 characters. If the timestamp (TSTAMP) option is set to <strong>Y</strong> in the RECOVERY MANAGER option set, the allowable prefix length is further reduced to 17 characters to allow the date and time to be appended to the log data set name. The output prefix value specified on this panel has the following restrictions: If you have a data sharing system and more than one copy job is being generated, then you must specify either the SSID, MBRNAME or MBRID symbolic. (These are the only symbolics guaranteed to be unique for each subsystem.) You can optionally specify the VCAT configuration option, but only if you also specify one of the approved symbolics. RECOVERY MANAGER supports symbolics for the output data set prefix, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ &amp;SSID—subsystem id</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ &amp;MBRNAME—data sharing member name (same as &amp;SSID if non-data-sharing)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ &amp;MBRID— data sharing member id (0 if non-data-sharing)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ &amp;VCAT—value of the SSID.VCAT configuration option (may not be unique across DB2 subsystems)</td>
</tr>
<tr>
<td>Device type</td>
<td>None</td>
<td>Specifies whether to write the offsite archive log copy to tape or disk</td>
</tr>
<tr>
<td>Unit</td>
<td>SYSALLDA (for disk devices)</td>
<td>Specifies the generic or esoteric name for the device type.</td>
</tr>
</tbody>
</table>
### Field | RECOVERY MANAGER default | Description
--- | --- | ---
Tape option | No limit | Optionally limits the amount of time a tape is kept. Use only one of the following values:
- **Retention period** — specifies the retention period in days for this copy. Valid numbers are 0 through 999.
- **Expiration date** — specifies the expiration date for this copy. The date must be in the format *yyddd* or *yyyy/ddd*.

TRTCH Compression | None | Specifies whether the hardware compression in the tape drive unit is to be enabled or disabled. If you do not specify TRTCH compression, a tape management system or operating system default may apply.

Stack archives on tape | Yes | Specifies whether to stack the archive log data set copies contiguously on a new tape volume.

Disk options | None | Optionally specifies information for disk storage, as follows:
- **SMS data class** — specifies a valid SMS data class name for the copies (1-8 characters)
- **SMS management class** — a valid SMS management class name for the copies (1-8 characters)
- **SMS storage class** — a valid SMS storage class name for the copies (1-8 characters)
- **Unit count** — specifies the number of units to be allocated for the output image copy data sets. Provide an integral number from 0 to 59. Zero (0) enables you to control the unit count with SMS if required. The default is no unit count at all.

---

**Creating a system resource recovery job (ARMBSRR)**

This procedure creates JCL that you can use in conjunction with copies of the archive logs to restore DB2 system resources at the recovery site.

You can also generate recovery simulation JCL to verify that you have all resources required for a system resource recovery. (For additional information, see “Recovery simulation” on page 294.)
Tip

BMC strongly recommends that you regenerate the JCL for ARMBSRR after you migrate to a newer version of RECOVERY MANAGER. This ensures that your JCL is able to take advantage of new features.

Before you begin

To perform this procedure, you need the following items:

- Wait for the ARMBARC job (or step) to finish. (See “Creating an archive log copy job” on page 315.)

  **Note**
  
  For a data sharing subsystem, there are multiple ARMBARC jobs (or multiple job steps).

- The following authorizations to execute ARMBSRR at the local site:
  - EXECUTE authority on the RECOVERY MANAGER DB2 plan
  - READ authority for the archive log data sets and BSDSs
  - APF authority for ARMBSRR
  - READ authority for the ICF catalog
  - READ authority for the archive history file
Preparing system resources for recovery

- The following information that is specific to building the ARMBSRR job:
  — If you will restore the archive logs to disk at the recovery site
  — If you will initialize the recovery site active logs from archive logs
  — if the archive copies will be cataloged at the recovery site (recommended)

- A GDG base that is established for the offsite JCL.

  Note
  Because the ARMBSRR JCL that is created by this procedure is executed regularly, you must create a GDG data set (ARMGDG) to receive the generated JCL that you must then send to the remote site. See the ARMGDG member in the RECOVERY MANAGER control library. If you are using SIMULATE YES, set up a GDG base that has a final node ending in SIM or .SIM. (For example, ARMGDG BASE SIM or ARMGDG BASE SIM.) This ensures that simulation jobs are not mistaken for recovery jobs. If you are using the BMC Recovery Management for DB2 solution and using ESTIMATE YES, set up a GDG base that has a final node ending in EST or .EST.

- A model Data Set Control Block (DSCB) name for GDG data sets. SYS1.MODEL can be used if it exists on your system.

- An archive history file data set (one for each member for data sharing subsystems)

- A job card that is stored in a data set that has a job name that includes the variables &##

Verify the following configuration option values (in the option set) used by ARMBSRR:

- The RECOVERY MANAGER plan
- The R+/CHANGE ACCUM plan
- Whether time stamps are to be used in the names of the archive log copies
- The version of DB2 that is installed
- The DB2 subsystem exit and load libraries
- The PACLOG libraries (optional)
- Whether the CATALOG parameter is used in IDCAMS specifications
- Work file database name (for data sharing groups only)
- The following DB2 subsystem parameters:
  — VCAT name
To create a system resource recovery job (ARMBSRR)

Start this procedure at the RECOVERY MANAGER Main Menu.

1. Access the System Resources Recovery panel, as follows:
   a. Select System Resources
   b. Select Disaster Recovery Prep
c Select Prepare to Recover the DB2 Subsystem

d  *(BMC Recovery Management for DB2 solution only)* A separate Mirror
Strategy panel appears to enable you to specify your mirroring strategy if you
are using the BMC Recovery Management for DB2 password. (For more
information, see the *Recovery Management for DB2 User Guide*).

The System Resource Recovery panel (ARMDR004) is displayed.

```
Command ===> _________________________________________________________________
Please type in the requested information. Then press Enter.

Extend Recovery Point at DR. . . . _ 1. Yes  2. No
Recover RM Repository. . . . . _ 1. Yes  2. No
Recover LogMaster Repository . _ 1. Yes  2. No
Recover Work File Database . _ 1. Yes  2. No
Library that contains DSNTEP2. CSGI.DB2V10M.DSNLOAD________________________
Maximum catalog recovery jobs. __ (1 - 32)
Image copy type. . . . . . . . _ 1. LP  2. LB  3. RP  4. RB
Synchronization file name. ____________________________________________
Data Collection dataset name . ____________________________________________
```

2 Specify the information on the panel and then press **Enter**. (For more information
about the entries on this panel, see “Field definitions—system resource recovery”
on page 325.)

Another System Resources Recovery panel (ARMDR04A) is displayed.

```
Command ===> _________________________________________________________________
Please type in the requested information. Then press Enter.

Archives Catalogued . . . . . _ 1. Yes  2. No
Restore archive copies to disk _ 0 (0 or 1 copy)
Number of logs to disk. . . . _ Logs ___ Hours ___ Days
Restore program . . . . . . . _ 1. Yes  2. No
Initialize Active Logs . . . . _ 1. Yes  2. No
Init Actives with DSNJLOGF . _ 1. Yes  2. No
Maximum log jobs per member. . _ (1-10)
Archive copy to use offsite. . _ 1. Offsite  2. Archive 2  3. Archive 1
BSDS Log Processing Unit . _ Logs ___ Hours 29 Days
```

3 Specify the information on the panel and then press **Enter**.

For more information about the entries on this panel, see “Field definitions—
system resource recovery” on page 325.

The Disaster Recovery - OffSite Copy Options panel is displayed.

4 Specify the information on the OffSite Copy Options panel, then press **Enter**. (For
more information about the entries on this panel, see “Field definitions—offsite
copy options” on page 332.)
The Disaster Recovery - System Resources Recovery panel (job card information) is displayed.

5 Specify the information on the job card information panel, then press Enter. (For more information about the entries on this panel, see “Field definitions—job card information” on page 332.)

6 At this point, you can generate the JCL to recover system resources, or you can wait until you have also set options for establishing a recovery point and made archive log copies and then generate JCL for all three steps into a single job.

   a To generate the job now, select Option 6, Prepare to Recover the DB2 Subsystem (ARMBSRR) and then press Enter to create JCL.

   b To generate a single job for all three disaster recovery steps, set all options, then select Option 9, Generate ALL Disaster Recovery Jobs, and then press Enter.

7 When the JCL Specification panel appears, enter a fully qualified output data set name. Be aware of the following information:

   ■ The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.

   ■ The job statement must contain a symbolic variable (&#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

    Note

If you selected Simulate Yes, RECOVERY MANAGER appends .SIM to the data set name if the final node does not end in SIM or .SIM. If you are using the BMC Recovery Management for DB2 solution and you selected Estimate Yes, RECOVERY MANAGER appends .EST to the data set name if the final node does not end in EST or .EST. You must have created these GDG bases previously.

8 Save the JCL data set.

Field definitions—system resource recovery

This section describes fields found on the System Resource Recovery panel.

Fields are listed in alphabetical order in the following table.
Table 33: System resource recovery fields

<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Archive copy to use offsite</td>
<td>1</td>
<td>Specifies which archive log data set is to be used at the recovery site, as follows: 1—recovery site archive data set copy 3 (created by ARMBARC) 2—local site archive copy 2 3—local site archive copy 1</td>
</tr>
<tr>
<td>Archives Cataloged</td>
<td>Yes</td>
<td>Specifies whether the archive log data sets will be cataloged at the recovery site</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the operating system and tape management catalogs will not be brought up to date at the recovery site, this option instructs RECOVERY MANAGER to generate all references to the archive logs with the unit and volume specified. Note: If you specify that the archive logs are to be restored to disk (Restore archive copies to disk), you cannot specify No in this field.</td>
</tr>
<tr>
<td>BSDS Log Processing Limit</td>
<td>No Limit (process all)</td>
<td>Directs RECOVERY MANAGER to process only the number of logs that you specify (1 through 9999), beginning with the most recent log  You can also specify the number of hours (1 through 999) or days (1 through 99) to process.  RECOVERY MANAGER issues a warning when it encounters missing or uncataloged archive logs in the BSDS. The value that you specify should be greater than the number of active logs.</td>
</tr>
<tr>
<td>Data Collection dataset name</td>
<td>userId.ssid.DCFILE</td>
<td>Specifies a data set to be used to store information about the system recovery during disaster recovery (when DB2 is not available)  When DB2 is restarted, the information from the file (as well as all additional recovery information from the system and application recovery) will be stored in DB2 tables.</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>--------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Extend Recovery Point at DR</td>
<td>No</td>
<td>Specifies whether RECOVERY MANAGER will extend the recovery point at the disaster recovery site. If you specify Yes, RECOVERY MANAGER uses ARMBSRR syntax option DREXTEND YES. You can only specify Yes if you also specify the following: ■ Simulate is No. ■ JCL Type is DR. ■ MAXLOGJOBS is 1. ■ Initialize Actives is not used. ■ Mirroring is not used. For more information, see “Extending the recovery point” on page 295.</td>
</tr>
<tr>
<td>Hardware mirroring level</td>
<td>0</td>
<td>Specifies the DASD mirroring level that is implemented, if any, as follows: ■ 0—no mirroring is in place (that is, a full recovery is required at the recovery site) ■ 1—the BSDS and active logs are being mirrored at the recovery site (Hardware Mirroring Level 1) ■ 2—the DB2 catalog and directory as well as the BSDS and active logs are being mirrored at the recovery site (Hardware Mirroring Level 2) For more information, see “Hardware mirroring support” on page 675. Note: Hardware mirroring levels are specified differently for the BMC Recovery Management for DB2 solution. For more information, see the Recovery Management for DB2 User Guide.</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Offsite data set copy</td>
<td>None</td>
<td>Specifies which copies of the BSDS and active logs are being mirrored at the recovery site, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 0—both copies of the BSDS and active logs are being mirrored.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 1—only Copy 1 of the BSDS and active logs are being mirrored (in this case, Copy 2 will be rebuilt).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 2—only Copy 2 of the BSDS and active logs are being mirrored (in this case, Copy 1 will be rebuilt).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Used with Hardware mirroring level.</td>
</tr>
<tr>
<td>Image copy type</td>
<td>4</td>
<td>Specifies the type of image copy to be used for the catalog and directory, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1—local primary (LP)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2—local backup (LB)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3—remote primary (RP)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4—remote backup (RB)</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> This option is ignored for systems with hardware mirroring Level 2.</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>---------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Initialize Active Logs                    | No                        | Specifies whether the active logs are to be populated with the contents of the archive log data sets during resource recovery. If you specify **Yes**, RECOVERY MANAGER populates the first n-1 active logs with the contents of the last n-1 archive log data sets, where the variable n is the number of active logs that are defined in the BSDS at the recovery site. The nth active log will be empty. This feature is not applicable for systems using hardware mirroring.  
  - If you specify that the active logs are not to be initialized, RECOVERY MANAGER initializes them to be empty.  
  - **Initialize Active Logs** and **Hardware Mirroring Level** are mutually exclusive options. Also, if you set the **Restore archive copies to disk** option to 1 or 2, RECOVERY MANAGER ignores the **Initialize Active Logs** parameter because it is unnecessary to copy archives to disk and to the active logs. |
| Initialize Actives with DSNJLOGF          | None                      | Generates DSNJLOGF steps to initialize actives when not initializing actives from archives **Initialize Active Logs** and **Initialize Actives with DSNJLOGF** are mutually exclusive options. |
| Maximum catalog recovery jobs             | 1                         | Specifies the maximum number of jobs (1 - 32) to use to recover the catalog and directory. **ARMBSRR** generates JCL to recover some of the catalog and directory spaces through the SYSDBASE space in the first grouping because these spaces must be recovered serially. Multiple, parallel recovery jobs are considered for the remaining table spaces and their indexes.  
  Multiple jobs are generated to recover the catalog table spaces only if  
  - You specify a value greater than 1  
  - All image copy information is found  
  - Image copies are not on the same stacked tape  
  **Note:** This parameter is ignored if you select Hardware Mirroring Level 2. |
### Field | MAX RECOVERY MANAGER Default | Description
--- | --- | ---
Maximum log jobs per member | 1 | Specifies the maximum number of jobs (1-10), as follows:
- Specifies the maximum number of jobs to use for initialization if you specified Yes in the Initialize active logs field.
- Specifies the maximum number of jobs to use for archive copies if you specified Yes in the Restore archive copies to disk field.

**Note:** Increasing the number of jobs decreases the amount of time it takes to initialize or copy the logs, provided there are a sufficient number of tape drives available.

Recover Log Master Repository | Yes | Recovers the Log Master for DB2 repository if available
If you choose not to recover the repository, this step is omitted from the recovery job.

Recover Work File Database | Yes | Recovers work file database (DSND807) objects during local or remote subsystem recovery
If you choose not to recover work file database objects, this step is omitted from the recovery job.

Library that contains DSNTEP2 | Yes | Specifies the library that contains DSNTEP2
**Tip:** BMC recommends that you choose No if your DSNLOAD library does not contain DSNTEP2.

Recover RM Repository | Yes | Recovers the BMC Common DB2 repository, RECOVERY MANAGER repository, and the R+/CHANGE ACCUM repository (if installed) and terminates BMC utilities
If you choose not to recover the repository, this step is omitted from the recovery job.

**Note:** You must recover the repository at the disaster recovery site to use the data collection feature of the BMC Recovery Management for DB2 solution.
<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remote Site has single LPAR</td>
<td>No</td>
<td>Generates JCL that does not include the SYSAFF cards designed to route different JCL steps to different members. Use this option if your system is data-sharing and all members are running on a single LPAR at the remote site. If you do not specify this option, RECOVERY MANAGER generates JCL for the remote site as if the remote site configuration matches the local configuration. If your system is non-data-sharing, this option is not available.</td>
</tr>
<tr>
<td>Restore archive copies to disk</td>
<td>0--</td>
<td>Specifies whether archive logs are to be restored to disk at the recovery site, as follows: 0— no copies to be restored 1— one copy to be restored Note: If you specify 0, this item is omitted from the recovery job. If you specify 1, the Disaster Recovery Offsite Copy 1 panel appears when you press Enter. With either choice, RECOVERY MANAGER automatically deletes and creates the archive history file to ensure that it does not contain old information. You can also specify the number of log data sets to restore to disk and the program to be used for restoration, as follows: Number of logs to disk—specifies how many log data sets will be restored to disk at the recovery site (1 through 999). You can also specify the number of hours (1 through 999) or days (1 through 99) to process. Restore program—specifies either PACLOG or RMGR (ARMBARC) (the default) as the program to be used for the restore. If you choose PACLOG, that utility must be installed at the recovery site.</td>
</tr>
<tr>
<td>Synchronization file name</td>
<td>userId.Ddate.Ttime.BMCSYNC</td>
<td>Specifies the fully-qualified name of the data set to be used to synchronize any generated jobs that run in parallel Do not use quotation marks around the name you enter.</td>
</tr>
</tbody>
</table>
Field definitions—offsite copy options

The following table describes fields found on the Offsite Copy Options panel.

**Table 34: Offsite copy fields**

<table>
<thead>
<tr>
<th>Fields</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk unit</td>
<td>SYSALLDA</td>
<td>Specifies the name of the disk drive unit to be used at the recovery site for restoring archives to disk</td>
</tr>
<tr>
<td>SMS data class</td>
<td>None</td>
<td>Specifies a valid SMS data class name for this copy (8 characters or less)</td>
</tr>
<tr>
<td>SMS management class</td>
<td>None</td>
<td>Specifies a valid SMS management class name for this copy (8 characters or less)</td>
</tr>
<tr>
<td>SMS storage class</td>
<td>None</td>
<td>Specifies a valid SMS storage class name for this copy (8 characters or less)</td>
</tr>
<tr>
<td>Compression</td>
<td>No</td>
<td>Compresses the data when restoring this copy to disk</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> You must select No when you have DASD hardware compression enabled.</td>
</tr>
</tbody>
</table>

Field definitions—job card information

The following table describes fields found on the Job Card Information panel.
Table 35: Job card information fields

<table>
<thead>
<tr>
<th>Fields</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| Alternate output GDG base Unit Volume Device type Alternate output model | None (BMC Recovery Management for DB2 solution only) | The alternate output fields are only available if the BMC Recovery Management for DB2 password is activated and if you have selected one of the mirroring strategies on a previous panel.

The alternate output base and models are used for JCL to restore both mirrored and nonmirrored application and system resources during a disaster recovery. (The primary output data sets contain only nonmirrored resources because RECOVERY MANAGER assumes the mirrored resources are already available at the recovery site.) Provide the name of a cataloged data set to define the model DCB for the output data set. See the definitions for the primary output fields for information about using the alternate output fields. |
| JCL data set name    | None                      | Specifies the fully qualified name of the data set name in which the recovery JCL will be stored. If you choose to use a GDG data set, you must enter the incrementing term (+1). |
**Field definitions—ARMBGPS groups panel**

The following table describes fields on the ARMBGPS groups panel for application groups created by ARMBGPS (groups created automatically using the subsystem recovery feature).

**Table 36: ARMBGPS groups fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| Generate RECOVER JCL| None                      | Generates application group recovery JCL. The recovery point is the restart RBA. The JCL will recover all groups with names in the following pattern: 
`groupOwner. groupNamePrefix??`

where ?? is a two-digit numerical suffix.
<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group name prefix</td>
<td>Last used</td>
<td>Specifies a character string to be used by RECOVERY MANAGER as a prefix in the group part of each group name. This value defaults to the string last entered at the Group name prefix prompt on the Build Recovery Groups panel. Note: Group name prefixes cannot be delimited.</td>
</tr>
<tr>
<td>Group owner</td>
<td>Last used</td>
<td>Specifies the owner of the subsystem groups created by ARMBGPS. This value defaults to the user ID last used at the Group owner prompt on the Build Recovery Groups panel.</td>
</tr>
<tr>
<td>JCL data set name</td>
<td>None</td>
<td>Specifies the fully qualified name of the data set name in which the recovery JCL will be stored. If you choose to use a GDG data set, you must enter the incrementing term (+1).</td>
</tr>
<tr>
<td>Primary output model</td>
<td>None</td>
<td>Specifies the name of the cataloged model DSCB for the specified GDG data set.</td>
</tr>
</tbody>
</table>

**RECOVERY MANAGER disaster recovery programs**

RECOVERY MANAGER provides the following programs for use at the local site to help you plan for disaster recovery:

**Table 37: Disaster recovery programs**

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARMBTSI</td>
<td>Inserts a row containing a user-specified timestamp into the RECOVERY MANAGER CRRDRPT table. The row is used to determine the RBA or LRSN to be used in recovery. For more information, see “ARMBTSI—Time stamp insertion” on page 719.</td>
</tr>
<tr>
<td>ARMBCOR</td>
<td>Manipulates the value of the ARMBSDR member in the CNTL data set to ensure that all data sharing members are processed. The JCL generated for ARMBCOR should not be modified. (ARMBCOR supports the feature to extend the recovery point at the disaster recovery site.) For more information, see “ARMBSRR—System resource recovery” on page 673.</td>
</tr>
<tr>
<td>Program</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ARMBCRC</td>
<td>Performs log analysis for coordinated disaster recovery to determine the RBA or LRSN required for recovery at a local site using a DB2 conditional restart. For more information, see “ARMBCRC—Conditional recovery to a timestamp” on page 425.</td>
</tr>
<tr>
<td>ARMBLOG</td>
<td>Executes a DB2 -ARCHIVE LOG command and then waits for the archive offload to complete. For more information, see “ARMBLOG—Archive log creation” on page 603.</td>
</tr>
<tr>
<td>ARMBARC</td>
<td>Makes recovery site copies of cataloged archive logs and identifies image copies of the catalog spaces that are registered in the DB2 log (SYSCOPY, SYSUTILX, SYSDBDXA, and DBD01). For more information, see “ARMBARC—Archive log data sets” on page 409. <strong>Note:</strong> If you have the PACLOG product installed at the local site, you can use it instead of ARMBARC. For more information, see the PACLOG for DB2 Reference Manual.</td>
</tr>
<tr>
<td>ARMBSDR</td>
<td>Finds the most recent BSDS and archive logs at the disaster recovery site (for each member if data sharing) and updates the BSDS. ARMBSDR also adds a new conditional restart control record to the BSDS. ARMBSRR generates ARMBSDR JCL. (ARMBSDR supports the feature to extend the recovery point at the disaster recovery site.) For more information, see “ARMBSRR—System resource recovery” on page 673.</td>
</tr>
<tr>
<td>ARMBSRR</td>
<td>Generates JCL for systems resource recovery using your cataloged archive logs. For more information, see “ARMBSRR—System resource recovery” on page 673.</td>
</tr>
<tr>
<td>ARMBGPV</td>
<td>Reports the available recovery resources and validates recoverability. For more information, see “ARMBGPV—Group recovery revalidation” on page 489.</td>
</tr>
<tr>
<td>ARMBGEN</td>
<td>Generates JCL for application data recovery. For more information, see “ARMBGEN—Backup and recovery JCL” on page 431.</td>
</tr>
<tr>
<td>ARMBWDC</td>
<td>Writes data collection information about recovery start and end times for actual, estimated, and simulated disaster recoveries of system resources. This program is only available with the BMC Recovery Management for DB2 solution. For more information, see “ARMBWDC—System recovery data collection” on page 725.</td>
</tr>
<tr>
<td>ARMBRDC</td>
<td>Reads data collection information about the disaster recovery jobs, populates the UTILITY_RUN table, and generates the data collection reports. This program is only available with the BMC Recovery Management for DB2 solution. For more information, see “ARMBRDC—Recovery data collection report” on page 615.</td>
</tr>
</tbody>
</table>
Running and restarting DB2 conditional restart recovery jobs

The following subsections describe the conditional restart recovery jobs generated by ARMBSRR and describes procedures for running and restarting them.

For more information about each batch program and for sample JCL, see “ARMBSRR—System resource recovery” on page 673.

Overview of ARMBSRR-generated jobs

ARMBSRR generates the following sets of jobs to perform a conditional restart recovery of a DB2 subsystem or data sharing group:

- Phase 1 jobs—run while DB2 is down
- Phase 2 jobs—run after DB2 is restarted in MAINT mode.
- Data collection jobs—run after application data recovery (Recovery Management solution only)

The jobs generated by ARMBSRR are written to a single PDS member or sequential file. ARMBSRR requires the job card to contain the symbolic variable &## to allow it to number the jobs it creates. The job card that ARMBSRR uses in the generated JCL is specified via the ARMJCIN DD statement.

Each job contains a comment to indicate whether it is a Phase 1, Phase 2, or data collection job and what its job number is within that phase. An example follows:

```/* DISASTER RECOVERY FOR SYSTEM RESOURCES - PHASE 2 JOB 2 */```

**Note**

If you are performing a recovery simulation, only a portion of the Phase 1 job set runs and Phase 2 does not run at all. If you are performing a recovery estimation, the ARMBWDC and ARMBRDC data collection jobs are run. You release the ARMBRDC job from HOLD after the successful completion of the application recovery. Both recovery simulation and estimation are features of the BMC Recovery Management for DB2 solution. For more information, see the Recovery Management for DB2 User Guide.

About Phase 1

There is at least one job per subsystem in Phase 1.
For data sharing there is at least one job per member. The jobs are numbered 1 to \( n \), where

- \( n \) is the number of members in the data sharing group
- 1 indicates a non-data-sharing environment

If you specify MAXLOGJOBS greater than one, RECOVERY MANAGER generates additional jobs for each subsystem to provide for parallel log copies to disk. The log copy jobs are numbered sequentially beginning with \((n + 1)\). A maximum of 32 total jobs is allowed.

**Note**

RECOVERY MANAGER performs stacked tape analysis prior to creating the Phase 1 JCL. The number of log copy jobs may vary based on this analysis and may possibly be less than you requested with the MAXLOGJOBS option.

When the JCL is submitted, the Phase 1 jobs begin executing immediately. If copying logs to disk, additional jobs are submitted to the internal reader at the end of the initial Phase 1 jobs. A Phase 2 job is placed on hold while the Phase 1 jobs execute. If you used the local subsystem recovery option to generate application recovery JCL as well, there is a second job 01 on hold that will be used to create application recovery JCL.

If you are using the BMC Recovery Management for DB2 solution, a data collection job is also placed on hold. Also, data collection information is written to a flat file during Phase 1 processing.

**Example - Phase 1 execution**

To illustrate, assume that you have a two-member data sharing system, MAXLOGJOBS 3, and a job name of BMCBSR&##. When the JCL is submitted, you see jobs 01 and 02 begin executing immediately. You also see a job 01 on hold—this is the first Phase 2 job. As one of its final steps, job 01 submits jobs 03 and 04 to copy logs for member 1. Job 01 itself also copies some of the logs, resulting in a total of 3 jobs that copy logs for member 1. Job 02 submits jobs 05 and 06 to copy some of the logs for member 2, and job 02 itself copies the remainder.
No synchronization between Phase 1 jobs is required. The only requirement is that they must all complete before starting DB2 in MAINT MODE and before releasing the Phase 2 job that is on hold.

Figure 29: Phase 1 execution (2 member data sharing, MAXLOGJOBS=3)

### About Phase 2

When Phase 1 jobs are complete, follow the instructions in the JCL for clearing the Coupling Facility for data sharing and starting DB2. You may then release the Phase 2 job to begin executing.

If only a single job is needed by Phase 2, it executes immediately.

Phase 2 is performed by:

- Multiple jobs for data sharing
  - For data sharing environments, there is at least one job per member.

- One job with multiple tasks, for DB2 Version 10 and later, when you specify a value for MAXCATJOBS greater than one
  - RECOVERY MANAGER uses the value of MAXCATJOBS for PARALLEL and TAPEUNITS to perform multiple tasks in one job.

**Example**

- A two-member data sharing group has at least two jobs (one for each member).
- A non-data sharing system with MAXCATJOBS=3 has one job that performs three tasks.
Some conditions can prevent concurrent jobs for catalog recovery such as stacked tape.

When multiple jobs are required for Phase 2, the first job that executes is the one that was placed on hold initially during Phase 1. It allocates a synchronization file that is used by the subsequent Phase 2 recovery jobs to monitor and synchronize the work between jobs. The first job then submits the actual Phase 2 recovery jobs. Once it has submitted the other jobs, it ends.

The first action of the Phase 2 first job is to submit a synchronization cleanup job also named 01. The synchronization cleanup job runs after Phase 2 recovery job 01 completes. If all jobs run successfully, the cleanup job then deletes the synchronization file. For data sharing groups, a Phase 2 job executes for each member and is routed to the system on which its corresponding member ran at the local site. There may also be additional jobs for catalog recovery as previously described. These jobs utilize the synchronization program and wait to execute at the appropriate time in the process.

Note

Note that the SYSAFF= needs to be changed for JES3 or if the members are run in a different system configuration than the local site.

If Phase 2 completes successfully, a DB2 STOP command is issued. You then start DB2 for normal access to begin the application recovery process.

At this point, if you have a job to generate application recovery JCL (ARMBGEN) on hold, you should release it when the DB2 start has completed successfully. Generating recovery JCL at this point is expected for Full Subsystem Local Recovery. (Disaster recovery procedures typically generate the JCL at the local site as part of the preparation process.)
If you are using the Recovery Management solution, the Phase 2 jobs should all complete before you release the data collection job. Also, data collection information is written to data collection tables during Phase 2 processing.

**Figure 30: Phase 2 execution**

**About data collection jobs**

For the Recovery Management solution only, data is collected about the recoveries throughout the disaster recovery process.

During Phase 1, the data about the system resource recoveries is written to a flat file. During Phase 2, the data is written to the data collection tables. After all application data is recovered, the data collection jobs run. These jobs consolidate all data into the tables and create a file of SQL statements that you can use to populate the data collection tables at the local site. For more information, see the *Recovery Management for DB2 User Guide*.

**Figure 31: Data collection**
Rerun or restart

If any of the generated jobs fail, you must first determine what caused the failure and correct the situation.

Then you should decide whether to rerun the entire job stream (by resubmitting the generated JCL) or restart the jobs at the point of failure. RECOVERY MANAGER provides an EDIT macro called ARMSBSRR to assist in rerunning or restarting the failed jobs.

Once the JCL created by RECOVERY MANAGER is split into separate members, you can decide whether you wish to rerun the phase that failed or restart the phase at the point of failure.

Note
Sometimes it is quicker to resubmit the generated JCL than to identify step restarts for each recovery job.

Creating separate jobs for restart

The ARMSBSRR macro separates the jobs contained in the ARMBSRR-generated JCL into distinct PDS members.

It uses the default prefix of JOB for the member names it creates. All Phase 1 jobs are placed into member JOBP1. If Phase 2 consists of only one job, the ARMSBSRR macro creates only one member for Phase 2: JOB001. If Phase 2 consists of multiple jobs, the Phase 2 jobs are placed into n + 1 members using sequential numbering. JOB000 is the Phase 2 restart job used to reset the synchronization file and monitor job completion, and JOB001 - JOB00n are the separate Phase 2 recovery jobs generated for DB2 catalog and directory recovery.

Before you begin

To use the ARMSBSRR macro, you must copy the ARMSBSRR member from the RMGR.CLIST library to a library in your SYSPROC concatenation.

To create separate jobs for restart

1. Copy the ARMSBSRR-generated JCL into a member in a PDS library.
2. Get into ISPF EDIT mode on the PDS member you just created.
3 Type ARMSBSRR `prefix` on the command line, where `prefix` is an optional 1 to 5 character prefix to use for the member name. If you do not specify `prefix`, ARMSBSRR creates members with a prefix of JOB.

---

**Note**

Be aware of the following information:

- Member `prefixPH1` contains the Phase 1 job(s). Members `prefix000 - prefix00 n` contain the Phase 2 jobs.

- When ARMSBSRR completes, it CANCELs out of ISPF EDIT mode, leaving the RECOVERY MANAGER-generated JCL intact.

---

**Phase 1 job failure**

This topic describes procedures to follow if a failure occurs during Phase 1.

1 Before you begin, determine what caused the job(s) to fail and correct the situation.

2 Execute the ARMSBSRR EDIT macro to isolate the Phase 1 job(s) into a separate JCL member (see “Creating separate jobs for restart” on page 342).

3 Either rerun or restart Phase 1, as follows:

   a Rerun Phase 1 by submitting member `prefixPH1` (JOBPH1 if you did not specify a prefix).

   b Restart Phase 1 by editing the job card and adding a RESTART= parameter in the failing step.

---

**Note**

If you are running in simulation mode, correct the problem and resubmit the job.

---

**Phase 2 job failure - rerunning Phase 2 jobs**

This section describes procedures to follow if a failure occurs during Phase 2, and you wish to run the job(s) again (as opposed to restarting the jobs).

1 Determine what caused the job(s) to fail and correct the situation.
You may need to issue the DB2 TERM UTILITY command if the failed step was a DSNUTILB execution.

Make a copy of the original ARMBSRR-generated JCL as a backup.

Edit the ARMBSRR-generated JCL. Delete all Phase 1 jobs from the JCL.

---

**Note**

Phase 1 jobs precede all Phase 2 jobs in the JCL, and each job contains a comment that indicates its phase. You can easily locate the first Phase 2 job by searching for the phrase PHASE 2 JOB 1 and then delete all JCL prior to that phrase in the file.

---

Submit the edited JCL, which then reruns Phase 2. It also deletes and reallocates the synchronization file.

### Phase 2 job failure - restarting Phase 2 jobs

This section describes procedures to follow if a failure occurs during Phase 2 and you wish to restart the jobs (as opposed to running them again from the beginning.) If there is more than one job in Phase 2, the jobs utilize program ARMBSYN to synchronize execution between the jobs.

1. For each recover job (1 - n), locate the job output and note the step name in which the job got the first return code greater than 4. Determine what caused the job(s) to fail and correct the situation.

---

**Note**

Some jobs may have completed successfully and do not need to be rerun.

---

2. If the original JCL is not in a PDS, copy it to a PDS.

3. Execute the ARMSBSRR EDIT macro to isolate the Phase 2 job(s) into a separate JCL member (see “Creating separate jobs for restart” on page 342).

4. If ARMSBSRR created only JOB001 (Phase 2 is not a set of jobs), skip to Step 6 on page 345. Otherwise, proceed to Step 5 on page 344.

5. Edit member JOB000 (or prefix000). ARMSBSRR has added the following card to the JCL directly after the job card:

```
// RESTART=ARM0002.ARMSYNC
```

Perform the following substeps:

a. Place a comma at the end of the job card and verify the restart card is immediately after the job card. Do not change the step name on the RESTART
The job is now ready to prepare the synchronization file for restart and clean up again at the end as needed.

b Submit this job before you submit the other Phase 2 recover jobs. It waits on the other jobs to complete before it completes.

6 Edit member JOB001 (or prefix001.) ARMSBSRR has added the following card to the JCL directly after the job card:

```
// RESTART=ARM?????
```

Perform the following substeps:

a Place a comma at the end of the job card and ensure the restart card is located immediately after the job card.

b Change the ARM???? on the restart card to the step name of the first step that received a return code greater than 4 (if any) in the original execution.

7 If the failing step was DSNUTILB, you may need to add RESTART to the parameter list on the DSNUTILB execution.

For example, if the subsystem ID is DBDF, the parameter list should look like this:

```
EXEC PGM=DSNUTILB,PARM='DBDF,,RESTART'
```

Note
Adding the RESTART parameter is required only if DSNUTILB was started and placed in a STOPPED status at the time of the failure.

8 If there was only one PHASE 1 job, you are now finished. Otherwise, repeat Step 5 on page 344 through Step 7 on page 345 for each job (2 - n), then proceed to Step 9 on page 345.

9 Submit member JOB000 if you have not already done so (see Step 5 on page 344).

10 Submit members JOB001 - JOB00 n that require restart due to previous failures.

Note
If for any reason you need to restart again, change the restart parms as required in JOB001 - JOB00n and resubmit JOB000 - JOB00 n.
Full subsystem recovery

This chapter describes the full subsystem recovery process.

Overview

This chapter discusses the use of the automation that is provided by the RECOVERY MANAGER product for the backup and local recovery of an entire DB2 subsystem.

(For information about disaster recovery, see “Recovering from a DB2 system disaster” on page 291) RECOVERY MANAGER provides an automated process to create groups and generate backup and recovery JCL for an entire DB2 subsystem or data sharing group (hereafter simply referred to as a subsystem).

---

**Note**

As a component of the BMC Recovery Management for DB2 solution, RECOVERY MANAGER supports:

- Hardware mirroring in full subsystem recoveries
- Conditional restart avoidance for faster more efficient recoveries

For more information, see the *Recovery Management for DB2 User Guide*.

RECOVERY MANAGER performs a log range analysis to identify objects that have not changed between the current time and the recovery point. This allows RECOVERY MANAGER to avoid the unnecessary recovery of unchanged objects and can dramatically improve processing time. RECOVERY MANAGER also uses log range analysis to avoid unnecessary backups, thus reducing backup time and resources when table spaces change infrequently.

Full local subsystem recovery is particularly useful for SAP applications. SAP comprises a large number of DB2 objects and generally is the only application in the subsystem. In addition, SAP performs a high degree of dynamic creation and deletion of DB2 objects, which impacts the contents of the DB2 catalog and directory. Its unique nature leads to the necessity of recovering the entire subsystem.
This chapter discusses the backup and recovery of all table spaces in a DB2 subsystem. You can perform all procedures through the RECOVERY MANAGER online interface or by using the batch programs. RECOVERY MANAGER provides the following programs:

- The ARMBGPS program divides all table spaces in the system into balanced groups for backup and recovery purposes. ARMBGPS also creates a delta group (00 group), which will initially be empty, but because the definition of the group is dynamic, will automatically pick up any newly-created objects. You decide how many groups are necessary to process your backups (and recoveries) in a timely fashion given the resources available on your system. If you are using the NGT Copy and NGT Recover utilities, indexes meeting a specified size threshold can be included in the backup and recovery jobs. For more information about ARMBGPS, see “ARMBGPS—Subsystem group split” on page 473.

- The ARMBGEN program can generate the following jobs:
  - Back up the entire subsystem
  - Back up only the new and changed objects
  - Recover the entire subsystem to a previous point in time
  - Simulate recovery of the entire subsystem to a previous point in time
  - Recover the new and changed objects to a previous point in time

For more information about ARMBGEN, see “ARMBGEN—Backup and recovery JCL” on page 431.

**Backup strategy**

In general, BMC recommends a full backup of the entire subsystem at regular intervals and full backups of the newly created spaces and the changed spaces on a
more frequent basis. If you are willing to retain the backups for longer periods of time, you can lengthen the interval between full subsystem backups.

**Example**

Example strategy:

- Sunday—full backups of the entire subsystem (full subsystem backup)
- Daily (except Sunday) --full backups of the new and changed spaces only (XUNCHANGED backup)

---

**Full subsystem backup**

When you make a backup of the entire subsystem, RECOVERY MANAGER uses the ARMBGPS program to split all table spaces in the subsystem into balanced groups.

By default, ARMBGPS creates eleven separate group (ten static groups and one dynamic delta group that picks up newly-created objects), but you can decide how many groups are necessary to back up your entire subsystem in a reasonable time frame. You can change the number of groups at any time to respond to changing backup windows or expansion or reduction of available resources. RECOVERY MANAGER then uses the ARMBGEN program to generate the JCL to back up all objects in all groups.

**Note**

You can perform the full subsystem backup using the RECOVERY MANAGER online interface or you can run the batch programs manually. For instructions on the procedure when using the online interface, see “Build subsystem groups and generate backup JCL” on page 351.

---

**XUNCHANGED backup**

RECOVERY MANAGER uses the ARMBGEN program with the XUNCHANGED option to determine which table spaces have changed and then to generate JCL to back up only the changed or new spaces in the groups that were created previously by ARMBGPS.

ARMBGEN does not generate a backup for spaces that are no longer defined to DB2.
You can perform the XUNCHANGED backup using the RECOVERY MANAGER online interface or you can run the batch programs manually. For information about ARMBGPS, see “ARMBGPS—Subsystem group split” on page 473.

Implementing the subsystem backup strategy

Use the following procedures to implement the full subsystem backup strategy by using the RECOVERY MANAGER batch programs.

To perform full subsystem backups

1 Run ARMBGPS to create balanced groups of all table spaces in the subsystem (sample syntax shown below). For instructions for using the online interface, see “Build subsystem groups and generate backup JCL” on page 351. (This step also creates the delta group, which excludes all other groups.)

```
SET CURRENT SQLID=RDAXXX;
BUILD GROUPS RDAXXX.ALLTS
MAXGROUPS 10
EXCLUDE BMC*.*;
```

RECOVERY MANAGER generates an EXCLUDE list based on plan dependencies for the repository.

2 Run ARMBGEN to generate JCL for the balanced groups and for any objects that are in the delta group (sample syntax shown below).

```
SET CURRENT SQLID = RDAXXX;
GENJCL BACKUP GROUP RDAXXX.ALLTS?? ;
```

3 Submit the ARMBGEN-generated JCL (created in Step 2 on page 350).

4 Submit JCL to copy the repository (including the R+/CHANGE ACCUM repository, if installed).

To perform XUNCHANGED backups

1 Add the XUNCHANGED parameter to the ARMBGEN JCL and run ARMBGEN to generate the backup JCL (sample syntax shown below).

```
SET CURRENT SQLID = RDAXXX ;
GENJCL BACKUP XUNCHANGED GROUP RDAXXX.ALLTS?? ;
```

2 Submit the ARMBGEN-generated JCL (created in Step 1 on page 350).
Build subsystem groups and generate backup JCL

This process automatically generates JCL to build groups for all application data within a subsystem (via the ARMBGPS program), then generates one backup job per group (via ARMBGEN).

These procedures show you how to build all JCL required for backup of the full subsystem using the RECOVERY MANAGER online interface.

Before you begin

- Set or verify the subsystem options for the copy utility you are going to use. For more information, see “RECOVERY MANAGER backup and recovery options” on page 94.

- Set or verify the RECOVERY MANAGER option set.

  **Note**

  ARMBGEN uses the job card and other information in the option set for the generated backup jobs.

- Run either the IBM utility, RUNSTATS, the NGT Copy utility with the RUNSTATS option, or NGT Copy with the NACTIVE option to obtain an accurate estimate of the number of pages that should be used for sizing purposes before you build the groups. Alternatively, you can set the NACTIVE option in SYSIBM.SYSTABLESPACE for any large spaces until statistics have been collected. If these steps are not done, the optimal balance of objects may not be achieved.

- Ensure you have the following authorizations:

  — EXECUTE authority for the RECOVERY MANAGER plan

  — UPDATE authority to replace existing groups
—EXECUTE authority on R+/CHANGE ACCUM plan if you are building R+/CHANGE ACCUM groups.

**To build subsystem recovery groups**

Start this procedure at the RECOVERY MANAGER Main Menu to build all JCL for subsystem backups.

1. Access the Build Recovery Groups panel as follows:
   a. Select **Subsystem recovery**.
   b. Select **Full Recovery groups**.

   The Build Recovery Groups panel is displayed.

   ARMUFS1 ==================== Build Recovery Groups ============================
   Command ===> _________________________________________________________________
   You are about to generate JCL to build RMGR recovery groups based on a sizing balance. Enter the fields and press Enter.
   Group owner . . . . . . . . . . . . . . RDAXXX__
   Group name prefix . . . . . . . . . . . TEST____________
   Group description . . . . . . . . . . . _________________________
   Maximum number of groups. . . . . . . . 10 (2-99)
   Build job for backup JCL generation . . 1 1. Yes 2. No
   Output data set. .  RDAXXX.ARM.JCL________________________________________
   Copy All Index Spaces . . 2 1. Yes 2. No 3. Auto
   Index Size Threshold ______ max bytes(4294967295K,4194303M,4095G)
   Include Clones. . . . . . . 2 1. Yes 2. No
   Output data set. .  ______________________________________________________

2. Enter information as required and press **Enter**. For field descriptions, see Table 38 on page 353.

3. On the JCL Specification panel, enter a fully qualified output data set name. Be aware of the following information:

   - The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.
   - The job statement must contain a symbolic variable (&#) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

4. *(optional)* Edit the control cards to exclude additional table spaces from the groups. The names may include wildcards. For syntax information, see “ARMBGPS—Subsystem group split” on page 473.
RECOVERY MANAGER automatically adds EXCLUDE statements for the BMC Common DB2, RECOVERY MANAGER, R+/CHANGE ACCUM, and Log Master repositories based on plan dependencies.

5 *(optional)* If you generated backup JCL, you can edit the ARMBGEN job to include a //ARMJCIN DD statement that points to a data set containing a job card to be used in the generated JCL. (The job card must include the variable &## in the job name so that ARMBGEN can increment the job numbers.) If you do not include the DD card, RECOVERY MANAGER uses the job card that you specified in the option set.

For more information about ARMBGEN and ARMJCIN, see “ARMBGEN—Backup and recovery JCL” on page 431.

6 Submit the JCL to create the subsystem groups (and the backup jobs for those groups). The subsystem groups are created and saved in the repository with the description, **GENERATED BY ARMBGPS**. The backup jobs are saved in the data set you specified in the **Output data set** field.

**Note**

RECOVERY MANAGER generates one backup job per subsystem group. These jobs use the copy utility and its options as specified in the subsystem backup options. (See “Subsystem recovery process” on page 356 for more information.)

7 To create copies of application data, submit the backup jobs created in **Step 6 on page 353**.

**Table 38: Subsystem recovery group fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group owner</td>
<td>Last value used</td>
<td>Specifies the TSO user ID to be used by RECOVERY MANAGER as the creator_ID part of each group name. For more information about authorization for creating groups, see “Group authorization” on page 90.</td>
</tr>
</tbody>
</table>

Chapter 9  Full subsystem recovery  353
<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group name prefix</td>
<td>Last value used</td>
<td>Specifies an alphanumeric prefix to be used as the group part of each group name. RECOVERY MANAGER adds a 2-digit number suffix to provide a unique name for each group. The prefix must not exceed 16 characters. The numerical suffixes provided by RECOVERY MANAGER start at 01 and continue up to the value you provide at the <strong>Maximum number of groups</strong> prompt. <strong>Note:</strong> Delimited entries are not allowed for the group name prefix.</td>
</tr>
<tr>
<td>Maximum number of groups</td>
<td>10</td>
<td>Specifies the maximum number of groups (2-99) to be created. The number of groups created will always be the maximum plus one because a delta group (00 group) is always created.</td>
</tr>
<tr>
<td>Build job for backup JCL generation</td>
<td>Yes</td>
<td>Builds a job to create backup JCL for the full subsystem.</td>
</tr>
<tr>
<td>Output data set</td>
<td>Last value used</td>
<td>Specifies the fully qualified name of a new or existing data set in which you want to place the backup JCL for the groups being created. New data sets are allocated on the work unit specified in the option set. If you are using a partitioned data set, be sure to include the member name. If you are using a generation data group (GDG), be sure to include the incrementing term (+1).</td>
</tr>
<tr>
<td>Copy All Index Spaces</td>
<td>No</td>
<td>Specifies whether to set group options that will cause index spaces to be included in the backup and recovery JCL when generated. This option is only available when RECOVER PLUS version 3.5 or later is selected as the recovery utility and NGT Copy is selected as the copy utility.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Yes—Back up all indexes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ No—(the default) Do not back up indexes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Auto—Back up indexes as large or larger than the size specified in the <strong>Index Size Threshold</strong> field.</td>
</tr>
</tbody>
</table>
### Recovery strategy

When an event occurs that creates the need to restore all table spaces to a prior point in time, RECOVERY MANAGER can create all of the jobs required to perform the recovery.

RECOVERY MANAGER uses DB2 RESTART to recover the subsystem to a point of consistency. It also uses ARMBGEN to create recovery JCL that excludes recovery of table spaces that have not changed in the interval of time between the selected

<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| Index Size Threshold         | Not applicable | Specifies the size threshold at which indexes are to be included in the backup and recovery JCL rather than rebuilt at recovery time. This option is used in conjunction with **Copy Index Space Auto**. Enter the size in the following format: `nnnnnnn t` where:  
  - `nnnnnnn` is an integer from 0-4194303 (depending on the size type specified and version of NGT Copy)  
  - `t` is the size type (M=Megabytes, K=Kilobytes, G=Gigabytes)  
  - M - Megabytes (the default). Valid range is 0-4194303.  
  - K - Kilobytes. Valid range is 0-4294967295.  
  - G - Gigabytes. Valid range is 0-4095. |
| Include Clones               | No                         | Specifies whether to create JCL to back up the cloned objects in the groups being created. Cloned objects are backed up separately from base objects. This option is available only when running on DB2 Version 10 or later and is not valid with compatibility mode. |
| Output data set (for clone groups) | Not applicable | Specifies the fully qualified name of a new or existing data set in which to place the backup JCL for the clones |
recovery point and the current time. The omission of needless recoveries allows the subsystem to be available again in the minimum amount of time.

### Subsystem recovery process

The following tasks are used to perform a subsystem recovery.

*Note*

As a component of the Recovery Management for DB2 solution, RECOVERY MANAGER supports conditional restart avoidance for faster, more efficient full subsystem local recoveries

For more information, see the *Recovery Management for DB2 User Guide*.

#### To recover a subsystem process

1. Select a point in time for recovery. This time is stated in date and time to the hundredths of seconds.

2. Execute the procedure, “Generating JCL for local recovery” on page 357 to create the JCL to recover the DB2 subsystem.

3. Stop all activity on the subsystem.

4. Submit the JCL created by Step 2 on page 356. The steps of this job execute the following programs:
   
   a. ARMBLOG—executes the appropriate DB2-ARCHIVE LOG command and waits for the archive offload to complete.
   
   b. ARMBTSI—inserts the desired point in time for recovery into the RECOVERY MANAGER CRRDRPT table.
   
   c. ARMBCRC--converts the point in time to an RBA or LRSN value for DB2 restart.
   
   d. ALPMAIN—Log Master searches for DDL and quiet points to be analyzed by ARMBSRR for catalog recovery. A Recovery Management password is required.
   
   e. ARMBSRR—creates recovery JCL for the DB2 catalog and directory and for the repository.
   
   f. ARMBLGR—if XUNCHANGED is requested, analyzes the log ranges and updates the ARMLRNG file (the RECOVERY MANAGER log range file).
The final step of this job stops DB2 again.

5 Execute the JCL generated by ARMBSRR in the previous step to recover the DB2 catalog and directory and the repository to the chosen point in time.

The JCL submits the first phase job(s) which execute immediately. It then submits a second job on hold. Follow the instructions in the JCL, which include starting DB2 after the first phase jobs complete and then releasing the second phase jobs. This is the same process used for disaster recovery to a coordinated recovery point. For more information on the JCL generated by ARMBSRR, see “ARMBSRR—System resource recovery” on page 673. For information about running and restarting ARMBSRR jobs, see “Running and restarting DB2 conditional restart recovery jobs” on page 337.

6 After the subsystem recovery, release the first held job to execute the ARMBGEN program.

The ARMBGEN program creates the required recovery JCL for all groups including the delta group created by ARMBGPS. ARMBGEN also reads the ARMLRNG file (the RECOVERY MANAGER log range file) and excludes from recovery any table spaces that are unchanged since the last backup.

7 Submit the generated recovery jobs.

Use the procedures described in “Generating JCL for local recovery” on page 357.

For more information about restarting synchronized ARMBGEN jobs, see “Restarting a recovery for a set of concurrent jobs” on page 223.

Generating JCL for local recovery

This process generates JCL to recover the entire DB2 system to a prior point in time.

The process is intended for a local recovery of a DB2 system that is currently active. It uses a conditional restart of the subsystem followed by recovery of all DB2 data. In addition, it generates revalidation JCL that ensures the recoverability of the objects and produces a report showing objects that were included in the backup, but that no longer exist.

RECOVERY MANAGER includes batch log range analysis in the JCL it generates to recover the subsystem. By identifying objects that have not changed (XUNCHANGED) between the current time and the recovery point, RECOVERY
MANAGER can avoid the unnecessary recovery of unchanged objects and can dramatically improve processing time. Be aware that XUNCHANGED processing does not occur for indexes unless they have the COPY YES attribute. For more information about XUNCHANGED processing, see “About XUNCHANGED processing in local subsystem recovery” on page 432. For more information about batch log analysis, see “ARMBLGR—Log range analysis” on page 597.

**Note**

As a component of the BMC Recovery Management for DB2 solution, RECOVERY MANAGER supports conditional restart avoidance for faster, more efficient full subsystem local recoveries.

For more information, see the *Recovery Management for DB2 User Guide*.

**Before you begin**

Make sure you have performed the following tasks:

- Build application groups for the subsystem and back them up (see “Build subsystem groups and generate backup JCL” on page 351). If this was not done prior to the recovery point, this process cannot be used.

- Set or verify the subsystem options for the utilities you are going to use. For more information, see “RECOVERY MANAGER backup and recovery options” on page 94.

- Set or verify the RECOVERY MANAGER option set information.

- Ensure that you have authority for the following:
  - EXECUTE authority for the RECOVERY MANAGER plan
  - DB2 -ARCHIVE LOG command
  - APF authorization for the following:
    - ARMBSRR
    - ARMBARC
    - ARMBTSI
    - ARMBCRC
    - ARMBGEN
    - ARMBLGR
    - ALPMAIN

**To generate JCL for local recovery**

Start this procedure at the RECOVERY MANAGER Main Menu.
1 Access the Local System Recovery panel, as follows:

   a Select **Subsystem recovery**.

   b Select **Local recovery**.

   The Local System Recovery panel (ARMUFS4) is displayed.

   **Note**
   This panel has different fields if you are using the BMC Recovery Management for DB2 solution password.

2 Enter information as required and press **Enter**.

   For more information, see Table 39 on page 360.

   Another Local System Recovery panel (ARMUFS4A) is displayed.

3 Enter information as required and press **Enter**.

   **Note**
   For more information, see Table 39 on page 360.
4 On the JCL Specification panel, enter a fully qualified output data set name. Be aware of the following information:

- The output data set is used for saving the JCL and must be cataloged. If not enclosed in quotes, the output data set will be prefixed by your TSO prefix.

- The job statement must contain a symbolic variable (\&) for the job number. For more information, see “Output data sets, job cards, and symbolic variables” on page 51.

Note

If the output JCL data set is a GDG, the product always uses SYS1.MODEL as the model data set name.

5 Save the JCL data set or submit the job as required.

Table 39: Local subsystem recovery fields

<table>
<thead>
<tr>
<th>Field</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group owner</td>
<td>Last value used</td>
<td>Specifies a valid TSO user ID to be used by RECOVERY MANAGER as the creator_ID part of each group name. See “Group authorization” on page 90 for more information about authorization for creating groups.</td>
</tr>
<tr>
<td>Group name prefix</td>
<td>Last value used</td>
<td>Specifies a character string to be used by RECOVERY MANAGER as a prefix in the group part of each group name. Note: Delimited entries are not allowed for the group name prefix.</td>
</tr>
<tr>
<td>Recover start range</td>
<td>Current time</td>
<td>Specifies the recovery time in the format yyyy-mm-dd hh.mm.ss If entered, you should choose a time prior to current for the start range.</td>
</tr>
<tr>
<td>Recover end range</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gen one job to convert</td>
<td>Yes</td>
<td>Generates a single job stream that converts the timestamps for all members of a data sharing system in a single execution. This feature simplifies scheduling and monitoring the timestamp conversion process in a data sharing environment. Selecting No generates a separate job for each data sharing member. Each job runs on the LPAR on which the member exists. Note: This option is only available on data sharing systems.</td>
</tr>
<tr>
<td>timestamp to rba</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Job card data set</td>
<td>Last value used</td>
<td>Specifies a fully-qualified name of the data set containing job card information. The job name must contain the symbolic variable &amp;##.</td>
</tr>
<tr>
<td>System Resource recovery JCL output data set name</td>
<td>Last value used</td>
<td>Specifies a fully-qualified name of a new or existing data set to be used for saving the recovery JCL for the system resources (catalog and directory).</td>
</tr>
<tr>
<td>Application data set RECOVER JCL output data set name</td>
<td>Last value used</td>
<td>Specifies a fully-qualified name of a new or existing data set to be used for saving the application data set recovery JCL.</td>
</tr>
<tr>
<td>Exclude Data Sharing Member</td>
<td>None</td>
<td>Excludes data sharing members from recovery and ignores these subsystems regardless of the status.</td>
</tr>
<tr>
<td>Bypass Quiesced Data Sharing Members</td>
<td>None</td>
<td>Bypasses quiesced data sharing members when calculating the recovery point.</td>
</tr>
<tr>
<td>Recover Work File Database</td>
<td>Yes</td>
<td>Recovers the work file database during subsystem recovery. If you choose not to recover the work file database, this step is omitted from the recovery job.</td>
</tr>
<tr>
<td>Library that contains</td>
<td>Yes</td>
<td>Specifies the library that contains DSNTEP2. <strong>Tip:</strong> BMC recommends that you choose <strong>No</strong> if your DSNLOAD library does not contain DSNTEP2.</td>
</tr>
<tr>
<td>Update history file with HISTONLY option</td>
<td>Last value used</td>
<td>Captures the copy information for the DB2 spaces DSNDB06.SYSCOPY, DSNDB01.DBD01, DSNDB01.SYSDBDXA, and DSNDB01.SYSUTILX in the RECOVERY MANAGER archive history file without copying the archive logs. RECOVERY MANAGER stores image copy registration information from the log in the archive history file so that it can optimize recovery JCL for the catalog and directory. The default is No. <strong>Note:</strong> Use this option if you made images copies of the catalog and directory within the log range of the log just archived. In this case, the history file does not yet contain a record of those copies.</td>
</tr>
<tr>
<td>Field</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Exclude unchanged from recovery</td>
<td>No</td>
<td>Exclude unchanged spaces from recovery. The objects that are marked as UNCHANGED are excluded from the recover JCL. Objects in WRITE PENDING status are included in the recovery even if they are still marked as UNCHANGED. This option can significantly reduce the time required for recovery by eliminating unnecessary processing.</td>
</tr>
<tr>
<td>Simulate Recovery</td>
<td>No</td>
<td>Simulates recovery at the local site and provides a way to help you prove that you can recover the DB2 data without performing an actual recovery. You can use recovery simulation to verify that needed recovery resources are valid and available and that log apply can be done. This option is a feature of the Recovery Management solution only, and both the RECOVERY MANAGER and NGT Recover components of the Recovery Management solution are required. For more information, see the Recovery Management for DB2 User Guide.</td>
</tr>
<tr>
<td>Include Clones</td>
<td>No</td>
<td>Specifies whether to create JCL to recover the cloned objects in the groups being created. Cloned objects are recovered separately from base objects. This option is available only when running on DB2 Version 10 or later and is not valid with compatibility mode. If you use clones, you need to be sure to specify Yes. Specifying Yes generates a separate ARMBGEN step for generating application recovery for only cloned objects. After the system recovery runs, the ARMBGEN JCL runs, which generates JCL to recover the applications. Be sure to run both jobs that are generated—for regular objects and for clones.</td>
</tr>
<tr>
<td>Output data set (for clone groups)</td>
<td>None</td>
<td>The name of an existing data set where you want to place the recover JCL for the cloned objects. This must be a fully qualified data set name that does not contain quotes.</td>
</tr>
</tbody>
</table>
**RECOVERY MANAGER subsystem recovery programs**

RECOVERY MANAGER provides the following programs for use at the local site to help you prepare for full subsystem recovery:

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARMBGPS</td>
<td>Automatically creates application groups for the entire subsystem. For more information, see “ARMBGPS—Subsystem group split” on page 473.</td>
</tr>
<tr>
<td>ARMBARC</td>
<td>Creates recovery site copies of archive log data sets and identifies image copies on the log. For more information, see “ARMBARC—Archive log data sets” on page 409.</td>
</tr>
<tr>
<td>ARMBLOG</td>
<td>Issues the appropriate DB2 -ARCHIVE LOG command and waits for the offload to complete. For more information, see “ARMBLOG—Archive log creation” on page 603.</td>
</tr>
<tr>
<td>ARMBTSI</td>
<td>Inserts a timestamp into the CRRDRPT table. For more information, see “ARMBTSI—Time stamp insertion” on page 719.</td>
</tr>
<tr>
<td>ARMBCRC</td>
<td>Performs log analysis for a subsystem point-in-time recovery. For more information, see “ARMBCRC—Conditional recovery to a timestamp” on page 425.</td>
</tr>
<tr>
<td>ARMBSRR</td>
<td>Generates JCL for subsystem resource recovery. For more information, see “ARMBSRR—System resource recovery” on page 673.</td>
</tr>
<tr>
<td>ARMBLGR</td>
<td>Writes log range summary information to the ARMLRNG file (the RECOVERY MANAGER log range file). For more information, see “ARMBLGR—Log range analysis” on page 597.</td>
</tr>
<tr>
<td>ARMBGEN</td>
<td>Generates JCL for application data recovery For more information, see “ARMBGEN—Backup and recovery JCL” on page 431.</td>
</tr>
<tr>
<td>ALPMAIN</td>
<td>Log Master searches for DDL and quiet points to be analyzed by ARMBSRR for catalog recovery. A Recovery Management password is required. For more information, see the <em>Log Master for DB2 User Guide</em> and the <em>Log Master for DB2 Reference Manual</em>.</td>
</tr>
</tbody>
</table>
Using RECOVERY MANAGER for BACKUP SYSTEM and RESTORE SYSTEM

RECOVERY MANAGER provides an automated method to generate jobs for IBM’s BACKUP SYSTEM and RESTORE SYSTEM for full volume backup and restore of a DB2 subsystem or data sharing group.

For a complete description of BACKUP SYSTEM and RESTORE SYSTEM, see the IBM DB2 for z/OS Utility Guide and Reference.

BACKUP SYSTEM

BACKUP SYSTEM copies the volumes on which the DB2 data and log information resides for either a DB2 subsystem or data sharing group.

You can use BACKUP SYSTEM to copy all data for a single application (for example, when DB2 is the database server for a resource planning solution). All data sets that you want to copy must be SMS-managed data sets. You can subsequently run RESTORE SYSTEM to recover the entire system.

RECOVERY MANAGER provides:

- Standalone JCL generation for BACKUP SYSTEM
- JCL generation to create a conditional restart record for log truncation based on BACKUP SYSTEM information from the BSDS

RESTORE SYSTEM

RESTORE SYSTEM recovers a DB2 subsystem or a data sharing group to a previous point in time.

To perform the recovery, RESTORE SYSTEM uses data that is copied by BACKUP SYSTEM. The data sets that you want to recover must be SMS-managed data sets.

You can run RESTORE SYSTEM from any member in a data sharing group, even one that is normally quiesced when any backups are taken. Any member in the data sharing group that is active at or beyond the log truncation point must be restarted, and its logs are truncated to the SYSPITR LRSN point. You can specify the SYSPITR LRSN point in the CRESTART control statement of the DSNJU003 (Change Log Inventory) utility. Any data sharing group member that is normally quiesced at the
time the backups are taken and is not active at or beyond the log truncation point does not need to be restarted.

RECOVERY MANAGER provides:

- Standalone JCL generation for RESTORE SYSTEM
- The option to generate JCL to recover objects in RECP, RBLD, GRECP and LPL status

**RECOVERY MANAGER process to generate standalone JCL**

The following topics describe how to access the panels you need to use to create the standalone JCL for BACKUP SYSTEM and RESTORE SYSTEM using RECOVERY MANAGER.

**To generate JCL for BACKUP SYSTEM and RESTORE SYSTEM**

Start this task at the RECOVERY MANAGER Main Menu.

1. Access the Recovery Preparation and JCL Generation panel (ARMFS01), as follows:
   a. Select Subsystem recovery.
   b. From Recovery Preparation and JCL Generation (Figure 32 on page 365), select option 6, 7, 8, or 9.

   **Figure 32: System Recovery Preparation and JCL Generation panel (ARMFS01)**

<table>
<thead>
<tr>
<th>Command</th>
<th>Select option. Then press Enter.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARMFS01</td>
<td>System Recovery Preparation and JCL Generation</td>
</tr>
<tr>
<td></td>
<td>Command ===&gt; ________________________</td>
</tr>
<tr>
<td>1. Full Recovery groups</td>
<td>- Build RMGR groups for data and generate backups</td>
</tr>
<tr>
<td>2. Repository</td>
<td>- Backup/Recover RMGR and Log Master Repositories</td>
</tr>
<tr>
<td>3. DB2 catalog</td>
<td>- Generate backup of DB2 catalog and directory</td>
</tr>
<tr>
<td>4. Local recovery</td>
<td>- Generate JCL to perform a local recovery</td>
</tr>
<tr>
<td>5. Disaster recovery</td>
<td>- Generate local site DR preparation JCL</td>
</tr>
<tr>
<td>6. Backup system</td>
<td>- Generate full volume backup of DB2 data and logs</td>
</tr>
<tr>
<td>7. Conditional restart</td>
<td>- Generate SYSPITR or SYSPITRT for Restore system</td>
</tr>
<tr>
<td>8. Restore system</td>
<td>- Generate full volume restore of DB2 data</td>
</tr>
<tr>
<td>9. Recover pending</td>
<td>- Generate recovery for RECP, RBLD, GRECP, LPL</td>
</tr>
</tbody>
</table>

**To generate BACKUP SYSTEM JCL**

1. Enter option 6 Backup System on the System Recovery Preparation and JCL Generation panel (ARMFS01).
The Backup System JCL Generation panel (ARMBS01) displays when you enter values to generate JCL for DSNUTILB BACKUP SYSTEM.

For a description of the options on this panel, see the *IBM DB2 for z/OS Utility Guide and Reference*.

**Figure 33: Backup System JCL Generation panel (ARMBS01)**

![Backup System JCL Generation panel (ARMBS01)](image)

Select options for BACKUP SYSTEM. Then press Enter.

- **Full**
  - 2 1. Yes 2. No (copy database and log pools)
- **Data only**
  - 2 1. Yes 2. No (copy only database pool)
- **Establish FC incr**
  - 2 1. Yes 2. No (establish persistent incr FlashCopy)
- **End FC incremental**
  - 2 1. Yes 2. No (end persistent incr FlashCopy)
- **Force**
  - 2 1. Yes 2. No (overwrite the oldest backup)
- **Dump**
  - 2 1. Yes 2. No (create a backup and copy to tape)
- **Dump only**
  - 2 1. Yes 2. No (copy existing backup to tape)
- **Relative backup**
  - ___ 0=Last to 99=99th previous
- **Dump class**
  - ________

Figure 33 on page 366 provides an example of the JCL that RECOVERY MANAGER generates for BACKUP SYSTEM.

```sql
//** ********************************************************************** */
//**               RECOVERY MANAGER - V11.2.00 - BMC SOFTWARE, INC.        */
//** ********************************************************************** */
//**               DSNUTILB - BACKUP                                       */
//** ********************************************************************** */
//** ARM0001 EXEC PGM=DSNUTILB,                                        */
//**      PARM='DEDL'                                                    */
//** STEPLIB   DD DISP=SHR,DSN=SYS3.DEDL.DSNEXIT                        */
//**          DD DISP=SHR,DSN=CSGI.DB2V91M.DSNLOAD                       */
//** SYSPRINT  DD SYSOUT=*                                             */
//** SYSDUMP   DD SYSOUT=*                                             */
//** JUTPRINT  DD SYSOUT=*                                             */
//** SYSSN    DD *                                                    */
//** BACKUP SYSTEM                                                      */
//** FULL                                                            */
```

**To create a conditional restart record**

1. Enter option 7 on the System Recovery Preparation and JCL Generation panel (ARMFS01)

The Log Truncation Point for Restore System panel (ARMRS01) displays where you enter values to use to generate DSNJU003 JCL to create a conditional restart record.

See the *IBM DB2 for z/OS Utility Guide and Reference* for a description of the options on this panel.
Select option for Log Truncation for RESTORE SYSTEM. Then press Enter.

Generate a step to create a new conditional restart control record.
Log truncation point and log truncation timestamp are mutually exclusive.
Log trunc point: ____________________ (all FF means no log truncation)
Log trunc timestamp: ___________________ (YYYY-MM-DD-HH.MM.SS.T)

<table>
<thead>
<tr>
<th>RBLP</th>
<th>DATA COMPLETE LRSN</th>
<th>COMPLETE DATE</th>
<th>COMPLETE TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>00199970F384</td>
<td>001999714716</td>
<td>2011/06/1</td>
<td>10:43:41:839345</td>
</tr>
<tr>
<td>001998F12090</td>
<td>001998FF2086</td>
<td>2011/06/03</td>
<td>12:41:22:264976</td>
</tr>
<tr>
<td>001998F12090</td>
<td>01998F16058</td>
<td>2011/06/03</td>
<td>07:26:22:862311</td>
</tr>
<tr>
<td>001994CA0B090</td>
<td>001994CB2144</td>
<td>2011/05/23</td>
<td>16:04:25:786646</td>
</tr>
<tr>
<td>001994988090</td>
<td>001994C007BC</td>
<td>2011/05/16</td>
<td>15:44:12:766085</td>
</tr>
<tr>
<td>001994988090</td>
<td>001994B916A4</td>
<td>2011/05/16</td>
<td>11:01:07:729698</td>
</tr>
<tr>
<td>001994988090</td>
<td>001994AC0B12</td>
<td>2011/05/16</td>
<td>10:52:30:733929</td>
</tr>
</tbody>
</table>

Figure 34 on page 367 provides an example of the conditional restart JCL that RECOVERY MANAGER generates.

Figure 34: Sample JCL for conditional restart for RESTORE SYSTEM

```/* *************************************************************** */
/*          RECOVERY MANAGER - V11.2.00 - BMC SOFTWARE, INC. */
/* *************************************************************** */
/* DNJU003 - CREATE CONDITIONAL RESTART CONTROL RECORD*/
/* *************************************************************** */
/* *************************************************************** */
//ARM0001 EXEC PGM=DNJU003
//STEPLIB DD DISP=SHR,DSN=SYS3.DEDL.DSNEXIT
//          DD DISP=SHR,DSN=CSGI.DB2V91M.DSNLOAD
//SYSUT1 DD DISP=SHR,DSN=DEDLCAT.BSDS01
//SYSUT2 DD DISP=SHR,DSN=DEDLCAT.BSDS02
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
CRESTART CREATE,SYSPITR=0C3847DA4000
/*
```

To create RESTORE SYSTEM JCL

1. Enter option 8 Restore System on the System Recovery Preparation and JCL Generation panel (ARMFS01).

The Restore System JCL Generation panel (ARMRS02) displays where you enter option to generate JCL for DSNUTILB RESTORE SYSTEM.

See the IBM DB2 for z/OS Utility Guide and Reference for a description of the options on this panel.

Figure 35: Restore System JCL Generation panel (ARMRS02)

```
ARMS02 =--------------------------------- Restore System JCL Generation ---------------------------------
Command ===> _________________________________________________________________

Select options for RESTORE SYSTEM. Then press Enter.

Log only . . . . . . 2  1. Yes  2. No (skip restore and only apply log)
Tape units . . . . .  ___  (1-255 maximum number of tape units to allocate)
From dump . . . . . 2  1. Yes  2. No (dump only the database pool to tape)
Dump class . . . . .  
```
Figure 36 on page 368 provides an example of the RESTORE SYSTEM JCL that RECOVERY MANAGER generates.

**Figure 36: Sample standalone JCL for RESTORE SYSTEM**

```bash
/* *************************************************************** */
/*                    RECOVERY MANAGER - V11.2.00 - BMC SOFTWARE, INC. */
/* *************************************************************** */
/*                    DSNUTILB - RESTORE SYSTEM */
/* *************************************************************** */
/* *************************************************************** */
// ARM0003 EXEC PGM=DSNUTILB,
//              PARM='DEDL'
//STEPLIB   DD DISP=SHR,DSN=SYS3.DEDL.DSNEXIT
//          DD DISP=SHR,DSN=CSGI.DB2V91M.DSNLOAD
//SYSPRINT  DD SYSOUT=*  
//SYSUDUMP  DD SYSOUT=*  
//UTPRINT   DD SYSOUT=*  
//SYSIN     DD *
//    RESTORE SYSTEM
//    TAPEUNITS 10
/*
```

To generate recovery JCL for RECP, RBLD, GRECP, LPL

1. Enter option 9 on the System Recovery Preparation and JCL Generation panel (ARMFS01) to generate JCL to recover objects in RECP, RBLD, GRECP, and LPL statuses. This JCL creates a group of objects in RECP, RBLD, GRECP, and LPL statuses and then generates JCL to recover those objects to current.
Modeling the DB2 logging environment

This chapter describes how to model the DB2 logging environment.

About the logging environment modeling tool

The RECOVERY MANAGER logging environment modeling tool allows you to examine and view different logging scenarios for a selected DB2 subsystem in order to optimize the amount of DASD space required by the archive logs.

In addition, you can display active log information and archive log information. By making use of the logging environment modeling tool’s optimization capabilities, you can dramatically reduce the amount of DASD space required by your logging environment.

The logging environment modeling tool has the following features:

- Displays current logging environment statistics, including:
  - Expected compression ratio if PACLOG is used
  - Days of log data recorded in the BSDS
  - Hours of log required on DASD
  - Highest logging rate
  - Active log size in cylinders
  - Number of active log pairs
  - Number of entries in BSDS
  - Hours of archive 2 on DASD
  - Hours of archive 1 on DASD
  - Total cylinders required by all logging resources

- Displays information regarding the active logs, including:
  - Total cylinders required
  - Average number of hours in each log
— Minimum number of hours in each log
— Start and end times of each log

- Displays information regarding the archive logs, including:
  — ARCHLOG1 details
  — ARCHLOG2 details
  — Audit details regarding archive copy synchronization

Note
RECOVERY MANAGER fully supports the maximum number of active logs (93) and archive logs (10,000) available in DB2.
All active logs can be displayed using the online interface. For performance reasons, only the most recent 100 archive logs are displayed using the online interface. All logs can be processed using the batch programs ARMBARC, ARMBSRR, ARMBEOL, and ARMBCRC.

Viewing and modeling logging environment statistics

The Model option of the logging environment modeling tool allows you to view information regarding the logging environment of a specified DB2 subsystem including the following:
- Days of log data recorded in the BSDS
- Hours of log required on DASD
- Highest logging rate
- Active log size in cylinders
- Number of active log pairs
- Number of entries in BSDS
- Hours of archive 2 on DASD
- Hours of archive 1 on DASD
- Total cylinders required by all logging resources

The Model option also allows you to change certain logging environment variables in order to see what effect those changes would have on the required amount of DASD and on other logging environment variables.

In addition, you can choose to have the logging environment modeling tool optimize your logging environment based upon the number of days of log data that you want to record in the BSDS and the number of hours of log that you require to be kept on DASD.
Before you begin

To view or model logging environment statistics, you must have the following authorizations:

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- Authority to use the Print Log Map utility

To view logging environment statistics

Start this procedure at the RECOVERY MANAGER Main Menu.

1. Select System resources, then press Enter.

2. Select Logging environment, and then press Enter.

   The Logging Environment panel is displayed.

   **Figure 37: Logging environment panel**

   ARMLGOOA ============== Logging environment - DEBC ==============
   Command ===> __________________________________________________________________
   
   Type information. Then press Enter.

   Option . . . 1. Model Display current statistics & model changes
   2. Actives Display active log information
   3. Archives Display archive log information
   DASD type. . . 1. 3380 Device type used in rate calculations
   2. 3390

3. Select Model, select or verify your DASD type, and then press Enter. The logging environment modeling tool begins analysis of the BSDS. This process may take one or two minutes. Press Enter again to continue.

   The Logging Environment Model panel is displayed.

   **Figure 38: Logging Environment Model panel**

   ARMLGO1A ============= Logging Environment Model for DGA3 =============
   Command ===> __________________________________________________________________
   
   Type information and press Enter.

   Source of highest logging rate ** . . . . . . . . . 1. BSDS 2. User 3. List
   Optimize fields marked * for DASD archives . . . . . . 1. Yes 2. No

   Current Model Prev
   Expected compression ratio . . . . . . . . . . . . . 0-95
   148 Days of log data recorded in BSDS . . . . . . 1-366
   0.10 Hours of log required on DASD . . . . . . . . . . 1-999
   2926 High logging rate for that period . . . . . . . . . 1-99999
   100 Active log size (cylinders each) . . . . . . . . . . . . . 1-99999
   3 Number of active log pairs . . . . . . . . . . . . . 1-3-93
   1000 Number of entries in BSDS . . . . . . . . . . . . . . . . . . 1-10000
   assuming 0 Hours of archive 2 on DASD . . . . . . . . . . . . 0-999 (0=tape)
   assuming 0 Hours of archive 1 on DASD
4 View the current statistics for your DB2 subsystem in the **Current** column. (For a list of field descriptions, see “Logging environment model field descriptions” on page 373.)

**To optimize logging environment statistics**

Start this procedure at the Logging Environment Model panel (For instructions on reaching this panel, see “To view logging environment statistics” on page 371).

1 Set or verify the **Source of highest logging rate** (For a description of this field, see “Logging environment model field descriptions” on page 373).

2 Specify **Yes** in the **Optimize fields marked * for DASD archives** field.

   This instructs the logging environment modeling tool to optimize the active log size, the number of active log pairs, the number of entries in the BSDS, and the number of hours ARCHLOG2 is retained on DASD.

3 Enter information in the following fields:

   a **Expected compression ratio**—this is the percentage that the archive logs are compressed if you are using PACLOG. Enter 0 if you are not using PACLOG (see “Logging environment model field descriptions” on page 373).

   b **Days of log data recorded in BSDS**—this is the number of days of log you want to have available for recovery (see “Logging environment model field descriptions” on page 373).

   c **Hours of log required on DASD**—this is the length of time you want recovery data available on DASD (see “Logging environment model field descriptions” on page 373).

4 Press **Enter**. The logging environment modeling tool optimizes all fields marked with an asterisk (*) and then calculates the required cylinders.

   You now have an optimized model against which you can compare other scenarios.

**To model logging environment statistics**

Start this procedure at the Logging Environment Model panel (for instructions on reaching this panel, see “To view logging environment statistics” on page 371).

1 Set or verify the **Source of highest logging rate** (for a description of this field, see “Logging environment model field descriptions” on page 373).
2 Specify No in the **Optimize fields marked * for DASD archives** field.

Doing so enables you to manually enter figures into the fields marked with an asterisk (*).

3 Enter information in all fields (see “Logging environment model field descriptions” on page 373).

---

**Note**

If you chose **BSDS** or **List** as your **Source of highest logging rate**, you do not have to enter the **High logging rate for that period**.

The logging environment modeling tool calculates the **Days of log data recorded in BSDS**, the **Hours of archive 1 on DASD**, and the total required cylinders with and without PACLOG.

You can compare this scenario to the previous model or make further changes and run the logging environment modeling tool again. The previous model is retained in the **Prev** column.

---

4 Press **Enter**.

---

**Logging environment model field descriptions**

The following table lists each of the fields on the Logging Environment Model panel alphabetically.

Descriptions of each field follow in the order in which they appear on the panel.

**Table 40: Modeling tool fields**

<table>
<thead>
<tr>
<th>Modeling tool option</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active log size</td>
<td>None</td>
<td>The active log size in cylinders If you choose to optimize the active log size (see “To optimize logging environment statistics” on page 372), then the size in cylinders is calculated as follows: 1 (one) plus [(number of days of data in BSDS multiplied by 24) x (the high logging rate) / (number of entries in the BSDS)] If you choose not to optimize the active log size (see “To model logging environment statistics” on page 372), then you must enter a value for active log size.</td>
</tr>
<tr>
<td>Current</td>
<td>None</td>
<td>Describes the current logging environment of your system</td>
</tr>
<tr>
<td>Modeling tool option</td>
<td>RECOVERY MANAGER default</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>--------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Cylinders saved by PACLOG         | None                     | Number of cylinders saved by using PACLOG compression  
This value is calculated by subtracting the cylinders required when using PACLOG from the cylinders required when not using PACLOG.                                                                                                                                                                                                                   |
| Days of log data recorded in BSDS | None                     | Displays the number of days of log data to be kept in the BSDS  
BMC recommends that you keep a few more days than is usually necessary to allow for periods of unexpectedly high activity. For example, if you want 14 days of log available for recovery, then you should ensure that you have 17 days of log available in the BSDS.  
If you choose to optimize the logging environment (see “To optimize logging environment statistics” on page 372), then you must enter the number of days you want to keep log data.  
If you choose not to optimize the active log size (see “To model logging environment statistics” on page 372), then this value is calculated as follows:  
(active log size multiplied by the number of entries in BSDS)  
divided by  
(the highest logging rate multiplied by 24)                                                                                                                                                                                                                           |
| Expected compression ratio        | None                     | The expected compression of an archive log when PACLOG is used  
Compression percentages of 70-90 percent can be expected in most circumstances. Enter 0 if you do not intend to use PACLOG.                                                                                                                                                                                                                                                                  |
<table>
<thead>
<tr>
<th>Modeling tool option</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| High logging rate for that period | None | Searches the BSDS to find the highest rate of log activity during any period of \( nn \) hours, where \( nn \) is the number of hours set in the *Hours of log required on DASD* field.  
You can also enter a logging rate manually or choose a logging rate from a list of high rates.  
To have the logging environment modeling tool find the highest rate, choose *BSDS* in the *Source of highest logging rate field* and press *Enter*.  
To enter the logging rate of your choice, choose *User* in the *Source of highest logging rate field* and enter the logging rate you want in this field.  
Valid values range from 1 to 99999 cylinders per hour.  
To select a logging rate from a list of the highest 15 logging rates, select *List* as the *Source of highest logging rate* and press *Enter*. When the list is displayed, enter *S* or / beside your choice and press *Enter*. |
| Hours of archive 1 on DASD | None | Shows the amount of time that ARCHLOG1 should be retained on DASD before migrating it to tape.  
This figure is determined by the number of hours you indicate in the *Hours of log required on DASD* field.  
The amount shown in the *Current* column is the number of hours currently being saved to DASD on your system. |
### Modeling tool option

<table>
<thead>
<tr>
<th>Modeling tool option</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| Hours of archive 2 on DASD           | None                      | Shows the amount of time that you want to retain ARCHLOG2 on DASD before migrating it to tape. The logging environment modeling tool sets this to one hour when you choose to optimize. You can also use the logging environment modeling tool to determine the impact on DASD usage if you are currently keeping more than the optimal number of hours. If you do not store either copy of the archive log on DASD, set this value to 0. To view the effect of keeping a larger number of hours of ARCHLOG2 on DASD:  
  - Select No in the **Optimize fields marked * for DASD archives** field  
  - Set or verify the following fields:  
    - **Expected compression ratio**  
    - **Days of log data recorded in BSDS**  
    - **Hours of log required on DASD**  
  - Set the **Hours of archive 2 on DASD** as desired  
  - Press Enter |
<p>| Hours of log required on DASD        | None                      | Displays the number of hours you want to retain log data on DASD. This value is the length of time that you want recovery data to be available on DASD. Valid values range from 1 to 999. If you do not archive to DASD, the value entered in this field, along with the logging rate for this time period, is used to determine the number of days of data which is stored in the BSDS. The value in the <strong>Current</strong> column is the value determined by the shortest length of time taken to fill all of the active log data sets. |
| Model                               | None                      | Changes the logging environment settings in order to determine the overall effect of the changes on DASD or logs |</p>
<table>
<thead>
<tr>
<th>Modeling tool option</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of active log pairs</td>
<td>None</td>
<td>Shows the number of active log pairs for the DB2 subsystem. The entry in the Current column shows your current DB2 log configuration. If you choose Yes in the Optimize fields marked * for DASD archives field, then the logging environment modeling tool sets the number of active log pairs to 3.</td>
</tr>
<tr>
<td>Number of entries in BSDS</td>
<td>None</td>
<td>Shows the number of entries kept in the BSDS. The entry in the Current column shows number of entries currently kept in the BSDS. The amount of DASD space occupied by these entries is only a few cylinders. If you choose Yes in the Optimize fields marked * for DASD archives field, the logging environment modeling tool sets this figure to 10000. If you do not currently keep the maximum number of entries allowed for your version of DB2 in the BSDS, you can determine how changing the number of entries affects the number of days of log data recorded in the BSDS by performing log environment modeling (see “To model logging environment statistics” on page 372).</td>
</tr>
<tr>
<td>Optimize fields marked * for DASD archives</td>
<td>Yes</td>
<td>Calculates the optimal size and number of DB2 archive log data sets. All fields marked with a * are included in the optimization process. To perform the optimization, select Yes in this field, enter data in the following fields, and then press Enter. ■ Hours of log required on DASD ■ Days of log data recorded in BSDS ■ Expected compression ratio The tool initializes marked fields as follows, then performs the optimization. ■ Number of active log pairs is set to 3 ■ Number of entries in BSDS is set to 10,000 ■ Hours of archive 2 on DASD is set to 1.</td>
</tr>
<tr>
<td>Prev</td>
<td>None</td>
<td>Displays the results of the previous model you created, allowing you to compare two different scenarios.</td>
</tr>
</tbody>
</table>

---

**Viewing and modeling logging environment statistics**

Chapter 10  Modeling the DB2 logging environment  377
<table>
<thead>
<tr>
<th>Modeling tool option</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source of highest logging rate</td>
<td>BSDS</td>
<td>Determines how the logging environment model obtains the highest logging rate: from the BSDS, user-entry or from a list of previous rates</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Specify <strong>BSDS</strong> to display the highest logging rate. The logging environment modeling tool searches the BSDS to find the highest rate of activity during any period of ( nn ) hours, where ( nn ) is the number set in the <strong>Hours of log required on DASD</strong> field.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Specify <strong>User</strong> if you want to enter the highest logging rate yourself in the <strong>High logging rate for that period</strong> field.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Specify <strong>List</strong> to select the highest logging rate from a list encompassing the highest rates of activity during any period of ( nn ) hours, where ( nn ) is the number you set in the <strong>Hours of log required on DASD</strong> field. When the list is displayed, enter / or S beside the rate you want and press Enter.</td>
</tr>
<tr>
<td>Total cylinders with PACLOG</td>
<td>None</td>
<td>Total number of cylinders required if you use PACLOG compression</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This value is calculated as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Amount required without PACLOG minus the percentage entered in the <strong>Expected compression ratio</strong> field.</td>
</tr>
<tr>
<td>Total cylinders without PACLOG</td>
<td>None</td>
<td>Total number of cylinders required if you do not use PACLOG compression</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This value is calculated as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- If you do not keep any archive log on DASD:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(size of the active log data sets) x (number of log data sets) x 2 (if you are using dual logs).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- If you keep archive data on DASD:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>All of the DASD for the active log data sets (formula above) + [(highest logging rate) x (number of hours of ARCLOG1) - (total hours of active log)] + the highest logging rate multiplied by the hours of ARCLOG2</td>
</tr>
</tbody>
</table>
Viewing active log information

The **Actives** option of the logging environment modeling tool allows you to view information regarding the active log.

**Before you begin**

To view active log information, you must have the following authorizations:

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- Authority to use the Print Log Map utility

**To view active log information**

Start this procedure at the RECOVERY MANAGER Main Menu.

1. Select **System resources**.

2. Select **Logging environment**.

   The Logging Environment panel is displayed.

3. Select **Actives**, select or verify your DASD type, and then press **Enter**. The logging environment modeling tool begins analysis of the BSDS. Press **Enter** again to continue.

   The Active Log Information panel is displayed.

   ARMLG02A =========== Active Log Information - DGA3 ========= Row 1 to 3 of 3
   Command ===> ________________________________________________ Scroll ===> PAGE
   Total cyls of actives (1 and 2) : 600
   Average hours of data in actives: 10.67
   Minimum hours of data in actives: 0.10 From: 2012-02-31 WED 12:31:20 To: 2012-02-31 WED 12:37:29
   Current active log copy 1 data sets
   Start time                 End time            Cyls    cyl/hr
   -----------------------   -----------------------   ------   --------
   2012-02-13 Thu 13:14:30   2012-02-13 Thu 18:20:35      100      19.60
   2012-02-13 Thu 18:20:35   2012-02-13 Thu 18:31:05      100     571.43
   2012-02-13 Thu 18:31:05   -- INVALID END TIME ---      100   UNKNOWN

   ******************************* Bottom of data *******************************

4. View the current information about the active log (for a list of field descriptions, see “Active log field descriptions” on page 380.)
Active log field descriptions

The following information describes each of the fields on the Active Log Information panel.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total cyls of actives (1 and 2)</td>
<td>The total number of cylinders allocated to active logs. This includes copy 1 data sets and copy 2 data sets if they exist.</td>
</tr>
<tr>
<td>Average hours of data in actives</td>
<td>For all log data recorded in the BSDS, the average number of hours for which data was available in the active logs. This number is based on the current number of active log sets.</td>
</tr>
<tr>
<td>Minimum hours of data in actives</td>
<td>For all log data recorded in the BSDS, the minimum number of hours for which data was available in the active logs. This is the shortest time span in which all of the active logs were filled.</td>
</tr>
<tr>
<td>Start time</td>
<td>The start time for each active log, in the format yyyy-mm-dd wkdd hh:mm:ss</td>
</tr>
<tr>
<td>End time</td>
<td>The end time for each active log in the same format as Start time.</td>
</tr>
<tr>
<td>Cyls</td>
<td>The number of cylinders which this log would occupy if on DASD.</td>
</tr>
<tr>
<td>cyl/hr</td>
<td>The logging rate in cylinders per hour for each active log.</td>
</tr>
</tbody>
</table>

Viewing archive log information

The Archives option of the logging environment modeling tool allows you to view information regarding the archive logs.

Before you begin

To view archive log information, you must have the following authorizations:

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- Authority to use the Print Log Map utility

To view archive log information

Start this procedure at the Logging Environment panel (see “To view active log information” on page 379.)
1 Select **Archives**, select or verify your DASD type, and then press **Enter**. The logging environment modeling tool begins analysis of the BSDS. Press **Enter** again to continue.

The Archive Log Information panel is displayed.

2 To view details about ARCHLOG1, select **Copy 1 detail** and press **Enter**.

3 To view details about ARCHLOG2, select **Copy 2 detail** and press **Enter**.

4 To audit the synchronization between all copies of ARCHLOG2 and ARCHLOG1, select **Audit sync** and press **Enter**.

### Archive log 1 and 2 detail field descriptions

The following table describes each of the fields on the Archive Log Copy 1 Information and Archive Log Copy 2 Information panels.

**Table 42: Archive log fields**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sort by</td>
<td>Sort the list of archive logs by start time, by cylinders per hour in descending order (largest to smallest), or by cylinders per hour in ascending order (smallest to largest)</td>
</tr>
<tr>
<td>Average cyls per hour</td>
<td>The average number of cylinders per hour for all of the logs in the list</td>
</tr>
<tr>
<td>Number of days covered</td>
<td>The number of days covered by all of the logs in the list; that is, from the start time of the first log to the end time of the last log</td>
</tr>
<tr>
<td>Start time</td>
<td>The start time for each archive log, in the format <strong>yyyy-mm-dd wkd hh:nn:ss</strong></td>
</tr>
<tr>
<td>End time</td>
<td>The end time for each archive log in the same format as <strong>Start time</strong>.</td>
</tr>
<tr>
<td>Cyls</td>
<td>The number of cylinders which this log would occupy if on DASD.</td>
</tr>
<tr>
<td>cyl/hr</td>
<td>The logging rate in cylinders per hour for each archive log.</td>
</tr>
</tbody>
</table>
Audit synchronization

You can verify the existence of an archive copy 1 data set for each archive copy 2 data in the BSDS. When you select this option, you will get one of the following results:

- An informational message number BMC80698I is given stating that the copies are synchronized. This means that there is an ARCHLOG1 data set that matches each ARCHLOG2 data set found in the BSDS.

- One or more messages with the number BMC80688W are given listing each ARCHLOG2 data set for which there is no matching ARCHLOG1 data set.
Accessing online Progress Reports

This chapter describes how to access online Progress Reports and time estimates.

About Progress Reports

RECOVERY MANAGER provides online information through Progress Reports so that you can:

- Gauge the progress of recoveries by database.tablespace or group
- Determine if any objects were not recovered when all recovery jobs are complete
- Identify objects that were missed in a recovery scenario
- Generate recover JCL for objects that were missed

Within a requested time range, you can use the group or an object pattern to display the following information:

- Objects recovered
- Objects not recovered
- Objects unchanged
- Objects changed
- Objects copied
- Objects not copied

Note

With a BMC Recovery for DB2 solution password, you can use the group or an object pattern to display backup and recover time estimates and to save time estimates in a data set.

Additionally, you can request JCL generation for the ARMBRPR batch program. For information about using the ARMBRPR batch program to produce Progress Reports, see “ARMBRPR — Progress Reports” on page 629.
Information in Progress Reports

Based on the group name or object pattern that you specify and the timestamp you enter, Progress Reports display the following information:

- Recovered and not recovered objects by percentage and in megabytes, gigabytes, and terabytes
  - Table space partitions recovered
  - Table space partitions not recovered
  - Index partitions recovered
  - Index partitions rebuilt
  - Index partitions not recovered or rebuilt
  - Total partitions recovered or rebuilt

- Changed and unchanged objects by percentage and in megabytes, gigabytes, and terabytes
  - Table space partitions changed
  - Table space partitions unchanged
  - Index partitions changed
  - Index partitions unchanged
  - Total partitions changed
  - Total partitions unchanged

- Copied and not copied objects by percentage and in megabytes, gigabytes, and terabytes
  - Table space partitions copied
  - Table space partitions not copied
  - Index partitions copied
  - Index partitions not copied
  - Total partitions copied
  - Total partitions not copied

On the online panels, you can select a category and display detail rows or the Group Object List panel.
Note

■ Clones are not supported.

■ Megabytes are taken from DB2 catalog real time statistics (SYSTABLESPACESTATS and SYSINDEXSPACESTATS). DSNUTILB RECOVER may clear the real time statistics in some cases. 2,147,483,647 is the largest number of megabytes that can be displayed.

■ Objects with an ending log range that is zeros or is greater than the specified timestamp are considered changed. Objects with no log ranges or that have log ranges that end prior or equal to the specified timestamp are considered unchanged.

With a BMC Recovery for DB2 solution password, Progress Report displays backup and recover time estimate information:

■ Backup elapsed time estimates for objects in a group or database/tablespace pattern
  Estimates are included for:
  — Standard copies
  — Snapshot copies
  — Cabinet copies
  Hybrid estimates are also included: total time estimates for cabinet and snapshot copies, standard and snapshot copies, and cabinet and standard copies.

■ Recover elapsed time estimates for objects in a group or database/tablespace pattern using a specified recovery point
  Estimates are included for:
  — Forward recoveries
  — Backout recoveries

■ Whether to save backup and recover time estimates in a user-specified data set

Viewing Progress Report information

This procedure describes how to view Progress Report information online.

1. From the RECOVERY MANAGER Main Menu, select 7. Progress Report, then press Enter.

2. Select the Processing mode.
The following processing modes are available, but options 5, 6, and 7 require a BMC Recovery for DB2 solution password.

- **1. Recovered**, the default value, displays information about recoveries after the timestamp that was entered.

- **2. Changed** displays information about objects that do not have log ranges after the timestamp that was entered.

- **3. Copied** displays information about objects that were copied after the timestamp that was entered.

- **4. Batch JCL** generates JCL for batch program ARMBRPR.

- **5. Backup Est** displays 'what-if' scenarios for backup elapsed time estimates.

- **6. Recover Est** displays 'what-if' scenarios for recover elapsed time estimates.

- **7. View Est** displays time estimate results and provides the option to save estimates in a user-specified file.

3 Enter **Start time** or **Recovery point** in the format **YYYY-MM-DD-HH.MM.SS**.

---

**Note**

Start time is not used when calculating backup elapsed time estimates. Recovery point is used for recovery elapsed time estimates.

---

4 Enter either a **Group owner** and **Group name** or **Database** and **Tablespace**, along with whether you want index information (**Include indexes**).

---

**Note**

Group owner cannot contain wildcard characters. Group name, Database, and Tablespace will accept wildcards.

---

**Viewing Progress Reports for recovered, changed, and copied objects**

This procedure describes how to view Progress Reports for recovered, changed, and copied objects.

1 From the RECOVERY MANAGER Main Menu, select **7. Progress Report**, then press **Enter**.

2 Select **1. Recovered**, **2. Changed**, or **3. Copied** on the ARMRPR01 panel.
3 Enter **Start time** in the format *YYYY-MM-DD-HH.MM.SS*.

4 Enter either a **Group owner** and **Group name** or **Database** and **Tablespace**, along with whether you want index information (**Include indexes**).

Summary information is provided in the following categories:

- TS Partitions
- IX Partitions
- Total Partitions

Each category has subcategories that show the number of objects and the percentage of the total number of objects. Additionally, each subcategory shows the number of MB (megabytes), GB (gigabytes), or TB (terabytes) and the percentage of the total.

The following examples display each of these options.

**Figure 39: Progress Report panel (ARMRPR01) with information about recoveries**

```
ARMRPR01 ================== Progress Report =========================
Command ===> _________________________________________________________________
Start time . . . . .  2015 - 12 - 10 - 08 . 43 . 54 (YYYY-MM-DD-HH.MM.SS)
Group owner . . . . . ___________             Database. . . . . . .  ACPDB4* __
Group name. . . . . . _____________________   Tablespace. . . . . .  ________
Include indexes . . . N (Y/N)                Include indexes . . .  Y (Y/N)
Simulated Recovery. . Y (Y/N)
Select one category below and press Enter - S=Detail G=Group Object List

TS Partitions          Objects            Total Bytes MB (MB, GB, TB)
- Recovered. . . 0     ( 0 % )    0.000    GB    ( 0 % )
- Not Recovered. 320   (100 % )    2.503    GB    (100 % )

IX Partitions
- Recovered. . . 0     ( 0 % )    0.000    GB    ( 0 % )
- Rebuilt. . . 0     ( 0 % )    55       GB    ( 0 % )
- Not Rec/Reb.. 352   (100 % )    0.568    GB    (100 % )

Total Partitions
- Recovered/Reb. 0     ( 0 % )    0.000    GB    ( 0 % )
- Not Rec/Reb.. 672   (100 % )    3.071    GB    (100 % )
```

**Figure 40: Progress Report panel (ARMRPR03) with information about changed objects**

```
ARMRPR03 ================== Progress Report
Command ===> _________________________________________________________________
Start time . . . . .  2014 - 01 - 01 - 07 . 20 . 18 (YYYY-MM-DD-HH.MM.SS)
Group owner . . . . . ___________             Database. . . . . . .  RMDDB4* __
Group name. . . . . . _____________________   Tablespace. . . . . .  *________
Include indexes . . . Y (Y/N)                Include indexes . . .  Y (Y/N)
Select one category below and press Enter - S=Detail G=Group Object List

TS Partitions          Objects            Total Bytes MB (MB, GB, TB)
- Recovered. . . 0     ( 0 % )    0.000    GB    ( 0 % )
- Not Recovered. 320   (100 % )    2.503    GB    (100 % )
```

Chapter 11  Accessing online Progress Reports  387
### Select one category below and press Enter - S=Detail G=Group Object List

<table>
<thead>
<tr>
<th>TS Partitions</th>
<th>Objects</th>
<th>Total Bytes MB (MB, GB, TB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>_ Changed. . .</td>
<td>37</td>
<td>105 MB (21%)</td>
</tr>
<tr>
<td>_ Unchanged.  .</td>
<td>417</td>
<td>380 MB (79%)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IX Partitions</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>_ Changed. . .</td>
<td>25</td>
<td>25 MB (5%)</td>
</tr>
<tr>
<td>_ Unchanged.  .</td>
<td>491</td>
<td>394 MB (95%)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Total Partitions</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>_ Changed. . .  .</td>
<td>62</td>
<td>130 MB (14%)</td>
</tr>
<tr>
<td>_ Unchanged. .  .</td>
<td>908</td>
<td>774 MB (86%)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IX Partitions</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>_ Copied . .  .</td>
<td>40</td>
<td>31 MB (7%)</td>
</tr>
<tr>
<td>_ Not Copied .</td>
<td>476</td>
<td>388 MB (93%)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Total Partitions</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>_ Copied . .  .</td>
<td>101</td>
<td>158 MB (17%)</td>
</tr>
<tr>
<td>_ Not Copied .</td>
<td>869</td>
<td>747 MB (83%)</td>
</tr>
</tbody>
</table>

#### Figure 41: Progress Report panel (ARMRPR04) with information about copied objects

|------------|------------------------|-----------------|-------------------|-------------------|-----------------------|---------------------|

Select one category below and press Enter - S=Detail G=Group Object List

<table>
<thead>
<tr>
<th>TS Partitions</th>
<th>Objects</th>
<th>Total Bytes MB (MB, GB, TB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>_ Copied . .  .</td>
<td>61</td>
<td>127 MB (26%)</td>
</tr>
<tr>
<td>_ Not Copied .</td>
<td>393</td>
<td>359 MB (74%)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IX Partitions</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>_ Copied . .  .</td>
<td>40</td>
<td>31 MB (7%)</td>
</tr>
<tr>
<td>_ Not Copied .</td>
<td>476</td>
<td>388 MB (93%)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Total Partitions</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>_ Copied . .  .</td>
<td>101</td>
<td>158 MB (17%)</td>
</tr>
<tr>
<td>_ Not Copied .</td>
<td>869</td>
<td>747 MB (83%)</td>
</tr>
</tbody>
</table>

5 In the **Total Bytes** field, enter MB, GB, or TB.

Megabytes (MB) are shown in whole numbers. Gigabytes (GB) and terabytes (TB) are shown with three places after the decimal point. ** indicates that the amount of data is too small to show in three decimal places in TB mode.

**Note**

The **Total Bytes** value is calculated from DB2 catalog real-time statistics. DSNUTILB RECOVER might clear the real-time statistics in some cases.

6 Enter **S** or **G** by a single nonzero item (such as Not Rec/Reb) on panel ARMRPR01, ARMRPR03, or ARMRPR04 to display the detail rows for that item on panel ARMRPR02.

- **S**—view the detail rows for a nonzero subcategory. Figure 42 on page 389 shows an example with detail about indexes rebuilt.
Multiple detail lines may be displayed for an object in the **Recovered** or **Copied** categories.

- **G**—view objects for a nonzero subcategory on a Group Object List panel. You can then generate backup or recover JCL or save the objects in a group.

**Note**

DEFINE NO objects that do not exist appear as not recovered, changed, or copied. Also, COPY NO indexes will also appear as not copied.

---

**Figure 42: Progress Report panel (ARMRPR02)**

<table>
<thead>
<tr>
<th>Command</th>
<th>OBJECT NAME</th>
<th>DSNUM</th>
<th>ICTYPE</th>
<th>STYPE</th>
<th>JOBNAME</th>
<th>TIMESTAMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>===&gt;</td>
<td>RDAWXPD5 PBGTS01</td>
<td>1</td>
<td>e</td>
<td>$IND1CB</td>
<td>2015-12-02-17.23</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RDAWXPD5 PBGTS01</td>
<td>1</td>
<td>e</td>
<td>$IND1CB</td>
<td>2015-12-02-17.24</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RDAWXPD5 PBGX01A</td>
<td>0</td>
<td>b</td>
<td>$IND1CB</td>
<td>2015-12-02-17.23</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RDAWXPD5 PBGX01A</td>
<td>0</td>
<td>b</td>
<td>$IND1CB</td>
<td>2015-12-02-17.24</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RDAWXPD5 PBGX01B</td>
<td>0</td>
<td>b</td>
<td>$IND1CB</td>
<td>2015-12-02-17.23</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RDAWXPD5 PBGX01B</td>
<td>0</td>
<td>b</td>
<td>$IND1CB</td>
<td>2015-12-02-17.24</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RDAWXPD5 PBR4TS01</td>
<td>1</td>
<td>e</td>
<td>$IND1CB</td>
<td>2015-12-02-17.23</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RDAWXPD5 PBR4TS01</td>
<td>1</td>
<td>e</td>
<td>$IND1CB</td>
<td>2015-12-02-17.24</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RDAWXPD5 PBR4TS01</td>
<td>2</td>
<td>e</td>
<td>$IND1CB</td>
<td>2015-12-02-17.23</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RDAWXPD5 PBR4TS01</td>
<td>3</td>
<td>e</td>
<td>$IND1CB</td>
<td>2015-12-02-17.24</td>
<td></td>
</tr>
</tbody>
</table>

Names longer than 8 characters are truncated on panel ARMRPR02. The truncation is indicated by >> at the end of the name. The full name can be seen by entering ZOOM in the command line, placing the cursor on the desired field, and pressing **Enter**. Another method of displaying the full name is to place the cursor on the desired field and press **PF4**.

### Generating JCL for ARMBRPR

This procedure describes how to generate JCL for ARMBRPR.

1. Select **4. Batch JCL** on panels to open the JCL Specification panel (ARMRPR0A).

   With a BMC Recovery for DB2 solution password, you have the option to generate syntax to estimate elapsed time for backups and recoveries and calculate I/O rates.

### Viewing information for backup time estimates

This procedure describes how to view information for backup time estimates.
1 Select **Backup Est** to display the Progress Report panel (ARMRPR05).

**Note**

Option 5 requires a BMC Recovery for DB2 solution password.

2 In the **Total Bytes** field, enter MB, GB, or TB.

Megabytes (MB) are shown in whole numbers. Gigabytes (GB) and terabytes (TB) are shown with three places after the decimal point. ** indicates that the amount of data is too small to show in three decimal places in TB mode.

**Note**

The **Total Bytes** value is calculated from DB2 catalog real-time statistics. DSNUTILB RECOVER might clear the real-time statistics in some cases.

3 Enter calculation options to generate estimates for different scenarios. Options include:

- I/O factor—valid values are 1-10000. The default is 100. A factor of 0 will calculate a new factor estimate.

  **Note**

  You can find the approximate I/O factor in NGT Copy output DD ACPPRTnn or SYSPRINT. The I/O factor is more accurate when copying large objects.

- DASD unit

- MAXTASKS—valid values are 1-32. The default is 1. MAXTASKS specifies the number of subtasks that NGT Copy can use for output.

- Outsize—valid values are 0-4294967295 for KB, 0-4194303 for MB, and 0-4095 for GB. The default is 0.

  **Note**

  Typically, the ARMBRPR batch program calculates values for the DASD I/O factor, Virt Tape I/O factor, and Tape I/O factor fields. If ARMBRPR does not calculate the values, the fields display 100 MB (the default).

**Figure 43: Progress Report panel (ARMRPR05) with information about backup time estimates**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ARMRPR05</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td></td>
<td>4</td>
<td></td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Recovered</td>
<td>Changed</td>
<td>Copied</td>
<td>Batch JCL</td>
<td>Backup Est</td>
<td>Recover Est</td>
<td>View Est</td>
<td></td>
</tr>
<tr>
<td>Start time</td>
<td>2014 - 12 - 02 - 11 . 33 . 33 (YYYY-MM-DD-HH.MM.SS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
If OUTSIZE is 0, the elapsed time estimates are generated as follows:

- Standard HH:MM:SS
- Snapshot HH:MM:SS
- Cabinet HH:MM:SS

If OUTSIZE is greater than 0, the elapsed time estimates are generated as follows:

- Cabinet HH:MM:SS Snapshot HH:MM:SS Hybrid Total HH:MM:SS

**Note**

**:**:** indicates that the time estimate exceeded 100 hours.

The estimates are saved in a temporary file. You can use Option 7 to view or save the accumulated results. Returning to the Main Menu deletes them from the temporary file.

4 Enter S or G under **Total Partitions** to display the detail rows for that item.

**WARNING**

Selecting G deletes all backup and recover time estimates produced in this session.

- S—view all objects.
- G—view objects on a Group Object List panel. You can then generate backup JCL or save the objects in a group.

**Note**

DEFINE NO objects that do not exist appear as not copied. COPY NO indexes also appear as not copied.
Viewing detailed information for recover time estimates

This procedure describes how to view detailed information for recover time estimates.

**Note**

1. Select **6. Recover Est** to display the Progress Report panel (ARMRPR06).

   **Note**
   Option 6 requires a BMC Recovery for DB2 solution password.

2. In the **Total Bytes** field, enter **MB**, **GB**, or **TB**.

   Megabytes (MB) are shown in whole numbers. Gigabytes (GB) and terabytes (TB) are shown with three places after the decimal point. ** indicates that the amount of data is too small to show in three decimal places in TB mode.

   **Note**
   The **Total Bytes** value is calculated from DB2 catalog real-time statistics. DSNUTILB RECOVER might clear the real-time statistics in some cases.

3. Enter calculation options to generate estimates for different scenarios. Options include:

   - **I/O factor**—valid values are 1-10000. The default is 100. A factor of 0 will calculate a new factor estimate.

     **Note**
     You can find the approximate I/O factor in NGT Copy output DD ACPPRTnn or SYSPRINT. The I/O factor is more accurate when copying large objects.

   - **DASD unit**

   - **Number of jobs**—valid values are 1-99. The default is 1.

   - **Rebuild indexes**—valid values are Y or N. The default is N.

   - **MAXLSORT**—valid values are 1-32. The default is 1. MAXLSORT specifies how many log sort tasks NGT Recover can run in parallel.
Note

Typically, the ARMBRPR batch program calculates values for the DASD I/O factor, Virt Tape I/O factor, and Tape I/O factor fields. If ARMBRPR does not calculate the values, the fields display 100 MB (the default).

Figure 44: Progress Report panel (ARMRPR06) with recover time estimates

<table>
<thead>
<tr>
<th>Command</th>
<th>Recover Time Estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recovery point</td>
<td>2016-05-05 11:33:33 (YYYY-MM-DD-HH.MM.SS)</td>
</tr>
<tr>
<td>Group owner</td>
<td>Database: RMDDB4*</td>
</tr>
<tr>
<td>Group name</td>
<td>Tablespace: X*</td>
</tr>
<tr>
<td>Include indexes</td>
<td>Y (Y/N)</td>
</tr>
<tr>
<td>Select category below and press Enter:</td>
<td>A=Detail G=Group Object List</td>
</tr>
<tr>
<td>Total Partitions</td>
<td>Objects</td>
</tr>
<tr>
<td>Total Bytes</td>
<td>MB</td>
</tr>
<tr>
<td>Calculation options:</td>
<td></td>
</tr>
<tr>
<td>DASD I/O factor</td>
<td>78</td>
</tr>
<tr>
<td>I/O factor</td>
<td>100 (0-10000)</td>
</tr>
<tr>
<td>DASD unit</td>
<td>Nbr of jobs</td>
</tr>
<tr>
<td>Rebuild indexes</td>
<td>N (Y/N)</td>
</tr>
<tr>
<td>MAXLSORT</td>
<td>1 (1-32)</td>
</tr>
<tr>
<td>Recover Elapsed Time Estimates: (use option 7 to view or save all estimates)</td>
<td>Forward 00:09:39 Backout 00:03:00</td>
</tr>
</tbody>
</table>

The elapsed time estimates are generated as follows:

- Forward HH:MM:SS
- Backout HH:MM:SS

Note

**:**:** indicates that the time estimate exceeded 100 hours.

The estimates are saved in a temporary file. You can use Option 7 to view or save the accumulated results. Returning to the Main Menu deletes them from the temporary file.

4 Enter S, G, A, B, or F under Total Partitions to display the detail rows for that item.

Warning

Selecting G deletes all backup and recover time estimates produced in this session.

- S—view all objects.
- G—view objects on a Group Object List panel. You can then generate backup or recover JCL or save the objects in a group.
**Note**

DEFINE NO objects that do not exist will appear as not copied. COPY NO indexes will also appear as not copied.

- **A**—view the BACKOUT AUTO time estimation, which is the sum of these estimates:
  - The time estimate for recovery of objects that can be backed out
  - The forward recovery time estimate for the recovery of objects that cannot be backed out

If objects cannot be recovered via backout or forward recovery, a message indicates that the objects are not eligible for BACKOUT AUTO recovery. You can view a list of BACKOUT AUTO exceptions, which are the same as backout and forward recovery exceptions.

- **B**—view objects with backout recovery exceptions.

  **Note**

  Objects cannot be backed out if they are LOB, XML, not logged, or have hash organization.

- **F**—view objects with forward recovery exceptions.

---

**Viewing and saving backup or recover time estimates**

This procedure describes how to view and save backup or recover time estimates.

1. Select **7. View Est** to display the Progress Report panel (ARMRPR07).

  **Note**

  Option 7 requires a BMC Recovery for DB2 solution password.

**Figure 45: View Backup/Recover Time Estimates panel (ARMRPR07) with information about backup/recover time estimates**

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Save backup/recover estimation results from options 5 and 6 Y (Y/N)</td>
<td></td>
</tr>
<tr>
<td>If output data set is partitioned, replace existing member N (Y/N)</td>
<td></td>
</tr>
<tr>
<td>Type output data set (including member name if partitioned): Data set name ___________________</td>
<td></td>
</tr>
</tbody>
</table>

***************BACKUP TIME ESTIMATE***************

Estimate Produced at: 2016-05-06-11.39.04.053313
2 Select Y or N to choose whether to save backup/recover estimation results from Options 5 and 6.

Y indicates that the elapsed time estimation results shown on the scrollable portion of the panel will be written in the specified data set.

3 Select Y or N to indicate whether to replace the existing member if the output data set is partitioned.

Y indicates that the output is an existing PDS member. N indicates that the output is a PDS member that does not exist.

4 Enter the output data set, including the member name if the data set is partitioned.

This name must be a valid cataloged data set name. If the data set is a PDS, you must specify a valid member name within parentheses.

--- Note ---

The output data set must exist.

--- Entry field descriptions for the Progress Report panels ---

The following table lists each of the fields on the Progress Report panels.

Descriptions of each field follow in the order in which they appear on the panel.
<table>
<thead>
<tr>
<th>Field name</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| Processing mode     | 1                        | Indicates the type of objects you want to see in the report or that you want to generate JCL  
  - 1 to see recovered/not recovered information.  
  - 2 to see changed/unchanged information.  
  - 3 to see copied/not copied information.  
  - 4 to generate JCL for program ARMBRPR that will report recovered, changed, and copied information.  
  **Note**: The following options require a BMC Recovery for DB2 solution password:  
  - 5 to see backup elapsed time estimates.  
  - 6 to see recover elapsed time estimates.  
  - 7 to see time estimate results and to save estimates.  
  **Note**: Information is reported at the partition level. |
| Start time          | None                     | Indicates the start time to beginning searching for recoveries, changes, and copies  
  Enter the start time in the format *YYYY-MM-DD-HH.MM.SS*.  
  Recovered, changed, or copied information is summarized from this point in time to the current time.  
  **Note**: Start time cannot be a point in the future. |
| Recovery point      | None                     | Indicates the recovery point used for recovery elapsed time estimates  
  Enter the recovery point in the format *YYYY-MM-DD-HH.MM.SS*.  
  **Note**: The recovery point cannot be a point in the future. |
<table>
<thead>
<tr>
<th>Field name</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
</table>
| Group owner and Group name | None                     | Specifies the group owner and group name for which you want Progress Report information. Enter the two-part name for an existing group. **Group owner** (creator ID) can be up to 8 characters. **Group name** can be up to 18 characters. **Group owner** and **Group name** must be enclosed in double quotes if they contain special characters or blanks. RECOVERY MANAGER summarizes the recovered, changed, or copied information for the objects in the group. **Note:**  
  - Group owner/Group name and Database/Tablespace are mutually exclusive.  
  - Group name accepts wildcard characters. Group owner does not accept wildcard characters. |
| Database and Tablespace    | None                     | Specifies the **Database** and **Tablespace** pattern for which you want Progress Report information. Type the **Database** and **Tablespace** explicit names or patterns. **Database** can be up to 8 characters. **Tablespace** can be up to 8 characters. RECOVERY MANAGER summarizes the recovery, changed, or copied information for the objects that satisfy the pattern. A pattern can include the following wildcard characters:  
  - * or %  
    Use an asterisk (*) or percent sign (%) as a wildcard that replaces zero to any number of characters. For example, the pattern RMDDB06.* selects all table spaces in database RMDDB06.  
  - ?  
    Use a question mark (?) as a wildcard that replaces only one character. For example, the pattern RMDBB?6.* selects all table spaces in databases RMDDB06 and RMDDB16.  
  - "  
    Use quotes (") to specify explicit names without wildcard expansion. For example, the pattern "RMDDB06","T?23" matches only object RMDDB06.T?23. To include one or more special characters in an explicit group name, delimit each part of the name with double quotes. **Note:** **Group owner/Group name** and **Database/Tablespace** are mutually exclusive. |
### Field name | RECOVERY MANAGER default | Description
--- | --- | ---
Include indexes | None | Specifies if indexes information is included
Enter Y to include indexes for table spaces matching the **Group owner/Group name** and **Database/Tablespace** pattern.
Enter N to exclude indexes. If indexes exist in a group, they will be included in processing.

Simulate recovery | N | Specifies if progress reporting is available for SIMULATE and INDEP OUTSPACE recoveries and rebuilds
Enter Y to specify that NGT Recover should insert rows into the BMCXCOPY table when each recovery or rebuild finishes; otherwise, enter N.

Detail rows | None | Displays the total number of objects and amount of data, including table spaces and indexes
- **S**
  Displays detail rows for each non-zero subcategory.
- **G**
  Displays detail rows for each non-zero subcategory on a group object list panel. You can generate backup or recover JCL or save the objects in a group.
  **WARNING:** Selecting **G** deletes all backup and recover time estimates produced in the session.
- **B**
  Displays objects with backout recovery exceptions.
- **F**
  Displays objects with forward recovery exceptions.

Total bytes | MB | Specifies objects displayed in **MB** (megabytes), **GB** (gigabytes), or **TB** (terabytes)
Megabytes (MB) are shown in whole numbers. Gigabytes (GB) and terabytes (TB) are shown with three places after the decimal point. ** indicates that the amount of data is too small to show in three decimal places in TB mode.
The **Total Bytes** value is calculated from DB2 catalog real-time statistics. DSNUTILB RECOVER might clear the real-time statistics in some cases.
<table>
<thead>
<tr>
<th>Field name</th>
<th>RECOVERY MANAGER default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calculation options</td>
<td></td>
<td>Specifies calculation options for generating elapsed time estimates:</td>
</tr>
</tbody>
</table>
|                         |                          | ■ I/O Factor
|                         |                          | Valid values are 1-10000. The default is 100. A factor of 0 will calculate a new factor estimate.                                    |
|                         |                          | **Note:** You can find the approximate I/O factor in NGT Copy output DD ACPPRTnn or SYSPRINT. The I/O factor is more accurate when copying large objects. |
|                         |                          | ■ DASD unit                                                                     |
|                         |                          | ■ MAXTASKS
|                         |                          | Valid values are 1-32. The default is 1. MAXTASKS specifies the number of subtasks that NGT Copy can use for output.                 |
|                         |                          | ■ Outsize
|                         |                          | Valid values are 0-4294967295 for KB, 0-4194303 for MB, and 0-4095 for GB. The default is 0.                                            |
|                         |                          | ■ Number of jobs
|                         |                          | Valid values are 1-99. The default is 1.                                                                                                |
|                         |                          | ■ Rebuild indexes
|                         |                          | Valid values are Y or N. The default is N.                                                                                              |
|                         |                          | ■ MAXLSORT
|                         |                          | Valid values are 1-32. The default is 1. MAXLSORT specifies how many log sort tasks NGT Recover can run in parallel.            |

The following are factors for output found by the ARMBRPR batch program for use as a guideline for backup and recover estimates. If these values have not been calculated by ARMBRPR, a default of 100 MB is displayed.

■ DASD I/O Factor
■ Virt Tape I/O Factor
■ Tape I/O Factor
This part presents reference information about the RECOVERY MANAGER batch programs and contains the following chapters:

- ARMBACT—Initialize active logs with DSNJLOGF
- ARMBARC—Archive log data sets
- ARMBCRC—Conditional recovery to a timestamp
- ARMBGEN—Backup and recovery JCL
- ARMBGIM—Impact analysis
- ARMBGPS—Subsystem group split
- ARMBGPV—Group recovery revalidation
- ARMBGRP—Group creation and maintenance
- ARMBLGR—Log range analysis
- ARMBLOG—Archive log creation
- ARMBLRLD—Log range formatting
- ARMBRDC—Recovery data collection report
- ARMBRID—Recover indoubt threads
- ARMBRPR — Progress Reports
- ARMBSDR—Extend recovery point at disaster recovery site
- **ARMBSET**—OBJECTSET processing
- **ARMBSRR**—System resource recovery
- **ARMBTSI**—Time stamp insertion
- **ARMBWDC**—System recovery data collection
ARMBACT—Initialize active logs with DSNJLOGF

This chapter describes ARMBACT—Initialize active logs with DSNJLOGF.

About ARMBACT

The ARMBACT program allows you to initialize all active logs for a specified SSID by calling DSNJLOGF. Each active log will be sent to DSNJLOGF. If a log has been formatted, ARMBACT will continue and process the next active log. All DSNJLOGF messages will be written to the ARMPRINT file.

Authorizations

The following authorizations are required to execute the ARMBACT program:

- Authorized Program Facility (APF) authorization for ARMBACT and the RECOVERY MANAGER load library
- READ authority for BSDS data sets
- ALTER authority for the active log data sets

Building the ARMBACT JCL

Building your own ARMBACT job involves creating JCL that includes the following statements:

- A JOB statement
An EXEC statement

Data definition (DD) statements that specify the use of the following libraries and data sets:
- RECOVERY MANAGER and DB2 load libraries
- Input data sets
- Output data sets

The descriptions in the following subsections provide more details.

Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends that you specify REGION=0M, which makes the amount of virtual storage needed to run the job automatically available when the ARMBACT job is executed. If REGION=0M is not allowed by your organization, specify REGION=4M.

Specifying the EXEC statement

The EXEC statement has the following format:

```c
//stepname EXEC PGM=ARMBACT,PARM='ssid,ARMOPTS=optionSet',
// REGION=4M
```

The variable ssid is the DB2 subsystem on which the program is running. If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

---

**Note**

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

---

The variable `optionSet` is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.
Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBACT to use. For example:

//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//        DD DISP=SHR,DSN=DSNEXIT
//        DD DISP=SHR,DSN=DSNLOAD

Specifying the ARMBACT data set DD statements

This topic describes the data sets that ARMBACT uses.

Each data set is specified by a ddname (data definition name). You must specify all required data sets in the JCL.

- **(required) ARMPRINT**
  The output for messages that are returned from RECOVERY MANAGER. RECOVERY MANAGER will write DSNJLOGF messages in the ARMPRINT output. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

- **ARMMSGs (required)**
  The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of hilvl.RMGR.ARMCTL(ARMMSGs). The data set must be allocated with DISP=SHR.

- **(optional) ARMERROR**
  The output for compiler run time errors. If compiler errors are detected and BMCERROR is not present in the JCL, the errors are printed in the JES log. The data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

- **(optional) ARMOPTS**
  The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARMOPTS is used as the default option set name.

  You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

  //ARMOPTS DD *
  ssid.configurationOption=value
  /*
Sample JCL

The following figure provides a sample of JCL for ARMBACT.

Figure 46: Sample ARMBACT JCL

```
//ARMD0000 EXEC PGM=ARMBACT,PARM='DEF0,ARMOPTS=JLW$OPTS'
// STEPLIB DD DISP=SHR,DSN=SCC.WJLW1111.LOAD
//        DD DISP=SHR,DSN=SCC.TEST1111.BMCLINK
//        DD DISP=SHR,DSN=ARM.WJLW1120.LOAD
//        DD DISP=SHR,DSN=SYS3.DEFQ.DSNEXIT
//        DD DISP=SHR,DSN=CSGI.DB2V10M.DSNLOAD
//        DD DISP=SHR,DSN=ARM.WJLW1120.CNTL(ARMMSGS)
//        DD SYSOUT=*  
//        DD SYSOUT=*  
```

Sample output

The following figure shows sample output for ARMBACT.

Figure 47: Sample ARMBACT output

```
** RECOVERY MANAGER FOR DB2 V11.2.00 - INITIALIZE ACTIVE LOGS 12/18/2014 18:26:45 **  
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.2.00  
(c) COPYRIGHT 1994-2015 BMC SOFTWARE, INC.  
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817, 5761676 AND 8880479  
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884  
BMC80223I MAINT: NO RECOVERY MANAGER PTFS APPLIED  
BMC80223I SOLUTION COMMON CODE V11.1.01  
BMC80223I MAINT: NO SCC PTFS APPLIED  
BMC80799I OPTION SET ARM$OPTS LAST UPDATED 10/20/2014 17:50:20 BY MVSVXV2  
BMC80799I DBC SSID = DC2A  STARTED TASK = DBCRI101  JOBID = 0263280  
DSNJ991I DSNJLOGF START OF LOG DATA SET PREFORMAT FOR JOB ARMBSRF4 ARMD0023  
DSNJ992I DSNJLOGF DDNAME = SYSUT1, LOG DATA SET NAME =  
```
Executing the JCL

This section describes special instructions or information required to run the ARMBACT JCL.

- Ensure that you have the appropriate authorizations. See “Authorizations” on page 403 for required authorizations.

- No restart is available for ARMBACT. You must resubmit the job after correcting any error conditions.
AR MBARC—Archive log data sets

The following section describes the AR MBARC—Archive log data sets.

About AR MBARC

The archive log copy program, AR MBARC, enables you to make up to two additional copies of the original archive log data sets for transport to the recovery site.

When an archive log data set is in use, AR MBARC waits until the file is available. AR MBARC can also be used to identify image copy information for the DB2 special spaces that are registered on the log. You can use AR MBARC at the recovery site to move archive data from tape to disk. In addition, AR MBARC supports syntax for the BMC PACLOG product.

Note

Image copy information for most table spaces in DB2 is registered in the SYSIBM.SYSCOPY table. However, several spaces have their image copies registered in the log. These spaces are DSNDB01.DBD01, DSNDB06.SYSCOPY, DSNDB01.SYSDBDXA, and DSNDB01.SYSUTILX. AR MBARC stores image copy registration information that is found in the log into the archive history file so that ARMBSRR can optimize recovery JCL for the catalog and directory.

AR MBARC checks the archive history file to identify archive log data sets that have not been processed, as well as those for which no recovery site copies have been made. It makes the requested number of copies for each log that has no recovery site copies and writes those copies to the media that you specify (tape or disk). RECOVERY MANAGER automatically catalogs the copies at the local site. Options enable you to limit the number of logs that RECOVERY MANAGER copies or processes.

The archive log history file data set name is found in RECOVERY MANAGER options. The data set is dynamically allocated. If the archive log history file is not found, RECOVERY MANAGER will create and initialize it.
All original archive log data sets that are required to be processed by ARMBARC must be cataloged. RECOVERY MANAGER does not process uncataloged archive log data sets.

The ARMBARC program creates a set of three data sets for each archive log data set that is copied, as follows:

- A copy of the Boot Strap Data Set (BSDS)
- A copy of the archive log data set
- A copy of the archive history file

New users of RECOVERY MANAGER must run ARMBARC or use the sample member ARMHSTEX in the .CNTL data set to create the history file for each DB2 subsystem.

**Authorizations**

The following authorizations are required to execute the ARMBARC program:

- Authorized Program Facility (APF) authorization for ARMBARC and the RECOVERY MANAGER load library
- READ authority for archive log data sets
- READ authority for BSDS data sets
- ALTER authority for the new archive log data sets to be created, if any
- ALTER authority for the archive history file
- ALTER authority for the active log data sets

**Building the ARMBARC JCL**

Building your own ARMBARC job involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
Data definition (DD) statements that specify the use of the following libraries and data sets:
- RECOVERY MANAGER and DB2 load libraries
- Input data sets
- Output data sets

The descriptions in the following topics provide more details.

**Specifying the JOB statement**

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends that you specify REGION=0M, which makes the amount of virtual storage needed to run the job automatically available when the ARMBARC job is executed. If REGION=0M is not allowed by your organization, specify REGION=4M.

**Specifying the EXEC statement**

The EXEC statement has the following format:

```plaintext
//stepname EXEC PGM=ARMBARC,PARM='ssid,ARMOPTS=optionSet',
//             REGION=0M
```

The variable `ssid` is the DB2 subsystem on which the program is running. If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

The variable `optionSet` is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

**Note**

- In a data sharing environment, you must run ARMBARC on each DB2 subsystem.

- The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.
Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBARC to use.

Example

```
//STEPLIB DD DISP=SHR,DSN=PRODUCT LOAD LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
```

Specifying the ARMBARC data set DD statements

This topic describes the data sets that ARMBARC uses.

Each data set is specified by a ddname (data definition name). You must specify all required data sets in the JCL.

- **(required) ARMIN**
  The input data set that contains one or more control statements. Attributes for this data set must be fixed-length records with a length of 80 (RECFM=F or FB, LRECL=80).

- **(required) ARMPRINT**
  The output for messages that are returned from RECOVERY MANAGER. RECOVERY MANAGER also echoes the contents of the ARMIN data set in the ARMPRINT output. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

- **(required) ARMMSGS**
  The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of hilvl.RMGR.ARMCNTL(ARMMSGS). The data set must be allocated with DISP=SHR.

- **(optional) BMCERROR**
  The output for compiler run time errors. If compiler errors are detected and BMCERROR is not present in the JCL, the errors are printed in the JES log. The data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.
(optional) ARMOPTS

The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

```
//ARMOPTS DD *
ssid, configurationOption=value /*
```

## Control cards and syntax

The ARMBARC syntax and option descriptions in this section are provided as control cards to be used when you build ARMIN input.

For information about running ARMBARC from the RECOVERY MANAGER online interface, see “Creating an archive log copy job” on page 315. For the control statement for ARMBARC, see Figure 48 on page 413.

---

**Note**

For more information on syntax rules and wildcard support, see “Syntax rules” on page 102.

---

**Figure 48: ARMBARC syntax**
Figure 49: ARMBARC syntax—copy specification

Figure 50: ARMBARC—archive options

* TAPE is not valid with ARCHIVE1
This option enables you to capture the copy information for the DB2 spaces, DSNDB06.SYSCOPY, DSNDB01.DBD01, DSNDB01.SYSDBDXA, and DSNDB01.SYSUTILX in the history file without copying any archive logs.

ARMBARC stores image copy registration information that is found in the log into the archive history file so that ARMBSRR can optimize recovery JCL for the catalog and directory.

Use the LIMIT global option to reduce the number of archive logs scanned to locate image copy information.

Copy specifications

The following options are used to define copy specifications for ARMBARC:
ARCHIVE1 and ARCHIVE2

Use these options to move archive log data from tape to disk at the recovery site. ARCHIVE1 is coded to move the first copy of the archive log from tape to disk. ARCHIVE2 is coded to move the second copy of the archive log from tape to disk. The logs are named according to their current names in the BSDS.

Be aware of the following items:
- The ARCHIVE2 option is only valid if there are dual archives in the BSDS.
- The DISK option (not the TAPE option) should be coded.

If you specify ARCHIVE1 and it is not available for one of the following reasons, RECOVERY MANAGER switches to use ARCHIVE2:
- Missing from the list
- Not cataloged
- Allocation fails

If ARCHIVE2 fails, ARMBARC writes error messages and ends with RC 8. If ARCHIVE2 works, ARMBARC writes warning messages, sets RC 4, and continues processing.

ARCHIVE3 and ARCHIVE4

Use these options to specify which recovery site archive log copies to make. In the control input data set, provide the keyword ARCHIVE3 to make the first copy. Provide both ARCHIVE3 and ARCHIVE4 to generate two copies.

PREFIX

Use this option to specify the data set prefix for each offsite archive log copy. In the control input data set, provide the keyword PREFIX after ARCHIVE3 and ARCHIVE4, followed by the prefix that you want to use.

Note

Because the archive log number is appended to the data set name, the name cannot exceed 35 characters. If the timestamp (TSTAMP) configuration option is set to Y in the RECOVERY MANAGER option set, the allowable prefix length is further reduced to 17 characters to allow the data and time to be appended to the log date set name.

Archive options

You must supply the following information for each copy that you request. The information may be different for each copy.
**DISK options**

**DISK**

DISK is the default unit for the archive log copies. You can specify the following options separately for each copy when you choose DISK as the output device.

**DATACLAS**

You can optionally specify a valid SMS data class name for the copies (not to exceed 8 characters). Use this option only when you use SMS. In the control input data set, provide the keyword DATACLAS followed by a data class name.

**MGMTCLAS**

You can optionally specify a valid SMS management class name for the copies (not to exceed 8 characters). Use this option only when you use SMS. In the control input data set, provide the keyword MGMTCLAS followed by a management class name.

**STORCLAS**

You can optionally specify a valid SMS storage class name for the copies (not to exceed 8 characters). Use this option only when you use SMS. In the control input data set, provide the keyword STORCLAS followed by a storage data class name.

**UNIT**

The unit name. The default unit name is SYSALLDA. If this unit designation is not valid or if you prefer to use another generic or esoteric name, you can specify one.

**UNITCNT**

This option specifies the number of units to be allocated for the output log copies. The default is to leave this option blank, which enables the unit count to be controlled by SMS. If you want to override the system value for this option, you can specify an integral number from 1 to 59.

**TAPE options**

**TAPE**

Specify tape as the output type. You must also provide a unit name.
EXPDT

You can optionally specify the expiration date for the copy data set. The date must be in the format yyyy/ddd, where:
- yyyy is the year
- ddd is the Julian date

The ARMBARC program does not provide a default. If you specify an expiration date, you cannot specify a retention period.

RETPD

You can optionally specify the retention period in days for the copy data set. The valid range is 0 through 999. The ARMBARC program does not provide a default. If you specify a retention period, you cannot specify an expiration date.

STACK

You can specify whether to stack the archive log data set copies contiguously on a new tape volume. In the control input data set, provide the keywords STACK NO to prevent stacking. STACK YES is the default.

TRTCH

You can specify whether the hardware compression in the tape drive unit is to be enabled or disabled. If you disable TRTCH compression (TRTCH NOCOMP), a tape management system or operating system default may apply. In the control input data set, provide the keywords TRTCH COMP or TRTCH NOCOMP.

UNIT

The unit name. This option is required with TAPE.

The default unit name is SYSALLDA. If this unit designation is not valid or if you prefer to use another generic or esoteric name, you can specify one.

**Note**

You can specify the following options separately for each copy when you choose tape as the output device.

**ZIIP option**

The ZIIP option specifies whether to attempt to use IBM System z Integrated Information Processors (zIIPs). RECOVERY MANAGER can use enclave service request blocks (SRBs) to enable zIIP processing automatically while running jobs.
Using zIIP processing can reduce the overall CPU time for RECOVERY MANAGER jobs.

You can specify one of the following values:

- **ENABLED** tells RECOVERY MANAGER to attempt to offload eligible processing to an available zIIP. If the zIIP is busy or not available, normal processing continues on a general-purpose processor.

- **DISABLED** tells RECOVERY MANAGER not to attempt to use zIIP processing.

To enable and use zIIP processing with RECOVERY MANAGER, you must:

- Have an installed authorized version of XBM or SUF
- Start and maintain an XBM subsystem in your environment
- Have a zIIP available in your environment

You can specify a particular XBM subsystem to use by specifying a value for the XBMID option (with a length of up to 8 characters), or RECOVERY MANAGER will discover an XBM subsystem that meets the requirements for zIIP processing.

XBM and SUF are licensed, installed, and maintained separately from RECOVERY MANAGER. You can use either XBM or SUF, depending on the license that you have obtained:

- A license for the full version of the XBM product authorizes you to use all features of XBM.

- A license for SUF authorizes you to use only the snapshot and zIIP-processing features of XBM.

**Filter options**

If you are using PACLOG in addition to ARMBARC, you can include the following optional specifications in your ARMBARC syntax.

**WARNING**

To avoid inadvertently removing records that you might need for nonrecovery purposes, consider using different filter options for each of your processed copies.

**FILTERIX**

Use FILTERIX to specify whether to remove index-related records from the archive log data set.

- Use FILTERIX NONE (the default) to keep all such records.
- Use FILTERIX ALL to filter out those records.
- Use FILTERIX EXCEPT \((ixspaceList)\) to exclude selected index spaces from filtering, thus allowing index recovery from log data.

---

**Note**

You can use an asterisk (*) as a wildcard to match a pattern in either or both the database and index space name.

---

**FILTERRECTYPE**

Use FILTERRECTYPE to specify whether to remove from the archive log other record types that are not needed for the DB2 forward recovery processes.

- Use FILTERRECTYPE NONE (the default) to keep all such records.
- Use FILTERRECTYPE ALL to filter out those records.

**FILTERTS**

Use FILTERTS and a DB2 table space specification to remove from the archive log all records that pertain to those table spaces. The list must be enclosed in parentheses and the items in the list must be separated by commas. You can use an asterisk (*) as a wildcard to match a pattern in either or both the database and table space names.

---

**WARNING**

If you use this option and you attempt to recover a table space that requires the DB2 log, the recovery will fail.

---

**Global options**

The following specifications are optional. However, when you first start to process log data sets, you should specify search limits. If you do not specify a limit, the ARMBARC program processes all archive log data sets that are currently registered in the BSDS that have not been processed. RECOVERY MANAGER searches the archive history file to make this determination.

**LIMIT**

This option specifies the limits for ARMBARC to use when searching for archive logs to process. You can express the limit as a number of hours, as a number of logs, or as a relative byte address (RBA) range.
When working at the disaster recovery (DR) site, you should change these limits.

**RBARANGE**

Use RBARANGE to limit the number of logs processed to those created within the specified RBA range. To process from a specific STARTRBA to the current time, specify an ENDRBA of FFFFFFFF0000.

**HOURS**

Use HOURS to limit the number of logs that are processed to those covering the last $n$ hours.

**LOGS**

Use LOGS to limit the number of logs that are processed by the specified number of logs. Log data sets are counted backwards with the most recent archive log data set being considered as the first data set. This value ranges from 1 to 9999.

---

**Sample JCL**

This section includes two samples of JCL for ARMBARC.

Sample 1 ([Figure 46 on page 406](#)) shows the JCL that creates copies of the archive log and updates the archive history file with image copy information for SYSCOPY, SYSUTILX, SYSDBDXA, and DBD01.

**Figure 53: Sample ARMBARC JCL—Archive log copies and history file**

```plaintext
//ARMD001 EXEC PGM=ARMBARC,PARM='DECI,ARMOPTS=ARM$OPTS',
//            REGION=4M
//STEPLIB  DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//        DD DISP=SHR,DSN=DSNEXIT
//        DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS  DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARMPRINT DD SYSOUT=*  
//BMCERROR DD SYSOUT=*  
//ARMIN     DD *
ARCHIVE3 PREFIX DECICAT.LOGCOPY3
   TAPE UNIT CART
   STACK YES
ARCHIVE4 PREFIX DECICAT.LOGCOPY4
   TAPE UNIT CART
   STACK YES
LIMIT LOGS 2
```
Sample 2 (Figure 54 on page 422) shows the JCL that only updates image copy information in the archive history file.

**Figure 54: Sample ARMBARC JCL—History only (HISTONLY)**

```
//ARM0001 EXEC PGM=ARMBARC,PARM='DECI,ARMOPTS=ARM$OPTS',
// REGION=4M
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARMPRINT DD SYSOUT=* 
//BMCERROR DD SYSOUT=* 
//ARMIN DD *
LIMIT LOGS 2
HISTONLY
```

**Sample output**

The following shows two samples of ARMBARC output.

Sample 1 (Figure 47 on page 406) shows the output from the job that copies the archive log and updates the archive history file with image copy information for SYSCOPY, SYSUTILX, SYSDBDXA, and DBD01.

**Figure 55: Sample ARMBARC output—Archive log copy**

```
** RECOVERY MANAGER FOR DB2 V11.2.00 - ARCHIVE LOG COPY 08/14/2014 09:43:27 **
(c) COPYRIGHT 1994-2015 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676

BMC80223I MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT: BPJ0197 BPJ0215 BPJ0219

ARCHIVE3 PREFIX DECICAT.LOGCOPY3
  TAPE UNIT CART
  STACK YES
ARCHIVE4 PREFIX DECICAT.LOGCOPY4
  TAPE UNIT CART
  STACK YES
LIMIT LOGS 2

BMC80649I BSDS ANALYSIS COMPLETE 08/14/2014 09:43:42

BMC80650I DECICAT.LOGCOPY3.B0087937 CREATED FROM DECICAT.ARCLG1.B0087937
BMC80650I DECICAT.LOGCOPY4.B0087937 CREATED FROM DECICAT.ARCLG1.B0087937
BMC80655I  397646722 BYTES PROCESSED - TOTAL
BMC80650I DECICAT.LOGCOPY4.A0087937 CREATED FROM DECICAT.ARCLG1.A0087937
BMC80650I DECICAT.LOGCOPY3.H0087937 CREATED FROM UPDATED ARCHIVE HISTORY
```
Sample 2 (Figure 56 on page 423) shows the output from the job that updates only image copy information in the archive history file using the HISTONLY option.

**Figure 56: Sample ARMBARC output—HISTONLY**

```plaintext
** RECOVERY MANAGER FOR DB2 V11.2.00 - ARCHIVE LOG COPY 08/14/2014 09:45:33 **
(c) COPYRIGHT 1994-2015 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676

BMC802231 MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC802231 SOLUTION COMMON CODE V11.1.00
BMC802231 MAINT: BPJ0197  BPJ0215  BPJ0219

  LIMIT LOGS 2
  HISTONLY

BMC80649I BSDS ANALYSIS COMPLETE 08/14/2014 09:45:46
BMC80653I DECICAT.ARCLG1.A0087938 PROCESSED
BMC80649I ARCHIVE LOG PROCESS COMPLETE 08/14/2014 09:45:54

BMC80653I DECICAT.ARCLG1.A0087938 PROCESSED
BMC80649I ARCHIVE LOG PROCESS COMPLETE 08/14/2014 09:46:03

BMC80571I PROGRAM COMPLETE RC = 0
```
Executing the JCL

This section describes special instructions or information required to run the ARMBARC JCL.

- Ensure that you have the appropriate authorizations. See “Authorizations” on page 410 for required authorizations.

- When you execute ARMBARC for the first time, you may not want to process all archive logs that are recorded in the bootstrap. To process only a recent subset of logs, use the LIMIT parameter (see “Global options” on page 420).

- On a data sharing subsystem, you must run ARMBARC on each member in order to copy the archive logs for each member. In addition to copying logs for each member, ARMBARC also records the image copy information for the special catalog and directory spaces in the history file for each member.

- Use the ARMBLOG program to issue the DB2 ARCHIVE LOG command and wait for offload processing to complete before running ARMBARC. See “ARMBLOG—Archive log creation” on page 603 for more information.

- No restart is available for ARMBARC. You must resubmit the job after correcting any error conditions.
ARMBCRC—Conditional recovery to a timestamp

This chapter describes ARMBCRC—Conditional recovery to a timestamp.

About ARMBCRC

RECOVERY MANAGER uses ARMBCRC to determine the RBA or LRSN of a timestamp for conditional restart recoveries and to perform full subsystem recoveries to a timestamp.

Full subsystem recoveries are often necessary for ERP applications such as SAP. The RBA or LRSN can also be used to perform coordinated disaster recoveries across multiple DB2 subsystems.

After you have established a timestamp-based recovery point using the ARMBTSI program (see “ARMBSI—Time stamp insertion” on page 719), the ARMBCRC program uses the timestamp that is recorded in the CRRDRPT table to analyze the DB2 log and determine the RBA to be used as the conditional restart control record ENDRBA. ARMBCRC reads active and archive logs as needed. If the program runs near the time of the established recovery point, it will normally read only the active log.

Note

- The timestamp recorded in the CRRDRPT table must be equal to or less than the current time. If you enter a timestamp greater than the current time (that is, a time in the future), ARMBCRC bypasses the entry and issues an informational message.

- ARMBCRC will convert all timestamps in the CRRDRPT table that have not already been converted.
Establishing a recovery point

You must run an ARMBTSI job to establish a recovery timestamp before running ARMBCRC.

The ARMBTSI program inserts a timestamp in the format required by ARMBCRC. The JCL to run this program can be generated online when you choose the Establish a Recovery Point option on the Disaster Recovery menu. For more information about ARMBTSI, see “ARMBTSI—Time stamp insertion” on page 719 or “Full subsystem recovery” on page 347.

Authorizations

The following authorizations are required to execute the ARMBCRC program:

- APF authorization for ARMBCRC and the RECOVERY MANAGER load library
- READ authority for the bootstrap data set (BSDS)
- ALTER authority for the active log data sets
- ALTER authority for the archive log data sets
- EXECUTE authority on the RECOVERY MANAGER DB2 plan

Building the ARMBCRC JCL

Building your own ARMBCRC job involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition (DD) statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Input data sets
  - Output data sets

The descriptions in the following subsections provide more details.
Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends that you specify REGION=0M, which makes the amount of virtual storage needed to run the job automatically available when the ARMBCRC job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

Specifying the EXEC statement

The EXEC statement has the following format:

```
//stepname EXEC PGM=ARMBCRC,
  PARM='ssid,DATASHARE=type,ARMOPTS=optionSet',REGION=0M
```

where ssid is the DB2 subsystem on which the program is running.

If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

**Note**

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

- type is data sharing type and can be one of the following:
  - GROUP
  - MEMBER
Note

- The default for data sharing systems is DATASHARE=MEMBER.
- If your system is data sharing and you want to run ARMBCRC on only one member, specify DATASHARE=MEMBER.
- If your system is data sharing and you want to run ARMBCRC on all members, specify DATASHARE=GROUP. Synchronization steps (ARMBSYN) will be generated and jobs for each data sharing member will be generated on the LPARs where they exist.
- If your system is not data sharing, do not specify the DATASHARE parameter.

- optionSet is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBCRC to use.

Example

```c
//STEPLIB   DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
       DD DISP=SHR,DSN=DSNEXIT
       DD DISP=SHR,DSN=DSNLOAD
```

Specifying the ARMBCRC data set DD statements

This section describes the data sets that ARMBCRC uses.

Each data set is specified by a `ddname` (data definition name). You must specify all required data sets in the JCL.

- (required) ARMPRINT
  This is the output for messages that are returned from RMGR. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.
- **(required) ARMMSGS**

  The RMGR messages data set created during RMGR installation with the default name of `hilvl.RMGR.ARMCNTL(ARMMSGS)`. The data set must be allocated with DISP=SHR.

- **(optional) ARMERROR**

  The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. The data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

- **(optional) ARMOPTS**

  The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

  You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

  ```
  //ARMOPTS DD *
  ssid.configurationOption=value
  /*
  ```

Sample JCL

The following figure provides a sample of JCL for ARMBCRC.

**Figure 57: Sample ARMBCRC JCL**

```jcl
//ARMOPTS DD *
ssid.configurationOption=value
/*
```

Sample output

The following figure provides a sample of output produced by ARMBCRC.

**Figure 58: Sample ARMBCRC output**

```
** RECOVERY MANAGER FOR DB2 V11.2.00 - CONVERT TIMESTAMP TO LRSN 08/13/2014 13:22:45 **
(c) COPYRIGHT 1994-2015 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
BMC80223I MAINT: NO RECOVERY MANAGER PTFs APPLIED
```
Executing the JCL

This section describes special instructions or information required to run the ARMCRC JCL.

- Ensure that the job owner has appropriate authority for the BSDS and log data sets. For required authorizations, see “Authorizations” on page 426.

- If your system is data sharing and you want to run ARMCRC on only one member, specify DATASHARE=MEMBER.

- No restart is available for ARMCRC. You must resubmit the job after correcting any error conditions.
ARMBGEN—Backup and recovery JCL

This chapter describes the ARMBGEN and ARMBGNR programs available in the RECOVERY MANAGER product. ARMBGEN generates JCL to back up or recover one or more application groups, and ARMBGNR copies the JCL to its final destination.

About ARMBGEN

ARMBGEN enables you to generate backup and recovery JCL for one or more groups offline instead of using a TSO session.

Note

JCL generation for application recovery considers BACKUP SYSTEM full volume backups as a valid backup for DSNUTILB recovery. NGT Recover does not support full volume backups so BACKUP SYSTEM backups are ignored if the recover utility is AFRMAIN.

You can use ARMBGEN in the following ways:

■ Code JCL to run ARMBGEN to create backup or recovery JCL. This method completely avoids using a TSO session.

■ Use online support that is provided by RECOVERY MANAGER to create ARMBGEN JCL. This approach requires only a short TSO session.

Note

When you generate recovery JCL, all groups specified in the recovery must have the same type of recovery point.

For more information about ARMBGEN online support, see “Generating recovery JCL in batch” on page 210.

You can optionally update the backup and recovery options for the specified group or groups. This is the GROUP UPDATE feature of the ARMBGRPN program and all
options are described in detail in “ARMBGRP—Group creation and maintenance” on page 507. If you do not change the options in the ARMBGEN syntax, ARMBGEN uses the options that are currently in effect for each group or groups.

**Note**
The changes that you make to backup and recovery options using the ARMBGEN program are not stored in the repository and are in effect only for the duration of the ARMBGEN execution. If you want the options to be saved with the specified groups and remain in effect for future backup and recoveries, set them using the ARMBGRP program or by using the online interface.

### About XUNCHANGED processing in local subsystem recovery

During local full subsystem recoveries, RECOVERY MANAGER uses the XUNCHANGED option of ARMBGEN to identify and exclude objects that have not changed between the recovery time and the current time.

This process can significantly reduce recovery time by avoiding unnecessary recoveries.

RECOVERY MANAGER first analyzes SYSCOPY and SYSLGRNX information to identify objects that appear to be unchanged and mark them as unchanged. After the DB2 catalog is recovered, RECOVERY MANAGER compares information in the DB2 catalog with the information stored in the RECOVERY MANAGER log range file that is built by program ARMBLGR during the preparation for local subsystem recovery. RECOVERY MANAGER does the following comparisons, which may result in an object that is marked unchanged being marked for recovery:

- For each table space identified as unchanged and for all indexes belonging to the unchanged table spaces, RECOVERY MANAGER compares the following values with those in the DB2 catalog. If any difference is found, RECOVERY MANAGER marks the table space or index for recovery.
  - DBID
  - PSID
  - PART
  - INSTANCE
  - IPREFIX
  - VCAT name
  - CREATE timestamp

- RECOVERY MANAGER runs a comparison to identify table spaces that exist in the DB2 catalog but that are not in the RECOVERY MANAGER log range file (a condition that means the table space was dropped after the recovery point).
RECOVERY MANAGER marks any table spaces found in this condition for recovery.

- RECOVERY MANAGER runs a comparison to identify indexes that exist in the DB2 catalog but that are not in the RECOVERY MANAGER log range file (a condition that means the index was dropped after the recovery point). RECOVERY MANAGER marks any indexes found in this condition for recovery.

- RECOVERY MANAGER identifies orphan VSAM data sets that were created after the recovery point and marks them for deletion.

Using ARMBGEN in full subsystem recovery

You can use ARMBGEN to provide more automation for the recovery of an entire DB2 subsystem.

Large applications such as SAP often require that the entire subsystem be included in the backup and recovery process. At the local site, the system resource recovery program, ARMBSRR, generates JCL to recover the subsystem to a prior point in time using a conditional restart. When ARMBSRR is completed, you can run the batch log range analysis program, ARMBLGR, to identify objects that have not changed between the recovery point and the current time. You can then generate application recovery JCL by using ARMBGEN and specifying the XUNCHANGED option. This action excludes unchanged objects from the recovery, thus improving recovery performance.

**Note**

RECOVERY MANAGER requires declared DB2 global temporary tables when generating JCL for unchanged analysis processing during local subsystem recovery. For more information, see “Creating required temporary tables” on page 64.

Using ARMBGEN in disaster recovery planning

You can use ARMBGEN to provide more automation for the recovery of your applications in a disaster recovery situation.

At the local site, the system resource recovery program, ARMBSRR, updates the archive history file with the end relative byte address (RBA) of the disaster recovery point. When ARMBSRR is completed, you can generate application recovery JCL by using ARMBGEN and specifying RESTARTRBA as the recovery type. ARMBGEN uses the end RBA, which was updated by the ARMBSRR job, to generate ready-to-run application recovery jobs that you can transport to the recovery site.
You can also use ARMBGEN to simulate and estimate recovery. Simulation can pinpoint any missing resources or tape copies that are not usable. Estimation can provide information about long-running objects and overall recovery time. (Simulation and estimation are only available with the BMC Recovery Management for DB2 solution.)

You might realize a significant improvement in data set sizing accuracy with this technique when the operating system catalog information is available at the local site but not at the recovery site.

About BACKOUT recovery

A BACKOUT recovery does not require image copies to perform a point-in-time recovery. Instead, it backs out the log records to undo or redo the changes that occurred between the selected point in time and the current point.

This method returns the spaces and indexes to the required state without the overhead of restoring image copies, or rebuilding or restoring indexes. In most cases, the BACKOUT recovery strategy is dramatically faster than traditional forward recovery. See the REORG PLUS for DB2 Reference Manual for more information about the BACKOUT option.

The backout to forward recovery strategy (BACKOUT AUTO) uses both the BACKOUT recovery and the traditional forward recovery functionality of NGT Recover for point-in-time recoveries. Using this strategy, RECOVERY MANAGER generates JCL for NGT Recover to first attempt to back out the spaces that need to be recovered. If any spaces cannot be backed out, NGT Recover automatically performs a forward recovery for those spaces. This option is only valid when you are using RECOVERY MANAGER as part of the BMC Recovery Management for DB2 solution.

You can also use BACKOUT when you choose DB2 RECOVER (DSNUTILB) as the recovery utility. The default value is NO. BACKOUT with DSNUTILB has the same restrictions as BACKOUT with NGT Recover.

If DSNUTILB is selected as the recovery utility and the DB2 version is less than Version 10, RECOVERY MANAGER changes BACKOUT to NO and continues.

Be aware of the following limitations:

- BACKOUT AUTO is invalid with DSNUTILB.
- If you specify BACKOUT AUTO or BACKOUT YES, you must choose one of the following recovery points:
  - TOQUIESCE
  - TOCOMMONRECPT
—TOLOGPOINT
—TOTIMESTAMP (Recovery Management solution only)

Recovery to CURRENT, TOCOPY, or TORESTARTRBA are not valid choices with a backout recovery.

- BACKOUT recovery requires that spaces be undamaged and not be in RECP, RECP*, RBDP, RBDP*, PSRCP, PSRBD, GRECP, WEPR, or STOPE status or have an LPL range. BACKOUT also cannot be used for the following spaces:
  - LOB spaces
  - NOT LOGGED spaces

- Table 44 on page 435 lists options that conflict with BACKOUT AUTO.

### Table 44: Options that conflict with BACKOUT AUTO

<table>
<thead>
<tr>
<th>Option</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALWAYS_REBUILD_INDEXES YES</td>
<td>BACKOUT AUTO overrides the request for index rebuilds. ARMBGEN ends with RC=4.</td>
</tr>
<tr>
<td>LOGSCAN YES</td>
<td>LOGSCAN cannot be specified with BACKOUT AUTO. The product issues an error message, and you must change one option or the other to continue.</td>
</tr>
<tr>
<td>OUTCOPY_BY_RECOVER YES</td>
<td>BACKOUT AUTO overrides OUTCOPY, and converts the request to the specified copy utility.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If you chose AFRMAIN as the copy utility, the product converts the request to DSNUTILB.</td>
</tr>
<tr>
<td>UNLOADKEYS_BUILDINDEX</td>
<td>BACKOUT AUTO overrides the UNLOADKEYS_BUILDINDEX option and proceeds with the backout.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If you specify BACKOUT YES with UNLOADKEYS, an error message is issued and you must change one option or the other to continue.</td>
</tr>
</tbody>
</table>

### Authorizations

The following authorizations are required to execute the ARMBGEN program:

- APF authorization for the RECOVERY MANAGER load library
- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- TYPE O (OPEN) authority for the groups (or SYSADM or system DBADM authority)
Building the ARMBGEN JCL

Building your own ARMBGEN job involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition (DD) statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Input data sets
  - Output data sets

The descriptions in the following subsections provide more details.

Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends you specify REGION=0M, which case makes the amount of virtual storage that is needed to run the job automatically available when the ARMBGEN job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

Specifying the EXEC statement

ARMBGEN is an Interactive System Productivity Facility (ISPF) program that is executed in TSO batch mode.

For more information about running TSO in batch mode, refer to the IBM TSO user guide.
Be aware of the following information:

- ARMBGEN requires an ISPF environment due to its use of ISPF services.
- IKJEFT1B is used instead of IKJEFT01 because the program returns the program completion code.

The EXEC statement has the following format:

```
//ARMDD001 EXEC PGM=IKJEFT1B,DYNAMNBR=250,
  PARM='ISPSTART PGM(ARMBGEN) PARM(ssid,ARMOPTS=optionSet),
  REGION=0M'
```

where

- The PARM contains the ISPSTART command for ARMBGEN.
- The variable ssid is the DB2 subsystem ID or data sharing group attach name where the group or groups reside.

If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

**Note**

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

- The variable optionSet is the name of an XML file that contains all of the product’s configuration option values.

The default option set for RECOVERY MANAGER is ARM$OPTS.

### Specifying the STEPLIB DD statement for ARMBGEN JCL

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBGEN to use.

**Example**

```
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// DD DISP=SHR,DSN=DSNLOAD
```
Specifying the ARMBGEN data set DD statements

This topic describes the data sets that ARMBGEN uses.

Each data set is specified by a *ddname* (data definition name). You must specify all required data sets in the JCL.

- **(required) ARMIN**
  The input data set that contains one or more control statements. Attributes for this data set must be fixed length records, with a record length of 80 (RECFM=F or FB, LRECL=80).

- **(required) ARMPRINT**
  The output for messages returned from RECOVERY MANAGER. RECOVERY MANAGER also echoes the contents of the ARMIN data set in the ARMPRINT output. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

- **(optional) ARMOPTS**
  The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

  You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

  ```
  //ARMOPTS DD *
  ssid.configurationOption=value */
  ```

- **(required) ARMMSGs**
  The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of *hilv*/RMGR.ARMCNTL(ARMMSGS). The data set must be allocated with DISP=SHR.

- **(optional) ARMERROR**
  The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. The data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

- **(required) ARMLOAD**
  The RECOVERY MANAGER load library. The data set should be allocated with DISP=SHR.
- **(optional) ARMJCIN**
  If coded, contains the job card that ARMBGEN uses in the generated JCL. Otherwise, the job card that is specified by the JCARD1-JCARD5 variables in the ARM$OPTS option set is used.
  
  For recovery JCL only, you must include the &# symbolic in the job name if recover option MAX_CONCURRENT_JOBS is greater than 1. The symbolic is not required if MAX_CONCURRENT_JOBS is 1.

  Attributes for this data set must be fixed length records, with a record length of 80 (RECFM=F or FB, LRECL=80).

- **(optional, used for local PIT recovery) ARMLGRNX**
  A dummy DD statement that instructs RECOVERY MANAGER to read the SYSLGRNX information from the RECOVERY MANAGER log range file (the ARMLRNG file) instead of SYSLGRNX. You should specify this DD statement when you use ARMBGEN as part of a job stream for a local PIT recovery of a full subsystem.

- **(for local PIT recovery only) ARMWPEND**
  The RECOVERY MANAGER input from a prior ARMBSRR invocation contains information about objects in write-pending status that need to be recovered. This data set is created as by ARMBSRR when the ARMWPEND data set is specified as input. *The data set should not be created by the user.* See “Specifying the ARMBSRR data set DD statements” on page 691 for more information about using ARMWPEND with ARMBSRR.

- **(required) ISPFILE**
  The temporary file that is used by ARMBGEN for creating the backup or recovery job. The use of a temporary file is highly recommended to avoid data set contention when the JCL is to be placed in a partitioned data set (PDS). The data set should be allocated in a previous step as a temporary PDS, DSORG=PO, LRECL=80, RECFM=FB, with space sufficient to hold the backup or recovery JCL. This data set should be referenced in the ARMBGEN step as DISP=(MOD, PASS)

- **(optional) ISPBKUP**
  The temporary file that is used by ARMBGEN for creating a backup copy of the alternate JCL that you can optionally generate for mirrored systems if you are using the Recovery Management *for DB2* solution. (The alternate JCL includes all objects, whether mirrored on not, and is used as a fallback in the event of mirror failure.) The use of a temporary file is highly recommended to avoid data set contention when the JCL is to be placed in a partitioned data set (PDS). The data set should be allocated in a previous step as a temporary PDS, DSORG=PO, LRECL=80, RECFM=FB, with space sufficient to hold the backup or recovery JCL. This data set should be referenced in the ARMBGEN step as DISP=(MOD, PASS)
- **(required) ISPSLIB**
  Required file for ISPF services. Use the RECOVERY MANAGER skeleton library created during RECOVERY MANAGER installation with the default name of `hilvl.RMGR.SLIB`. The data set should be allocated with DISP=SHR.

- **(required) ISPTABL**
  Required file for ISPF services. Use the RECOVERY MANAGER table library created during RECOVERY MANAGER installation with the default name of `hilvl.RMGR.TLIB`. The data set should be allocated with DISP=SHR.

- **(required) ISPMLIB**
  Required file for ISPF services. Use the RECOVERY MANAGER message library created during RECOVERY MANAGER installation with the default name of `hilvl.RMGR.MLIB`. The data set should be allocated with DISP=SHR.

- **(required) ISPTLIB**
  Required file for ISPF services. This data set should be a concatenation of a temporary data set followed by the RECOVERY MANAGER table library, `hilvl.RMGR.TLIB`. The temporary data set should be a PDS with a minimum space of TRK(1,1). Its DCB characteristics should be identical to the RECOVERY MANAGER panel library.

- **(required) ISPPLIB**
  Required file for ISPF services. This data set should be a temporary PDS data set with a minimum space of TRK(1,1), RECFM=FB, LRECL=80.

- **(required) ISPPROF**
  Required file for ISPF services. This data set should be a temporary PDS data set with a minimum space of TRK(1,1), RECFM=FB, LRECL=80.

- **(required) ISPCTL0**
  Required file for ISPF services. This data set should be a temporary sequential data set with a minimum space of CYL(1,1), RECFM=FB, LRECL=80.

- **(required) ISPCTL1**
  Required file for ISPF services. This data set should be a temporary sequential data set with a minimum space of CYL(1,1), RECFM=FB, LRECL=80.

- **(required) ISPLOG**
  Required file for ISPF services. This data set should be a temporary sequential data set with a minimum space of CYL(1,1), RECFM=VB, LRECL=125.

- **(required) SYSTSIN**
  Required file for batch TSO execution. This data set should be a dummy data set (DD DUMMY). 

---

**RECOVERY MANAGER for DB2 User Guide**

440
- *(required)* SYSTSPRT
  Required file for batch TSO execution. This data set should be a SYSOUT data set (DD SYSOUT).

- *(required)* SYSTERM
  Required file for batch TSO execution. This data set should be a SYSOUT data set (DD SYSOUT).

- *(required)* ISPLIST
  Required file for batch TSO execution. This data set should be a SYSOUT data set (DD SYSOUT).

### Building the ARMBGNR JCL

ARMBGEN is followed by a step that conditionally executes the ARMBGNR program to copy the JCL that is created in ISPFILE to its final destination.

ARMBGNR is used instead of IEBGENER because it provides ENQ/DEQ (enqueue/dequeue) support when copying data sets.

Building your own ARMBGNR job involves creating JCL that includes the following statements:

- An EXEC statement

- Data definition (DD) statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Input data sets
  - Output data sets

The descriptions in the following subsections provide more details.

### Specifying the EXEC statement

The EXEC statement has the following format:

```
//ARMD0003 EXEC PGM=ARMBGNR,COND=(4,LT),PARM='MEMBER'
//              REGION=0M
```

The COND parameter ensures that the step will run only if the previous ARMBGEN step has been completed with a condition code of 4 or less.
The EXEC statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. BMC recommends that you specify REGION=0M, which makes the amount of virtual storage needed to run the job automatically available when the ARMBGNR job is executed. If REGION=0M is not allowed by your organization, specify REGION=4M.

The MEMBER parameter is optional and causes each generated job to be copied into separate members. This option only works for certain types of job streams and might impact performance. For more information, see “Separating jobs from a multi-job batch job stream” on page 77.

Specifying the STEPLIB DD statement for ARMBGNR JCL

The STEPLIB DD statement identifies the RECOVERY MANAGER load library that you want ARMBGGEN to use.

```
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// DD DISP=SHR,DSN=DSNLOAD
```

Specifying the ARMBGNR data set DD statements

This topic describes the data sets that ARMBGNR uses.

Each data set is specified by a `ddname` (data definition name). You must specify all required data sets in the JCL.

- **(required) ARMPRINT**
  
  The output for messages returned from RECOVERY MANAGER. RECOVERY MANAGER also echoes the contents of the ARMIN data set in the ARMPRINT output. ARMPRINT may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

- **(required) ARMMSGS**
  
  The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of `hilvl.RMGR.ARMCTRL(ARMMSGS)`. The data set must be allocated with DISP=SHR.

- **(optional) ARMERROR**
  
  The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. The
data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

■ **(required) SYSUT1**
The temporary PDS that is referenced in the ARMBGEN step with the ISPFILE DD statement.

■ **(required) SYSUT2**
Contains the backup or recovery JCL that is created by ARMBGEN. The data set should be LRECL=80,RECFM=FB, BLKSIZE with space sufficient to hold the generated JCL. The data set may be allocated to SYSOUT.

■ **(optional) SYSUT3**
The temporary PDS that is referenced in the ARMBGEN step with the ISPBKUP DD statement for mirrored systems.

■ **(optional) SYSUT4**
Contains the alternate recovery JCL that is created by ARMBGEN for mirrored systems. The data set should be LRECL=80,RECFM=FB, BLKSIZE with space sufficient to hold the generated JCL. The data set may be allocated to SYSOUT.

**ARMBGEN syntax and option descriptions**

The ARMBGEN syntax and option descriptions in this section are the control statements that you use when you build your own job.

For information about running ARMBGEN from the RECOVERY MANAGER online interface, see “Generating a batch ARMBGEN job interactively” on page 164. See “Syntax rules” on page 102 for more information on syntax rules and wildcard support.

**Syntax diagrams**

ARMBGEN control statement syntax for SET CURRENT SQLID is shown in the following figure.

**Figure 59: ARMBGEN control statement—SET CURRENT SQLID**

```
SET CURRENT SQLID = sqlid ;
```
ARMBGEN control statement syntax for SET OPTIONS is shown in Figure 60 on page 444. Syntax descriptions can be found on the pages that are shown in parentheses.

Figure 60: ARMBGEN control statement—SET OPTIONS

ARMBGEN syntax and option descriptions

444  RECOVERY MANAGER for DB2 User Guide
ARMBGEN control statement syntax for GENJCL is shown in Figure 61 on page 445. Syntax descriptions can be found on the pages that are shown in parentheses.

Figure 61: ARMBGEN control statement—GENJCL
Option descriptions

This section contains descriptions of syntax options.

**SET CURRENT SQLID**

In the ARMIN input data set, optionally provide the control statement SET CURRENT SQLID = sqlId to set the SQLID to be used in the reporting and revalidation.

The SQLID defaults to your user ID.

You can have multiple SET CURRENT statements in the control data set. The SET CURRENT SQLID statement is in effect for all statements that follow it until another SET CURRENT SQLID statement is issued.

**SET OPTIONS**

In the ARMIN input data set, provide the optional SET OPTIONS control statement. You can have multiple SET OPTIONS statements in the control data set.

Each SET OPTIONS statement applies to the GENJCL control statement immediately following it.

**SIMULATE**

Simulates disaster recovery for the objects in the group or groups. The recovery simulation feature simulates all aspects of recovery up to, but not including, the actual I/O. You might find disaster recovery simulation useful in reducing your disaster recovery testing costs. The default is NO.

SIMULATE is a feature of the Recovery Management solution and requires the solution password.

You must specify the following options to use SIMULATE YES:

<table>
<thead>
<tr>
<th>Option</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECOVER TO</td>
<td>Specify one of the following values:</td>
</tr>
<tr>
<td></td>
<td>■ CURRENT</td>
</tr>
<tr>
<td></td>
<td>■ COMMON RECOVERY POINT</td>
</tr>
<tr>
<td></td>
<td>■ SPECIFIC LOGPOINT</td>
</tr>
<tr>
<td></td>
<td>■ RESTART RBA</td>
</tr>
<tr>
<td></td>
<td>■ TIMESTAMP</td>
</tr>
</tbody>
</table>
Option | Specification
--- | ---
SITETYPE | Specify one of the following values:
  - RECOVERY
  - LOCAL

SIMULATE RECOVERY | YES

*a If you specify Sitetype=Recovery, you must have RP or RB copies.

---

**Note**

- SIMULATE YES is valid only when NGT Recover is the recovery utility.
- When you are using SIMULATE YES, you cannot set a value for **Max concurrents jobs** greater than 1.
- SIMULATE YES is not valid with the NGT Recover options LOGSCAN YES or UNLOADKEYS/BLDINDEX YES. If these options are specified for the groups for which you are simulating recovery, ARMBGEN issues an error message and does not generate the JCL.
- If you specify SIMULATE YES for a group that uses DSNUTILB as the recovery utility, any JCL that is generated will not execute due to JCL logic that is generated by simulation mode.
- ARMBGEN simulates the recovery of application resources only. The ARMBSRR program can simulate the recovery of the DB2 system resources. Online support for both system and application recovery simulation is also available. For more information, see the *Recovery Management for DB2 User Guide*.

**ESTIMATE**

Estimate application resource recovery for a remote site. When you specify this option, ARMBGEN produces an estimate of the amount of time the recovery of your application resources will take, considering the options that you set and the unique configuration of your subsystem, including number of table spaces, size of table spaces, and more. The default is NO.

This option is only available when you are using RECOVERY MANAGER as a component of the BMC Recovery Management for DB2 solution. For more information about estimation and simulation, see the *Recovery Management for DB2 User Guide*.

**BACKOUT**

You can specify the BACKOUT option whether you are using NGT Recover or DB2 RECOVER (DSNUTILB) as the recovery utility. The BACKOUT
option invokes the backout strategy for point-in-time recovery by using log points (TOLOGPOINT, TOQUIESCE) or a timestamp (Recovery Management solution only). This strategy assumes that spaces are undamaged and that you require a reset to a point in time. The spaces are used along with the log records between the point in time and the current point to back out to the required state. LOGSORT is required to properly order the log records. The default is NO when using RECOVERY MANAGER alone or when DSNUTILB is the recovery utility. The default is AUTO when using RECOVERY MANAGER as part of the BMC Recovery Management for DB2 solution.

Specify one of the following choices:
- **NO**—do not perform backout recovery
- **YES**—perform backout recovery for all objects in the group
- **AUTO**—use the backout-to-forward recovery strategy and not valid with DSNUTILB (For more information, see the *Recovery Management for DB2 User Guide*.)

Be aware of the following information when using BACKOUT:

- If you do not code the BACKOUT option at all in the JCL and you are using AFRMAIN as the recovery utility, ARMBGEN defaults to
  - BACKOUT AUTO if you are using RECOVERY MANAGER as part of the Recovery Management solution
  - BACKOUT NO if you are using RECOVERY MANAGER outside the solution.

- If you are using DSNUTILB as the recovery utility, BACKOUT AUTO is invalid. If you specify DSNUTILB using the online interface, the program overrides BACKOUT AUTO and converts the request to BACKOUT NO during JCL generation.

- If you specify BACKOUT AUTO or BACKOUT YES, you must choose one of the following recovery points:
  - TOQUIESCE
  - TOCOMMONRECPT
  - TOLOGPOINT
  - TOTIMESTAMP

(Recovery to CURRENT, TOCOPY, or TORESTARTRBA are not valid choices with a backout recovery.)

See “About BACKOUT recovery” on page 434 for a list of options that conflict with BACKOUT AUTO.
ANALYZE

This NGT Recover option prints a recovery plan before executing that plan. Specify one of the following choices:

- **YES**—prints a recovery plan and generates JCL for recovery. YES is the default value. Information that is printed includes
  - Names of any required image copy data sets
  - Names of any required log data sets
  - Log ranges, if any
  - Phases that will occur during execution
  - Number of log pages to be read
  - Record sizes for index sort work data sets
  - Steps to occur within each phase

- **NO**—prints only the information in the first four bullets and generates JCL for recovery.

- **ONLY**—terminates after the information is printed.

RECOVERY MANAGER sets ANALYZE to NO if the recovery utility is DSNUTILB.

SITETYPE

Use this option to specify whether the JCL created by ARMBGEN is to be run at the local site (the default) or at a recovery site. If you specify SITETYPE RECOVERY, ARMBGEN automatically sets the following group options to Yes:

- **DELETE_STOGROUP_OBJ**
- **REDEFINE_VCAT_OBJ**

If you specify DB2 Recover (DSNUTILB) as the recovery utility for the group and if you specify the DSNUTILB site type in the DSNUTILB-specific options, then the SITETYPE option is ignored. Also, if you use NGT Recover alternate resources and you select the RP or RB copy as your first choice for the image copy in the recovery options, RECOVERY MANAGER considers the site type to be RECOVERY regardless of what you specify here. For more information about setting recovery utility options using the online interface, see “Setting utility options” on page 125.

UNRECOVER_RC

Specify a return code that ARMBGEN should issue if it encounters an object that cannot or will not be recovered.
Valid values are 0 (the default) through 254. A value of zero causes no return code to be issued and values (even those higher than 4) do not affect the step execution. This return code does not control the overall job return code. If RECOVERY MANAGER encounters errors other than unrecoverable data, it sets the return code based on the most severe condition code of all of the types of errors found.

An object is considered to be unrecoverable by ARMBGEN if it exists in any of the following states:

- ALTER IX
- BAD TYPE
- DS LEVEL
- NOTAVAIL
- TSREORP
- TS STAT

For more information object status, see “RECOVERY MANAGER object exception status” on page 839.

**RESOLVE_INFLIGHTS**

Specify whether inflight units of work are to be resolved during a recovery to a user-specified RBA or LRSN. This option is only valid for a recovery using TOLOGPOINT or TOLOGMARK and is available only with the BMC Recovery Management for DB2 solution.

**JCLTYPE**

Specify whether the JCL is for use at the remote or local site. This option is used for mirroring support only. The default is the value specified in SITETYPE.

ARMBGEN only verifies that resources are actively mirrored if all of the following conditions are true:

- The JCLTYPE is DR
- Mirroring level 2 or higher has been established at the subsystem level
- The recovery is TORESTARTRBA
- The BMC Recovery Management for DB2 solution password is in effect

**Note**

JCLTYPE DR does not turn on the Delete Stogroup Objects option. The Delete Stogroup Objects option must be set in the group options.
REPORTIX

Specify whether to create a report showing all indexes that are associated with table spaces in the group, but that are not included in the recovery because they are not in the group.

REPORTLOBS

Specify whether to create a report showing LOB-related spaces that are associated with table spaces in the group, but that are not included in the recovery because they are not in the group.

REPORTXML

Specify whether to create a report showing XML-related spaces that are associated with table spaces in the group, but that are not included in the recovery because they are not in the group.

REPORTRI

Specify whether to create a report showing all spaces related by referential integrity with table spaces in the group, but that are not included in the recovery because they are not in the group.

REPORTHISTORY

For DB2 Version 10 and later, specify whether to create a report showing spaces that are associated with table spaces in the group by a history (versioning) relationship, but that are not included in the recovery because they are not in the group.

REPORTARCHIVE

Specify whether to create a report showing archive table spaces that meet both of these conditions:

- Are associated with table spaces in the group
- Are not included in the recovery because they are not in the group

CLONES ONLY

Specify whether to include only clones in the processing. Non-cloned objects will be excluded. This option is available only when running on DB2 Version 10 or later and is not valid with compatibility mode.

SYNC

Specify whether to use synchronization steps to control execution of multi-job JCL. SYNC YES generates synchronization steps, which ensure that the jobs are run automatically in the most efficient way possible. SYNC NO causes multi-job JCL to be generated without the synchronization steps. You must run the jobs manually in the correct sequence. The default is YES.
This option requires the MAX_CONCURRENT_JOB option to be greater than 1 and is incompatible with UNLOADKEYS_BUILDINDEX=YES, which requires synchronization steps. SYNC is not available in the online interface.

LOGONLY

Specify whether the recovery is a log only recovery. The default value is NO.

Note

Use this option only when the target application data sets exist at the disaster recovery site.

LOGONLY is not allowed with the following options:
- SIMULATE YES
- TOCOPY
- BACKOUT

OBJECTSET

Specify whether NGT Recover is to use the repository to identify the objects in the group, which means that you do not need to regenerate the recovery JCL when objects in the group change. The default value is YES.

RECOVERY MANAGER generates the TABLESPACE OBJECTSET option and the INDEX OBJECTSET option in the recovery JCL.

GENJCL

You can have multiple GENJCL statements in the ARMIN data set to produce backup or recover JCL for each of your groups.

RECOVER

Specify that the JCL is for recovery. Specify XUNCHANGED to exclude unchanged objects from the recover JCL. Also specify the type of recovery you want.

XUNCHANGED

For local point-in-time recoveries, this option instructs ARMBGEN to analyze SYSLGRNX information and identify those objects that have not been changed between the recovery time and the current time.

For full subsystem recoveries (local point-in-time recoveries via a conditional restart), ARMBGEN analyzes the RMGR log range file (ARMLGRNX DD) to identify unchanged objects.
The objects that are marked as UNCHANGED are excluded from the recover JCL. Objects in WRITE PENDING status are included in the recovery even if they are still marked as UNCHANGED. This option can significantly reduce the time required for recovery by eliminating unnecessary processing.

Updates are not logged for indexes unless they were created with the COPY YES attribute. XUNCHANGED processing does not occur for indexes unless they have the COPY YES attribute.

This option is ignored if you select a recovery to the current time or if you specify SITETYPE=RECOVERY.

**Note**

Organizations with heavy update activity may experience contention on SYSLGRNX when this option is used. Objects with large numbers of log ranges may increase the elapsed time of JCL generation. See “Avoid RUNSTATS on BMCLGRNX” on page 70 for additional information.

Use the NGT Copy MODIFY option or the IBM MODIFY utility to remove old entries. For more information, see the BMC Next Generation Technology Copy for DB2 for z/OS Reference Manual or the IBM DB2 utility guide.

**TOCURRENT**

This type of recovery is the default. When you specify a recovery to the current time, the most recent full and incremental copies are used to recover the spaces. Then, the DB2 log records are used to make the spaces current. The RECOVER syntax defaults to TOCURRENT.

**TOLOGPOINT**

Specify this type of recovery when you want to recover to a log point (RBA or LRSN). You must supply the value (a 12-digit hexadecimal number).

**TORESTARTRBA**

Specify this type of recovery when making preparations at the local site to recover applications at a recovery site. When you specify this type of recovery, ARMBGEN retrieves the RBA stored by ARMBSRR. This action ensures that the recovery JCL will be ready for use at a recovery site after the system-resource recovery jobs generated by ARMBSRR are run.

This option is for local site use only and should not be used at a recovery site as part of a disaster recovery.
**Note**
The analysis for this type of recovery is the same as a recovery to an RBA. However, because the actual recovery is done after a conditional restart, RMGR generates JCL to perform recovery to current.

**TOCOPY**
Specify this type of recovery to recover to a specific full image copy or an incremental image copy registered in SYSIBM.SYSCOPY. You identify the copy by specifying a number from 0 through 99, where 0 represents the most recent copy made and 99 represents the 100th previous copy.

For the purpose of identifying the copy, a DFSMS concurrent copy registered in SYSIBM.SYSCOPY is counted as an image copy.

When you specify OBJECTSET syntax for NGT Recover with TOCOPY syntax, RECOVERY MANAGER generates the following syntax:

```sql
RECOVER OBJECTSET creator.name
TOCOPY LASTCOPY(-1)
```

**TOFULLCOPY**
Specify this type of recovery to recover only to a specific full image copy registered in SYSIBM.SYSCOPY. You identify the copy by specifying a number from 0 through 99, where 0 represents the most recent copy made and 99 represents the 100th previous copy. RECOVERY MANAGER ignores any existing incremental image copies when searching for the specified previous copy.

For the purpose of identifying the copy, a DFSMS concurrent copy registered in SYSIBM.SYSCOPY is counted as an image copy.

**Note**
You cannot specify TOFULLCOPY when you specify OBJECTSET because this is not supported by NGT Recover.

**TOQUIESCE**
Specify this type of recovery when you want to recover each object to a specified quiesce point registered in SYSIBM.SYSCOPY by the DB2 QUIESCE command.

Specify the quiesce point by using the keyword MINUS \( n \), where \( n \) is a number from 0 through 99. The default is MINUS 0, the most recent quiesce point.
When you specify OBJECTSET syntax for NGT Recover with TOQUIESCE syntax, RECOVERY MANAGER generates the following syntax:

```
RECOVER OBJECTSET creator.name
TOLOGPOINT LASTQUIESCE(0)
```

**TOCOMMONRECPT**

Specify this type of recovery when you want to recover all objects to the same point. Similar to recover to quiesce, you identify the common point by specifying a number from 0 through 99, where 0 is the most recent common point.

This option considers only recovery points that are common to all table spaces in the group.

**TOTIMESTAMP**

This option requires the BMC Recovery Management for DB2 solution password and that the recovery utility is NGT Recover. The option is valid on both data sharing systems and non-data-sharing systems. The timestamp recovery feature of BMC Recovery Management for DB2 solution uses inflight resolution technology to perform a consistent point-in-time recovery to any user-specified timestamp. The Recovery Management solution translates the timestamp to a log point, recovers the objects, then resolves all inflight units of work for both data sharing and non-data-sharing systems. Specify the timestamp in the format yyyy-mm-dd-hh.mm.ss.tttttt. For more information about timestamp recovery, see the Recovery Management for DB2 User Guide.

**TOLOGMARK**

Specify this type of recovery to recover to a Log Master for DB2 log mark, which is registered in the Log Master ALPMARK table. You identify the log mark by specifying the log mark name and a version number from 0 through 99, where 0 represents the most recent log mark made and 99 represents the 100th previous log mark. The generated JCL will recover to the log point represented by the log mark.

This option requires use of a BMC Recovery Management for DB2 solution password and use of NGT Recover version 9.1.00 and later as the recovery utility.

**BACKUP**

Use the command GENJCL BACKUP to create backup JCL for the specified group. The copy utility, type of copy, and all options for the group are retrieved from the repository.

You can have multiple GENJCL BACKUP statements in the ARMIN data set.
XUNCHANGED

When you use this option, ARMBGEN analyzes SYSCOPY and SYSLGRNX information to exclude spaces that have not changed since the last backup. This option can significantly reduce the time required for backing up by eliminating unnecessary processing.

You must decide the frequency of full system backups versus backups that exclude unchanged spaces. For example, you could make backups that exclude unchanged objects on a daily basis, while performing a full system backup once per week.

GROUP

You must provide a group name in the form creator.name. You can use a wildcard pattern to specify multiple groups in a single GENJCL statement. If RECOVERY MANAGER finds an explicit group name for a group created by the ARMBGPS program, RECOVERY MANAGER processes the entire set of groups.

ALTLOAD

This option enables you to specify a load library that contains RECOVERY MANAGER. This load library is generated in steplibs.

UPDATE

Update group backup and recovery option settings for the specified group or groups. This is the GROUP UPDATE feature of the ARMBGRP program and all options are described in detail in “ARMBGRP—Group creation and maintenance” on page 507.

WARNING

The changes that you make to backup and recovery options using the ARMBGEN program are not stored in the repository and are in effect only for the duration of the ARMBGEN execution. If you want the options to be saved with the specified groups and remain in effect for future backup and recoveries, set them using the ARMBGRP program or by using the online interface.

RECOVER_OPTIONS

Specifies the recovery options to be updated. For a syntax diagram of the recover options, see “Syntax for updating group options” on page 546. For descriptions of all options that you can update, see “Copy and recover utility options” on page 847.
COPY_OPTIONS

Specifies the copy options to be updated. For a syntax diagram of the copy options, see “Syntax for updating group options” on page 546. For descriptions of all options that you can update, see “Copy and recover utility options” on page 847.

Sample JCL

The following figure provides a sample of ARMBGEN JCL that recovers to a restart RBA.

Figure 62: Sample ARMBGEN JCL—Recover to restart RBA

```
//ARMJCL1 JOB 5220,ARMQA
/**
** DOCUMENTATION
** ARMQA SAMPLE
** GENERATED BY RDAJBM ON 12/08/14 AT 10:20
**
** *****************************************************************
** RECOVERY MANAGER - V11.2.00 - BMC SOFTWARE, INC. *
** *****************************************************************
** *****************************************************************
** *****************************************************************
** CREATE TEMP DATA SET FOR FILE TAILORING *
** *****************************************************************
** ARM0000 EXEC PGM=IEFBR14
** TEMP1     DD DISP=(MOD,DELETE),
**            DSN=RMD.WK.TMPISPF.D120814.T102006,
**            UNIT=WORK,SPACE=(CYL,(10,5)),
**            DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120,DSORG=PS)
** ARM0001 EXEC PGM=IEBGENER
** SYSPRINT DD SYSPRINT=*
** SYSIN DD DUMMY
** SYSUT1 DD DUMMY,
**          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120,DSORG=PS)
** SYSUT2 DD DISP=(NEW,CATLG,DELETE),
**          DSN=RMD.WK.TMPISPF.D120814.T102006,
**          UNIT=WORK,SPACE=(CYL,(10,5)),
**          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120,DSORG=PS)
** *****************************************************************
** BATCH JCL GENERATION *
** FOR SELECTED GROUPS *
** *****************************************************************
** ARM0003 EXEC PGM=IKJEFT1B,DYNAMNBR=250,
**          PARM='ISPSTART PGM(ARMBGEN) PARM(DEC2,ARMOPTS=ARM$OPTS)',
**          REGION=4M
** STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
** DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
** DD DISP=SHR,DSN=DSNEXIT
** DD DISP=SHR,DSN=DSNLAD
** ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
** ARMPRINT DD SYSPRINT=*
** ARMERROR DD SYSPRINT=*
** ARMIN DD *
** SET CURRENT SQLID = ARMQA
** SET OPTIONS SITETYPE LOCAL
```
Figure 63 on page 458 provides a sample of ARMBGEN JCL that recovers to current.

Figure 63: Sample ARMBGEN JCL—Recover to current with UPDATE option

```plaintext
/* *************************************************************** */
/*              COPY TEMP DATA SET TO TARGET DATA SET              */
/* *************************************************************** */
// ARM0004 EXEC PGM=ARMBGNR,COND=(4,LT),REGION=4M
// STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// DD DISP=SHR,DSN=DSNLOAD
// ARMMSGSS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGSS)
// ARMPRINT DD SYSOUT=*
// ARMERROR DD SYSOUT=*
// SYSUT1 DD DISP=(OLD,DELETE,KEEP),
// DSN=RMD.WK.TMPISPF.D120814.T102006
// SYSUT2 DD DISP=SHR,DSN=ROAJB.M.ACT11.O.JCL(SAMPLE4)
/* *************************************************************** */
/*          RECOVERY MANAGER           - BMC SOFTWARE, INC.        */
/* *************************************************************** */
/*             CREATE TEMP DATA SET FOR FILE TAILORING             */
/* *************************************************************** */
// ARM0000 EXEC PGM=IEFBR14
// TEMP1 DD DISP=(MOD,DELETE),
// DSN=RMD.WK.TMPISPF.D120318.T160802.
// UNIT=WORK,SPACE=(CYL,(10,5)).
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120,DSORG=PS)
// ARM0001 EXEC PGM=IEBGENER
// SYSPRINT DD SYSOUT=*
// SYSIN DD DUMMY
// SYSUT1 DD DUMMY,
// DSN=RMD.WK.TMPISPF.D120318.T160802.
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120,DSORG=PS)
```
SYSUT2 DD DISP=(NEW,CATLG,DELETE),
   DSN=RMD.WK.TMPISPF.D120318.T160802,
   UNIT=WORK,SPACE=(CYL,(10,5)),
   DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120,DSORG=PS)
//* *************************************************************** */
//*                      BATCH JCL GENERATION                      */
//* FOR SELECTED GROUPS                                       */
//* *************************************************************** */
ARM0003 EXEC PGM=IKJEFT1B,DYNAMNBR=250,
   PARM="ISPSTART PGM(ARMBGEN) PARM(DEC2,ARMOPTS=ARM$OPTS)".
   REGION=4M
STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
          DD DISP=SHR,DSN=DSNEXIT
          DD DISP=SHR,DSN=DSNLOAD
ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
ARMPRINT DD SYSOUT=*   
ARMERROR DD SYSOUT=*   

ARMIN DD *
   SET CURRENT SQLID = RDAJBM
   SET OPTIONS SITETYPE LOCAL
   BACKOUT NO
   JCLTYPE LOCAL  
   GENJCL
   RECOVER
   TOCOPY MINUS 0
   GROUP "ARMAQ","SAMPLE"
   UPDATE RECOVER_OPTIONS
   RECOVER_UTILITY AFRMAIN
   COPY_AFTER_LP YES
   COPY_AFTER_LB NO
   COPY_AFTER_RP YES
   COPY_AFTER_RB NO
   */
   ISPFILE DD DISP=(MOD,KEEP),
   DSN=RMD.WK.TMPISPF.D120318.T160802
   ISPsLIB DD DISP=SHR,DSN=BMCARM.TEST.DBSLIB
   ISPTABL DD DISP=SHR,DSN=BMCARM.TEST.DBTLIB
   ISPMLIB DD DISP=SHR,DSN=BMCARM.TEST.DBMLIB
   DD DISP=SHR,DSN=SYS1.PROD.ISPMLIB
   ISPTLIB DD DISP=(DELETE),UNIT=WORK,SPACE=(TRK,(1,1,2)),
   DBC=(RECFM=FB,LRECL=80,BLKSIZE=3120,DSORG=PO)
   DD DISP=SHR,DSN=BMCARM.VTEST.DBTLIB
   DD DISP=SHR,DSN=SYS1.PROD.ISPTLIB
   ISPPLIB DD DISP=(NEW,DELETE),UNIT=WORK,SPACE=(TRK,(1,1,2)),
   DBC=(RECFM=FB,LRECL=80,BLKSIZE=3120,DSORG=PO)
   ISPLOG DD DISP=(DELETE),UNIT=WORK,SPACE=(CYL,(1,1)),
   DBC=(RECFM=FB,LRECL=80,BLKSIZE=3120,DSORG=PS)
   DD DISP=SHR,DSN=DSN&LOG
   SYSTSIN DD DUMMY
   SYSTSPRT DD SYSOUT=*   
   SYSTERM DD SYSOUT=*    
   ISPBLD DD SYSOUT=*  
   */
   COPY TEMP DATA SET TO TARGET DATA SET */
   COPY TEMP DATA SET TO TARGET DATA SET */
   ARM0004 EXEC PGM=ARMBGNR,COND=(4,LT)
   STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS

Chapter 15 ARMBGEN—Backup and recovery JCL 459
Sample output

The following figure provides a sample of ARMBGEN output for a recovery to a restart RBA.

Figure 64: Sample ARMBGEN output—Recover to restart RBA

```
** RECOVERY MANAGER FOR DB2 V11.2.00 - BATCH JCL GENERATION 02/18/2014 15:46:28 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.2.00
BMC80223I MAINT:  NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT:  BPJ0021 BPJ0023 BPJ0029 BPJ0031 BPJ0035 BPJ0036
BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910
SET CURRENT SQLID = RDAJBM
;
BMC80570I COMMAND COMPLETE RC = 0
SET OPTIONS SITETYPE LOCAL
BACKOUT NO
JCLTYPE LOCAL
;
BMC80570I COMMAND COMPLETE RC = 0
GENJCL
RECOVER
torestartrba
GROUP "ARMQA":"SAMPLE"
;
BMC80477I SSID=DEC2, RESTARTRBA=002787D50FFF
BMC80526I PROCESSING - GROUP ARMQA.SAMPLE
BMC80539W NOTDEFINED - PRECLUDES RECOVERING - ARMTN22.INDEXNAMEQL18C22EX 0
BMC80539W NOTDEFINED - PRECLUDES RECOVERING - ARMBGN22.TN22EX 0
BMC80539W DEFER - PRECLUDES RECOVERING - ARMBGN22.TN22N10 0
BMC80539W DEFER - PRECLUDES RECOVERING - ARMTN22.IXN22N10 0
BMC80570I COMMAND COMPLETE RC = 4
BMC80571I PROGRAM COMPLETE RC = 4
```
1. The 80477I message indicates the log point to which the spaces will be recovered.

2. The 80539W DEFER message indicates that no recover JCL is generated for the named spaces because those spaces were defined as DEFINE NO.

Figure 65 on page 461 provides a sample of ARMBGEN output for a recovery to current.

**Figure 65: Sample ARMBGEN output—Recover to current with UPDATE option**

```plaintext
** RECOVERY MANAGER FOR DB2 V11.2.00 - BATCH JCL GENERATION 08/12/2014 15:22:10 **
(c) COPYRIGHT 1994-2015 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676

BMC80223I MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT: BPJ0197 BPJ0215 BPJ0219

BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910
SET OPTIONS SITETYPE LOCAL BACKOUT NO ;
BMC80570I COMMAND COMPLETE RC = 0
GENJCL RECOVER TCURRENT GROUP "ARMQA"."BGEN 24A0";
UPDATE RECOVER_OPTIONS
COPY_AFTER_LP YES
COPY_AFTER_LB NO
COPY_AFTER_RP YES
COPY_AFTER_RB NO ;
BMC80526I PROCESSING - GROUP ARMQA.BGEN 24A0
BMC80539W DEFER - PRECLUDES RECOVERING - ARMTN24.IXNL0VL
BMC80539W DEFER - PRECLUDES RECOVERING - ARMBGN24.TN24N10
BMC80539W DEFER - PRECLUDES RECOVERING - ARMTN24.IXNL0VL
BMC80539W WORKFILE - PRECLUDES RECOVERING - DSNDB07.DSN32K01
BMC80539W WORKFILE - PRECLUDES RECOVERING - DSNDB07.DSN32K04
BMC80539W WORKFILE - PRECLUDES RECOVERING - DSNDB07.DSN4K01
BMC80539W WORKFILE - PRECLUDES RECOVERING - DSNDB07.DSN4K02
BMC80539W WORKFILE - PRECLUDES RECOVERING - DSNDB07.DSN4K03
BMC80570I COMMAND COMPLETE RC = 4
BMC80571I PROGRAM COMPLETE RC = 4
```

Figure 66 on page 461 provides a sample of ARMBGEN Output with the SIMULATE option.

**Figure 66: Sample ARMBGEN output—Recover with SIMULATE option**

```plaintext
** RECOVERY MANAGER FOR DB2 V11.2.00 - BATCH JCL GENERATION 08/14/2014 12:11:52 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.2.00 **
(c) COPYRIGHT 1994-2015 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884

BMC80223I MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT: BPJ0197 BPJ0215 BPJ0219

BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910
SET CURRENT SQLID = ARMQA
```
Figure 67 on page 462 provides a sample of ARMBGEN Output with the ESTIMATE option.

Figure 67: Sample ARMBGEN output—Recover with ESTIMATE option
 Executing the JCL

This section describes special instructions or information required to run the ARMBGEN JCL.

- Ensure that you have the appropriate authorizations. For required authorizations, see “Authorizations” on page 435.

- Be sure to perform the following actions before running ARMBGEN:
  — Make image copies
  — Run ARMBGPV
  — Run ARMBSRR if using TORESTARTRBA

- No restart is available for ARMBGEN. You must resubmit the job after correcting any error conditions.

**Note**

Restart procedures are available for the JCL generated by ARMBGEN. For more information, see “Restarting failed recovery jobs” on page 222.
ARMBGIM—Impact analysis

This chapter describes the implementation of ARMBGIM—Impact analysis.

About ARMBGIM

The ARMBGIM program is used to perform plan and package impact analysis to determine what plans or packages (and therefore what applications) are potentially unavailable during a planned backup and recovery.

RECOVERY MANAGER determines the impact information by analyzing the SYSIBM.SYSPLANDEP and SYSIBM.SYSPACKDEP tables for plans and packages respectively.

You can generate printed plan and package impact analysis reports using the ARMBGIM program in the following ways:

- You can code JCL to run ARMBGIM to create an impact analysis report. This completely avoids using a TSO session.
- You can use online support provided by RECOVERY MANAGER to automatically create ARMBGIM JCL. This approach requires only a short TSO session. For more information, see “Using plan and package impact analysis and reporting” on page 139.

Authorizations

The following authorizations are required to execute the ARMBGIM program:

- APF authorization for the RECOVERY MANAGER load library
- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- TYPE O (OPEN) authority for the group (or SYSADM or system DBADM authority)
Building the ARMBGIM JCL

Building your own ARMBGIM job to perform plan and package impact analysis involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition (DD) statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Input data sets
  - Output data sets

The descriptions in the following subsections provide more details.

Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends you specify REGION=0M, which makes the amount of virtual storage that is needed to run the job automatically available when the ARMBGIM job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

Specifying the EXEC statement

The EXEC statement has the following format:

```
//stepname EXEC PGM=ARMBGIM, REGION=0M,
// PARM='ssid,ARMOPTS=optionSet'.
```

The variable ssid is the DB2 subsystem or data sharing group attach name where the RECOVERY MANAGER group resides. If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.
Note
The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

The variable optionSet is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBGIM to use. For example:

```plaintext
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//         DD DISP=SHR,DSN=DSNEXIT
//         DD DISP=SHR,DSN=DSNLOAD
```

Specifying the ARMBGIM data set DD statements

This topic describes the data sets that ARMBGIM uses.

Each data set is specified by a `ddname` (data definition name). You must specify all of the required data sets in the JCL.

- *(required)* ARMIN
  The input data set that contains one or more control statements. Attributes for this data set must be fixed length records, with a record length of 80 (RECFM=F or FB, LRECL=80).

- *(required)* ARMPRINT
  The output data set for messages that are returned from RECOVERY MANAGER. RECOVERY MANAGER also echoes the contents of the ARMIN data set in the ARMPRINT output. ARMPRINT may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.
- *(optional)* ARMOPTS
  The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM SOPTS is used as the default option set name.
  You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

  ```
  //ARMOPTS DD *
  ssid.configurationOption=value
  ```

- *(required)* ARMMSGS
  The RECOVERY MANAGER messages data set, created during RECOVERY MANAGER installation with the default name of `hilvl/RMGR.ARMCNTL(ARMMSGS)`. The data set must be allocated with DISP=SHR.

- *(optional)* ARMERROR
  This statement specifies the output data set for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

---

**ARMBGIM syntax and option descriptions**

The ARMBGIM syntax and option descriptions in this section are the control statements that you use when you build the ARMIN input.

For information about generating ARMBGIM JCL from the RECOVERY MANAGER online interface, see “Using plan and package impact analysis and reporting” on page 139.

**Note**

See “Syntax rules” on page 102 for more information on syntax rules and wildcard support.

ARMBGIM control statement syntax for SET CURRENT SQLID is shown in Figure 68 on page 468. Syntax descriptions can be found on the pages that are shown in parentheses.

**Figure 68: ARMBGIM control statement—SET CURRENT SQLID**

```
SET CURRENT SQLID = sqlid ;
```
ARMBGIM control statement syntax for REPORT GROUP is shown in Figure 69 on page 469. Syntax descriptions can be found on the pages that are shown in parentheses.

**Figure 69: ARMBGIM control statement—REPORT GROUP**

```
REPORT GROUP creator.name

  PLANS
    YES
    NO

  PACKAGES
    YES
    NO

```

**SET CURRENT SQLID**

In the ARMIN input data set, optionally provide the control statement SET CURRENT SQLID = sqlid to set the SQLID to be used in the reporting and revalidation.

The SQLID defaults to your user ID.

You can have multiple SET CURRENT statements in the control data set. The SET CURRENT SQLID statement is in effect for all statements that follow it until another SET CURRENT SQLID statement is issued.

**REPORT GROUP**

This control statement is required.

You must provide a group name in the form `creator.name`. You can use a wildcard pattern to specify multiple groups. You can repeat the REPORT GROUP statement for as many groups as you want to process.

**PLANS**

Determines whether to produce impact analysis reports for plans that are affected by the group (PLANS YES) or not (PLANS NO). The default is YES.

**PACKAGES**

Determines whether to produce impact analysis reports for packages that are affected by the group (PACKAGES YES) or not (PACKAGES NO). The default is YES.
Sample JCL

The following figure shows a sample of JCL for ARMBGIM.

Figure 70: Sample ARMBGIM JCL

```plaintext
//* *************************************************************** */
//*          RECOVERY MANAGER             BMC SOFTWARE, INC.        */
//* *************************************************************** */
//* *************************************************************** */
//ARM0000 EXEC PGM=ARMBGIM,  
//             PARM="DEC2,ARMOPTS=ARM$OPTS",  
//             REGION=4M  
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS   
//          DD DISP=SHR,DSN=DSNEXIT     
//          DD DISP=SHR,DSN=DSNLOAD   
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)  
//ARMPRINT DD SYSOUT=**  
//ARMEERROR DD SYSOUT=**  
//ARMIN DD *  
REPORT GROUP "ARMQA"."ARMBGIM"  
  PLANS YES  
  PACKAGES YES  
:  
/*
```

Sample output

The following figure shows a sample of output for ARMBGIM.

Figure 71: Sample ARMBGIM output

```plaintext
** RECOVERY MANAGER FOR DB2 V11.1.00 - GROUP IMPACT REPORT 08/14/2012 09:18:18 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.1.00

(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884

BMC80223I MAINT:  NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT:  BPJ0197 BPJ0215 BPJ0219
BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910

REPORT GROUP "ARMQA"."ARMBGIM"
  PLANS YES
  PACKAGES YES
: 
PLAN NAMES
-----------
ACAM420I
ACAM510I
ACAM510T
ACAM520I
ACAM520T
```
Executing the JCL

This section describes special instructions or information required to run the ARMBGIM JCL.

- Ensure that the SQLID used has appropriate authority for the groups. For required authorizations, see “Authorizations” on page 465.

- ARMBGIM can be executed before or during a recovery to identify the impact of the outage.

- No restart is available for ARMBGIM. You must resubmit the job after correcting any error conditions.
ARMBGPS—Subsystem group split

This chapter describes the implementation of ARMBGPS—Subsystem group split.

About ARMBGPS

Use the ARMBGPS program to create a set of groups to be used for backup and recovery.

ARMBGPS automatically divides the objects identified by the group definition into multiple balanced groups. These groups can then be treated as independent units for backup and recovery purposes.

---

**WARNING**

ARMBGPS is designed to drive the backup process and is not intended to be run at recovery time.

---

The patterns that you use for group definition determines the groups created.

---

**WARNING**

You must make a full image copy of the full subsystem directly after running ARMBGPS. Only then can you can make full or incremental copies of all or part of the subsystem and make use of the XUNCHANGED option of ARMBGEN. For more information about XUNCHANGED, see “ARMBGEN—Backup and recovery JCL” on page 431. How frequently you run ARMBGPS depends on the volatility of the DB2 system’s objects. If table spaces are created frequently or significantly change size frequently, run ARMBGPS each time before you make full copies of the entire subsystem. Never run ARMBGPS more frequently than the full copy cycle. The groups created by ARMBGPS must match the grouping of the full copies of the full system.

With each new release of RECOVERY MANAGER, you must rebuild your ARMBGPS groups for subsystem recovery.

---

If you use INCLUDE patterns for group definition, ARMBGPS builds groups for a subset of the DB2 subsystem. These groups are called ARMBGPS application groups.
Creating ARMBGPS application groups is a new way to create groups in RECOVERY MANAGER versions 9.2.00 and later and simplifies the method for generating multiple jobs for an application. (Recovery Groups is an entry on the Object Selection panel (ARMUS001) to generate ARMBGPS JCL for these application groups.)

ARMBGPS returns LOB, XML and History objects with their related base table space, not by using their explicit name.

You can create ARMBGPS JCL in the following ways:

- Manually code JCL to run ARMBGPS. This method completely avoids using a TSO session.
- Use online support that is provided by RECOVERY MANAGER to create ARMBGPS JCL. For more information, see “Build subsystem groups and generate backup JCL” on page 351.

After you have created your groups with ARMBGPS, you can then use ARMBGEN to generate one backup job and one recover job per group. For more information, see “ARMBGEN—Backup and recovery JCL” on page 431. You can also use the online interface to generate backup and recover JCL.

**Delta groups**

ARMBGPS builds the 00 delta group with an exclude for each group in the set of groups at the same time that it creates all of the groups.

These groups are handled by ARMBGEN so that each group is recovered in a separate job.

The definition is dynamic so that the delta group will contain any object that is not part of the other BGPS groups.

The delta group is always created as the number 00 group using the ARMBGPS prefix. After initial creation, the delta group is maintained by dynamic group processing. You can then use ARMBGEN to generate recover JCL for those table spaces and their indexes.

**How ARMBGPS builds multiple groups**

ARMBGPS identifies all objects belonging to a particular DB2 subsystem and builds up to \( nn \) RECOVERY MANAGER groups (where \( nn \) is a user-specified maximum number of groups set with the MAXGROUPS option) plus a delta or 00 group.
The size of these groups is based on a sizing split determined in the NACTIVE column of SYSIBM.SYSTABLESPACE. If NACTIVE = -1, the sizing defaults to 10 pages.

By default, RECOVERY MANAGER generates a single job stream (consisting of one job per group) to create the groups. You can optionally specify storing each separate job into a separate member of a partitioned data set. Doing so enables you to control when the jobs are submitted. For more information, see “Separating jobs from a multi-job batch job stream” on page 77.

Each group will be recovered in a single job. No further job-splitting will occur.

**Note**

You should use either RUNSTATS or NGT Copy with the RUNSTATS option to obtain an accurate estimate of number of pages required for sizing. Otherwise, you can manually update the NACTIVE value to more accurately reflect the status of your system.

---

**Group names and defaults**

ARMBGPS builds groups by using the creator and group name prefix that you supply in the GROUPS keyword and then attaching a suffix of 00 through \( nn \), where \( nn \) is the maximum number of groups (MAXGROUPS).

The groups are then saved in the repository.

For example, if you use the creator and group name of CCB.ALLTS, ARMBGPS will build CCB.ALLTS00, CCB.ALLTS01, CCB.ALLTS02, CCB.ALLTS03, and so on.

If the group name already exists in the repository, RECOVERY MANAGER deletes all objects from the existing group and then adds the newly identified set of objects. RECOVERY MANAGER keeps the existing set of group utility options, but deletes and re-adds the group definition in case the EXCLUDE list has changed. All groups with the suffix 01 through \( nn \) are static groups with the list of all objects generated at create time. Group 00 will have the INCL and EXCL patterns as well as an EXCL for each additional group 01 through \( nn \).

**Note**

ARMBGPS deletes any groups that have a numeric suffix greater than the number specified by MAXGROUPS. It interprets these groups as having been left over from a prior run which used a higher value for MAXGROUPS.
Exclusions

ARMBGPS automatically excludes the DB2 catalog (DSNDB06), the directory (DSNDB01), the work file databases, and the databases declared AS TEMP at creation time.

You can also exclude table spaces from the set of groups by using the EXCLUDE statement and indicating one or more table space names or patterns (subsystem groups, not application groups).

EXCLUDE_GROUP is also available to exclude groups using one or more group names or patterns.

ARMBGPS excludes LOB, XML, and History objects with their related base table space, not by using the explicit name of the LOB, XML, or History object. See the following sections for more information.

Note

BMC recommends that you exclude the repository and back it up separately from the full subsystem. If you generate the ARMBGPS via the online interface, RECOVERY MANAGER automatically generates the necessary EXCLUDE statements.

Inclusions

ARMBGPS provides the INCLUDE and INCLUDE_GROUP syntax that you can use to optionally specify the pattern that you want to use to generate the list of objects or groups in your group. The default pattern is *.*.

ARMBGPS returns LOB, XML, and History objects with their related base table space, not by using the explicit name of the LOB, XML, or History object. See the following sections for more information.

ARMBGPS does not support indexes so any INCLUDE GROUP must be defined by table space only with no indexes included.

LOB objects

ARMBGPS automatically identifies LOB-related spaces and keeps them together in the same group (regardless of size) to ensure that they are recovered together. LOB-related spaces must be recovered together so that they are not placed in pending status.
**XML objects**

ARMBGPS automatically identifies XML-related spaces and keeps them together in the same group to ensure that they are recovered together. XML-related spaces must be recovered together so that they are not placed in pending status.

**Temporal or History tables**

For DB2 Version 10 and later, when you are creating groups with ARMBGPS, the history table space is included in the same group with its related parent or base table.

**Archive-enabled tables**

For DB2 Version 11 and later, when you are creating groups with ARMBGPS, the archive table space is included in the same group with its related parent or base table.

**Indexes**

Indexes are not included in groups that are built by ARMBGPS and are either rebuilt or recovered with the associated group, as follows:

- If you use the IBM DSNUTILB COPY and RECOVER utilities, indexes are always rebuilt. ARMBGPS creates new groups with a default group profile that includes the INDEX ALL, REBUILD INDEXES, and MAXJOBS=1 options. (You can set other options as required using the online interface.) These options are used when you generate backup and recovery JCL for the groups.

- If you use the BMC NGT Copy and NGT Recover utilities, you can choose whether to rebuild or recover the indexes. ARMBGPS can optionally create new groups with a default group profile that includes the INDEX ALL, RECOVER INDEXES, and MAXJOBS=1 options. These options are used when you generate backup and recovery JCL for the groups. See “Setting options for index recovery” on page 477 for more information about backing up and recovering indexes for ARMBGPS groups.

**Setting options for index recovery**

If you use the BMC NGT Copy and NGT Recover utilities, you can optionally set options to back up and recover indexes for groups that are created by ARMBGPS by specifying options in the ARMBGPS syntax.
ARMBGPS uses the utilities that you specify in the subsystem-level defaults. For more information about setting subsystem-level options, see “Setting subsystem-level options” on page 98.

Use the following ARMBGPS syntax to back up and recover indexes:

```plaintext
COPY_OPTIONS
 COPY_IX auto|yes
 IX_SIZE nnnnnnnnnn IX_SIZE_TYPE x
```

For more information see the ARMBGPS syntax diagram in “ARMBGPS syntax and option descriptions” on page 481 or the option definitions in “BUILD GROUPS” on page 483.

**Index backup for ARMBGPS groups**

When you specify the COPY_OPTIONS syntax for ARMBGPS, RECOVERY MANAGER backs up indexes that meet the specified size threshold (if specified). The backup JCL for the groups is generated with the following options:

- COPY INDEXSPACES AUTO
- COPY INDEXES ALL
- IXSIZE nnnnnnnnnn
- IXSIZET x

For more information about these copy options, see the *BMC Next Generation Technology Copy for DB2 for z/OS Reference Manual*.

**Index recovery for ARMBGPS groups**

When you specify the COPY_OPTIONS syntax for ARMBGPS, RECOVERY MANAGER recovers the indexes when possible or rebuilds the indexes if they cannot be recovered. The recovery JCL for the groups is generated with the following options:

- INDEXLOG AUTO
- RECOVER INDEXES ALL

For more information about these recovery options, see the *BMC Next Generation Technology Recover for DB2 for z/OS Reference Manual*. 
RECOVERY MANAGER does not include indexes when revalidating recovery resources. Because many users copy only their largest indexes and because the indexes will be automatically rebuilt if no copies exist, it is not necessary to include missing index copies in an exception report.

Revalidating ARMBGPS groups

You can use the ARMBGPV program to verify the recoverability of the ARMBGPS groups.

Authorizations

The following authorizations are required to execute the ARMBGPS program:

- APF authorization for the ARMBGPS program and the RECOVERY MANAGER load library
- EXECUTE authority for the RECOVERY MANAGER DB2 plans
- TYPE A (ALL) authority on the groups (or SYSADM or system DBADM authority)

Building the JCL

Building your own ARMBGPS job to perform multiple group creation involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Input data sets
  - Output data sets

The descriptions in the following subsections provide more details.
Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends that you specify REGION=0M, which makes the amount of virtual storage that is needed to run the job automatically available when the ARMBGPS job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

Specifying the EXEC statement

The EXEC statement has the following format:

```
//stepname EXEC PGM=ARMBGPS,
  //             PARM='
ssid,ARMOPTS=optionSet',
  //             REGION=0M
```

The variable ssid is the DB2 subsystem or group attach name where the RECOVERY MANAGER groups reside. If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

**Note**

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

The variable `optionSet` is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBGPS to use. For example:

```
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
  // DD DISP=SHR,DSN=DSNEXIT
  // DD DISP=SHR,DSN=DSNLOAD
```
Specifying the ARMBGPS data set DD statements

This subsection describes the data sets that ARMBGPS uses.

Each data set is specified by a *ddname* (data definition name). You must specify all required data sets in the JCL.

- **(required) ARMIN**
  The input data set that contains one or more control statements. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

- **(required) ARMPRINT**
  The output for messages that are returned from RECOVERY MANAGER. RECOVERY MANAGER also echoes the contents of the ARMIN data set in the ARMPRINT output. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

- **(optional) ARMOPTS**
  The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

  You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

  ```
  //ARMOPTS  DD *
  ssid.configurationOption=value
  /*
  ```

- **(required) ARMMSGS**
  The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of `hilv/RMGR.ARMCNTL(ARMMSGS)`. The data set must be allocated with DISP=SHR.

- **(optional) ARMERROR**
  The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

ARMBGPS syntax and option descriptions

The ARMBGPS syntax and option descriptions in this section are the control statements that you use when you build ARMIN input.
Note
See “Syntax rules” on page 102 for more information on syntax rules and wildcard support.

Figure 72: ARMBGPS control statement—SET CURRENT SQLID

```
SET CURRENT SQLID = sqlid ;
```

Figure 73: ARMBGPS control statement—BUILD GROUPS

```
BUILD GROUPS creator.prefixName
  INCLUDE databaseName.tableSpaceName
  INCLUDE_GROUP groupName
  EXCLUDE databaseName.tableSpaceName
  EXCLUDE_GROUP groupName
  MAXGROUPS nn
  DESCRIPTION "text"

COPY_OPTIONS
  COPY_IX NO
    IX_SIZE nnnnnnnnn
    IX_SIZE_TYPE K
    M
    G
  YES
  AUTO
```

* Note: Group name prefixes cannot be delimited.
SET CURRENT SQLID

In the ARMIN input data set, optionally provide the control statement SET CURRENT SQLID =sqlid to set the SQLID to be used in the group creation.

The SQLID defaults to your user ID.

You can have multiple SET CURRENT statements in the control data set. The SET CURRENT SQLID statement is in effect for all statements that follow it until another SET CURRENT SQLID statement is issued.

BUILD GROUPS

This control statement is required when building a set of groups based on a very large number of table spaces.

The creator.prefixName that you specify becomes the prefix of each group name, to which RECOVERY MANAGER adds a numeric suffix in the range of 00 to nn, where nn is the value that you specify with the MAXGROUPS option.

INCLUDE

Use this optional control statement to include one or more table spaces in the set of groups. You may enter one or more individual table spaces with wildcard patterns.

If you do not use INCLUDE, RECOVERY MANAGER assumes that all table spaces should be included and uses the *.* pattern.

ARMBGPS returns LOB, XML and History objects with their related base table space, not by using their explicit name.

ARMBGPS does not support indexes so any INCLUDE GROUP must be defined by table space only with no indexes included.

EXCLUDE

Use this optional control statement to exclude one or more table spaces from inclusion in the set of groups. You may enter one or more individual table space names or wildcard patterns.

If you exclude table spaces from the full subsystem groups, you should create a separate group for backup and recovery of these objects, because the full subsystem recovery will not include them.
ARMBGPS returns LOB, XML and History objects with their related base table space, not by using their explicit name.

**INCLUDE_GROUP**

Use this optional control statement to include one or more groups in the set of groups. You may enter one or more individual groups with wildcard patterns.

**EXCLUDE_GROUP**

Use this optional control statement to exclude one or more groups from inclusion in the set of groups. You may enter one or more individual group names or wildcard patterns.

If you exclude groups from the full subsystem groups, you should create a separate group for backup and recovery of these objects, because the full subsystem recovery will not include them.

**MAXGROUPS**

Use this optional statement to set the maximum number of groups that RECOVERY MANAGER will create. Valid values are 2 through 99. The default is 10.

RECOVERY MANAGER creates the maximum number of groups that you specify plus an additional one, the delta group (00 group), which is always created. The delta group automatically picks up any new objects created since the BGPS groups were generated.

**DESCRIPTION**

The description of the groups is optional. Text can be up to 25 characters and must be enclosed in single quotes (’’) or double quotes (" ").

**COPY_OPTIONS**

Use this optional control statement to include copied indexes in the groups.

**COPY_IX**

Specify whether to set group options that will include index spaces in the backup and recovery JCL that is generated for the ARMBGPS groups. NO is the default. YES specifies to include all indexes. AUTO specifies including indexes as large or larger than the size specified by the IX_SIZE option.

This option is available only if

- NGT Recover is specified as the recover utility
- NGT Copy is specified as the copy utility
**IX_SIZE**

Specify the size threshold at which you want indexes backed up rather than rebuilt. This option is used in conjunction with COPY_ix AUTO. You can enter the size as follows:

- 0-4194303 if using M as the IX_SIZE_TYPE
- 0-4294967295 if using K as the IX_SIZE_TYPE
- 0-4095 if using G as the IX_SIZE_TYPE

**IX_SIZE_TYPE**

Specify the unit of measure for the threshold size, as follows:

- K - Kilobytes
- M - Megabytes
- G - Gigabytes

### Sample JCL

The following figure shows sample JCL for ARMBGPS.

**Figure 74: Sample ARMBGPS JCL**

```plaintext
//*************************************************************** // RECOVERY MANAGER - V11.1.00 - BMC SOFTWARE, INC. //*************************************************************** /*// BATCH GROUP CREATION */ //*************************************************************** //*************************************************************** /*EXEC PGM=ARMBGPS, PARM='DEC2,ARMOPTS=ARM$OPTS', REGION=4M */ //STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS // DD DISP=SHR,DSN=DSNEXIT // DD DISP=SHR,DSN=DSNLOAD //ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS) //ARMPRINT DD SYSOUT=** //ARMERROR DD SYSOUT=** //ARMIN DD * BUILD GROUPS RDAJBM.DEC2SAMP MAXGROUPS 10 EXCLUDE BMCACA32.ACAREPOS , BMCARM.BMCARMCR , BMCARM.BMCARMA , BMCARM.BMCARMG , BMCARM.BMCARMG , BMCARM.BMCARMGP , BMCARM.BMCARMGS , BMCARM.BMCARMOP , BMCARM.BMCARMSF , BMCALP73.ALPURSP , BMCALP73.ALPWHSP 
```
The following figure shows sample output for ARMBGPS.

**Figure 75: Sample ARMBGPS output**

** RECOVERY MANAGER FOR DB2 V11.1.00 - BATCH GROUP SPLIT 08/14/2012 12:53:00 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.1.00 **
(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884

BMC80223I MAINT:  NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT:  BPJ0197  BPJ0215  BPJ0219

BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910
SET CURRENT SOLID = ARMQA;
BMC80570I COMMAND COMPLETE RC = 0
BUILD GROUPS RDAJBM.DEC2SAMP
     MAXGROUPS 10
EXCLUDE
     DESCRIPTION 'DEC2 SAMPLE GROUPS' ;
BMC80868I 1898 SPACES DO NOT HAVE DB2 CATALOG STATISTICS. TOTAL SPACES = 2809
BMC80570I COMMAND COMPLETE RC = 0
BMC80571I PROGRAM COMPLETE RC = 0
1

** RECOVERY MANAGER FOR DB2 V11.1.00 - BATCH GROUP SPLIT 08/16/2012 13:09:09 **
(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676

BMC80223I MAINT:  NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT:  BPJ0197  BPJ0215  BPJ0219

BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910
BUILD GROUPS RDAJBM.DEC2SAMP
     MAXGROUPS 10
EXCLUDE
     BMCACA32.ACAREPOS
     , BMCARM.BMCARMCR

Executing the JCL

This section describes special instructions or information required to run the ARMBGPS JCL.

- Ensure that the job owner has the appropriate authorizations. See “Authorizations” on page 479.

- No restart is available for ARMBGPS. You must resubmit the job after correcting any error conditions.
ARMBGPV—Group recovery revalidation

This chapter describes ARMBGPV—Group recovery revalidation.

About ARMBGPV

If a group contains mirrored objects, ARMBGPV verifies that the mirrors are valid and reports exceptions if you are using the BMC Recovery Management for DB2 solution. It can also verify mirroring for system resources.

Note

- ARMBGPV revalidation is used for all the recovery resource reports to ensure the recoverability of groups.
- If you have a BMC Recovery for DB2 solution password, RECOVERY MANAGER writes all recoverability exceptions to the new RECOVERY_EXCEPTIONS table for later reporting to BMC Workbench for DB2. For more information, see “Job history: BMCARM.BMCRMD_RECOVERY_EXCEPTIONS table” on page 807.

In addition, you can request detailed printed reports, as follows:

- Object recoverability
- Recovery resources required by the group
- Tape volumes needed for recovery (pick list)
- Archived data sets required for recovery

You can also:

- Specify whether the reports are for the local site or a recovery site
- Specify the recovery point for analysis

You can generate printed batch group reports by using the ARMBGPV program in the following ways:
- Code JCL to run ARMBGPV. This method completely avoids using a TSO session.

- Use online support provided by RMGR to automatically create ARMBGPV JCL. This method requires only a short TSO session. For more information, see “Group recovery revalidation and reporting” on page 133.

## Using ARMBGPV in disaster recovery planning

You can use ARMBGPV to provide more automation for the recovery of applications in a disaster recovery situation. At the local site, the system resource recovery program, ARMBSRR, updates the archive history file with the end RBA of the last archived log. When ARMBSRR is completed, you can audit your disaster recovery group by using ARMBGPV.

## Using ARMBGPV for mirror revalidation

If you are using the BMC Recovery Management for DB2 solution, you can use the ARMBGPV program to revalidate the mirror status of the subsystem objects and application groups. ARMBGPV works with SUF to verify whether objects are successfully mirrored and report exceptions.

### Revalidating mirroring for system objects

The REVALIDATE MIRROR SYSTEM command verifies and reports mirroring exceptions for the following system objects:
- BSDS
- Active logs
- Catalog and directory
- The repository (BMC, RECOVERY MANAGER, and CHANGE ACCUM)
- Log Master repository

The exceptions are written to the ARMXCEPT file.

To use this command, you must have:

- Applied the BMC Recovery Management for DB2 solution password
- Established mirroring level 2 or higher at the subsystem level

For information about setting subsystem-level options, see “Setting subsystem-level options” on page 98.
Revalidating mirroring for groups

The REVALIDATE GROUP command verifies and reports mirroring exceptions for one or more user-specified groups in addition to revalidating group objects.

Informational messages regarding data sets that could not be verified as being mirrored are written to the ARMPRINT file.

To use this command, you must have:

- Applied the BMC Recovery Management for DB2 solution password
- Established mirroring level 2 or higher at the subsystem level
- Specified JCLTYPE DR
- Specified TORESTARTRBA as the revalidation point
- Set the group-level mirror strategy in the group general recover options

For information about setting subsystem-level options, see “Setting subsystem-level options” on page 98.

Authorizations

The following authorizations are required to execute the ARMBGPV program:

- APF authorization for the RECOVERY MANAGER load library
- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- EXECUTE authority to run report recovery on DSNDB06.
- TYPE A (ALL) authority for the group if you intend to save changes (or SYSADM or system DBADM authority)
- TYPE O (OPEN) authority for the group if you do not intend to save changes
- DISPLAY DATABASE authority to list all objects in exception status when you attempt to revalidate a group of such objects

Building the ARMBGPV JCL

Building your own ARMBGPV job involves creating JCL that includes the following statements:
A JOB statement

An EXEC statement

Data definition statements that specify the use of the following libraries and data sets:
- RECOVERY MANAGER and DB2 load libraries
- Input data sets
- Output data sets

The descriptions in the following subsections provide more details.

**Specifying the JOB statement**

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends you specify REGION=0M, which makes the amount of virtual storage needed to run the job automatically available when the ARMBGPV job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

**Specifying the EXEC statement**

The EXEC statement has the following format:

```bash
//stepname EXEC PGM=ARMBGPV,
//             PARM='ssid,ARMOPTS=optionSet',
//             REGION=0M
```

The variable ssid is the DB2 subsystem ID or data sharing group attach name where the RECOVERY MANAGER groups reside. If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

**Note**

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.
The variable `optionSet` is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

**Specifying the STEPLIB DD statement**

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load and exit libraries that you want ARMBGPV to use. For example:

```
//STEPLIB   DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
```

**Specifying the ARMBGPV data set DD statements**

This section describes the data sets ARMBGPV uses.

Each data set is specified by a `ddname` (data definition name).

- **(required) ARMIN**
  
  The input data set containing one or more control statements. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

- **(required) ARMPRINT**
  
  The output for messages returned from RECOVERY MANAGER. RECOVERY MANAGER also echoes the contents of the ARMIN data set in the ARMPRINT output. ARMPRINT may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

- **(optional) ARMOPTS**
  
  The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

  You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

  ```
  //ARMOPTS  DD *
  ssid.configurationOption=value
  /*
  ```
- **(required) ARMMSGS**
  The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of `hilvl.RMGR.ARMCNTL(ARMMSGS)`. The data set must be allocated with `DISP=SHR`.

- **(optional) ARMERROR**
  The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a data control block (DCB) of `LRECL=121, RECFM=VB`.

- **(required) ARMXCEPT**
  The output for the ARMBGPV recoverability report that is requested by the `RECOVERABILITY YES` option. This data set may be allocated to SYSOUT or to a data set with a DCB of `LRECL=121, RECFM=VB`.

- **(required) ARMRESRC**
  The output for the ARMBGPV resources report requested by the `RESOURCES YES` option. This data set may be allocated to SYSOUT or to a data set with a DCB of `LRECL=121, RECFM=VB`.

- **(required) ARMRCALL**
  The output for the ARMBGPV recall report requested by the `RECALL YES` option. This data set may be allocated to SYSOUT or to a data set with a DCB of `LRECL=121, RECFM=VB`.

- **(optional) ARMPICK**
  The output for the ARMBGPV pick list report requested by the `PICKLIST YES` option. This data set may be allocated to SYSOUT or to a data set with a DCB of `LRECL=121, RECFM=VB`.

- **ARMRSTOR**
  ARMBGPV writes SAMS RESTORE commands to the ARMRSTOR DD. You can use the SAMS RESTORE commands as input to restore migrated data sets. You do not set an option to have RECOVERY MANAGER create the ARMRSTOR DD.

---

**ARMBGPV syntax and option descriptions**

The ARMBGPV syntax and option descriptions in this topic are the control statements that you use when you build ARMIN input.

For information about running ARMBGPV from the RECOVERY MANAGER online interface, see “Revalidating and reporting on groups in batch” on page 134.
See “Syntax rules” on page 102 for more information on syntax rules and wildcard support.

ARMBGPV control statement syntax for SET CURRENT SQLID is shown in Figure 76 on page 495. Syntax descriptions can be found on the pages that are shown in parentheses.

**Figure 76: ARMBGPV control statement—SET CURRENT SQLID**

```plaintext
SET CURRENT SQLID = sqlid ;
```
ARMBGPV options statement syntax is shown in Figure 77 on page 496. Syntax descriptions can be found on the pages that are shown in parentheses.

Figure 77: ARMBGPV options statement

---

$^{*}$ The BACKOUT option default is NO when using RMGR alone and for DSNUTILS and AUTO when using RMGR as part of the Recovery Management for DB2 solution. AUTO is not valid with DSNUTILS.
ARMBGPV control statement syntax is shown in Figure 78 on page 497.

**Figure 78: ARMBGPV control statement**

---

**SET CURRENT SQLID**

In the ARMIN input data set, optionally provide the control statement `SET CURRENT SQLID = sqlId` to set the SQLID to be used in the reporting and revalidation.

The SQLID defaults to your user ID.

You can have multiple `SET CURRENT` statements in the control data set. The `SET CURRENT SQLID` statement is in effect for all statements that follow it until another `SET CURRENT SQLID` statement is issued.

**SET OPTIONS**

In the ARMIN input data set, provide the optional `SET OPTIONS` control statement.
If you do not code a SET OPTIONS control card, you will, by default, receive all possible reports.

**CLONES ONLY**

Specify whether to include only clones in the revalidation processing. This option is available only when running on DB2 Version 10 or later and is not valid with compatibility mode.

**RECOVERABILITY**

Validates the recoverability of each object in the group to the selected recovery point. This validation enables you to verify recoverability by identifying recoverability exceptions (for example, data sets for which no copies were found). The report is written to the data set that is defined by the ARMXCEPT DD statement. The default is YES.

**RESOURCES**

Reports the resources needed for the recovery of a group. These resources include image copy data sets, log data sets, and R+/CHANGE ACCUM files. Use this report as a checklist of data sets that must be taken offsite in preparation for disaster recovery.

If the current subsystem is a member of a data sharing group, the resources report is for the group. Otherwise, the report is for the current subsystem. The report is written to the data set that is defined by the ARMRESRC DD statement. The default is YES.

**RECALL**

Produces a report of the archived data sets that are needed for recovery. The report is written to the data set that is defined by the ARMRCALL DD statement. The default is YES.

**UNRECOVER_RC**

Specify a return code that ARMBGPV should issue if it encounters an unrecoverable object during revalidation. The UNRECOVER_RC option is ignored if you choose the RECOVERABILITY NO option.

Valid values are 0 (the default) through 254. A value of zero causes no return code to be issued and values (even those higher than 4) do not affect the step execution. This return code does not control the overall job return code. If RECOVERY MANAGER encounters errors other than unrecoverable data, it sets the return code based on the most severe condition code found.

Unrecoverable objects are listed in the exception report (ARMXCEPT) if you request it using the RECOVERABILITY YES option, whether or not you specify UNRECOVER_RC.
An object is considered to be unrecoverable by ARMBGPV if it exists in any of the following states or if it has missing recovery resources:

- ALTER IX
- BAD TYPE
- DS LEVEL
- LOG GONE
- NOCOPIES
- NOTAVAIL
- NOTCTLG
- TSREORP
- TS STAT

**Note**
The following statuses are unique to ARMBGPV:

- LOG GONE — indicates that one or more of the logs that are required for recovery are missing
- NOTCTLG — one or more recovery resources are uncataloged

For more information about the object exception status codes, see “RECOVERY MANAGER object exception status” on page 839.

**PICKLIST**

Produces a report of the tape volumes that are needed for recovery. This report enables tape operators to locate the required tapes prior to a recovery and is written to the data set defined by the ARMPICK DD statement. The default is YES.

**SITETYPE**

Determines whether the selected reports are generated by using the local site or recovery site resources. The default is LOCAL.

If you specify DB2 Recover (DSNUTILB) as the recovery utility for the group and if you specify the DSNUTILB site type in the DSNUTILB-specific options, then the SITETYPE option is ignored. Also, if you use NGT Recover alternate resources and you select the RP or RB copy as your first choice for the image copy in the recovery options, RECOVERY MANAGER considers the site type to be RECOVERY regardless of what you specify here.

**LOGCOPY**

Specifies which copy of the archive log (1, 2, or 3) to report or verify. If SITETYPE LOCAL is used, the value is always set to 1. If SITETYPE RECOVERY is used, the default is 3.
BACKOUT

Validates the availability of resources required for a BACKOUT AUTO, BACKOUT NO, or BACKOUT YES recovery. This option is available when using NGT Recover or DB2 RECOVER (DSNUTILB) as the recovery utility.

BACKOUT AUTO is only available when you are using the Recovery Management solution password. The default is AUTO if you are using RECOVERY MANAGER as part of the Recovery Management solution. Otherwise, the default is NO.

You must specify QUIESCE or SPECIFIC LOGPOINT as the recovery point to use BACKOUT AUTO or BACKOUT YES. If you specify TOCURRENT, RECOVERY MANAGER overrides the BACKOUT AUTO and converts the option to BACKOUT NO.

JCLTYPE

Specify whether the JCL is for use at the remote or local site. This option is used for mirroring support only. The default is the value specified in SITETYPE. For information about mirror revalidation, see “Revalidating mirroring for groups” on page 491.

RESOLVE_INFLIGHTS

Specify whether inflight units of work are to be resolved during a recovery to a user-specified RBA or LRSN. This option is only valid for a recovery using TOLOGPOINT and is available only with the Recovery Management for DB2 solution. For more information, see the timestamp recovery chapter in the Recovery Management for DB2 User Guide.

REVALIDATE GROUP

Specify one or more groups to be revalidated in the form creator.name.

You can use a wildcard pattern to specify multiple groups. You can repeat the REVALIDATE GROUP statement for as many groups as you want to process. If you are using the BMC Recovery Management for DB2 solution, this command can revalidate the mirroring status of the groups. For more information, see “Revalidating mirroring for groups” on page 491.

Choose one of the following recovery points for validation.

TOCURRENT

This revalidation point is the default. When you specify a revalidation to the current time, the most recent full and incremental copies are used to
revalidate the spaces, in addition to the DB2 log records that are used to make the spaces current. The RECOVER syntax defaults to TOCURRENT.

**TOCOPY**

Specify this value to revalidate to a specific full image copy or an incremental image copy registered in SYSIBM.SYSCOPY. You identify the copy by specifying a number from 0 through 99 where 0 represents the most recent copy made and 99 represents the 100th previous copy.

For the purpose of identifying the copy, a DFSMS concurrent copy registered in SYSIBM.SYSCOPY is counted as an image copy.

**TOFULLCOPY**

Specify this value to revalidate only to a specific full image copy registered in SYSIBM.SYSCOPY. You identify the copy by specifying a number from 0 through 99 where 0 represents the most recent copy made and 99 represents the 100th previous copy. RECOVERY MANAGER ignores any existing incremental image copies when searching for the specified previous copy.

For the purpose of identifying the copy, a DFSMS concurrent copy registered in SYSIBM.SYSCOPY is counted as an image copy.

**TOQUIESCE**

Specify this value when you want to revalidate each object to a specified quiesce point registered in SYSIBM.SYSCOPY by the DB2 QUIESCE command. Similarly, to revalidate to an image copy, you identify the quiesce point by specifying a number from 0 through 99, where 0 is the most recent quiesce point.

**TOLOGPOINT**

Specify this value when you want to revalidate to a log point (RBA or LRSN). You must supply the value (a 12-digit hexadecimal number).

**TORESTARTRBA**

Specify this type of recovery when making preparations at the local site to recover applications at a recovery site. When you specify TORESTARTRBA, ARMBGEN retrieves the recovery point RBA stored by the ARMBSRR program and performs the same analysis as a recovery to an RBA. Because the actual recovery is performed after a conditional restart, the generated JCL is for a recover to the current point in time. This option is for use at the local site only; it should not be used at a recovery site as part of a disaster recovery.
TOTIMESTAMP

Valid with the BMC Recovery Management for DB2 solution only. Revalidates recovery to a specified timestamp. You must specify the timestamp in the format yyyy-mm-dd-hh.mm.ss.tttttt. For more information about timestamp recovery, see the Recovery Management for DB2 User Guide.

TOLOGMARK

Specify this value to revalidate a Log Master for DB2 log mark, which is registered in the Log Master ALPMARK table. You identify the log mark by specifying the log mark name and a version number from 0 through 99, where 0 represents the most recent log mark made and 99 represents the 100th previous log mark.

This option requires use of a BMC Recovery Management for DB2 solution password.

REVALIDATE MIRROR SYSTEM

Specify this command to revalidate mirrored system resources. Exceptions are written to the ARMXCEPT file. For more information, see “Revalidating mirroring for system objects” on page 490.

Sample JCL

The following figure shows a sample of JCL for ARMBGPV.

Figure 79: Sample ARMBGPV JCL

```plaintext
/* *************************************************************** */
/* *************************************************************** */
/*          RECOVERY MANAGER             BMC SOFTWARE, INC.        */
/* *************************************************************** */
/*                   BATCH REVALIDATION REPORT                     */
/*                      FOR SELECTED GROUPS                        */
/* *************************************************************** */
/* *************************************************************** */
ARM0000 EXEC PGM=ARMBGPV,
  PARM='DEC2,ARMOPTS=ARM$OPTS',
  REGION=4M
  STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
  DD DISP=SHR,DSN=DSNEXIT
  DD DISP=SHR,DSN=DSNLOAD
  ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
  ARMPRINT DD SYSOUT=*
  ARMERROR DD SYSOUT=*
  SYSPRINT DD DUMMY
  ARMRRESRC DD SYSOUT=*
  ARMXCEPT DD SYSOUT=*
  ARMPICK DD SYSOUT=*
  ARMRCALL DD SYSOUT=*
  ARMIN DD *
SET OPTIONS
```
Sample output

The following figure shows a sample group revalidation report generated by ARMBGPV.

** **
** RECOVERY MANAGER FOR DB2 V11.1.00 - GROUP REVALIDATION 08/14/2012 13:23:03 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.1.00 **

(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884

BMC80223I MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT: BPJ0197 BPJ0215 BPJ0219

BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910

SET CURRENT SOLID = ARMQA;
BMC80570I COMMAND COMPLETE RC = 0
SET OPTIONS
JCLTYPE LOCAL
BACKOUT NO
;
BMC80570I COMMAND COMPLETE RC = 0
REVALIDATE GROUP "ARMQA"."SAMPLE"
TOCURRENT
;
BMC80570I COMMAND COMPLETE RC = 0
BMC80571I PROGRAM COMPLETE RC = 0

Figure 81 on page 503 shows the recovery resources required to recover the group.

** **
** RECOVERY MANAGER FOR DB2 V11.1.00 - GROUP REVALIDATION 08/14/2012 13:23:03 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.1.00 **

BMC80522I RECOVERY RESOURCES

IMAGE COPIES = LP  ARCHIVE LOGCOPY = 1  CHG ACCUM FILES = LP

------------------------------------------- LOCAL SITE - GROUP ARMQA.SAMPLE
-------------------------------------------

<table>
<thead>
<tr>
<th>SPACE NAME</th>
<th>INSTANCE</th>
<th>PART</th>
<th>RESOURCE PART</th>
<th>DATE</th>
<th>TIME</th>
<th>VOLSER</th>
<th>DSN</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARMBGN22.TNL0VL</td>
<td>1</td>
<td>0000</td>
<td>FULL REF 0000</td>
<td>12-08-14</td>
<td>10:11:50</td>
<td>134997</td>
<td></td>
</tr>
<tr>
<td>RDAJBM.ARMBGN22.TNL0VL.LP00.T101125</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARMBGN22.TNL1VL</td>
<td>1</td>
<td>0000</td>
<td>FULL REF 0000</td>
<td>12-08-14</td>
<td>10:11:52</td>
<td>134997</td>
<td></td>
</tr>
<tr>
<td>ARMBGN22.TNL1VL.LP00.T101125</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RDAJBM.ARMBGN22.TNL1VL.LP00.T101125</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARMBGN22.TN22EX</td>
<td>1</td>
<td>0000</td>
<td>FULL REF 0000</td>
<td>12-08-14</td>
<td>10:11:53</td>
<td>134997</td>
<td></td>
</tr>
</tbody>
</table>
Figure 82 on page 504 lists any missing recovery resources that would prevent objects in the group from being recovered.

** Figure 82: Sample ARMBGPV ARMXCEPT DD output **

** RECOVERY MANAGER for DB2 V11.1.00 - GROUP REVALIDATION 08/14/2012 13:23:03 **

** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.1.00 **

BMC80523I RECOVERABILITY EXCEPTIONS

IMAGE COPIES = LP ARCHIVE LOGCOPY = 1 CHG ACCUM FILES = LP

------------------------------------ LOCAL SITE - GROUP ARMQA.SAMPLE -------------------

---------------------------------------- MISSING RESOURCE(S) FROM MVS CATALOG -------------------
** RECOVERY MANAGER FOR DB2 V11.1.00 - GROUP REVALIDATION 02/18/2012 15:05:01 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.1.00 **
BMC80525I VOLSER PICK LIST
700617

Figure 83 on page 505 shows a volume pick list. This list is blank if no volumes are included in the group.

Figure 83: Sample ARMPICK ARMPRINT DD output

Executing the JCL

This section describes special instructions or information required to run the ARMBGPV JCL.

- Ensure that the SQLID used has appropriate authority for the group(s). See “Authorizations” on page 491 for required authorizations.

- If you specified TORESTARTRBA, run ARMBGPV after you have created backups and run ARMBSRR.

- No restart is available for ARMBGPV. You must resubmit the job after correcting any error conditions.
ARMBGRP—Group creation and maintenance

This chapter describes ARMBGRP—Group creation and maintenance.

About ARMBGRP

Use the ARMBGRP program to perform the following functions:
- Create groups
- Rename groups
- Update group utility options
- Delete groups
- Report the contents of groups
- Query repository and groups for subsystem ID

Creating groups

Use the CREATE GROUP command to create a group based on exceptions, indexes, index space names, plans, packages, storage groups, user-defined SQL, table name, table spaces, or volumes.

You can optionally copy an existing group in order to retain the same backup and recovery options and authorizations.

With dynamic grouping, you can create an empty group (a group for which no objects are found that meet the definition). After you create the objects and then open the group, the group is automatically populated. This ability enables you to prepare to back up and recover objects that do not yet exist. For example, if you know that a new application is being added to your system, you can set up the groups in advance. When the objects are created, dynamic grouping will automatically find and add them to the backup and recovery jobs.
Groups created by volume are static groups, not dynamic groups. Groups created by exception are static groups by default, but when creating groups online or in batch using VIA EXCEPTION, you can specify DYNAMIC to enable dynamic grouping. For example, to enable dynamic group, you can use the following syntax:

```plaintext
//ARMIN     DD *
CREATE GROUP USERID.EXCEPTION
  DESCRIPTION "EXCEPTION STATUS OBJECTS"
  REPLACE YES
  VIA EXCEPTION
  DYNAMIC
  RECOVERYPEND
  CHECKPEND
  COPYEND
  LPL
STOPPENDERRORRANGE
/*
```

With RECOVERY MANAGER version 11.1 and later, you can use multiple VIA statements in the CREATE GROUP syntax. For example, you can use the following syntax:

```plaintext
//ARMIN     DD *
CREATE GROUP RDAJTR.TEST_BYDEF2
  REPLACE YES
  DESCRIPTION 'CREATE BY DEF '
  VIA TABLE
    JTR.T40N
  INCLUDEIX NO
  VIA TABLESPACE
    ARMDBJTR.TS40P1
  INCLUDEIX NO;
... ...
/*
```

You cannot use the VIA VOLUMES option multiple times. VIA EXCEPTION is also not allowed multiple times unless you have enabled dynamic grouping.

**Unicode support**

RECOVERY MANAGER supports DB2 objects with Unicode names (both those that can be translated to EBCDIC and those that cannot).

RECOVERY MANAGER online panels and reports make use of EBCDIC characters. Any Unicode characters that cannot be translated into EBCDIC are represented with hexadecimal substitution characters.
About table space table index exception and volume groups

For groups made via table space name pattern, table name pattern, index name pattern, and exception status, ARMBGRP performs a DB2 catalog search.

For volume groups, ARMBGRP supports two different methods of obtaining information. You can create the group by performing catalog searches or by using the volume’s VSAM volume data set (VVDS).

■ Catalog search method

The catalog search method uses the DB2 and Integrated Catalog Facility (ICF) catalogs to locate the DB2 objects for the desired subsystem or data sharing group. This method does not require the volume to be online and can be executed after a volume failure has occurred. This method is ideal for an ad hoc recovery after an unexpected volume failure.

■ VVDS method

The VVDS method uses the VSAM "table of contents" located on the volume and the DB2 catalog to identify objects that belong to the specified DB2 subsystem or data sharing group. This method requires the volume to be online and available at the time that the job executes.

WARNING

Recovering with a group that was created by the VVDS method does not include objects that were created or moved to the volume after the group was generated.

Note

Creating a volume group interactively can take significantly longer than using the batch process. The interactive process assumes that the volume is available and searches the operating system and DB2 catalogs for the required information.

About user-defined SQL groups

For groups created using user-defined SQL, RECOVERY MANAGER uses the ARMSQL DD statement and the VIA SQL syntax.

For more information about the ARMSQL DD statement, see “Specifying the ARMBGRP data set DD statements” on page 516.

For more information about the VIA SQL syntax, see “CREATE GROUP” on page 530.
RECOVERY MANAGER also allows you to enter dynamic SQL inline in the VIA SQL syntax between the #BEGINSQL and #ENDSQL options. You can enter multiple VIA SQL statements with the inline SQL in a CREATE GROUP statement.

The VIA SQL syntax supports a limited number of key words for including related objects. Subselects are supported. The same SQL restrictions apply in batch as in the online group creation by SQL (“Creating a new group” on page 113).

**About repository groups**

Use the VIA REPOSITORY option to create a group containing the repository objects of RECOVERY MANAGER (with R+/CHANGE ACCUM and Log Master, if they are present in your system).

RECOVERY MANAGER obtains the tables and indexes that make up the repository of each product from the plan listed in the option file of each product.

The default option file names for each product are as follows:

- ARM$OPTS (RECOVERY MANAGER)
- AFR$OPTS (NGT Recover)
- ALP$OPTS (Log Master)

Creating a repository group can streamline the backup and recovery of the repositories, especially if you are using NGT Copy or Online Consistent Copy as the backup utility. RECOVERY MANAGER uses the OBJECTSET option of NGT Copy, which means that you do not need to regenerate the backup JCL when objects in the group change. For more information, see “OBJECTSET support” on page 169.

**Note**

If you change the plan names for RECOVERY MANAGER, Log Master, or R+/CHANGE ACCUM, you should rebuild the repository group to prevent problems from occurring.

Also, when you migrate to a new version of DB2, you need to create new repository groups to backup the objects of the repository.

**Copying groups**

Use the LIKE parameter to copy an existing group.

RECOVERY MANAGER verifies that the new name does not already exist, and then copies the group into the repository. The new group retains all utility options setting of the original group, and you can optionally retain the authorizations and objects of the original group as well.
Use the following syntax to copy a group:

```
CREATE GROUP creator2.name2 LIKE creator1.name1
    RETAIN AUTH yes/no
    RETAIN OBJECTS yes/no
```

If you want to copy an existing group but also include additional objects, you can use the LIKE parameter when creating a new group with the VIA TABLESPACE, VIA TABLE, VIA SQL, or VIA INDEX options.

**Example**

```
CREATE GROUP creator2.name2
    VIA TABLE SPACE tablespaceName
    LIKE creator1.name1
    RETAIN AUTH yes/no
    RETAIN OBJECTS yes/no
```

### Renaming groups

Use the RENAME GROUP command to rename one or more groups.

RECOVERY MANAGER verifies that the new names do not already exist, and then writes the new and old names to the ARMRENAM file.

The RENAME GROUP command uses the following format:

```
RENAME GROUP creator1.name1 NEWNAME creator2.name2
```

### Wildcard support for RENAME GROUP

Wildcard support for the RENAME GROUP command differs from wildcard support use for the other commands.

For RENAME GROUP, wildcard characters are supported as follows:

- Use characters * (asterisk) and % (percent) to match multiple characters.
- Use the character ? (question mark) to match a single character.

Be aware of the following rules that apply to RENAME GROUP only:

- Using a multiple-character wildcard (* or %) in the NEWNAME parameter causes all characters in either the creator or name portion of the group name to be retained, starting at the position of the wildcard.
Note

The NEWNAME creator and name cannot both begin with a multiple-character wildcard.

- Using a single-character wildcard (?) in the NEWNAME field causes the corresponding character in the existing group name to be retained.

- If the GROUP creator contains a multiple character wildcard, the NEWNAME creator must also contain a multiple character wildcard.

- If the GROUP name contains a multiple character wildcard, the NEWNAME name must also contain a multiple character wildcard.

- No further processing on the group name is performed after a multiple-character wildcard is encountered. If you use a multiple-character wildcard, that wildcard must be the last character in the creator or name.

Sample wildcard usage for renaming groups

The following table contains examples of using wildcards when renaming groups.

Table 45: Sample wildcard usage for renaming groups

<table>
<thead>
<tr>
<th>Function</th>
<th>Command</th>
<th>Previous GroupName(s)</th>
<th>New Group Name(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rename a single group</td>
<td>RENAME GROUP USERABC.PAYROLL</td>
<td>USERABC.PAYROLL</td>
<td>USERXYZ.PAYROLL</td>
</tr>
<tr>
<td>Assign a new creator name to multiple groups</td>
<td>RENAME GROUP USERABC.*</td>
<td>USERABC.PAYROLL</td>
<td>USERXYZ.PAYROLL</td>
</tr>
<tr>
<td>using multiple wildcards</td>
<td>NEWNAME USERXYZ.*</td>
<td>USERABC.TIMECARD</td>
<td>USERXYZ.TIMECARD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>USERABC.INVENTORY</td>
<td>USERXYZ.INVENTORY</td>
</tr>
<tr>
<td>Change the name of multiple groups using</td>
<td>RENAME GROUP USERABC.??TEST</td>
<td>USERABC.ABCTEST</td>
<td>USERABC.ABCPROD</td>
</tr>
<tr>
<td>single-character wildcards</td>
<td>NEWNAME USERABC.??PROD</td>
<td>USERABC.XYZTEST</td>
<td>USERABC.XYZPROD</td>
</tr>
<tr>
<td></td>
<td></td>
<td>USERABC.LMNTEST</td>
<td>USERABC.LMNPROD</td>
</tr>
</tbody>
</table>

Updating group options

You can update many group options in batch mode using ARMBGRP.
See “UPDATE GROUP option description” on page 559 for a complete listing. The changes that you make are stored in the repository and remain in effect for the group until you change them again either online or in batch mode.

You can update the options for multiple groups with a single command if you use wildcard characters in the group name. See “Syntax rules” on page 102 for more information on syntax rules and wildcard support.

---

**Note**

You can remove any group-level value by entering RESET as the value for the recover or copy option. RESET causes the option to default to the value set at the subsystem level. If no subsystem value exists, the option defaults to the product level.

---

### Deleting groups

You can delete multiple groups with a single command by using wildcards in the group name.

(See “Syntax rules” on page 102 for more information.)

---

**Note**

Deleting a group removes only the groups definition from the repository and has no affect on the objects within the group.

---

### Reporting group information

You can generate reports about one or more groups, as follows:

- Objects within the group (written to file ARMOBJS)
- Group authorizations (written to file ARMAUTH)
- Group definition (written to file ARMDEFN)
- Recover utility options (written to file ARMRCVR)
- Copy utility options (written to file ARMCOPY)

Generate reports for multiple groups with a single command by using wildcards in the group name. For more information, see “Syntax rules” on page 102.
Exporting and migrating groups

RECOVERY MANAGER now enables you to convert group definitions into CREATE GROUP syntax and write them to a specified data set. You can then either keep the data set as a backup or use it to migrate the converted definitions to another DB2 subsystem.

The export and migration of group definitions involves the following steps:

1. Use the online interface to prepare ARMBGRP syntax and JCL.
2. Execute the ARMBGRP batch program.

This change includes adds the following:

- ARMEXPRT DD
- EXPORT GROUP and MIGRATE_TO_SSID commands
- The following options on the Group List panel (ARMGP001):
  - X to export the specified groups
  - XA to export all groups
- Batch Group Export/Migrate JCL Generation panels

Using the query function

You can use ARMBGRP to issue SQL query statements for the following information:

- Missing Objects (lists objects not in any group)
- Multiple Objects (lists objects in more than one group)
- Group Information (lists all groups, their creators, and other information)
- Group Objects (lists all groups and all objects contained in those groups)
- Group Authorization (lists all groups and associated authorization information)
- BMCTABLES (lists the names of the BMC tables used by the RECOVERY MANAGER plan)
- Groups that contain specified table spaces, and optionally indexes
Authorizations

The following authorizations are required to execute the ARMBGRP program:

- EXECUTE authority for the RECOVERY MANAGER DB2 plan
- READ access to the VVDS for the volumes (if using the volume VVDS method)
- TYPE A (ALL) authority for the groups (or SYSADM or system DBADM authority)
- Authorized Program Facility (APF) authorization on ARMBGRP (required when you use the background method and replace an existing group of the same name)
- APF authorization for the RECOVERY MANAGER load library

Building the ARMBGRP JCL

Building your own ARMBGRP job involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition statements that specify the use of the following libraries or data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Input data sets
  - Output data sets

The descriptions in the following subsections provide more details.

Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends that you specify REGION=0M, which makes the amount of virtual
storage that is needed to run the job automatically available when the ARMBGRP job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

### Specifying the EXEC statement

The EXEC statement has the following format:

```
//stepname EXEC PGM=ARMBGRP,
  //     PARM='ssid,ARMOPTS=optionSet',
  //     REGION=0M
```

The variable ssid is the DB2 subsystem or group attach name where the RECOVERY MANAGER groups reside. If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

**Note**

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

The variable optionSet is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

### Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBGRP to use. See the following example:

```
//STEPLIB   DD DISP=SHR,DSN=SCC.WJLW1210.LOAD
  //     DD DISP=SHR,DSN=SCC.TEST1210.LGCLINK
  //     DD DISP=SHR,DSN=SCC.TEST1210.BMCLINK
  //     DD DISP=SHR,DSN=ARM.WJLW1210.LOAD
  //     DD DISP=SHR,DSN=SYS3.DEFQ.DSNEXIT
  //     DD DISP=SHR,DSN=CSGI.DB2V10M.DSNLOAD
```

### Specifying the ARMBGRP data set DD statements

This topic describes the data sets that ARMBGRP uses.

Each data set is specified by a *ddname* (data definition name). You must specify all required data sets in the JCL.
■ **(required) ARMMSGs**

The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of `hilvl.RMGR.ARMCNTL(ARMMSGs)`. The data set must be allocated with DISP=SHR.

■ **(optional) ARMOPTS**

The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

```
//ARMOPTS DD *
ssid.configurationOption=value
/*
```

■ **(required) ARMPRINT**

The output for messages that are returned from RECOVERY MANAGER. RECOVERY MANAGER also echoes the contents of the ARMIN data set in the ARMPRINT output. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

The following figure shows sample ARMPRINT for the EXPORT GROUP command with a wildcard pattern:

```
EXPORT GROUP PUBLIC.Q187*;
BMC80591I PROCESSING GROUP PUBLIC.Q1876526
BMC80591I PROCESSING GROUP PUBLIC.Q1878348
BMC80591I PROCESSING GROUP PUBLIC.Q1878802
BMC80591I PROCESSING GROUP PUBLIC.Q1878802_IX
BMC80591I PROCESSING GROUP PUBLIC.Q1878802_IXONLY
BMC80570I COMMAND COMPLETE RC = 0
```

■ **(required) ARMERROR**

The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

■ **(optional) ARMEXPRT**

This data set to which CREATE GROUP syntax for one or more data sets is written. The ARMEXPRT data set is required for using the EXPORT GROUP or MIGRATE_TO_GROUP `ssid` commands. This data set can be either sequential or a partitioned data set (PDS). If it is a PDS, you must specify the member name. Attributes for this data set must be fixed-length records, with a length of 80 (RECFM=F or FB, LRECL=80).
(required) ARMIN
The input data set that contains one or more control statements. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

(optional) ARMOBJS
The output for the Group Object report. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

(optional) ARMAUTH
The output for the Group Authorization report. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

(optional) ARMDEFN
The output for the Group Definition Report. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

(optional) ARMRCVR
The output for the Recover Utility Options Report. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

(optional) ARMCOPY
The output for the Copy Utility Options Report. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

(optional) ARMQUERY
The output for query functions. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

(optional) ARMRENAM
The output data set that stores the new and old group names when the RENAME command is used. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

(required for creating groups with user-defined SQL) ARMSQL
The input for groups created with user-defined SQL. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80). However, SQL syntax is limited to the first 72 bytes.

The following example of how you can set up ARMSQL is taken from the ARMBGRP member in the .ARMSAMP data set that was created during installation:

```
//** FILE USED TO CREATE GROUP VIA SQL
//ARMSQL    DD *
SELECT DBNAME, NAME FROM SYSIBM.SYSTABLESPACE
WHERE DBNAME LIKE 'ARMDB%'
/*
//** OR PDS FILE USED TO CREATE GROUP VIA SQL
```
When you use ARMSQL and SQL, follow these guidelines:

- To create a group by partition, select from SYSIBM.SYSTABLEPART or SYSIBM.SYSINDEXPART.

- To create a group that is not by partition, select from SYSIBM.SYSTABLE or SYSIBM.SYSINDEX.

- Specify TS, IX, or SG as the first variable. RECOVERY MANAGER checks to make sure that 'TS', 'IX', or 'SG' follows the SELECT statement.

- Blank lines are excluded.

- You may enter up to 16,000 characters.

- Only one SQL statement is accepted.

- Semicolons are not allowed.

- If not specified, the partition number is set to 0.

- The IX name length and number of partitions follow the rules of the DB2 version that you are using.

- The SQL statements must be exactly as shown in the examples in the following table and cannot deviate except in the WHERE clause.

<table>
<thead>
<tr>
<th>Object type</th>
<th>SQL statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>TS</td>
<td>SELECT 'TS', DBNAME, NAME FROM SYSIBM.SYSTABLESPACE WHERE DBNAME LIKE 'QZU%'</td>
</tr>
<tr>
<td></td>
<td>SELECT 'TS', DBNAME, TNAME, PARTITION FROM SYSIBM.SYSTABLEPART WHERE DBNAME='QZUDPT22' AND TNAME='QZUS0122' AND PARTITION IN (4090, 4092, 4094, 4096)</td>
</tr>
<tr>
<td>IX</td>
<td>SELECT 'IX', CREATOR, NAME, CREATOR, NAME FROM SYSIBM.SYSINDEXES WHERE DBNAME = 'R92DB59'</td>
</tr>
<tr>
<td></td>
<td>SELECT 'IX', IXCREATOR, IXNAME, IXCREATOR, IXNAME, PARTITION FROM SYSIBM.SYSINDEXPART WHERE IXCREATOR='QZU' AND PARTITION &gt; 100 AND PARTITION &lt;= 200</td>
</tr>
<tr>
<td>SG</td>
<td>SELECT 'SG', CREATOR, NAME, CREATOR, NAME FROM SYSIBM.SYSSTOGROUP WHERE NAME = 'SG1234'</td>
</tr>
</tbody>
</table>

For more information, see “CREATE GROUP” on page 530.
- \textit{VVVVVV} (required for the volume VVDS method only)

Required when you use the VVDS method to create volume groups. The ddname must be V vvvvvv where the variable vvvvvv is the volume name. The data set name must be the volume’s VVDS name. The data set should be allocated with DISP=SHR.

### CREATE GROUP syntax and option descriptions

The ARMBGRP syntax and option descriptions in this subsection are the control statements used for ARMIN input.

For more information about online support for creating groups, see “Creating and working with groups” on page 105.

\textit{Note}

For more information on syntax rules and wildcard support, see “Syntax rules” on page 102.

ARMBGRP syntax for building a group based on catalog search (exceptions, indexes, user-defined SQL, table name, table spaces, or volume) is shown in “Syntax for creating groups: Catalog search method” on page 521.

ARMBGRP syntax for building a volume group based on the VVDS is shown in “Syntax for creating volume groups: VVDS method” on page 544.
Syntax for creating groups: Catalog search method

The syntax in the following figure is used for creating groups based on catalog search (exceptions, indexes, user-defined SQL, table name, table spaces, or volume).

Figure 84: ARMBGRP—CREATE GROUP
Figure 85: ARMBGRP—CREATE GROUP—tablespace, table, and exception options

```
tablespace, table, and exception options

VIA TABLESPACE
    databaseName.tablespaceName[ owner ]

VIA TABLE
    creatortableName

VIA EXCEPTION
    ALL
    ADVISORYREBUILD
    ADVISORYREORG
    AUXILIARYCHECK
    AUXILIARYWARN
    CHECKPEND
    COPYEND
    DYNAMIC
    ERRORRANGE
    GRECP
    INFORMATIONALCOPY
    LPL
    PERSISTENTREADONLY
    REBUILDPEND
    RECOVERPEND
    REORGPEND
    REPLICATIONONLY
    STOPPEDERRORRANGE
```
CREATE GROUP syntax and option descriptions

**tablespace, table, and exception options**

- `LIKE creator2.name2`
- `RETAINT AUTH`:
  - NO
  - YES
- `RETAINT OBJECTS`:
  - YES
  - NO
- `BYPART`:
  - NO
  - YES
- `EXCLUDEPARTS*`:
  - `nnnn`
  - ALL
- `EXCLUDE`:
  - `databaseName.tableSpaceName[,owner]`
- `EXCLUDEIX*`:
  - `indexName`
- `DSNUM*`:
  - `n`
  - `n:m`
  - ALL
- `EXCLUDE DSNUM*`:
  - `n`
  - `n.m`
- `INCLUDERI`:
  - NO
  - YES
- `INCLUDEIX`:
  - NO
  - YES
- `INCLUDELOB`:
  - NO
  - YES
  - BASE
  - ONLY
- `INCLUDEXML`:
  - NO
  - YES
  - BASE
  - ONLY
- `INCLUDEHISTORY`:
  - NO
  - YES
  - BASE
  - ONLY
- `INCLUDEARCHIVE`:
  - NO
  - YES
  - BASE
  - ONLY

* For use with VIA TABLESPACE only
**Figure 86: ARMBGRP Control Statement—CREATE GROUP—SQL options**

**SQL options**

VIA SQL

LIKE creator2.name2

RETAIN AUTH

NO

YES

RETAIN OBJECTS

YES

NO

#BEGINSQL – SQL statements - #ENDSQL

INCLUDERI

NO

YES

INCLUDEIX

NO

YES

INCLUDELOB

NO

YES

BASE

ONLY

INCLUDEXML

NO

YES

BASE

ONLY

INCLUDEHISTORY

NO

YES

BASE

ONLY

**Figure 87: ARMBGRP Control Statement—CREATE GROUP—volume options**

**volume options**

VIA VOLUMES

(volumeName)

EXCLUDE

databaseName$tableSpaceName [owner]

tableName

EXCLUDEIX

indexName

* For use with VIA TABLESPACE only
Figure 88: ARMBGRP Control Statement—CREATE GROUP—index options

CREATE GROUP syntax and option descriptions

Chapter 19 ARMBGRP—Group creation and maintenance 525
Figure 89: ARMBGRP Control Statement—CREATE GROUP—indexspace options

```plaintext
CREATE GROUP syntax and option descriptions
```

Figure 90: ARMBGRP Control Statement—CREATE GROUP—repository options

```plaintext
CREATE GROUP syntax and option descriptions
```
Figure 91: ARMBGRP Control Statement—CREATE GROUP—package options
Figure 92: ARMBGRP Control Statement—CREATE GROUP—plan options

- VIA PLAN
  - planName
  - EXCLUDE
    - planName

- BYPART
  - NO
  - YES

- INCLUDERI
  - NO
  - YES

- INCLUDEIX
  - NO
  - YES

- INCLUDELOB
  - NO
  - YES
  - BASE
  - ONLY

- INCLUDEXML
  - NO
  - YES
  - BASE
  - ONLY

- INCLUDEHISTORY
  - NO
  - YES
  - BASE
  - ONLY
CREATE GROUP syntax and option descriptions

Figure 93: ARMBGRP Control Statement—CREATE GROUP—stogroup options

```
stogroup options

VIA STOGROUP (stogroupName)

BYPART
  NO
  YES

EXCLUDE
  stogroupName

INCLUDERI
  NO
  YES

INCLUDEIX
  NO
  YES

INCLUDELOB
  NO
  YES
  BASE
  ONLY

INCLUDEXML
  NO
  YES
  BASE
  ONLY

INCLUDEHISTORY
  NO
  YES
  BASE
  ONLY
```

Figure 94: ARMBGRP Control Statement—CREATE GROUP—group options

```
group options

VIA GROUP (groupName)

EXCLUDEGROUP (groupName)
```
CREATE GROUP option descriptions (catalog search)

This section contains descriptions of the options used with the CREATE GROUP command when using the catalog search method.

CREATE GROUP

This control statement is required when you create a volume, repository, or table space group based on the catalog search method.

You can repeat the CREATE GROUP statement for as many groups as you want to create, using the format creator.name.

DESCRIPTION

Optionally, you can provide a description of the group. Text can be up to 25 characters and must be enclosed in double quotes (" ").

REPLACE

Set REPLACE YES to automatically update an existing group when object changes are detected. The default is NO.

CLONES ONLY

Use this option to include only clone objects in the exception group. This option is available only when running on DB2 Version 10 or later and is not valid with compatibility mode. CLONES ONLY syntax is valid only for group creation using VIA EXCEPTION and VIA VOLUMES. This syntax is ignored for all other types of group creation.

LIKE

LIKE copies an existing group. All backup and recovery options as well as authorizations and objects can optionally be copied to the new group that you are creating. Enter the name of the existing group in the format creator2.name2. For more information, see “Copying groups” on page 510.

Note

- Only one LIKE parameter is allowed in each CREATE command.
- To use LIKE, your primary or secondary AUTHID must have authority to open creator2.name2 (or else you must have SYSADM or system DBADM authority). The name of the new group that you are creating cannot be the same as creator2.name2.
RETAIN AUTH

This option is valid in conjunction with the LIKE statement. RETAIN AUTH YES causes the new group to retain all of the authorizations granted in the existing group. RETAIN AUTH NO (the default) causes the new group to be created with no group authorizations.

RETAIN OBJECTS

This option is valid in conjunction with the LIKE statement. RETAIN OBJECTS YES (the default) causes the new group to retain all of the objects and object definitions currently within the existing group. RETAIN OBJECTS NO causes the new group to be created without including the objects from the existing group.

VIA TABLESPACE, TABLE, and EXCEPTION

Use this option to specify creating groups by table spaces, tables, or exceptions.

VIA TABLESPACE

When creating a group by table space, you can include any number of table space names or wildcard patterns. The table space is named by databaseName. tableSpaceName[, owner], where owner is optional.

VIA TABLE

When creating a group by table name, you can include any number of table names or wildcard patterns.

VIA EXCEPTION

When creating a group by exception status, you can include all exceptions (ALL), or one or more of the following exception types. The default is RECOVERPEND.

- ADVISORYREBUILD
- ADVISORYREORG
- AUXILIARYCHECK
- AUXILIARYWARN
- CHECKPEND
- COPYPEND
- DYNAMIC
- ERRORRANGE
- GRECP
- INFORMATIONALCOPY
- LPL
- PERSISTENTREADONLY
Groups created in batch using VIA EXCEPTION are static groups by default, but you can enable dynamic grouping using the DYNAMIC option. For more information, see “Creating groups” on page 507.

**LIKE**

When used in conjunction with VIA TABLESPACE, VIA TABLE, or VIA EXCEPTION, the LIKE option builds a group using the table name, table space, or table space name pattern that you specified and copies the attributes of an existing group. The backup and recovery options of the existing group and optionally the authorizations and objects of the existing group are copied to the new group that you are creating. Enter the name of the existing group in the format creator2.name2. For more information, see “Copying groups” on page 510.

Only one LIKE parameter is allowed in each CREATE command. The LIKE parameter can be coded either before or after the VIA TABLESPACE or VIA EXCEPTION option. To use LIKE, your primary or secondary AUTHID must have authority to open creator2.name2 (or else you must have SYSADM or system DBADM authority). The name of the new group that you are creating cannot be the same as creator2.name2.

**RETAIN AUTH**

This option is only valid in conjunction with the LIKE statement. RETAIN AUTH YES causes the new group to retain all of the authorizations granted in the existing group. RETAIN AUTH NO causes the new group to be created with no group authorizations.

**RETAIN OBJECTS**

This option is valid in conjunction with the LIKE statement. RETAIN OBJECTS YES (the default) causes the new group to retain all of the objects and object definitions currently within the existing group. RETAIN OBJECTS NO causes the new group to be created without including the objects from the existing group.

**BYPART**

Use this option to add table spaces to the group by partition. This option is not valid with nonpartitioned table spaces.
DSNUM

Use this option with VIA TABLESPACE to include a specified table space by partition.

--- Note ---
You cannot use DSNUM with the BYPART option.

Valid values for DSNUM are:

- $n$—a partition number.
- $n:m$—a range of partitions starting with $n$ and ending with $m$, where $n$ and $m$ are partition numbers.
- ALL—includes all partitions.

--- Tip ---
BMC Software recommends that you use DSNUM ALL with PBG table spaces, when the number of partitions can change.

EXCLUDEPARTS

Use EXCLUDEPARTS to exclude partitions when creating a group by partition using the BYPART YES option. This option is only valid with BYPART YES. You must specify the EXCLUDEPARTS parameter before the EXCLUDE table space parameter. The value of the EXCLUDEPARTS parameter remains in effect for all subsequent EXCLUDE parameters until you specify a new EXCLUDEPARTS value. EXCLUDEPARTS replaces the EXCLUDEALLPARTS parameter.

--- Note ---
You cannot use EXCLUDEPARTS for groups built with VIA TABLE syntax.

Valid values for EXCLUDEPARTS are:

- $nnnn$—the number of the specific partition that you want to exclude from the group. The table space to which the partition belongs is specified in the subsequent EXCLUDE statement. The partition number is an integer from 1 to 4096.

In the following example, partition 4 of table space DB1.TS1 is excluded from the group. All other partitions of DB1.TS1 are included.

```
BYPART YES
EXCLUDEPARTS 4
EXCLUDE DB1.TS1
```
- **ALL**—excludes all partitions of the table space or spaces specified by the subsequent **EXCLUDE** option.

In the following example, all partitions of table space DB1.TS1 and DB2.TS2 are excluded from the group:

```
BYPART YES
EXCLUDEPARTS ALL
EXCLUDE DB1.TS1
EXCLUDE DB2.TS2
```

**EXCLUDE**

Enter table space name (or wildcard pattern) to exclude those spaces from the group when you use VIA TABLESPACE. The table space is named by `databaseName.tableSpaceName[.owner]`, where `owner` is optional. Use **EXCLUDE** in conjunction with **EXCLUDEPARTS** to exclude specific partitions from the table spaces in the group.

When you use VIA TABLE syntax, you can also use exclude with `tableName`. Enter the table name or table name list to exclude tables from the group.

**EXCLUDE DSNUM**

Use this option with VIA TABLESPACE to exclude a specified table space by partition.

*Note*

You cannot use **EXCLUDE DSNUM** with the **BYPART** option.

Valid values for **EXCLUDE DSNUM** are:

- **n**—a partition number
- **n:m**—a range of partitions starting with **n** and ending with **m**, where **n** and **m** are partition numbers

**EXCLUDEIX**

Use this option with VIA TABLESPACE to exclude the specified indexes from the group.

**INCLUDE**

Use this option to include all table spaces associated by referential integrity in the group.

**INCLUDEIX**

Use this option to include all associated indexes in the group.
INCLUDELOB

Use this option to add all table spaces that are associated by LOB columns with the objects in the group. Doing so ensures that both the base table space and the LOB table space are included in the group.

INCLUDEXML

Use this option to add all table spaces that are associated by XML columns with the objects in the group. Doing so ensures that all XML-related objects are included in the group and will be processed together.

INCLUDEHISTORY

Use this option for DB2 Version 10 and later to add all of the objects that are associated by a history (versioning) relationship to those specified in the group. The objects are also referred to as temporal objects and history objects.

INCLUDEARCHIVE

Use this option for DB2 Version 11 and later to add all of the objects that are associated by an archive relationship to those specified in the group.

VIA SQL

Use this option to specify objects by using a user-defined SQL SELECT statement in the ARMSQL DD statement.

For more information, see “Specifying the ARMBGRP data set DD statements” on page 516.

You can also enter dynamic SQL inline in the VIA SQL syntax using #BEGINSQL and #ENDSQL. This option allows multiple VIA SQL statements with multiple SQL syntax in the same CREATE GROUP syntax as shown in the following example:

```sql
//ARMIN DD *
CREATE GROUP RDAJTR.BY_SQL01
REPLACE YES
DESCRIPTION 'SQL BY BATCH'
VIA SQL
#BEGINSQL
SELECT 'TS', DBNAME, NAME
FROM SYSIBM.SYSTABLESPACE
WHERE DBNAME = 'ARMDBJTR'
#ENDSQL
INCLUDERIN NO
INCLUDEIX NO
INCLUDELOB NO
VIA SQL
#BEGINSQL
SELECT 'TS', DBNAME, NAME
FROM SYSIBM.SYSTABLESPACE
WHERE DBNAME = 'ARMDBLOB'
#ENDSQL
```
LIKE

When used in conjunction with VIA SQL, the LIKE option builds a group using the name pattern that you specified and copies the attributes of an existing group. The backup and recovery options of the existing group and optionally the authorizations and objects of the existing group are copied to the new group that you are creating. Enter the name of the existing group in the format creator2.name2. For more information, see “Copying groups” on page 510.

Be aware of the following information:

- Only one LIKE parameter is allowed in each CREATE command.
- The LIKE parameter can be coded either before or after the VIA SQL option.
- To use LIKE, your primary or secondary AUTHID must have authority to open creator2.name2 (or else you must have SYSADM or system DBADM authority). The name of the new group that you are creating cannot be the same as creator2.name2.

RETAIN AUTH

This option is only valid in conjunction with the LIKE statement. RETAIN AUTH YES causes the new group to retain all of the authorizations granted in the existing group. RETAIN AUTH NO causes the new group to be created with no group authorizations.

RETAIN OBJECTS

This option is valid in conjunction with the LIKE statement. RETAIN OBJECTS YES (the default) causes the new group to retain all objects and object definitions currently within the existing group. RETAIN OBJECTS NO causes the new group to be created without including the objects from the existing group.

#BEGINSQL ... #ENDSQL

Use #BEGINSQL and #ENDSQL to enclose SQL statements within the VIA SQL syntax. SQL must begin with syntax #BEGINSQL and must end with #ENDSQL.
INCLUDER

Use this option to include all table spaces associated by referential integrity in the group.

INCLUDEIX

Use this option to include all associated indexes in the group.

INCLUDELOB

Use this option to add all table spaces that are associated by LOB columns with the objects in the group. Doing so ensures that both the base table space and the LOB table space are included in the group.

INCLUDEXML

Use this option to add all table spaces that are associated by XML columns with the objects in the group. Doing so ensures that all XML-related objects are included in the group and will be processed together.

INCLUDEHISTORY

Use this option for DB2 Version 10 and later to add all of the objects that are associated by a history (versioning) relationship to those specified in the group. The objects are also referred to as temporal objects and history objects.

VIA VOLUMES

When creating a group by volume, enter any number of volume names separated by commas. You cannot use wildcards in volume names.

Tip

BMC recommends that you specify all volumes to be included in the group in the same statement. The time needed to search the system and DB2 catalogs is independent of the number of volumes.

Following is sample JCL for VIA VOLUMES:

```sql
CREATE GROUP USERID.DEV240
REPLACE YES
VIA VOLUMES (DEV240)
EXCLUDE DBNAME1.TSNAME1
EXCLUDEIX CREATOR1.IXNAME1
;
```

EXCLUDE

Enter table space name (or wildcard pattern) to exclude those spaces from the group when you use VIA VOLUMES. The table space is named by `databaseName. tableSpaceName[.owner]`, where `owner` is optional.
EXCLUDEIX

Use this option with VIA VOLUMES to exclude the specified indexes from the group. The index is specified by creator.ixname.

VIA INDEX

When creating a group by index, you can include any number of index names or wildcard patterns.

LIKE

When used in conjunction with VIA INDEX, the LIKE option builds a group using the indexes or index name pattern that you specified and copies the attributes of an existing group. The backup and recovery options of the existing group and optionally the authorizations and objects of the existing group are copied to the new group that you are creating. Enter the name of the existing group in the format creator2.name2. For more information, see “Copying groups” on page 510.

Note

■ Only one LIKE parameter is allowed in each CREATE command.

■ The LIKE parameter can be coded either before or after the VIA INDEX option.

■ To use LIKE, your primary or secondary AUTHID must have authority to open creator2.name2 (or else you must have SYSADM or system DBADM authority). The name of the new group that you are creating cannot be the same as creator2.name2.

RETAIN AUTH

This option is only valid in conjunction with the LIKE statement. RETAIN AUTH YES causes the new group to retain all of the authorizations granted in the existing group. RETAIN AUTH NO causes the new group to be created with no group authorizations.

RETAIN OBJECTS

This option is valid in conjunction with the LIKE statement. RETAIN OBJECTS YES (the default) causes the new group to retain all objects and object definitions currently within the existing group. RETAIN OBJECTS NO causes the new group to be created without including the objects from the existing group.
BYPART YES/NO

Use this option to add indexes to the group by partition. This option is not valid with nonpartitioned index spaces.

DSNUM

Use this option with VIA TABLESPACE to include a specified index by partition.

**Note**
You cannot use DSNUM with the BYPART option.

Valid values for DSNUM are:
- *n*—a partition number.
- *n:m*—a range of partitions starting with *n* and ending with *m*, where *n* and *m* are partition numbers.
- ALL—includes all partitions.

EXCLUDEPARTS

Use EXCLUDEPARTS to exclude partitions when creating a group by partition (using the BYPART YES option). See “CREATE GROUP” on page 530 for more information.

EXCLUDE

Enter index space names or wildcards patterns to exclude those spaces from the group. Use EXCLUDE in conjunction with EXCLUDEPARTS to exclude specific partitions from the group.

EXCLUDE DSNUM

Use this option with VIA TABLESPACE to exclude a specified index by partition.

**Note**
You cannot use EXCLUDE DSNUM with the BYPART option.

Valid values for EXCLUDE DSNUM are:
- *n*—a partition number.
- *n:m*—a range of partitions starting with *n* and ending with *m*, where *n* and *m* are partition numbers.
**VIA INDEXSPACE**

When creating a group by index space, you can include any number of index space names or wildcard patterns.

The index space is named by `databaseName. indexSpaceName`.

**VIA REPOSITORY**

Use the VIA REPOSITORY option to create a group containing the repository objects of RECOVERY MANAGER, R+/CHANGE ACCUM, and Log Master (if they are present in your system). RECOVERY MANAGER obtains the tables and indexes that make up the repository of each product from the plan listed in the options file of each product.

**VIA PACKAGE**

Use this option to specify creating groups by package. The package name is specify as `collid. package. version`. Following is sample JCL:

```plaintext
CREATE GROUP RDAJTR.BY_PKG1
REPLACE YES
DESCRIPTION 'CREATE BY PACKAGE'
VIA PACKAGE
   JTRC111D.ARM*.ARM1110*
   EXCLUDE
   JTRC111D.ARMQAAA.*
   INCLUDEI YES
   INCLUDEIX YES
   INCLUDELOB YES
   INCLUDEXML YES
   INCLUDEHISTORY YES
   BYPART NO
;  
```

**BYPART**

Use this option to add tables spaces to the group by partition. This option is not valid with nonpartitioned table spaces.

**DSNUM**

Use this option with VIA TABLESPACE to include a specified index space by partition.

---

**Note**

You cannot use DSNUM with the BYPART option.

---

Valid values for DSNUM are:

- `n`—a partition number.
n:m—a range of partitions starting with n and ending with m, where n and m are partition numbers.

- ALL—includes all partitions.

**EXCLUDE**

Enter the package name to exclude those spaces from the group

**EXCLUDE DSNUM**

Use this option with VIA TABLESPACE to exclude a specified index space by partition.

*Note*

You cannot use EXCLUDE DSNUM with the BYPART option.

Valid values for EXCLUDE DSNUM are:

- n—a partition number.
- n:m—a range of partitions starting with n and ending with m, where n and m are partition numbers.

**INCLUDERI**

Use this option to include all objects associated by referential integrity in the group.

**INCLUDEIX**

Use this option to include all associated indexes in the group.

**INCLUDELOB**

Use this option to add all objects that are associated by LOB columns with the objects in the group. Doing so ensures that both the base table space and the LOB table space are included in the group.

**INCLUDEXML**

Use this option to add all objects that are associated by XML columns with the objects in the group. Doing so ensures that all XML-related objects are included in the group and will be processed together.

**INCLUDEHISTORY**

Use this option for DB2 Version 10 and later to add all of the objects that are associated by a history (versioning) relationship to those specified in the group. The objects are also referred to as temporal objects and history objects.
VIA PLAN

Use this option to specify creating groups by plan names or patterns.

EXCLUDE

Enter the package name to exclude those spaces from the group

BYPART

Use this option to add tables spaces to the group by partition. This option is not valid with nonpartitioned table spaces.

INCLUDERI

Use this option to include all objects associated by referential integrity in the group.

INCLUDEIX

Use this option to include all associated indexes in the group.

INCLUDELOB

Use this option to add all objects that are associated by LOB columns with the objects in the group. Doing so ensures that both the base table space and the LOB table space are included in the group.

INCLUDEXML

Use this option to add all objects that are associated by XML columns with the objects in the group. Doing so ensures that all XML-related objects are included in the group and will be processed together.

INCLUDEHISTORY

Use this option for DB2 Version 10 and later to add all of the objects that are associated by a history (versioning) relationship to those specified in the group. The objects are also referred to as temporal objects and history objects.

VIA STOGROUP

Use this option to specify creating groups by storage group names or patterns. Following is sample JCL:

```
CREATE GROUP RDAJTR.BY_STG1
REPLACE YES
DESCRIPTION 'CREATE BY STORGROUP'
VIA STOGROUP
JTR*
EXCLUDE
JTRXBMFC
```
BYPART

Use this option to add objects to the group by partition. This option is not valid with nonpartitioned table spaces.

EXCLUDE

Enter storage group name or pattern to exclude those spaces from the group.

INCLUDERI

Use this option to include all objects associated by referential integrity in the group.

INCLUDEDIX

Use this option to include all associated indexes in the group.

INCLUDELOB

Use this option to add all objects that are associated by LOB columns with the objects in the group. Doing so ensures that both the base table space and the LOB table space are included in the group.

INCLUDEXML

Use this option to add all objects that are associated by XML columns with the objects in the group. Doing so ensures that all XML-related objects are included in the group and will be processed together.

INCLUDEHISTORY

Use this option for DB2 Version 10 and later to add all of the objects that are associated by a history (versioning) relationship to those specified in the group. The objects are also referred to as temporal objects and history objects.

VIA GROUP

When creating a group from another group or groups, you can include any number of group names or wildcard patterns.
**Note**

RECOVERY MANAGER restricts the name change of any group with a group definition to avoid calling itself or getting into a loop. This is validated when creating groups.

Following is an example of VIA GROUP syntax:

```
CREATE GROUP RDAJTR.BY_GRP20
REPLACE YES
DESCRIPTION 'TEST BGRP BY GROUP'
VIA GROUP
   "RDAJTR".*BY_TS*"
EXCLUDEGROUP
   "RDAJTR".*BY_TSXX"
```

**EXCLUDEGROUP**

Enter group name or pattern to exclude those groups from the new group.

**Syntax for creating volume groups: VVDS method**

The syntax in the following figure is used for creating volume or table space groups via the VVDS method.

**Figure 95: ARMBGRP — CREATE VOLUME GROUP**

```
CREATE VOLUME GROUP volume_name ;
```

**CREATE GROUP option descriptions (VVDS method)**

This section contains descriptions of the options used with the CREATE GROUP command when using the VVDS method.

**CREATE VOLUME GROUP**

This control statement is required when creating a volume group based on the VVDS.

Specify the volume ID of the DASD volume. (You cannot use a list of volumes.)

The group name is generated by RECOVERY MANAGER in the following format:
userid. Vvvvvyyyyyydddhmmm

The variables in the name are as follows:

- **userid** is the owner of the job
- **vvvvvv** is the volume name
- **yyyy** is the year
- **ddd** is the Julian day
- **hh** is the hour
- **mm** is the minute

**Note**
You must include a DD card in the JCL referencing the VVDS data set name.

You can repeat the CREATE VOLUME GROUP statement for all volumes for which you would like groups created.

**UPDATE GROUP syntax and option descriptions**

Use the UPDATE GROUP command to modify existing groups in batch mode. You can change the following types of options for the selected group or groups:

- **Recover Options**
  - General Recover Options
  - NGT Recover options
  - DSNUTILB recover options
  - Work File options
  - Output data set options

- **Copy Options**
  - General Copy Options
  - NGT Copy options
  - DSNUTILB copy options
  - NGT Recover OUTCOPY options
  - Output data set options
Syntax for updating group options

The syntax in the following figure is used to update the utility options of the specified group or groups.

You can remove all group-level values by entering RESET as the value for each group option. This causes the option to default to the value set at the subsystem level. If no subsystem value exists, the option defaults to the product level.

Figure 96: ARMBGRP control statement—UPDATE GROUP

```
UPDATE GROUP creator.name

RECOVER_OPTIONS  recover_options
COPY_OPTIONS      copy_options
```
Figure 97: ARMBGRP control statement—Recover options

```
RECOVER.Utility    AFRMAIN    CHECK.Utility    ACKMAIN
                  DSNUTILB    DSNUTILB
                  RESET      RESET

COPY.Utility      ACPMAIN    REGION_SIZE mmm - CHECK_PEND_ACTION
                  DSNUTILB
                  RESET

REDEFINE_VCAT.OBJ YES        COPY_AFTER_copyType
                   NO         YES
                   RESET     NO
                   RESET

DELETE_STOGROUP.OBJ YES        REUSE
                       NO         YES
                       RESET     NOSCR
                       RESET

MAX_CONCURRENT_JOBS nn        ALWAYS_REBUILD_INDEXES
                   RESET

INDEX_ALL          YES        MIRROR
                   NO         YES
                   RESET     NO
                   RESET

LIMIT_SYSCOPY.SEARCH nn        CHECKPOINT
                   RESET

EARLYRECALL        YES        MAXLOGS
                   NO         nn
                   RESET
                   RESET
```

Chapter 19 ARMBGRP—Group creation and maintenance 547
Figure 98: ARMBGRP control statement—Recover options

- **UNLOADKEYS_BUILDDINDEX**: YES/NO/RESET
- **OUTCOPY_BY_RECOVER**: NO/ASCODED/BYPART/RESET
- **REBUILD_INDEX_SHRLEVEL**: REFERENCE/CHANGE
- **OUTCOPY_MAXPRIM**: nnn/RESET
- **OUTCOPY_OUTSIZE**: YES/NO/RESET
- **ALTERNATE_RESOURCES**: NO/YES/AUTO/RESET
- **LOGSCAN**: YES/NO/RESET
- **ON_ERROR_CONTINUE**: nnnnnnn
- **DIAGNOSTIC_MESSAGES**: YES/NO/RESET
- **OPTIMIZE_FOR**: I/O (CPU, BALANCE, ELAPSED, RESET)
- **MAXSORT**: nnn/RESET
- **MAXLSORT**: nnn/RESET
- **KSORTSHARE**: YES/NO/RESET
- **MSGLEVEL**: STANDARD/SNAP/HW
  - OBJECT_SUMMARY
  - PLAN_SUMMARY
  - VSAM
  - RESET
Figure 99: ARMBGRP control statement—Recover options

```
recover_options

  ALTERNATE_COPY_copyType  n  ALTERNATE_ACT1  n  RESET

  ALTERNATE_ARC1  n  ALTERNATE_ARC2  n  RESET

  ALTERNATE_CHANGE_ACCUM_copyType  n  XBMID  xbmID  RESET

  DSNUTILB_SITE_TYPE  LOCAL  RECOVERY  DSNUTILB_SORTKEYS  YES
  NO
  RESET

  DSNUTILB_REPORT  YES  DSNUTILB_UPDATE  ALL
  NO
  RESET
  ACCESSPATH
  SPACE
  NONE
  RESET

  DSNUTILB_KEYCARD  YES  DATASET_SIZING  CATALOG
  NO  DEFAULTS
  RESET
  BMCSTTS
  RESET

  WORKFILE_MAX_PRIMARY  nn  WORKFILE_ALLOCATION_TYPE  CYL
  RESET
  TRACK
  RESET

  WORKFILE_WORK_UNIT  unit  WORKFILE_PRIMARY_ALLOC  nnnn
  RESET
  WORKFILE_WORKDDN
  YES
  NO
  RESET

  WORKFILE_SECONDARY_UNIT  nnnn
  RESET
```
Figure 100: ARMBGRP control statement—Recover options

```
recover_options

WORKFILE_WORKPREFIX prefix —— RECOVER_OUTPUT_copyType_DSN  dsn
  RESET

RECOVER_OUTPUT_copyType_DSN_FOR_REC_PLUS  dsn
  RESET

RECOVER_OUTPUT_copyType_UNIT  unit
  RESET

RECOVER_OUTPUT_copyType_TAPE
  YES
  NO
  RESET

RECOVER_OUTPUT_copyType_VOL_COUNT  nnn
  RESET

RECOVER_OUTPUT_copyType_CATALOG
  YES
  NO
  RESET

RECOVER_OUTPUT_copyType_MODEL_DSN
  YES
  NO
  RESET

RECOVER_OUTPUT_copyType_STACK
  YES
  NO
  CABINET*
  RESET

RECOVER_OUTPUT_copyType_MAX_PRIMARY  nnnn
  RESET

RECOVER_OUTPUT_copyType_ALLOC_TYPE
  CYL
  TRACK
  RESET

RECOVER_OUTPUT_copyType_PRIMARY_ALLOC  nnnn
  RESET

RECOVER_OUTPUT_copyType_SECONDARY_ALLOC  nnnn
  RESET

RECOVER_OUTPUT_copyType_SMS_STORAGE  nnn
  RESET
```

* For use with VIA TABLESPACE only
Figure 101: ARMBGRP control statement—Recover options

```
recover_options

- RECOVER_OUTPUT_copyType_SMS_DATA
  - sms_class
  - RESET

- RECOVER_OUTPUT_copyType_SMS_MGMT
  - sms_class
  - RESET

- RECOVER_OUTPUT_copyType_RETENTION
  - nnnn
  - RESET

- RECOVER_OUTPUT_copyType_EXPIRATION
  - yyyy/ddd
  - RESET

- RECOVER_OUTPUT_copyType_EATTR
  - OPT
  - NO
  - RESET
```
Figure 102: ARMBGRP control statement—Copy options

* Recovery Management solution only
Figure 103: ARMBGRP control statement—Copy options
Figure 104: ARMBGRP control statement—Copy options

```
copy_options

<table>
<thead>
<tr>
<th>CHECKERROR</th>
<th>nnn</th>
<th>CHECKTSLEVEL</th>
<th>n</th>
<th>RESYNC</th>
<th>YES</th>
<th>NO</th>
<th>RESET</th>
</tr>
</thead>
<tbody>
<tr>
<td>number</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACPGDG_DATA_SET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAX_TASKS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UNIT_COUNT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RUNSTATS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>REPORT_STATISTICS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UPDATE_DB2_CATALOG</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NONE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ACCESSPATH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPACE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UPDATE_BMCSTATS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SQUEEZE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUPPORT_FULL_COPY_DDS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUPPORT_OUTSIZE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUTSIZE_THRESHOLD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ON_ERROR_BADSTATUS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ON_DUPLICATES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ERROR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DELETE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ON_ERROR_EXISTS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>END</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SKIP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ON_ERROR_NOTSUPPORTED</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>END</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSNUTILB_FULL_COPY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>YES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUTO</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSNUTILB_SHRLEVEL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HW</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VSAM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSNUTILB_FULL_CONCURRENT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>END</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SKIP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESET</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
Figure 105: ARMBGRP control statement—Copy options

```
copy_options

DSNUTILB_FULL_INCREMENTAL_PERCENT nnn

RESET

DSNUTILB_GROUP YES

NO

RESET

DSNUTILB_COPY_PARALLEL YES

NO

RESET

OUTCOPY_EARLYRECALL YES

NO

RESET

OUTCOPY_EARLYCAT YES

NO

RESET

OUTCOPY_USEACCUM YES

NO

RESET

OUTCOPY_ANALYZE YES

NO

ONLY

RESET

OUTCOPY_MAXDRIVES nnn

RESET

OUTCOPY_SORDEVT device

TOLOGPOINT CURRENT

LASTQUIESCE

SPECIFIC_RBA

CURRENT

LASTSHUTDOWN

LASTCOMMONQUIESCE

RESET

TIMESTAMP

LOGMARK

OUTCOPY_MSGLEVEL STANDARD

OUTCOPY_COPY_copyType nnn

RESET

OBJECT_SUMMARY

PLAN_SUMMARY

RESET

OUTCOPY_ACT1 n

RESET

OUTCOPY_ACT2 n

RESET

OUTCOPY_ARC1 n

RESET

OUTCOPY_ARC2 n

RESET

OUTCOPY_CHANGE_ACCUM_copyType n

RESET
```
Figure 106: ARMBGRP control statement—Copy options

```
copy_options
    OUTCOPY_copyType_DSN  dsn  OUTCOPY_copyType_ENCIIPHER  YES  NO  RESET
    OUTCOPY_copyType_UNIT  unit  OUTCOPY_copyType_TAPE  YES  NO  RESET
    OUTCOPY_copyType_CATALOG  YES  OUTCOPY_copyType_EATTR  OPT  NO  RESET
    OUTCOPY_copyType_STACK  YES  OUTCOPY_copyType_MAX_PRIMARY  nnnn  RESET
    OUTCOPY_copyType_ALLOCATION_TYPE  CYL  OUTCOPY_copyType_RETENTION  nnnn  RESET
    OUTCOPY_copyType_EXPIRATION  yyyy/ddd  OUTCOPY_copyType_VOL_COUNT  nnn  RESET
    OUTCOPY_copyType_MODELS_DSN  dsn  OUTCOPY_copyTypePRIMARY_ALLOC  nnnn  RESET
    OUTCOPY_copyTypeSECONDARY_ALLOC  dsn  OUTCOPY_copyType_SMS STORAGE  sms_class  RESET
    OUTCOPY_copyType_SMS_DATA  sms_class  OUTCOPY_copyType_SMS_MGMT  sms_class  RESET
    OUTCOPY_copyType_ACP_GDG  dsn  OUTCOPY_copyType_MIGRATE  NO  HSM  ML2  RESET
```

*Recovery Management solution only
Figure 107: ARMBGRP control statement—Copy options

- FULLDDN_copyType DSN: $dsn$
  - RESET
  - FULLDDN_copyType ENCIPHER:
    - YES
    - NO
    - RESET

- FULLDDN_copyType UNIT: $unit$
  - RESET
  - FULLDDN_copyType TAPE:
    - YES
    - NO
    - RESET

- FULLDDN_copyType VOL_COUNT: $nnn$
  - RESET
  - FULLDDN_copyType CATALOG:
    - YES
    - NO
    - RESET

- FULLDDN_copyType EATTRR:
  - OPT
  - NO
  - RESET
  - FULLDDN_copyType STACK:
    - YES
    - NO
    - CABINET
    - RESET

- FULLDDN_copyType MODEL DSN: $dsn$
  - RESET
  - FULLDDN_copyType_MAX_PRIMARY: $nnnn$
  - RESET

- FULLDDN_copyType_ALLOCATION_TYPE:
  - CYL
  - TRACK
  - RESET

- FULLDDN_copyTypePRIMARY_ALLOC: $nnnn$
  - RESET

- FULLDDN_copyTypeSECONDARY_ALLOC: $nnnn$
  - RESET

- FULLDDN_copyType_SMS_STORAGE:
  - $sms_class$
  - RESET

- FULLDDN_copyType_SMS_DATA:
  - $sms_class$
  - FULLDDN_copyType_SMS_MGMT:
    - $sms_class$
    - RESET
Figure 108: ARMBGRP control statement—Copy options

```
copy_options
  └── FULLDDN_copyType_RETENTION [nnnn]  └── FULLDDN_copyType_EXPIRATION [yyyy/ddd]
      REST
  └── FULLDDN_copyType_ACP_GDG [dsn]  └──
      REST
  └── FULLDDN_copyType_MIGRATE [NO]  └── FULLDDN_copyType_DSSNAP [YES]
      [HSM]  └── [AUTO]
      [ML2]  └── [NO]
  └── BIGDDN_copyType_DSN [nnn]  └── BIGDDN_copyType_ENCIPHER [YES]
      REST  └── NO  └── RESET
  └── BIGDDN_copyType_UNIT [unit]  └── BIGDDN_copyType_TAPE └──
      REST
  └── BIGDDN_copyType_VOL_COUNT [nnn]  └── BIGDDN_copyType_EATTR [OPT]
      REST  └── NO  └── RESET
  └── BIGDDN_copyType_CATALOG [YES]  └── BIGDDN_copyType_STACK [YES]
      [NO]  └── [RESET]
      [RESET]
  └── BIGDDN_copyType_MODEL_DSN [dsn]  └── FULLDDN_copyType_MAX_PRIMARY [nnnn]
      REST  └──
  └── BIGDDN_copyType_ALLOCATION_TYPE [CYL]  └──
      [TRACK]  └──
      [RESET]  └──
  └── BIGDDN_copyType_PRIMARY_ALLOC [nnnn]  └──
      REST
```

* Recovery Management solution only
UPDATE GROUP option description

This section contains descriptions of the options used with the UPDATE GROUP command.

**Note**

You can remove any group-level value by entering RESET as the value for the recover or copy option. RESET causes the option to default to the value set at the subsystem level. If no subsystem value exists, the option defaults to the product level.
UPDATE GROUP

This option enables you to update group backup and recovery option settings in batch mode.

Use the format UPDATE GROUP creator.name.

RECOVER_OPTIONS

Specifies the recovery options that are to be updated. See the following topics for descriptions of the recovery options:

- “General recovery options” on page 847
- “NGT Recover recover options” on page 851
- “DSNUTILB recover options” on page 860
- “Work file recover options” on page 862
- “Output recover options” on page 864

COPY_OPTIONS

Specifies the copy options that are to be updated. See the following topics for descriptions of the copy options:

- “General copy options” on page 868
- “NGT Recover copy options” on page 870
- “DSNUTILB copy options” on page 881
- “NGT Recover OUTCOPY copy options” on page 883
- “Output copy options” on page 888
- “FULLDDN copy options” on page 892
- “BIGDDN copy options” on page 897

RENAME GROUP syntax and option descriptions

Use the RENAME GROUP command to change the name of existing groups in batch mode.

Syntax for renaming groups

The syntax in the following figure is used to rename the specified group or groups.
Figure 110: ARMBGRP control statement—RENAME GROUP

RENAMGROUP creator.name —→ NEWNAME creator.name — ;

RENAME GROUP option descriptions

This section contains a description of the RENAME GROUP command.

RENAME GROUP

Specify the name of the group or groups that you want to rename.

You can use wildcard characters to rename multiple groups. See "Wildcard support for RENAME GROUP" on page 511 for more information.

NEWNAME

Specify the new name for the group or groups. You can use wildcard characters to rename multiple groups. See "Wildcard support for RENAME GROUP" on page 511 for more information.

Note

A report of the groups processed by the rename command is written to the ARMRENAM file.

DELETE GROUP syntax and option descriptions

Use the DELETE GROUP command to delete existing groups in batch mode.

When you delete a group, the group definition is removed from the repository. The objects within the group are not affected.

Syntax for deleting groups

The syntax in the following figure is used to update the utility options of the specified group or groups.
DELETE GROUP option descriptions

This section contains a description of the DELETE GROUP command.

DELETE GROUP

This option enables you to delete one or more groups in batch mode.

You can use wildcard characters to delete multiple groups. See “Syntax rules” on page 102 for more information.

Note

When you delete a group, the group definition is removed from the repository. The objects within the group are not affected.

REPORT GROUP syntax and option descriptions

Use the REPORT GROUP command to generate reports on the characteristics of one or more groups in batch mode.
Syntax for reporting on groups

The syntax in the following figure is used to report on the details of the specified group or groups.

Figure 112: ARMBGRP control statement—REPORT GROUP

REPORT GROUP option descriptions

This section contains a description of options used with the REPORT GROUP command.

REPORT GROUP

This option enables you to generate reports on the characteristics of one or more groups in batch mode.

You can use wildcard characters to report on multiple groups. For more information, see “Syntax rules” on page 102.

OBJECTS

Reports all objects contained within the groups that you specify and sends the output to the ARMOBJS file. This information includes the mirrored volume summary, if the groups are mirrored and if you are using the BMC Recovery Management for DB2 solution.

AUTHORIZATION

Reports on the authorizations for all groups that you specify and sends the output to the ARMAUTH file.
DEFINITION

Reports the definition of each group that you specify and sends the output to the ARMDEFN file.

RECOVER OPTIONS

Reports the recovery options for each group that you specify and sends the output to the ARMRCVR file.

COPY OPTIONS

Reports the copy options for each group that you specify and sends the output to the ARMCOPY file.

QUERY syntax and option descriptions

Use the QUERY option to issue SQL queries for information relating to RECOVERY MANAGER groups, such as:

- Objects not contained in any group
- Objects contained in more than one group
- List of all groups and creators
- List of all objects within each group
- List of all groups and authorizations
- List of all BMC tables included in the RECOVERY MANAGER plan
- List the groups that contain certain table spaces and indexes
Syntax for querying groups

The syntax in the following figure is used to issue SQL queries regarding the specified criteria.

Figure 113: ARMBGRP control statement—QUERY option descriptions

This topic contains a description of options used with the QUERY command.

QUERY

This option enables you to issue SQL queries for information relating to RECOVERY MANAGER groups in general. All output is written to the ARMQUERY file.

MISSING OBJECTS

Lists all objects in the subsystem that are not contained within any groups.
**Note**
Groups built by ARMBGPS for full subsystem recovery, without the INCLUDE parameter, are ignored by this query.

INDEXES

Indicates whether to include indexes in the MISSING OBJECTS query.

**DBNAME databaseName**

Indicates whether to include objects for a specified database in the MISSING OBJECTS query. DBNAME allows a list of database names separated by commas and allows wildcard values in the database names.

MULTIPLE OBJECTS

Lists any objects in the subsystem that are contained in more than one group.

**Note**
Groups built by ARMBGPS for full subsystem recovery, without the INCLUDE parameter, are ignored by this query.

**DBNAME databaseName**

Indicates whether to include objects for a specified database in the MISSING OBJECTS query. DBNAME allows a list of database names separated by commas and allows wildcard values in the database names.

GROUP INFO

Lists all groups, their creators, and other related information.

*.:* is the default value.

**GROUP INFO RDAJ*.:***

Lists all groups, their creators, and other related information.

This option allows wildcard values and delimited group names.

GROUP OBJECTS

Lists all groups and the objects contained in each group.

The default value is *.:*.

**GROUP OBJECTS RMD.***

Lists all groups and the objects contained in each group.
This option allows wildcard values and delimited group names.

**GROUP AUTH**

Lists all groups with their associated authorizations.

The default value is *.*

**GROUP AUTH PUBLIC.ACP***

Lists all groups with their associated authorizations.

This option allows wildcard values and delimited group names.

**BMCTABLES**

Lists the names of all BMC tables used by the RECOVERY MANAGER plan.

**OBJECTS**

Lists the groups that include certain table spaces and indexes.

**DBNAME databaseName**

Indicates whether to include objects for a specified database in the OBJECTS query. DBNAME allows a list of database names separated by commas and allows wildcard values in the database names.

**INDEXES**

Indicates whether to include indexes in the OBJECTS query

### EXPORT GROUP syntax and option descriptions

Use the EXPORT GROUP command to convert group definitions into CREATE GROUP syntax and write them to a specified data set.

You can then either keep the data set as a backup or use it to migrate the converted definitions to another DB2 subsystem.

For more information about the migration process, see “MIGRATE_TO_SSID syntax and option descriptions” on page 568.
Syntax for exporting groups

The following syntax converts group definitions to CREATE GROUP syntax and writes them to a specified data set:

```
EXPORT GROUP creator.name ;
```

EXPORT GROUP command

This command enables you to convert group definitions to CREATE GROUP syntax and write them to a specified data set.

The EXPORT GROUP command is followed by a group name in the format `creator.name`. The group name can be explicit or contain wildcards to process all groups that match the pattern that you specified.

MIGRATE_TO_SSID syntax and option descriptions

Use the MIGRATE_TO_SSID command to migrate exported group definitions to another DB2 subsystem.

To use this command, you must first export the group definitions. For more information about the export process, see “EXPORT GROUP syntax and option descriptions” on page 567.

Syntax for migrating groups

The syntax in the following figure is used to migrate group definitions from a specified data set to a DB2 subsystem.

```
MIGRATE_TO_SSID ssid ;
```
MIGRATE_TO_SSID option descriptions

This command enables you to migrate group definitions that you have exported by using the EXPORT GROUP command to a target DB2 subsystem.

The MIGRATE_TO_SSID command is followed by an SSID that identifies the target DB2 subsystem. The SSID must be different from that in the ARMBGRP EXEC PARM.

The target subsystem must comply with the following guidelines:

- The target subsystem must be active on the same LPAR as that on which ARMBGRP operates.
- The DB2 version of the target subsystem must be supported by RECOVERY MANAGER.
- RECOVERY MANAGER must be installed on the target subsystem.

To use this command, you must first export the group definitions to a specified data set. For more information, see “EXPORT GROUP syntax and option descriptions” on page 567.

RECOVER SCOPE syntax and option descriptions

Use the RECOVER_SCOPE option on the NGT Recover product’s OPTION command to recover objects based on specified criteria, thereby defining the scope of the recovery.

ALL

NGT Recover recovers all specified indexes. It does not bypass any spaces that have changed since the specified recovery point.

Note

ALL overrides all other options. For example, NGT Recover changes RECOVER_SCOPE (RECP, ALL, LPL) to RECOVER_SCOPE (ALL).

UPDT

NGT Recover recovers only spaces that have changed since a specified recovery point.
NGT Recover recovers objects based on their specified statuses.

The following are valid statuses:
- ARDBP
- AREO
- ACHKP
- AREST
- AUXW
- CHKP
- COPY
- GRECP
- ICOPY
- LPL
- LSTOP
- PRO
- LSTOP
- PRO
- PSRBD
- RBDP
- RECP
- REFP
- RELDP
- REORP
- RESTP
- RO
- RREPL
- RW
- STOP
- STOPE
- STOPP
- UT
- UTRO
- UTRW
- UTUT
- WEPR

**Note**

NGT Recover filters out duplicate statuses.
Syntax for `RECOVER_SCOPE`

The following syntax bypasses objects based on specified criteria.

```
| OPTIONS | RECOVER_SCOPE | ( | ALL  |
|         |              |   | UPDT |
|         |              |   | status |
```

`RECOVER_SCOPE` option descriptions

This option instructs RECOVERY MANAGER to recover objects based on specified criteria, thereby defining the scope of the recovery.

```
RECOVER_SCOPE
```

This option instructs RECOVERY MANAGER to recover objects based on specified criteria, thereby defining the scope of the recovery.

REBUILD_SCOPE syntax and option descriptions

If you specify the REBUILD_SCOPE option on the OPTIONS command in NGT Recover, all of the REBUILD command statements in the SYSIN data set instruct RECOVERY MANAGER to rebuild objects based on specified criteria, thereby defining the scope of the rebuild organization.

```
ALL
```

RECOVERY MANAGER recovers all specified indexes. It does not bypass any spaces that have changed since the specified recovery point.

Note

ALL overrides all other options. For example, RECOVERY MANAGER changes REBUILD_SCOPE (RECP,ALL,LPL) to REBUILD_SCOPE (ALL).

```
PEND
```

RECOVERY MANAGER rebuilds only indexes that are in a RBDP, PSRBD, RBDP*, RECP, or AREO* status.
**Note**

PEND overrides all options except for ALL. For example, RECOVERY MANAGER changes REBUILD_SCOPE (GRECP, PEND, RECP) to REBUILD_SCOPE (PEND).

**status1, status2, ...**

The following are valid status values:

- ARDBP
- AREO
- ACHKP
- AREST
- AUXW
- CHKP
- COPY
- GRECP
- ICOPY
- LPL
- LSTOP
- PRO
- LSTOP
- PRO
- PSRBD
- RBDP
- RECP
- REFP
- RELDP
- REORP
- RESTP
- RO
- RREPL
- RW
- STOP
- STOPE
- STOPP
- UT
- UTR
- UTRW
- UTUT
- WEPR
Note
RECOVERY MANAGER filters out duplicate statuses.

Syntax for `REBUILD_SCOPE`

The following syntax bypasses objects based on specified criteria.

```
REBUILD_SCOPE = ( ALL, PEND, status1, status2, ... )
```

`REBUILD_SCOPE` option descriptions

This topic contains a description of the `REBUILD_SCOPE` option.

`REBUILD_SCOPE`

This option instructs RECOVERY MANAGER to rebuild objects based on specified criteria, thereby defining the scope of the rebuild organization.

Sample JCL and output

This section includes samples of JCL and output for ARMBGRP.

`CREATE GROUP`

The following figure shows sample JCL for using the catalog search method to create table space groups.

**Figure 114: CREATE GROUP sample JCL**

```
//ARMBGRPc EXEC PGM=ARMBGRP,
// PARM='DEC2,ARMOPTS=ARM$OPTS',
// REGION=QM
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARMERROR DD SYSOUT=*  
```
Figure 115 on page 574 shows sample output created when using the catalog search method to create table space groups.

** Figure 115: CREATE GROUP sample output **
Figure 116 on page 575 shows the output when creating a group using the VIA EXCEPTION parameter. The Group Type in the Group Objects Report will be RP, BA (BGPS) or BG (BGPS). Group Type will be blank for all other types of groups.

Figure 116: CREATE GROUP VIA EXCEPTION sample output

** RECOVERY MANAGER FOR DB2 V11.1.00 - BATCH GROUP PROCESSING 08/07/2012 10:10:1
(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5762234

BMC802231 MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC802231 SOLUTION COMMON CODE V11.1.00
BMC802231 MAINT: NO SCC PTFS APPLIED
BMC803091 CONNECTED TO DB2 SSID = DECI VERSION 910
CREATE GROUP VIA SQL

The following figure shows the JCL for creating a group with user-defined SQL.

Figure 117: CREATE GROUP VIA SQL sample JCL

```sql
/* CREATION OF ARMQA.SQLGRP00 */
SET CURRENT SQLID = ARMQA ;
```

```sql
SELECT 'TS', DBNAME, TSNAME, PARTITION FROM SYSIBM.SYSTABLEPART
WHERE DBNAME LIKE 'ARMMULT%'
  AND TSNAME <> 'TS080NL'
  AND TSNAME <> 'TS256NL'
/
```

```
//ARMRP1 EXEC PGM=ARMBGRP,
//             PARM='&SSID,ARMOPTS=ARM$OPTS',
//             REGION=0M,TIME=1440
//          INCLUDE MEMBER=ARMBGRP2
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
//ARMMGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMGS)
//ARMIN DD *
//ARMERROR DD SYSOUT=*
//ARMPRINT DD SYSOUT=*
//ARMSQL DD *
```

```sql
SELECT 'TS', DBNAME, TSNAME, PARTITION FROM SYSIBM.SYSTABLEPART
WHERE DBNAME LIKE 'ARMMULT%'
  AND TSNAME <> 'TS080NL'
  AND TSNAME <> 'TS256NL'
/*
```

```sql
//ARMRP1 EXEC PGM=ARMBGRP,
//             PARM='&SSID,ARMOPTS=ARM$OPTS',
//             REGION=0M,TIME=1440
//          INCLUDE MEMBER=ARMBGRP2
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
//ARMMGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMGS)
//ARMIN DD *
//ARMERROR DD SYSOUT=*
//ARMPRINT DD SYSOUT=*
//ARMSQL DD *
```

SELECT 'TS', DBNAME, TSNAME, PARTITION FROM SYSIBM.SYSTABLEPART
WHERE DBNAME LIKE 'ARMMULT%'
  AND TSNAME <> 'TS080NL'
  AND TSNAME <> 'TS256NL'
/*
*/

```
//ARMRP1 EXEC PGM=ARMBGRP,
//             PARM='&SSID,ARMOPTS=ARM$OPTS',
//             REGION=0M,TIME=1440
//          INCLUDE MEMBER=ARMBGRP2
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
//ARMMGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMGS)
//ARMIN DD *
//ARMERROR DD SYSOUT=*
//ARMPRINT DD SYSOUT=*
//ARMSQL DD *
```
CREATE GROUP ARMQA.SQLGRP00
REPLACE YES
VIA SQL
INCLUDERI NO
INCLUDEIX YES
;

Figure 118 on page 578 shows the output created when creating a group with user-defined SQL.

**Figure 118: CREATE GROUP VIA SQL sample output**
CREATE GROUP (Volume, VVDS)

The following figure shows the JCL for using the VVDS method to create volume groups.

**Figure 119: CREATE GROUP (Volume, VVDS method) JCL**

```
//ARMCRGVL EXEC PGM=ARMBGRP,  
 //                         PARM='DEC2,ARMOPTS=ARM$OPTS',  
 //                         REGION=0M  
 //STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS  
 // DD DISP=SHR,DSN=DSNEXIT  
 // DD DISP=SHR,DSN=DSNLOAD  
 //ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)  
 //ARMERROR DD SYSOUT=  
 //ARMPRINT DD SYSOUT=  
 //VSMSRLS DD DISP=SHR,DSN=SYS1.VVDS.VSMSRLS  
 //ARMIN DD *  
 SET CURRENT SQLID = ARMQA;  
 CREATE VOLUME GROUP SMSRLS
```

**Figure 120 on page 580** shows the output created when using the VVDS method to create volume groups.

**Figure 120: CREATE GROUP (Volume, VVDS method) output**

```
** RECOVERY MANAGER FOR DB2 V11.1.00 - BATCH GROUP PROCESSING 08/10/2012 14:24:16 **  
(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.  
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676  
BMC80223I MAINT: NO RECOVERY MANAGER PTFS APPLIED  
BMC80223I SOLUTION COMMON CODE V11.1.00  
BMC80223I MAINT: BPJ0197 BPJ0215 BPJ0219  
BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910  
SET CURRENT SOLID = ARMQA;  
BMC80570I COMMAND COMPLETE RC = 0  
CREATE VOLUME GROUP SMSRLS  
:  
BMC80587I PROCESSING VOLUME SMSRLS FOR GROUP ARMQA.VSMSRLS20060781503  
BMC80589I TABLE SPACE ARMBRP01.TRPI9P DSNUM 1 INCLUDED IN GROUP  
BMC80589I TABLE SPACE ARMBRP01.TRPI9P DSNUM 2 INCLUDED IN GROUP  
BMC80589I TABLE SPACE ARMBRP01.TRPI9P DSNUM 3 INCLUDED IN GROUP  
BMC80589I TABLE SPACE ARMBRP01.TRPI9P DSNUM 4 INCLUDED IN GROUP
```
CREATE GROUP (volume, catalog search method)

The following figure shows the JCL for using the catalog search method to create a volume group for a single volume.

**Figure 121: CREATE GROUP (volume, catalog search method) JCL**

```plaintext
//ARMCRL2 EXEC PGM=ARMGRP,
//                PARM='DEC2,ARMOPTS=ARM$OPTS',
//                REGION=0M
//STEPLIB  DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//         DD DISP=SHR,DSN=DSNEXIT
//         DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS  DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARMERROR DD SYSOUT=*  
//ARMPRINT DD SYSOUT=*  
//ARMDEFN  DD SYSOUT=*  
//ARMIN    DD *
CREATE GROUP ARMQA."SAMPLE_VOLUME"
REPLACE YES
VIA VOLUMES ( AUSS21 )
```

**Figure 122 on page 581** shows the output for volume group creation.

**Figure 122: CREATE GROUP (volume, catalog search method) output**

```plaintext
** RECOVERY MANAGER FOR DB2 V11.1.00 - BATCH GROUP PROCESSING 08/26/2012 13:04:26 **
(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
BMC80223I MAINT:  NO RECOVERY MANAGER PTFs APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT:  BPJ0197  BPJ0215  BPJ0219
BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910

CREATE GROUP ARMQA."SAMPLE_VOLUME"
REPLACE YES
VIA VOLUMES ( AUSS21 )

BMC80590I INDEX ARMTPL1.IXLBSC1 DSNUM 0 INCLUDED IN GROUP
BMC80590I INDEX ARMTN1.IXPLBC1 DSNUM 0 INCLUDED IN GROUP
BMC80590I INDEX ARMTN1.IXPLBC1 DSNUM 0 INCLUDED IN GROUP
BMC80590I INDEX ARMTN1.IXPLBC1 DSNUM 0 INCLUDED IN GROUP
```
UPDATE GROUP

The following figure shows the JCL for updating the utility options for a group.

**Figure 123: UPDATE GROUP sample JCL**

```sql
//ARMBGRPU EXEC PGM=ARMBGRP,
// PARM="DEC2,ARMOPTS=ARM$OPTS",
// REGION=0M
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//DD DISP=SHR,DSN=DSNEXIT
//DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
```
Figure 124 on page 583 the output created when updating group utility options.

Figure 124: UPDATE GROUP sample output

** RECOVERY MANAGER FOR DB2 V11.2.00 - BATCH GROUP PROCESSING 08/11/2014 08:53:41 **
** BMC802201 RECOVERY MANAGEMENT FOR DB2 V11.2.00 **

(c) COPYRIGHT 1994-2015 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884

BMC802231 MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC802231 SOLUTION COMMON CODE V11.2.00
BMC802231 MAINT: BPJ0197 BPJ0215 BPJ0219

BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910

UPDATE GROUP RMD.TSRG40A
RECOVER_OPTIONS
RECOVER_UTILITY AFRMAIN
CHECK_UTILITY DSNUTILB
COPY_UTILITY DSNUTILB
CHECK_PEND_ACTION CHECK
REPAIR_COPY_PEND NO
REDEFINE_VCAT_OBJ YES
COPY_AFTER_LP NO
COPY_AFTER_LB NO
COPY_AFTER_RP NO
COPY_AFTER_RB NO
DELETE_STOGROUP_OBJ NO
REUSE YES
MAX_CONCURRENT_JOBS 01
ALWAYS_REBUILD_INDEXES NO
INDEX_ALL NO
UNLOADKEYS_BUIDINDEX NO
OUTCOPY_BY_RECOVER BYPART
ALTERNATE_RESOURCES NO
ALTERNATE_COPY_SB 0
ALTERNATE_COPY_FC 0
ALTERNATE_COPY_LP 0
ALTERNATE_COPY_LB 0
ALTERNATE_COPY_RP 0
ALTERNATE_COPY_RB 0
ALTERNATE_ACT1 0
ALTERNATE_ACT2 0
ALTERNATE_ARC1 0
ALTERNATE_ARC2 0
XBMID X8MB
DSNUTILB_SITE_TYPE LOCAL
DATASET_SIZING CATALOG
WORKFILE_MAX_PRIMARY 3
WORKFILE_ALLOCATION_TYPE CYL
MIRROR NO
RENAME GROUP

The following figure shows the JCL for renaming a set of groups.

Figure 125: RENAME GROUP sample JCL

```
//ARMRENAM EXEC PGM=ARMBGRP,
/// PARM='DEC2,ARMOPTS=ARM$OPTS',
/// REGION=OM
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
/// DD DISP=SHR,DSN=DSNEXIT
/// DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=PRODUCT_CNTL.LIBS(ARMMSGS)
//ARMERROR DD SYSOUT=* 
//ARMPRINT DD SYSOUT=* 
//ARMDEFN DD SYSOUT=* 
//ARMRENAME DD SYSOUT=* 
//ARMIN DD *
SET CURRENT SQLID = ARMQA : 
RENAME GROUP ARMQA.SAMPLE_RESP NEWNAME ARMQA.SAMPLE_NEW ;
```

Figure 126 on page 584 shows the ARMRENAM DD output created when renaming a group.

Figure 126: RENAME GROUP - sample output

```
** RECOVERY MANAGER FOR DB2 V11.1.00 - BATCH GROUP PROCESSING 02/19/2012 15:09:46 **
(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676

BMC802231 MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC802231 SOLUTION COMMON CODE V11.1.00
BMC802231 MAINT: BPJ0197 BPJ0215 BPJ0219

BMC803091 CONNECTED TO DB2 SSID = DEC2 VERSION 910
SET CURRENT SQLID = ARMQA : 
BMC805701 COMMAND COMPLETE RC = 0
   RENAME GROUP ARMQA.SAMPLE_RESP NEWNAME ARMQA.SAMPLE_NEW
BMC805911 PROCESSING GROUP ARMQA.SAMPLE_RESP
BMC805701 COMMAND COMPLETE RC = 0
```
DELETE GROUP

The following figure shows the JCL for deleting the group called ARMQA.SAMPLE_NEW.

**Figure 127: DELETE GROUP sample JCL**

```plaintext
//ARMDELET EXEC PGM=ARMBGRP,
  //      PARM='DEC2,ARMOPTS=ARM$OPTS',
  //      REGION=0M
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//     DD DISP=SHR,DSN=DSNEXIT
//     DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARMERROR DD SYSOUT=* 
//ARMPRINT DD SYSOUT=* 
//ARMDEFN DD SYSOUT=* 
//ARMRENAM DD SYSOUT=* 
//ARMIN DD *
SET CURRENT SQLID = ARMQA ;
DELETE GROUP ARMQA.SAMPLE_NEW ;
```

**Figure 128 on page 585** shows the output for deleting the group called ARMQA.SAMPLE_NEW.

**Figure 128: DELETE GROUP sample Output**

```plaintext
**  RECOVERY MANAGER FOR DB2 V11.1.00 - BATCH GROUP PROCESSING 02/19/2012 15:09:48 **
(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676

BMC80223I MAINT:  NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT:  BPJ0197 BPJ0215 BPJ0219
BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910

SET CURRENT SQLID = ARMQA ;
BMC80570I COMMAND COMPLETE RC = 0

DELETE GROUP ARMQA.SAMPLE_NEW :
BMC80591I PROCESSING GROUP ARMQA.SAMPLE_NEW
BMC80570I COMMAND COMPLETE RC = 0

BMC80531I 2 COMMANDS WERE PROCESSED
BMC80571I PROGRAM COMPLETE RC = 0
```

REPORT GROUP

The following figure shows the JCL for creating a report that shows the recover and copy options for a group.
Figure 129: REPORT GROUP sample JCL

```plaintext
//ARMBGRP EXEC PGM=ARMBGRP,
//          PARM='DEDL,ARMOPTS=ARM$OPTS',
//          REGION=0M
//STEPLIB   DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS   DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARMERROR  DD SYSOUT=*  
//ARMPRINT  DD SYSOUT=*  
//ARMCOPY   DD SYSOUT=*  
//ARMRCVR   DD SYSOUT=*  
//ARMDEFN   DD SYSOUT=*  
//ARMIN     DD *
SET CURRENT SQLID = ARMQA;
REPORT GROUP ARMQA.SAMPLE_RESP
RECOVER OPTIONS
COPY OPTIONS
OBJECTS
DEFINITION
```

Figure 130 on page 586 shows the output for creating a report that shows the recover and copy options for a group.

Figure 130: REPORT GROUP sample output

```
** RECOVERY MANAGER FOR DB2 V11.2.00 - BATCH GROUP PROCESSING 03/02/2014 13:18:07 **
(c) COPYRIGHT 1994-2015 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676

BMC80223I MAINT:  NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.2.00
BMC80223I MAINT:  BPJ0197  BPJ0215  BPJ0219

BMC80309I CONNECTED TO DB2 SSID = DEDL VERSION 910
SET CURRENT SQLID = ARMQA;
BMC80570I COMMAND COMPLETE RC = 0
REPORT GROUP ARMQA.SAMPLE_RESP
RECOVER OPTIONS
COPY OPTIONS
OBJECTS
DEFINITION

BMC80591I PROCESSING GROUP ARMQA.SAMPLE_RESP
BMC80570I COMMAND COMPLETE RC = 0
2 COMMANDS WERE PROCESSED
BMC80531I PROGRAM COMPLETE RC = 0
** RECOVERY MANAGER FOR DB2 V11.2.00 - GROUP COPY OPTIONS REPORT 03/02/2014 13:18:08 **

* ** *** Group Name ARMQA.SAMPLE_RESP *** ** *
GENERAL BACKUP OPTIONS:
Copy Utility: DB2 Copy (DSNUTILB) Quiesce before: Yes
Quiesce after: Yes  Quiesce group: No
Quiesce write: Yes
Copy Index Spaces: No  Copy Index Size Threshold: 0 M
Copy all indexes in a table space: No  Output copy type: LP LB
Scope Setting: SCOPE Undefined
```
NGT Copy SPECIFIC OPTIONS:
Full copy: Yes                                Empty: Yes
Cumulative: Yes                               Keep previous: Yes
Readtype: Random                              Auto read percent: Not specified
Maximum incremertals: Not specified           Incremental percent: Not specified
Full percent: Not specified                   Minimum pages: Not specified
Day of the week: Not specified                NACTIVE: Not specified
DSSNAP: Not specified                         Shrlevel: Reference
XBMID: Not specified                          Group: No
Separate by partition: No                     Restartmod: Not specified
Checktslevel: Not specified                   Checkerror: Not specified
Resync: Not specified                         Squeeze: Not specified
Maximum Tape Tasks: Not specified             # of read/write buffers: Not specified
Unit Count: Not specified                     Maximum Total Tasks: Not specified
Report Statistics: No                         RUNSTATS: No
Update BMCSTATS Table: No                    Support DB2 catalog: All
ON ERROR BADSTATUS: Not specified             Support OUTSIZE option (BIGDDN): No
ON ERROR ICEXISTS: Not specified              OUTSIZE threshold for BIGDDN: Not specified
ON ERROR NOTSUPPORTED: Not specified          ON DUPLICATEDS: Not specified
ACPGDG data set: Not specified                ON ERROR NOTSUPPORTED: Not specified
After Initialize Phase: Continue              Compess: Not specified
Start message: Not specified                  System backup:
Use COPY IMAGECOPY to make output type: Not specified

DB2 COPY (DSNUTILB) OPTIONS:
Full copy: Yes                                Concurrent/DFSMS: No
Changelimit Incremental percent: Not specified
Changelimit Full percent: Not specified       Group: No
Shrlevel: Reference                           Max parallel objects: 2
Copy objects in parallel: Yes                 No

NGT Recover OUTCOPY OPTIONS:
EARLYRECALL: Yes                              EARLYCAT: Yes
ANALYZE: Yes                                  MAXDRIVES: 0
SORTDEVT: Not specified                       MSGLEVEL: Object Summary
MAXPRIM:
900                                             AUTOSIZE: Yes
TOLOGPOINT: Current                           TOLOGPOINT: Current
Alternate Resources:                          Alternate Resources:
Image copies:                                 Image copies:
Local primary: Not specified                  Local backup: Not specified
Recovery primary: Not specified               Recovery backup: Not specified
IBM Flashcopy: Not specified                  System backup:
Not specified                                 Not specified
Logs:
Active log copy 1: Not specified              Active log copy 2: Not specified
Archive log copy 1: Not specified             Archive log copy 2: Not specified
Change accums:
Local primary: Not specified                  Local backup: Not specified
Recovery primary: Not specified               Recovery backup: Not specified

OUTPUT DATA SET OPTION DESCRIPTIONS:
Local Primary Copy Options:
Data set name: BMCARM.DBB.&TYPES.DNUM.DDAY.TIME
Model data set name: BMCARM.REGR.MODEDeD.CB.LP
Unit: WORK                                     Tape: No
Volume count: Not specified                    Catalog: Yes
Stack: No                                      Retention period: 2
Max Primary allocation: 2                     Migrate: NO
Primary allocation: 22                        Secondary allocation: 42
Allocation type: Track                         SMS storage class: ARMBBRP1
SMS data class: COPYCLAS                       SMS management class: MIG14
Expiration date: 1999/000                      ACP GDG: Not specified
Encryption: Not specified
Local Backup Copy Options:
Data set name: BMCARM.&DB.&TS.&TYPE&DSNUM.D&DAY.T&TIME
Model data set name: Not specified
Unit: SYSALLDA
Volume count: Not specified
Stack: No
Max Primary allocation: 0
Primary allocation: 10
Allocation type: Cylinder
SMS data class: Not specified
Expiration date: 1999/000
Encryption: Not specified

** RECOVERY MANAGER FOR DB2 V11.2.00 - GROUP RECOVER OPTIONS REPORT 03/02/2014 13:18:08 **

* ** *** Group Name_ ARMQA.SAMPLE_RESP *** ** *

GENERAL RECOVERY OPTIONS:
Rcvr Util: DB2 Recover (DSNUTILB)
Copy Util: DB2 Copy (DSNUTILB)
Unloadkeys/Buildindex: No
Max concurrent jobs: 1
What action when Check Pending: Check
Make copies after recovery for: LP

NGT Recover OPTIONS:
CHECKPOINT: Not specified
EARLYCAT: Yes
Unloadkeys/Buildindex: No
Dynamic sortworks: No
XBMID: Not specified
MAXPRIM: 900
Alternate Resources: No
Max Key Sort: Not specified

DB2 RECOVER OPTIONS:
SORTKEYS: No
REPORT: Yes
KEYCARD: No
WORK FILE OPTION DESCRIPTIONS:
Work unit: SYSALLDA
Allocation type: Cylinder
Primary allocation: 22
Work prefix: RMD.QA

OUTPUT DATA SET OPTION DESCRIPTIONS:
Local Primary Copy Options:
Data set name: BMCARM.&DB.&TS.&TYPE&DSNUM.D&DAY.T&TIME
Data set name(for NGT Recover by part): &USER.&DB.&TS.&TYPE&DATE
Model data set name(DCB): Not specified
Expiration date: 1999/000
Unit: SYSALLDA
Volume Count: Not specified
Stack: No
Max Primary allocation: 3
Primary allocation: 22
Secondary allocation: 42
SMS storage class: ARMBGPR1
SMS management class: MIG14
Group Name    ARMQA.SAMPLE_RESP
Date Created  2014-03-02-13.11.17
Created by RMGRUS
Date Updated  2014-03-02-13.18.06
Updated by RMGRUS
Description   SAMPLE GROUP RESP
Create product ARM
Group type RP
Number objects 25

TYPE T  DSNUM  0  INSTANCE 1  CLONED N
BMARM.BMCARMCR
TYPE T  DSNUM  0  INSTANCE 1  CLONED N Object Level Options
BMARM.BMCARMGA
TYPE T  DSNUM  0  INSTANCE 1  CLONED N
BMARM.BMCARMGC
TYPE T  DSNUM  0  INSTANCE 1  CLONED N
BMARM.BMCARMGD
TYPE T  DSNUM  0  INSTANCE 1  CLONED N
BMARM.BMCARMGF
TYPE T  DSNUM  0  INSTANCE 1  CLONED N
BMARM.BMCARMGO
TYPE T  DSNUM  0  INSTANCE 1  CLONED N
BMARM.BMCARMGP
TYPE T  DSNUM  0  INSTANCE 1  CLONED N Object Level Options
BMARM.BMCARMGS
TYPE T  DSNUM  0  INSTANCE 1  CLONED N
BMARM.BMCARMOP
TYPE T  DSNUM  0  INSTANCE 1  CLONED N
BMARM.BMCARMGF
TYPE T  DSNUM  0  INSTANCE 1  CLONED N
BMARM.BMCESTM
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.CRX1
TYPE I  DSNUM  0  INSTANCE 1  CLONED N Object Level Options
BMARM.GAUT
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.GCOM
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.GDEF
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.GPRO
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.IX IX_PART
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.IX IXP_SORT
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.IX JOB
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.IX KEYSORT
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.IX PHASE
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.IX TS
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.IX TS PART
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.IX TSP SORT
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
BMARM.IX UTIL RUN
TYPE I  DSNUM  0  INSTANCE 1  CLONED N
Figure 131 on page 590 shows the JCL used to see the definition for a group defined with dynamic SQL.

**Figure 131: REPORT GROUP sample JCL to see the SQL that defines a group**

```
//ARMBGRP EXEC PGM=ARMBGRP,
//          PARM='DEDL,ARMOPTS=ARM$OPTS',
//          REGION=0M
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARMERROR DD SYSOUT=*  
//ARMPRINT DD SYSOUT=*  
//ARMCOPY DD SYSOUT=*  
//ARMRCVR DD SYSOUT=*  
//ARMOBJS DD SYSOUT=*  
//ARMDEFN DD SYSOUT=*  
//ARMIN DD *
SET CURRENT SQLID = ARMQA;
REPORT GROUP PUBLIC.Q1545928
DEFINITION
```

Figure 132 on page 590 shows the output displaying the SQL used to define a group.

**Figure 132: REPORT GROUP sample output displaying SQL for a user-defined group**

```
** RECOVERY MANAGER FOR DB2 V11.2.00 - GROUP DEFINITION REPORT 05/04/2014**
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.2.00**

Group Name    PUBLIC.Q1545928
Type      Dsnum  Indexes   RI    By Part   LOBs   XML   History
Clone
SQL      0      Y         Y     Y         B      O     N         N
SELECT 'TS', DBNAME, NAME FROM SYSIBM.SYSTABLESPACE WHERE DBNAME = 'RMDDB48'
```

**QUERY**

This is the JCL for creating an SQL about the BMC tables. In the sample below, the BMCTABLES option is used to find all table spaces used by RECOVERY MANAGER plan.

**Figure 133: QUERY sample JCL**

```
//RDAJBM4R JOB (PARM),"RECOVERY.MANAGER",CLASS=Q,MSGCLASS=X,
// NOTIFY=&SYSUID
/** QUERY BMCTABLES
```
The following sample shows the output for a QUERY command.

** Figure 134: QUERY sample output **

```sql
** RECOVERY MANAGER FOR DB2 V11.2.00 - QUERY REPORT 06/17/2014 08:43:58 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.2.00 **
** BMC80220I BMC RECOVERY FOR DB2 V11.2.00 **

BMC TABLE NAMES FOR RECOVERY MANAGER PLAN JLWC112D:

<table>
<thead>
<tr>
<th>DBNAME</th>
<th>TSNAME</th>
<th>CREATOR</th>
<th>TABLE NAME</th>
<th>SYNONYM</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMCARM</td>
<td>BMCARMCR</td>
<td>BMCARM</td>
<td>CRRDRPT</td>
<td>BMCARM_CRRDRPT</td>
</tr>
<tr>
<td>BMCARM</td>
<td>BMCESTM</td>
<td>BMCARM</td>
<td>BMCRMD_IXP_SORT</td>
<td>BMCARM_IXP_SORT</td>
</tr>
<tr>
<td>BMCARM</td>
<td>BMCESTM</td>
<td>BMCARM</td>
<td>BMCRMD_IX_PART</td>
<td>BMCARM_IX_PART</td>
</tr>
<tr>
<td>BMCARM</td>
<td>BMCESTM</td>
<td>BMCARM</td>
<td>BMCRMD_JOB</td>
<td>BMCARM_JOB</td>
</tr>
<tr>
<td>BMCARM</td>
<td>BMCESTM</td>
<td>BMCARM</td>
<td>BMCRMD_JOB_RESTART</td>
<td>BMCARM_JOB_RESTART</td>
</tr>
<tr>
<td>BMCARM</td>
<td>BMCESTM</td>
<td>BMCARM</td>
<td>BMCRMD_KEYSORT</td>
<td>BMCARM_KEYSORT</td>
</tr>
<tr>
<td>BMCARM</td>
<td>BMCESTM</td>
<td>BMCARM</td>
<td>BMCRMD_PHASE</td>
<td>BMCARM_PHASE</td>
</tr>
<tr>
<td>BMCARM</td>
<td>BMCESTM</td>
<td>BMCARM</td>
<td>BMCRMD_TS</td>
<td>BMCARM_TS</td>
</tr>
</tbody>
</table>
```

The following figure shows the JCL for creating an SQL query about groups in the subsystem. In the sample below, the MISSING OBJECTS option is used to find all table spaces not included in any group.

** Figure 135: QUERY sample JCL **

```sql
//RDAJBMR4 JOB (PARM),RECOVERY.MANAGER',CLASS=Q,MSGCLASS=X, NOTIFY=&SYSUID
//* QUERY MISSING OBJECTS
//ARMDD001 EXEC PGM=ARMBGRP,
//    PARM='DGE,ARMOPTS=ARM$OPTS',
//    REGION=4M
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARMERROR DD SYSOUT=*
//ARMPRINT DD SYSOUT=*
//ARMQUERY DD SYSOUT=*
//ARMIN DD *
QUERY
MISSING OBJECTS
```

Chapter 19  ARMBGRP—Group creation and maintenance  591
The following figure shows the output for a QUERY command.

**Figure 136: QUERY sample output**

```
** RECOVERY MANAGER FOR DB2 V11.2.00 - QUERY REPORT 06/17/2014 15:13:55 **
(c) COPYRIGHT 1994-2015 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
BMC80223I MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.2.00
BMC80223I MAINT: BPJ0197 BPJ0215 BPJ0219

OBJECTS NOT IN A GROUP:
TABLESPACE  BMCARMTP.RMDTEMP1
TABLESPACE  BMCARMTP.RMDTEMP2
TABLESPACE  BMCARMTP.RMDTEMP3
TABLESPACE  BMCARMTP.RMDTEMP4
TABLESPACE  DSNDB06.SYSALTER
TABLESPACE  DSNDB06.SYSCOPY
TABLESPACE  DSNDB06.SYSCOPY
TABLESPACE  DSNDB06.SYSDBAUT
TABLESPACE  DSNDB06.SYSEBCDC
TABLESPACE  DSNDB06.SYSGPAUT
TABLESPACE  DSNDB06.SYGRTNS
TABLESPACE  DSNDB06.SYSJAUXA
TABLESPACE  DSNDB06.SYSJAUXB
TABLESPACE  DSNDB06.SYSJAVA
TABLESPACE  DSNDB06.SYSSEQ
TABLESPACE  DSNDB06.SYSSEQ2
TABLESPACE  DSNDB06.SYSUSER
TABLESPACE  DSNDB06.SYSVIEWS
```

The following figure shows the JCL for creating an SQL query where the OBJECTS option is used to find what groups include the specified table spaces and indexes.

**Figure 137: QUERY OBJECTS sample JCL**

```
//RDAJBMR4 JOB (PARM),'RECOVERY.MANAGER',CLASS=Q,MSGCLASS=X,
       NOTIFY=&SYSUID
// ** QUERY OBJECTS
//ARMOOD001 EXEC PGM=ARMBGRP,
       PARM='DGE,ARMOPTS=ARM$OPTS',
       REGION=4M
//STEPLIB     DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//             DD DISP=SHR,DSN=DSNEXIT
//             DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS     DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARERROR DD SYSOUT=* 
//ARPRINT DD SYSOUT=* 
//ARQUERY DD SYSOUT=* 
//ARMIN DD *
QUERY OBJECTS
       DBNAME ARMBJTR.TS4*,
       JTRTEST*,*
       INDEXES
```

The following figure shows the output for a QUERY OBJECTS command.

**Figure 138: QUERY OBJECTS sample output**

```
** RECOVERY MANAGER FOR DB2 V11.2.00 - QUERY REPORT 06/17/2014 14:53:01 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.2.00 **
```
The following figure shows sample JCL for exporting group definitions to a specified data set and then migrating those definitions to a group on another DB2 subsystem.

```
//ARMD000 EXEC PGM=ARMBGRP,
   //   PARM='DEFQ,ARMOPTS=JLW$OPTS',
   //   REGION=4M
//STEPLIB DD DISP=SHR,DSN=SCC.WJLW1210.LOAD
//   DD DISP=SHR,DSN=SCC.TEST1210.LGCLINK
//   DD DISP=SHR,DSN=SCC.TEST1210.BMCLINK
//   DD DISP=SHR,DSN=ARM.WJLW1210.LOAD
//   DD DISP=SHR,DSN=SYS3.DEFQ.DSNEXIT
//   DD DISP=SHR,DSN=CSGI.DB2V10M.DSNLOAD
//ARMMGS DD DISP=SHR,
//   DSN=ARM.WJLW1210.CNTL(ARMMGS)
//ARMPRINT DD SYSOUT=*
//ARMEERROR DD SYSOUT=*  
//ARMEXPRT DD DISP=SHR,
//   DSN=RDAJLW.ARMBGRP.EXPORT(ARMEXPRT)
//ARMIN DD *
SET CURRENT SQLID = RDAJLW;
EXPORT GROUP "PUBLIC"."Q1876526";
EXPORT GROUP "PUBLIC"."Q1878348";
EXPORT GROUP PUBLIC.Q1878802*;
MIGRATE_TO_SSID DIZ3;
MIGRATE_TO_SSID DEHE;

//ARMIN DD *
SET CURRENT SQLID = RDAJLW;
EXPORT GROUP "PUBLIC"."Q18789926";
* UNSUPPORTED DB2 VERSION FOR SSID DEBH BYPASSED AS A TARGET SSID
* SOURCE SSID DEFQ BYPASSED AS A TARGET SSID
* RECOVERY MANAGER MUST BE INSTALLED ON ALL TARGET SSID
MIGRATE_TO_SSID DEFR;
* INACTIVE SSID DBAC BYPASSED AS A TARGET SSID
```

The following figure shows a sample of valid combinations for the EXPORT GROUP and MIGRATE_TO_SSID syntax:

```
//ARMIN DD *
EXPORT GROUP "PUBLIC"."Q1876526";
*/
```
The following figure shows sample ARMPRINT for the EXPORT GROUP command with a wildcard pattern:

```
EXPORT GROUP PUBLIC.Q187*:
BMC80591I PROCESSING GROUP PUBLIC.Q1876526
BMC80591I PROCESSING GROUP PUBLIC.Q1878348
BMC80591I PROCESSING GROUP PUBLIC.Q1878802
BMC80591I PROCESSING GROUP PUBLIC.Q1878802_IX
BMC80591I PROCESSING GROUP PUBLIC.Q1878802_IXONLY
BMC80570I COMMAND COMPLETE RC = 0
```

The following figure shows sample output written to the ARMEXPRT DD:

```
CREATE GROUP "PUBLIC"."Q1876526"
   DESCRIPTION "test for jlw"
   REPLACE YES
   * LIKE "PUBLIC"."Q1876526"
   * RETAIN AUTH YES
   * RETAIN OBJECTS YES
   VIA TABLESPACE RMDDB4*.*.*
   BYPART NO
   INCLUDEIX NO
   INCLUDERI NO
   INCLUDELOB NO
   INCLUDEXML NO
   INCLUDEHISTORY NO
   INCLUDEARCHIVE NO;
CREATE GROUP "PUBLIC"."Q1878348"
   DESCRIPTION ""
   REPLACE YES
   * LIKE "PUBLIC"."Q1878348"
   * RETAIN AUTH YES
   * RETAIN OBJECTS YES
   VIA INDEX ACP.*
   BYPART NO;
```
REBUILD_SCOPE and RECOVER_SCOPE

The following figure shows sample JCL for bypassing objects based on specified criteria, thereby defining the scope of the rebuild and recovery organization.

```plaintext
BMC RECOVER
OPTIONS:
CHECKPOINT: Not specified
EARLYRECALL:
YES
EARLYCAT: Yes
OUTCOPY by Recover:
MAXLOGS:
MSGLEVEL: Object
UNLOADKEYS/Buildindex: No
SORT PRODUCT: OBSOLETE
Sort Product: OBSOLETE
SUMMARY
OPTION
Dynamic sortworks: Yes
Diagnostic messages:
No
Rebuild Index Shrlvel:
Reference
Rebuild Scope: (RECP)
Recover Scope: (ALL)
XBMID: Not specified
OUTCOPY by Recover:
ASCODED
MAXPRIM: 0
AUTOSIZE:
Yes
SNAP: Not
SNAP: Not
specified
specified
Max Key Sort: Not
Max Key Sort: Not
specified
specified
KSORTSHARE: Not specified
KSORTSHARE: Not specified
Max Log Sort: Not
specified
specified
ON ERROR CONTINUE:
10
Alternate Resources:
No
```

Executing the JCL

This topic describes special instructions or information required to run the ARMBGRP JCL.

- Ensure that the job owner has the appropriate authorizations (see “Authorizations” on page 515).

- If you are using the VVDS method, ensure that you allocated the data set with DISP=SHR on the `VVV VVV DD card`.

- No restart is available for ARMBGRP. You must remove the control cards for commands that successfully completed during the initial run, and then resubmit the job.
ARMBLGR—Log range analysis

This chapter describes ARMBLGR—Log range analysis.

About ARMBLGR

The ARMBLGR program performs log range analysis for a subsystem and captures information for each part of every table space and every index in the subsystem.

This information is saved in the RECOVERY MANAGER log range file (the ARMLRNG file).

The analysis is required by ARMBGEN when performing a local point-in-time recovery using a conditional restart. You can dramatically improve processing time by avoiding the unnecessary recovery of objects that have not changed between the current time and the recovery point. Use the ARMBLGR program to perform batch log range analysis prior to a local point in time (PIT) recovery of the full subsystem in order to identify objects that have not changed. Be aware that unchanged processing does not occur for indexes unless they have the COPY YES attribute.

---

**Note**

ARMBLGR writes its analysis to the RECOVERY MANAGER log range file (the ARMLRNG file). Each subsystem and data sharing group has a defined log range file that is tracked by the .LRNG option in the RECOVERY MANAGER option set. (Data sharing members share the same log range file.)

The log range file is defined during product install. If for any reason, you need to redefine the file, allocate a 50-cylinder file as FB, LRECL 392.

You can create the ARMBLGR JCL in the following ways:

- Manually code JCL to run ARMBLGR. This method completely avoids using a TSO session.

- Use online support that is provided by RECOVERY MANAGER to create ARMBLGR JCL. For more information, see “Generating JCL for local recovery” on page 357.
Authorizations

The following authorizations are required to execute the ARMBLGR program:

- APF authorization for the ARMBLGR program and the RECOVERY MANAGER load library
- EXECUTE authority for the RECOVERY MANAGER DB2 plan

Building the JCL

Building your own ARMBLGR job to perform analysis of log ranges involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Input data sets
  - Output data sets

The descriptions in the following subsections provide more details.

Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends that you specify REGION=0M, which makes the amount of virtual storage that is needed to run the job automatically available when the ARMBLGR job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.
Specifying the EXEC statement

The EXEC statement has the following format:

```plaintext
//stepname EXEC PGM=ARMBLGR,
//             PARM='ssid,REDO,ARMOPTS=optionSet',
//             REGION=0M
```

where:

- **ssid** is the DB2 subsystem or group attach name where the RECOVERY MANAGER groups reside.

  If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

  **Note**
  
  The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

- **The REDO parameter is used if two or more consecutive catalog point-in-time recoveries are performed without running application recoveries generated by ARMBGEN. Without this PARM, RECOVERY MANAGER deletes the information in the RECOVERY MANAGER log range file (the ARMLRNG file) before writing new information. If consecutive recoveries are performed and REDO is specified, the information in the file is merged with the information currently in SYSLGRNX. If you need to perform consecutive recoveries, be sure to regenerate the recovery JCL from the online panels using the new recovery point in time.**

- **optionSet** is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBLGR to use. For example:

```plaintext
//STEPLIB   DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
```
Specifying the ARMBLGR data set DD statements

This subsection describes the data sets that ARMBLGR uses.

Each data set is specified by a ddname (data definition name). You must specify all required data sets in the JCL.

- **(required) ARMPRINT**

  The output for messages that are returned from RECOVERY MANAGER. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

- **(optional) ARMOPTS**

  The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

  You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

  ```
  //ARMOPTS  DD *
  ssid.configurationOption=value
  /*
  ```

- **(required) ARMMSGS**

  The RMGR messages data set created during RECOVERY MANAGER installation with the default name of hlq/RMGR.ARMCNTL(ARMMSGS). The data set must be allocated with DISP=SHR.

- **(optional) ARMERROR**

  The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

Sample JCL

The following figure shows sample JCL for ARMBLGR.

**Figure 139: Sample ARMBLGR JCL**

```bash
//ARMBGLR JOB 5220
//*
//ARMD001 EXEC PGM=ARMBLGR,
//   PARM='DEC2,ARMOPTS=ARM$OPTS',
//   REGION=4M,COND=(4,LT)
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
```
Sample output

The following figure shows sample output for ARMBLGR.

** RECOVERY MANAGER FOR DB2 V11.1.00 - LOG RANGE ANALYSIS 08/16/2012 13:48:05 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.1.00 **
(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884

BMC80223I MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT: BPJ0197 BPJ0215 BPJ0219
BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910
BMC80799I LRNG FILE IS BMCARM.DEC2.LOGRANGE
BMC80571I PROGRAM COMPLETE RC = 0

Executing the JCL

This section describes special instructions or information required to run the ARMBLGR JCL.

- Ensure that the job owner has the appropriate authorizations. See “Authorizations” on page 598.
- No restart is available for ARMBLGR. You must resubmit the job after correcting any error conditions.
ARMBLOG—Archive log creation

This chapter describes ARMBLOG—Archive log creation.

About ARMBLOG

The batch archive log creation program, ARMBLOG, issues an archive log command and waits until the command is completed, as follows:

- In a non-data-sharing subsystem, the program issues an -ARCHIVE LOG command.
- In a data sharing subsystem, the SCOPE(GROUP) parameter is added if ARMBLOG is issued with the DATASHARE=GROUP parameter specified.

You can specify the maximum number of times to check the Print Log Map output to determine if the new archive log has been created. You can also specify the maximum number of seconds to wait between Print Log Map invocations.

Authorizations

The following authorizations are required to execute the ARMBLOG program:

- APF authorization for the RECOVERY MANAGER load library
- Authority to issue the -ARCHIVE LOG command
- READ authority for BSDS data sets
Building the JCL

Building your own ARMBLOG job involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition statements that specify the RECOVERY MANAGER and DB2 load libraries

The descriptions in the following subsections provide more details.

Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends you specify REGION=0M, which makes the amount of virtual storage that is needed to run the job automatically available when the ARMBLOG job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

Specifying the EXEC statement

The EXEC statement has the following format:

```
//stepname EXEC PGM=ARMBLOG,
//  PARM='ssid,tries,wait,ARMOPTS=optionSet,MODEQ=yes_no,DATASHARE=dstype',
//  REGION=0M
```

- The variable `ssid` is the DB2 subsystem on which the program is executing.
  If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

**Note**

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.
The variable `tries` is the maximum number of times to check the Print Log Map output. The default is 5.

The variable `wait` is the number of seconds to wait before issuing another Print Log Map command. The default is 180 seconds (three minutes).

The variable `optionSet` is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is `ARM$OPTS`.

The parameter `MODEQ= yes_no` can be `MODEQ=YES` or `MODEQ=NO`, as follows:

— `MODEQ=NO` (the default) causes the following commands to be generated:

- ARCHIVE LOG command for non-data-sharing members
- ARCHIVE LOG SCOPE(GROUP) for data sharing systems

— `MODEQ=YES` causes the ARCHIVE LOG MODE(QUIESCE) command to be generated for both data sharing and non-data-sharing systems

The parameter `DATASHARE= dstype` can be `DATASHARE=GROUP` or `DATASHARE=MEMBER`, as follows:

— `DATASHARE=GROUP` (the default) issues the ARCHIVE LOG command with SCOPE(GROUP).
— `DATASHARE=MEMBER` issues the ARCHIVE LOG command on the specified subsystem only.

**Specifying the STEPLIB DD statement**

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBLOG to use. For example:

```
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//      DD DISP=SHR,DSN=DSNEXIT
//      DD DISP=SHR,DSN=DSNLOAD
```

**Specifying the ARMBLOG data set DD statements**

This subsection describes the data sets ARMBLOG uses.

Each data set is specified by a `ddname` (data definition name). You must specify all required data sets in the JCL.
(required) ARMPRINT
The output for messages that are returned from RECOVERY MANAGER. This data set may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

(optional) ARMOPTS
The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

```plaintext
//ARMOPTS DD *
ssid.configurationOption=value /*
```

(required) ARMMSGS
The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of hilv/RLMGR.ARMCTRL(ARMMSGS). The data set must be allocated with DISP=SHR.

(optional) ARMERROR
The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

Sample JCL

The following figure shows sample JCL for ARMBLOG. The JCL indicates the maximum number of times to check the Print Log Map for the new archive as 2 and the number of minutes to wait between tries as 2.

Note
If the new log has not been created after the maximum number of tries, ARMBLOG issues the following message and ends with RC=8:

BMC80879E OFFLOAD TASK DID NOT COMPLETE, RETRIES EXHAUSTED

Figure 141: Sample ARMBLOG JCL

```plaintext
//ARM0001 EXEC PGM=ARMBLOG,
//      PARM='DHX,ARMOPTS=ARM$OPTS',
//      REGION=4M
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
```
Sample output

The following figure shows a sample of ARMBLOG output.

Figure 142: Sample ARMBLOG output

** RECOVERY MANAGER FOR DB2 V11.1.00 - BATCH ARCHIVE LOG 02/27/2012 11:52:11 **

(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 562581 AND 5761676

BMC802231 MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC802231 SOLUTION COMMON CODE V11.1.00
BMC802231 MAINT: NO SCC PTFS APPLIED

BMC803091 CONNECTED TO DB2 SSID = DHX1 VERSION 910

BMC808771 PRINT LOG MAP ISSUED RC = 0
BMC808761 DHX1 LAST LOG=DSNDHX.DHX1.ARCLG1.D09058.T1136375.A0004361
BMC808761 DHX2 LAST LOG=DSNDHX.DHX2.ARCLG1.D09058.T1136377.A0015847
BMC808771 -ARCHIVE LOG SCOPE(GROUP) ISSUED RC = 0
BMC808761 DHX1 NOT OFFLOADED LAST LOG=DSNDHX.DHX1.ARCLG1.D09058.T113665.A0004361
BMC808761 DHX2 OFFLOADED LAST LOG=DSNDHX.DHX2.ARCLG1.D09058.T1152236.A0015848
BMC808791 START LRSN = C3CFA3358FC3 END LRSN = C3CFA6BBDEEE
BMC808781 WAITING 5 SECONDS PRIOR TO RETRYING COMMAND
BMC808761 DHX1 OFFLOADED LAST LOG=DSNDHX.DHX1.ARCLG1.D09058.T1152236.A0004362
BMC808791 START LRSN = C3CFA3358FC3 END LRSN = C3CFA6BBDEEE
BMC808761 DHX2 OFFLOADED LAST LOG=DSNDHX.DHX2.ARCLG1.D09058.T1152236.A0015848
BMC808791 START LRSN = C3CFA3358FC3 END LRSN = C3CFA6BBDEEE
BMC808781 ARCHIVE OFFLOAD COMPLETE
BMC805711 PROGRAM COMPLETE RC = 0

Executing the JCL

This section describes special instructions or information required to run the ARMBLOG JCL.

- Ensure that you have the appropriate authorizations. See “Authorizations” on page 603 for required authorizations.

- On a data sharing system, if you want to archive a log on all members, you only need to run ARMBLOG on one member with the DATASHARE=GROUP parameter specified. If you want to archive a log on some (but not all) members, you must run ARMBLOG on each specific subsystem with the DATASHARE=MEMBER parameter specified. This parameter is ignored for non-data-sharing subsystems.
- No restart is available for ARMBLOG. You must resubmit the job after correcting any error conditions.
This chapter describes ARMBLRD—Log range formatting.

About ARMBLRD

The ARMBLRD program reads the log range file and then prints it in a readable format to the ARMTRACE file.

The output is much easier to read and analyze than the unformatted log range file.

ARMBLRD also prints the contents of the ARMWPEND file, which is created by ARMBSRR and is used by ARMBGEN during Full Subsystem Local PIT recovery.

Note
Both reports are intended for use only in problem diagnosis under the instructions of BMC Customer Support.

Authorizations

The following authorizations are required to execute the ARMBLRD program:

- APF authorization for the RECOVERY MANAGER load library
- READ authority for the log range file

Building the JCL

Building your own ARMBLRD job to print the log range file for the subsystem-based groups involves creating JCL that includes the following statements:

- A JOB statement
An EXEC statement

Data definition statements that specify the use of the following libraries and data sets:

- RECOVERY MANAGER load libraries
- Input data sets
- Output data sets

The descriptions in the following subsections provide more details.

**Specifying the JOB statement**

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends that you specify REGION=0M, which makes the amount of virtual storage that is needed to run the job automatically available when the ARMBLRD job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

**Specifying the EXEC statement**

The EXEC statement has the following format:

```
//stepname EXEC PGM=ARMBLRD
PARM=(ssid, pitRba, ARMOPTS=optionSet)
//           REGION=4M
```

where

- `ssid` identifies the subsystem that corresponds to the log range file used as input. ARMBLRD accesses DB2 and includes the database name and table space name that correspond to the DBID and PSID in each row of the output of the TRACE DD.

If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

**Note**

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.
- *pitRba* is an optional parameter that allows you to limit the log range analysis by specifying a recovery point RBA or LRSN.

**Example**

You can specify the PARM using the subsystem ID, the recovery point RBA or LRSN, or both. See the following examples:

- By subsystem ID:
  
  `PARM='DEBC'`

- By RBA/LRSN:
  
  `PARM=',0000088161E2'`

- By both subsystem ID and RBA/LRSN:
  
  `PARM='DEBC,0000088161E2'`

- *optionSet* is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

### Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library that you want ARMBLRD to use. For example:

```plaintext
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
/// DD DISP=SHR,DSN=DSNEXIT
/// DD DISP=SHR,DSN=DSNLOAD
```

### Specifying the ARMBLRD data set DD statements

This subsection describes the data sets that ARMBLRD uses.

Each data set is specified by a *ddname* (data definition name). You must specify all required data sets in the JCL.

- *(required) ARMPRINT*
  
  The output for error messages that are returned from the program. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.
(required) ARMLGRNX
Points ARMBLRD to the RECOVERY MANAGER log range file (the ARMLRNG file). The log range file name is specified in the ARMSOPTS options file as variable ssid.LRNG. The data set must be allocated with DISP=SHR.

(required) ARMWPEND
The RECOVERY MANAGER write pending file created by ARMBSRR processing. The data set must be allocated with DISP=SHR.

(required) ARMTRACE
The output for the trace messages and for the formatted log range information.

(required) ARMMGS
The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of hilvl.RMGR.ARMCNTL(ARMMSGS). The data set must be allocated with DISP=SHR.

(optional) ARMOPTS
The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

```
//ARMOPTS DD *
ssid.configurationOption= value
/*
```

Sample JCL

The following figure shows a sample of JCL for ARMBLRD.

**Figure 143: Sample ARMBLRD JCL**

```
//STEP1   EXEC PGM=ARMBLRD,  
  //*          PARM='DECI,0000088161E2,ARMOPTS=ARM $OPTS',  
  //          REGION=4M  
//STEPLIB   DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS  
//          DD DISP=SHR,DSN=DSNEXIT  
//          DD DISP=SHR,DSN=DSNLOAD  
//ARMMSGS   DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)  
//ARMLGRNX  DD DISP=SHR,DSN=BMCARM.DECI.LGRNX  
//ARMWPEND  DD DISP=SHR,DSN=BMCARM.DECI.WPEND  
//ARMERROR  DD SYSOUT=*  
//ARMTRACE  DD SYSOUT=*  
//ARMPRINT  DD SYSOUT=*  
//
```
Sample output

The following figure shows sample output for ARMBLRD. The formatted log range information is included in the ARMTRACE DD and the sample shows a portion of the log range information from the trace file:

**Figure 144: Sample ARMTRACE output from ARMBLRD**

| ELSTRAC: 15:37:00 - Trace Enabled - 02/19/2012 |
| ELSTRAC: 15:37:01 - |
| ELSTRAC: 15:37:01 - RECOVERY MANAGER FOR DB2 V11.1.00 |
| ELSTRAC: 15:37:01 - Supports DB2 Version V9.1.00 through V10.1.00 |
| ELSTRAC: 15:37:01 - |
| ELSTRAC: 15:37:01 - MAINT: NO RECOVERY MANAGER PTFS APPLIED |
| ELSTRAC: 15:37:01 - SOLUTION COMMON CODE V11.1.00 |
| ELSTRAC: 15:37:01 - MAINT: BPJ0088 |
| ELSTRAC: 15:37:01 - |
| ELSTRAC: 15:37:01 - no TRCIN DD |
| ARMSETU: 15:37:01 - RM class init --> Trace enabled |
| ARMSETU: 15:37:01 - Found product code --> ARM |
| ARMSETU: 15:37:01 - Found SSID |
| ARMSETU: 15:37:01 - Passed security check |

<table>
<thead>
<tr>
<th>DBID</th>
<th>PSID</th>
<th>DSNUM</th>
<th>DBNAME</th>
<th>NAME</th>
<th>TYPE</th>
<th>STARTRBA</th>
<th>ENDRBA</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>2</td>
<td>0</td>
<td>DSNDB04</td>
<td>MQT22</td>
<td>TS</td>
<td>00001DD6D5A8</td>
<td>00001E60137E</td>
</tr>
<tr>
<td>2004-04-22-17.45.18.250057</td>
<td>DECICAT</td>
<td>N</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

Executing the JCL

This section describes special instructions or information required to run the ARMBLRD JCL.

- Ensure that the job owner has the appropriate authorizations. See “Authorizations” on page 609.

- No restart is available for ARMBLRD. You must resubmit the job after correcting any error conditions.
ARMBRDC—Recovery data collection report

This chapter describes the ARMBRDC — Recovery data collection report.

About ARMBRDC

The ARMBRDC program reports data collected during actual, simulated, and estimated disaster recoveries.

It is only available with the Recovery Management for DB2 solution. For more information about data collection, see the Recovery Management for DB2 User Guide. ARMBRDC stores the data collected during recoveries in the UTILITY_RUN table and generates the recovery summary reports. See “Recovery history: UTILITY_RUN table” on page 774 for more information about the UTILITY_RUN table.

Note

The ARMBRDC program cleans up the data collection tables each time it is run by deleting the oldest entries. No more than 12 recoveries for each recovery type (actual, simulated, and estimated) are stored. The delete recoveries are reported in a recovery history clean-up report.

Authorizations

The following authorizations are required to execute the ARMBRDC program:

- APF authorization for the ARMBRDC program and the RECOVERY MANAGER load library
- EXECUTE authority on the RECOVERY MANAGER DB2 plan
Building the JCL

Building your own ARMBRDC job to generate JCL to recover the DB2 subsystem involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Input data sets
  - Output data sets

The descriptions that follow provide more details.

Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends you specify REGION=0M, which makes the amount of virtual storage needed to run the job automatically available when the ARMBRDC job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

Specifying the EXEC statement

The EXEC statement has the following format:

```bash
//stepname EXEC PGM=ARMBRDC,
//              PARM='ssid,ARMOPTS=optionSet',
//              REGION=0M
```
The variable ssid is the DB2 subsystem or group attach name where the RECOVERY MANAGER groups reside. If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

**Note**

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

The variable `optionSet` is the name of an XML file that contains all of the product's configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

### Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the Recovery Management load library and DB2 load libraries that you want ARMBRDC to use. For example:

```plaintext
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
```

### Specifying the ARMBRDC data set DD statements

This subsection describes the data sets that ARMBRDC uses.

Each data set is specified by a `ddname` (data definition name). You must specify all required data sets in the JCL.

- **(required) ARMIN**
  
  The input data set that contains one or more control statements. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

- **(required) ARMPRINT**
  
  The output for messages that are returned from RECOVERY MANAGER. RECOVERY MANAGER also echoes the contents of the ARMIN data set in the ARMPRINT output. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.
■ **(optional) ARMOPTS**

The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

```
//ARMOPTS  DD *
ssid.configurationOption=value /*
```

■ **(required) ARMMSGS**

The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of hilvl/RMGR.ARMICNTL(ARMMSGS). The data set must be allocated with DISP=SHR.

■ **(optional) ARMERROR**

The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. The data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

■ **(optional) ARMRRPT**

Contains the data collection reports. The data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

### ARMBRDC syntax and option descriptions

The ARMBRDC syntax and option descriptions in this section are the control statements that you use when you build ARMIN input.
Figure 145 on page 619 shows the ARMBRDC syntax.

Figure 145: ARMBRDC control statement

![ARMBRDC control statement diagram](image)

**DCTOKEN**

DCTOKEN *token* is an identifier used by RECOVERY MANAGER to differentiate different types of data collection information. The only DCTOKEN value currently used by RECOVERY MANAGER is DRECOVER.

**PROCESS RECOVERY**

Processes recovery data collection information generated and stored by NGT Recover and RECOVERY MANAGER. ARMBRDC collects and consolidates data from the JOB entries and then stores the data in the corresponding recovery entry in the UTILITY_RUN table. It also includes the objects with the longest recovery time from the actual recovery and shows the estimated time to recover each of those objects as a comparison.

*Note*

If no active recovery exists in the UTILITY_RUN table, ARMBRDC reports on the most recent recovery that matches the token name.

**COMPARE ACTUAL ESTIMATE**

Generates a report that compares the last actual recovery to the last estimated recovery.

**LAST ACTUAL**

Generates a report showing data collected from the last *n* number of actual recoveries.
LAST ESTIMATE

Generates a report showing data collected from the last \( n \) number of estimated recoveries.

LAST SIMULATE

Generates a report showing data collected from the last \( n \) number of simulated recoveries.

Sample JCL

The following figure shows a sample of disaster recovery JCL for ARMBRDC.

**Figure 146: Sample ARMBRDC JCL**

```
//STEP01   EXEC PGM=ARMBRDC,PARM='DEBN,ARMOPTS=ARM
$OPTS',REGION=4M
//STEPLIB   DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS   DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARMPRINT  DD SYSOUT=* 
//ARMERROR  DD SYSOUT=* 
//SYSPRINT  DD SYSOUT=* 
//SYSTEM    DD SYSOUT=* 
//ARMRRPT   DD SYSOUT=* 
//ARMIN     DD *
PROCESS RECOVERY DCTOKEN DRECOVER
```

Sample output

The following figure shows sample output for the ARMBRDC program.

--- Note ---

In this report ** indicates greater than 99.

** Figure 147: Sample ARMBRDC output - Recovery Data Collection report **

```
** RECOVERY MANAGER FOR DB2 V11.1.00 - Recovery Report(s) 02/20/2012 14:04:55 **

Report: Process the Recovery

------------------------------------------
Recovery report for last ACTUAL on CK (UID=1223)
Start: 2012-02-20 13:51:30.000000
End: 2012-02-20 14:02:15.127932
Actual Elapsed Time : 00:10:45
```
Estimated Elapsed Time: 00:03:37

Avg time to recover Tablespace: 9
Estimated Avg time to recover Tablespace: 40

Total Tablespaces: 22       Total Number Jobs: 13
Total Indexspaces: 0        Total Bytes Recovered: 32112640

Top 10 Tablespaces in this recovery:

<table>
<thead>
<tr>
<th>DBNAME</th>
<th>TSNAME</th>
<th>START TS</th>
<th>END TS</th>
<th>Estimated Elapsed</th>
<th>Actual Elapsed</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMDDB59</td>
<td>L59P252B</td>
<td>2012-02-20-14.01.33</td>
<td>00:00:43</td>
<td>00:00:00</td>
<td>Merge SP/No IC</td>
<td></td>
</tr>
<tr>
<td>RMDDB59</td>
<td>LS59S25B</td>
<td>2012-02-20-14.01.24</td>
<td>00:00:43</td>
<td>00:00:09</td>
<td>Merge SP/No IC</td>
<td></td>
</tr>
<tr>
<td>RMDDB59</td>
<td>LS59N25A</td>
<td>2012-02-20-14.00.43</td>
<td>00:00:43</td>
<td>00:00:09</td>
<td>Merge SP/No IC</td>
<td></td>
</tr>
<tr>
<td>RMDDB59</td>
<td>LS59N25B</td>
<td>2012-02-20-14.00.46</td>
<td>00:00:43</td>
<td>00:00:38</td>
<td>Merge SP/No IC</td>
<td></td>
</tr>
<tr>
<td>RMDDB59</td>
<td>LS59S25A</td>
<td>2012-02-20-14.00.55</td>
<td>00:00:43</td>
<td>00:00:38</td>
<td>Merge SP/No IC</td>
<td></td>
</tr>
<tr>
<td>RMDDB59</td>
<td>L59P252A</td>
<td>2012-02-20-14.00.58</td>
<td>00:00:43</td>
<td>00:00:08</td>
<td>Merge SP/No IC</td>
<td></td>
</tr>
<tr>
<td>RMDDB59</td>
<td>L59P251A</td>
<td>2012-02-20-14.00.42</td>
<td>00:00:43</td>
<td>00:00:12</td>
<td>Merge SP/No IC</td>
<td></td>
</tr>
<tr>
<td>RMDDB59</td>
<td>L59P251B</td>
<td>2012-02-20-14.00.49</td>
<td>00:00:43</td>
<td>00:00:14</td>
<td>Merge SP/No IC</td>
<td></td>
</tr>
<tr>
<td>RMDDB59</td>
<td>L59P251A</td>
<td>2012-02-20-13.55.40</td>
<td>00:00:43</td>
<td>00:00:02</td>
<td>Merge SP/No IC</td>
<td></td>
</tr>
<tr>
<td>RMDDB59</td>
<td>L59P252A</td>
<td>2012-02-20-13.55.43</td>
<td>00:00:43</td>
<td>00:00:02</td>
<td>Merge SP/No IC</td>
<td></td>
</tr>
</tbody>
</table>

Report: Recovery History Clean-up

01 Actual Recoveries found - None deleted
01 Recovery Simulations found - None deleted
01 Recovery Estimations found - None deleted

**Executing the JCL**

This section describes special instructions or information required to run the ARMBRDC JCL.
- Ensure that the job owner has the appropriate authorizations. See “Authorizations” on page 615.

- No restart is available for ARMBRDC. You must resubmit the job after correcting any error conditions.
ARMBRID—Recover indoubt threads

This chapter describes ARMBRID—Recover indoubt threads.

About ARMBRID

The recover indoubt threads program, ARMBRID, displays indoubt threads, parses the output, and executes Recover Indoubt commands as needed.

ARMBSRR always generates a step at the beginning of the Phase 2 job to process indoubt threads and invoke ARMBRID.

Authorizations

The following authorizations are required to execute the ARMBRID program:

- APF authorization for the ARMBRID program and the RECOVERY MANAGER load library
- EXECUTE authority on the RECOVERY MANAGER plan

Building the JCL

Building your own ARMBRID job involves creating JCL that includes the following statements:

- A JOB statement
An EXEC statement

Data definition statements that specify the use of the following libraries and data sets:
- RECOVERY MANAGER and DB2 load libraries
- Output data sets

**Specifying the JOB statement**

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends that you specify REGION=0M, which makes the amount of virtual storage that is needed to run the job automatically available when the ARMBRID job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

**Specifying the EXEC statement**

The EXEC statement has the following format:
```
//stepname EXEC PGM=ARMBRID,PARM='ssid,ARMOPTS=optionSet',
//             REGION=0M
```

The variable `ssid` is the DB2 subsystem on which the program is executing. If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

*Note*

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

The variable `optionSet` is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARMOPTS.
Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBRID to use. For example:

```
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// DD DISP=SHR,DSN=DSNLOAD
```

Specifying the ARMBRID data set DD statements

This subsection describes the data sets ARMBRID uses.

Each data set is specified by a *ddname* (data definition name). You must specify all required data sets in the JCL.

- *(required)* ARMPRINT
  
  The output for messages that are returned from RECOVERY MANAGER. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

- *(optional)* ARMOPTS
  
  The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

  You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

  ```
  //ARMOPTS DD *
  ssid.configurationOption=value /*
  ```

- *(required)* ARMMSGS
  
  The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of `hilvl.RMGR.ARMCNTL(ARMMSG)`. The data set must be allocated with DISP=SHR.

- *(optional)* ARMERROR
  
  The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.
Sample JCL

The following figure shows sample JCL for ARMBRID.

Figure 148: Sample ARMBRID JCL

```plaintext
//** *************************************************************** */
//** *************************************************************** */
//** RECOVER INDOUBT TRANSACTIONS (IF ANY): DEDL */
//** *************************************************************** */
//** *************************************************************** */
//ARM00001 EXEC PGM=ARMBRID,PARM='DEDL,ARMOPTS=ARM$OPTS',REGION=4M,COND=(4,LT)
//STELIB  DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//         DD DISP=SHR,DSN=DSNEXIT
//         DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARMPRINT DD SYSOUT=*
//ARMERROR DD SYSOUT=*  
```

Sample output

The following figure shows sample output for ARMBRID.

Figure 149: Sample ARMBRID output

```plaintext
** RECOVERY MANAGER FOR DB2 V11.1.00 - RECOVER INDOUBT TRANSACTIONS 02/20/2012 14
** BMC802201 RECOVERY MANAGEMENT FOR DB2 V11.1.00
(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884

BMC80223I MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT: NO SCC PTFS APPLIED

BMC80309I CONNECTED TO DB2 SSID = DEDL VERSION 910
-DISPLAY THREAD(*) TYPE(INDOUBT)
DSNV401I *DEDL DISPLAY THREAD REPORT FOLLOWS -
DSNV420I *DEDL NO INDOUBT THREADS FOUND
DSN9022I *DEDL DSNVDT '-DISPLAY THREAD' NORMAL COMPLETION

BMC80571I PROGRAM COMPLETE RC = 0
```

Executing the JCL

This section describes special instructions or information required to run the ARMBRID JCL.

- Ensure that you have the appropriate authorizations. See “Authorizations” on page 623 for required authorizations.
No restart is available for ARMBRID. You must resubmit the job after correcting any error conditions.
ARMBRPR — Progress Reports

This chapter describes ARMBRPR — Progress Reports.

About ARMBRPR

ARMBRPR provides reports to gauge the progress of backups and recoveries for the requested set of objects.

Note

Backup and recovery elapsed time estimates are available only with a Recovery Management solution password.

ARMBRPR issues the -ACCESS DB(*) SP(*) MODE(STATS) command, which externalizes real-time statistics to provide more accurate reports.

ARMBRPR produces the following reports:

- ARMRECOV — Objects Recovered Within Requested Time Range
- ARMNOREC — Objects Not Recovered Within Requested Time Range
- ARMUNCHG — Objects Unchanged Within Requested Time Range
- ARMCHANG — Objects Changed Within Requested Time Range
- ARMCOPYD — Objects Copied Within Requested Time Range
- ARMNOCOP — Objects Not Copied Within Requested Time Range
- ARMOERR — Objects Not Eligible For Backout Recovery
- ARMFWERR — Objects Not Eligible For Forward Recovery
- ARMSUMRY — Summary

For information about using the RECOVERY MANAGER online interface to produce these reports, see “Accessing online Progress Reports” on page 383.
Note

- Clones are not supported.

- Megabytes are taken from DB2 catalog real time statistics (SYSTABLESPACESTATS and SYSINDEXSPACESTATS). DSNUTILB RECOVER may clear the real time statistics in some cases. 2,147,483,647 is the largest number of megabytes that can be displayed.

- Objects with an ending log range that is zeros or is greater than the specified timestamp are considered changed. Objects with no log ranges or that have log ranges that end prior or equal to the specified timestamp are considered unchanged.

Authorizations

The following authorizations are required to execute the ARMBRPR program:

- APF authorization for the ARMBRPR program and the RECOVERY MANAGER load library

- EXECUTE authority on the RECOVERY MANAGER plan

Building the JCL

The ARMBRPR JCL includes the following statements:

- A JOB statement

- An EXEC statement

- Data definition statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Input data sets
  - Output data sets

The descriptions in the following subsections provide more details.
Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends that you specify REGION=0M, which makes the amount of virtual storage that is needed to run the job automatically available when the ARMPR job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

Specifying the EXEC statement

The EXEC statement has the following format:

```
//stepname EXEC PGM=ARMBRPR,
   PARM='ssid,ARMOPTS=optionSet',
   REGION=0M
```

The variable ssid is the DB2 subsystem or group attach name on which the program is executing. If you do not provide a ssid, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

**Note**

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

The variable optionSet is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMRPR to use. For example:

```
//STEPLIB DD DISP=SHR,DSN=PRODUCT_LOAD.LIBS
//       DD DISP=SHR,DSN=DSNEXIT
//       DD DISP=SHR,DSN=DSNLOAD
```
Specifying the ARMBRPR data set DD statements

This section describes the data sets that ARMBRPR uses.

Each data set is specified by a *ddname* (data definition name). You must specify all required data sets in the JCL.

- **(required) ARMIN**
  The input data set that contains one or more control statements. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

- **(required) ARMPRINT**
  The output for messages that are returned from RECOVERY MANAGER. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

- **(optional) ARMOPTS**
  The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.
  
  You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

  ```
  //ARMOPTS DD *
  ssid.configurationOption=value
  /*
  ```

- **(required) ARMMGS**
  The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of hilv/RMGR.ARMCNTL(ARMMGS). The data set must be allocated with DISP=SHR.

- **(optional) ARMERROR**
  The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

- **ARMSUMRY**
  The output for the Summary report

- **(optional) ARMRECOV**
  The output for the Objects Recovered Within Requested Time Range report
(optional) ARMNOREC
The output for the Objects Not Recovered Within Requested Time Range report

(optional) ARMUNCHG
The output for the Objects Unchanged Within Requested Time Range report

(optional) ARMCHANG
The output for the Objects Changed Within Requested Time Range report

(optional) ARMCOPYD
The output for the Objects Copied Within Requested Time Range report

(optional) ARMNOCOP
The output for the Objects Not Copied Within Requested Time Range report

(optional) ARMBOERR
The output for the Objects Not Eligible For Backout Recovery report.
This output provides detailed information from backout recovery time estimates.

(optional) ARMFWERR
The output for the Objects Not Eligible For Forward Recovery report
This output provides detailed information from forward recovery time estimates.

(optional) ARMAUERR
The output for the Backout Auto Estimation Exceptions report
This output provides detailed information from backout auto estimates.

Syntax diagrams

The ARMBRPR syntax is generated by RECOVERY MANAGER. Sample JCL is provided in the ARMBRPR$ member in the sample data set.
Figure 150: ARMBRPR syntax statements

```
SET CURRENT SQILD = sqild ;

REPORT

GROUP creator.name

INCLUDEINDEXES

NO

YES

DBNAME database.tablespace

INCLUDEINDEXES

NO

YES

START_TIME startTime

SIMULATED_RECOVERY

NO

YES

TOTAL_FORMAT

MB

GB

TB

CALCULATE_IO_FACTOR

DASD

VIRTUAL_TAPE

TAPE

RECOVERY_ESTIMATES

IO_FACTOR

NUMBER_JOBS

MAXSORT

REBUILD

BACKUP_ESTIMATES

IO_FACTOR

MAXTASKS

OUTSIZE
```
Option descriptions

This section contains descriptions of syntax options.

**SET CURRENT SQLID**

In the ARMIN input data set, optionally provide the control statement SET CURRENT SQLID = sqlid to set the SQLID to be used for SQL execution.

The SQLID defaults to your user ID.

You can have multiple SET CURRENT statements in the control data set. The SET CURRENT SQLID statement is in effect for all statements that follow it until another SET CURRENT SQLID statement is issued.

**REPORT**

Indicates a Progress Report request.

**GROUP**

Indicates the group name in the form creator.name. Group creator cannot contain wildcard characters. Group name can include wildcards.

**INCLUDEINDEXES**

Indicates if indexes are included. Valid values are NO (the default) and YES.

**DBNAME**

Indicates the database.tablespace. The database.tablespace can include a wildcard pattern.

**INCLUDEINDEXES**

Indicates whether indexes are included. Valid values are NO (the default) and YES.

**START_TIME**

Indicates the recovery start time. The input is a timestamp in the format YYYY-MM-DD-HH.MM.SS to start searching for recoveries, copies and changes. START_TIME is required.

**SIMULATE_RECOVERY**

Indicates whether progress reporting is available for SIMULATE and INDEP OUTSPACE recoveries and rebuilds. If this option is enabled, NGT Recover
inserts rows into the BMCXCOPY table when each recovery or rebuild finishes. Valid values are NO (the default) and YES.

**TOTALS_FORMAT**

Indicates the value for total bytes. Valid values are MB (megabytes), GB (gigabytes), and TB (terabytes).

**CALCULATE_IO_FACTOR**

RECOVERY MANAGER will calculate and store I/O rates for use by backup and recovery time estimation. CALCULATE_IO_FACTOR is optional.

---

**Note**

The approximate I/O factor can be found in NGT Copy output DD ACPPRTnn or SYSPRINT. The I/O factor will be more accurate when copying large objects.

---

**DASD**

Indicates the I/O rate for a DASD unit.

**VIRTUAL_TAPE**

Indicates the I/O rate for a Virtual Tape unit.

**TAPE**

Indicates the I/O rate for a Tape unit.

**BACKUP_ESTIMATES**

Indicates backup elapsed time estimates for standard, snapshot, and cabinet copies. The following options are used to estimate elapsed time.

**IO_FACTOR**

Valid values are 1-10000. The default is 100.

**MAXTASKS**

Indicates the number of subtasks that NGT Copy can use for output. Valid values are 1-32. The default is 1.

**OUTSIZE**

Valid values are 0-4294967295 for KB, 0-4194303 for MB, 0-4095 for GB. The default is 0.
If OUTSIZE is 0, the elapsed time estimates are generated as follows:

- Standard HH:MM:SS
- Snapshot HH:MM:SS
- Cabinet HH:MM:SS

If OUTSIZE is greater than 0, the elapsed time estimates are generated as follows:

- Cabinet HH:MM:SS Snapshot HH:MM:SS Hybrid Total HH:MM:SS

**RECOVER_ESTIMATES**

Indicates elapsed time estimates for forward and backout recoveries. The following options are used to estimate elapsed time.

**IO_FACTOR**

Valid values are 1-10000. The default is 100.

**NUMBER_JOBS**

Indicates the number of jobs. Valid values are 1-99. The default is 1.

**MAXLSORT**

Indicates how many log sort tasks NGT Recover can run in parallel. Valid values are 1-32. The default is 1.

**REBUILD**

Indicates whether estimate to rebuild indexes is included. Valid values are NO, the default, and YES.

---

**Sample ARMBRPR JCL**

The following figure shows sample JCL for ARMBRPR.

A sample job is located in the ARMBRPR$SAMP member in the .ARMSAMP data set that was created during installation.

**Figure 151: Sample ARMBRPR JCL**

```plaintext
//ARMBRPR JOB (PARM), 'RECOVERY.MANAGER', CLASS=Q, MSGCLASS=X,
// NOTIFY=&SYSSID
/* *********************************************** */
```

---
Sample ARMBRPR output

The following examples show sample ARMBRPR output.

Figure 152: Sample ARMBRPR ARMPRINT output

** RECOVERY MANAGER FOR DB2 V11.2.00 - PROGRESS REPORT 08/21/2014 13:3
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.2.00

(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 576
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884

BMC80223I MAINT:  NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V10.1.01
BMC80223I MAINT:  NO SCC PTFS APPLIED

BMC80309I CONNECTED TO DB2 SSID = DEFR VERSION 1010 MODE = NFM

638 RECOVERY MANAGER for DB2 User Guide
Figure 153: ARMRECOV sample output — Objects Recovered Within Requested Time Range

Figure 154: ARMNOREC sample output—Objects Not Recovered Within Requested Time Range
** Recovery Manager for DB2 V11.2.00 - Progress Report 08/21/2014 13:39:27 **

** BMC802201 Recovery Management for DB2 V11.2.00 **

Recovery start time 2014-08-21-13.29.49.000000
Group RMD.CRC5

Indexes not recovered/rebuilt within requested time range

<table>
<thead>
<tr>
<th>Object Name</th>
<th>IX Space</th>
<th>DSNUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMD.IC46N041</td>
<td>IC46N041</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>1</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>2</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>3</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>4</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>5</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>6</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>7</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>8</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>9</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>10</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>11</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>12</td>
</tr>
<tr>
<td>RMD.IN46N041</td>
<td>IN46N041</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IN46P041</td>
<td>IN46P041</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IN46P041</td>
<td>IN46P041</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IN46S041</td>
<td>IN46S041</td>
<td>0</td>
</tr>
</tbody>
</table>

Recovery start time 2014-08-21-13.29.49.000000
Group RMD.CRC5

Indexes unchanged within requested time range

<table>
<thead>
<tr>
<th>Object Name</th>
<th>IX Space</th>
<th>DSNUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMD.IC46N041</td>
<td>IC46N041</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 155: ARMUNCHG sample output—Objects Unchanged Within Requested Time Range
** RECOVERY MANAGER FOR DB2 V11.2.00 - PROGRESS REPORT
08/21/2014 13:39:27 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.2.00

RECOVERY START TIME 2014-08-21-13.29.49.000000
GROUP RMD.CRCS5

TABLESPACES UNCHANGED WITHIN REQUESTED TIME RANGE

<table>
<thead>
<tr>
<th>OBJECT NAME</th>
<th>DSNUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMDDB46.TS46N04</td>
<td>0</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>1</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>2</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>3</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>4</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>5</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>6</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>7</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>8</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>9</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>10</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>11</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>12</td>
</tr>
<tr>
<td>RMDDB46.TS46S04</td>
<td>0</td>
</tr>
</tbody>
</table>

RECOVERY START TIME 2014-08-21-13.29.49.000000
GROUP RMD.CRCS5

INDEXES UNCHANGED WITHIN REQUESTED TIME RANGE

<table>
<thead>
<tr>
<th>OBJECT NAME</th>
<th>IX SPACE</th>
<th>DSNUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMD.IC46N041</td>
<td>IC46N041</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>1</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>2</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>3</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>4</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>5</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>6</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>7</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>8</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>9</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>10</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>11</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>12</td>
</tr>
<tr>
<td>RMD.IN46N041</td>
<td>IN46N041</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IN46P041</td>
<td>IN46P041</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IN46S041</td>
<td>IN46S041</td>
<td>0</td>
</tr>
</tbody>
</table>
** Figure 156: ARMCHANG sample output—Objects Changed Within Requested Time Range **

** RECOVERY MANAGER FOR DB2 V11.2.00 - PROGRESS REPORT 08/21/2014 13:39:27 **

** BMC802201 RECOVERY MANAGEMENT FOR DB2 V11.2.00 **

** RECOVERY START TIME 2014-08-21-13.29.49.000000 **

GROUP RMD.CRCS5

TABLESPACES CHANGED WITHIN REQUESTED TIME RANGE

<table>
<thead>
<tr>
<th>OBJECT NAME</th>
<th>DSNUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMDDB46.TS46N01</td>
<td>0</td>
</tr>
<tr>
<td>RMDDB46.TS46N02</td>
<td>0</td>
</tr>
<tr>
<td>RMDDB46.TS46N03</td>
<td>0</td>
</tr>
<tr>
<td>RMDDB46.TS46P01</td>
<td>1</td>
</tr>
<tr>
<td>RMDDB46.TS46P01</td>
<td>2</td>
</tr>
<tr>
<td>RMDDB46.TS46P01</td>
<td>3</td>
</tr>
<tr>
<td>RMDDB46.TS46P01</td>
<td>4</td>
</tr>
<tr>
<td>RMDDB46.TS46P02</td>
<td>1</td>
</tr>
<tr>
<td>RMDDB46.TS46P02</td>
<td>2</td>
</tr>
<tr>
<td>RMDDB46.TS46P02</td>
<td>3</td>
</tr>
<tr>
<td>RMDDB46.TS46P02</td>
<td>4</td>
</tr>
<tr>
<td>RMDDB46.TS46P02</td>
<td>5</td>
</tr>
<tr>
<td>RMDDB46.TS46P02</td>
<td>6</td>
</tr>
<tr>
<td>RMDDB46.TS46P02</td>
<td>7</td>
</tr>
<tr>
<td>RMDDB46.TS46P02</td>
<td>8</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>1</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>2</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>3</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>4</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>5</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>6</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>7</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>8</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>9</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>10</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>11</td>
</tr>
<tr>
<td>RMDDB46.TS46P03</td>
<td>12</td>
</tr>
<tr>
<td>RMDDB46.TS46S01</td>
<td>0</td>
</tr>
<tr>
<td>RMDDB46.TS46S02</td>
<td>0</td>
</tr>
<tr>
<td>RMDDB46.TS46S03</td>
<td>0</td>
</tr>
<tr>
<td>RMDDB46.XT460000</td>
<td>1</td>
</tr>
<tr>
<td>RMDDB46.XT460001</td>
<td>1</td>
</tr>
<tr>
<td>RMDDB46.XT460002</td>
<td>1</td>
</tr>
<tr>
<td>RMDDB46.XT460003</td>
<td>1</td>
</tr>
<tr>
<td>RMDDB46.XT460003</td>
<td>2</td>
</tr>
<tr>
<td>RMDDB46.XT460003</td>
<td>3</td>
</tr>
<tr>
<td>RMDDB46.XT460003</td>
<td>4</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>1</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>2</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>3</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>4</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>5</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>6</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>7</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>8</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>9</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>10</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>11</td>
</tr>
<tr>
<td>RMDDB46.XT460004</td>
<td>12</td>
</tr>
<tr>
<td>RMDDB46.XT460005</td>
<td>1</td>
</tr>
<tr>
<td>RMDDB46.XT460005</td>
<td>2</td>
</tr>
<tr>
<td>RMDDB46.XT460005</td>
<td>3</td>
</tr>
<tr>
<td>RMDDB46.XT460005</td>
<td>4</td>
</tr>
</tbody>
</table>

Sample ARMBRPR output
### Sample ARMBRPR output

<table>
<thead>
<tr>
<th>OBJECT NAME</th>
<th>IX SPACE</th>
<th>DSNUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMD.IC46N011</td>
<td>IC46N011</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IC46N021</td>
<td>IC46N021</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IC46N031</td>
<td>IC46N031</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>1</td>
</tr>
<tr>
<td>RMD.IC46P021</td>
<td>IC46P021</td>
<td>1</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>1</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>2</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>3</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>4</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>5</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>6</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>7</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>8</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>9</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>10</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>11</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>12</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>13</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>14</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>15</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>16</td>
</tr>
<tr>
<td>RMD.IC46P031</td>
<td>IC46P031</td>
<td>17</td>
</tr>
</tbody>
</table>

**RECOVERY START TIME**: 2014-08-21-13.29.49.000000

**GROUP**: RMD.CRC5

**INDEXES CHANGED WITHIN REQUESTED TIME RANGE**
** Figure 157: ARMCOPYD sample output—Objects Copied Within Requested Time Range **

** RECOVERY MANAGER FOR DB2 V11.2.00 - PROGRESS REPORT 08/21/2014 13:39:27 **

** BMC802201 RECOVERY MANAGEMENT FOR DB2 V11.2.00 **

RECOVERY START TIME 2014-08-21-13.29.49.000000

GROUP RMD.CRCS5

TABLESPACES COPIED WITHIN REQUESTED TIME RANGE

<table>
<thead>
<tr>
<th>OBJECT NAME</th>
<th>DSNUM</th>
<th>ICTYPE</th>
<th>STYPE</th>
<th>JOBNAME</th>
<th>TIMESTAMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMDB46.TS46N01</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.10.113878</td>
</tr>
<tr>
<td>RMDB46.TS46N01</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.31.05.794891</td>
</tr>
<tr>
<td>RMDB46.TS46N02</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.10.921262</td>
</tr>
<tr>
<td>RMDB46.TS46N02</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.31.05.803218</td>
</tr>
<tr>
<td>RMDB46.TS46N03</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.10.125532</td>
</tr>
<tr>
<td>RMDB46.TS46N03</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.31.05.785655</td>
</tr>
<tr>
<td>RMDB46.TS46N03</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.33.50.727419</td>
</tr>
<tr>
<td>RMDB46.TS46N03</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.32.32.363668</td>
</tr>
<tr>
<td>RMDB46.TS46N03</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.32.32.356191</td>
</tr>
<tr>
<td>RMDB46.TS46P02</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.32.32.349329</td>
</tr>
<tr>
<td>RMDB46.TS46P02</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.32.32.340573</td>
</tr>
<tr>
<td>RMDB46.TS46P02</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.32.32.795955</td>
</tr>
<tr>
<td>RMDB46.TS46P02</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.32.32.803917</td>
</tr>
<tr>
<td>RMDB46.TS46P02</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.32.32.907561</td>
</tr>
<tr>
<td>RMDB46.TS46P02</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.32.33.141120</td>
</tr>
<tr>
<td>RMDB46.TS46P03</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.32.33.348520</td>
</tr>
</tbody>
</table>
Chapter 25  ARMBRPR — Progress Reports

Sample ARMBRPR output
**Sample ARMBRPR output**

```
RMDB46.XT460008  0  F  RMDR10N1  2014-08-21-13.30.11.201409
RMDB46.XT460008  0  F  RMDR10N1  2014-08-21-13.31.06.255313
RMDB46.XT460009  0  F  RMDR10N1  2014-08-21-13.29.49.000000
RMDB46.XT460009  0  F  RMDR10N1  2014-08-21-13.30.11.590126

INDEXES COPIED WITHIN REQUESTED TIME RANGE

<table>
<thead>
<tr>
<th>OBJECT NAME</th>
<th>IX SPACE</th>
<th>DSNUM</th>
<th>ICTYPE</th>
<th>STYPE</th>
<th>JOBNAME</th>
<th>TIMESTAMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMD.IC46N011</td>
<td>IC46N011</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.966863</td>
</tr>
<tr>
<td>RMD.IC46N011</td>
<td>IC46N011</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.31.06.950454</td>
</tr>
<tr>
<td>RMD.IC46N021</td>
<td>IC46N021</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.915000</td>
</tr>
<tr>
<td>RMD.IC46N021</td>
<td>IC46N021</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.958077</td>
</tr>
<tr>
<td>RMD.IC46N031</td>
<td>IC46N031</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.12.183687</td>
</tr>
<tr>
<td>RMD.IC46N031</td>
<td>IC46N031</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.12.183687</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>0</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>2</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>3</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>4</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>5</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>6</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>7</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>8</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>9</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>10</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>11</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>12</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>13</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>14</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
<tr>
<td>RMD.IC46P011</td>
<td>IC46P011</td>
<td>15</td>
<td>F</td>
<td></td>
<td>RMDR10N1</td>
<td>2014-08-21-13.30.11.917751</td>
</tr>
</tbody>
</table>
```

646  *RECOVERY MANAGER for DB2 User Guide*
** Recovery Manager for DB2 V11.2.00 - Progress Report 08/21/2014 13:39:27 **

** BMC802201 RECOVERY MANAGEMENT FOR DB2 V11.2.00 **

** RECOVERY START TIME 2014-08-21-13.29.49.000000 GROUP RMD.CRCS5 **

### TABLESPACES NOT COPIED WITHIN REQUESTED TIME RANGE

<table>
<thead>
<tr>
<th>OBJECT NAME</th>
<th>DSNUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMDDB46.TS46N04</td>
<td>0</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>1</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>2</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>3</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>4</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>5</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>6</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>7</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>8</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>9</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>10</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>11</td>
</tr>
<tr>
<td>RMDDB46.TS46P04</td>
<td>12</td>
</tr>
<tr>
<td>RMDDB46.TS46S04</td>
<td>0</td>
</tr>
</tbody>
</table>

** RECOVERY START TIME 2014-08-21-13.29.49.000000 GROUP RMD.CRCS5 **

### INDEXES NOT COPIED WITHIN REQUESTED TIME RANGE

<table>
<thead>
<tr>
<th>OBJECT NAME</th>
<th>IX SPACE</th>
<th>DSNUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMD.IC46N041</td>
<td>IC46N041</td>
<td>0</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>1</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>2</td>
</tr>
<tr>
<td>RMD.IC46P041</td>
<td>IC46P041</td>
<td>3</td>
</tr>
</tbody>
</table>

---

**Figure 158: ARMNOCOP sample output—Objects Not Copied Within Requested Time Range**

Sample ARMNOCOP output — Objects Not Copied Within Requested Time Range

---

Chapter 25  ARMBRPR — Progress Reports  647
### Figure 159: ARMSUMRY sample output—Summary

<table>
<thead>
<tr>
<th>TABLESPACE PARTITIONS</th>
<th>OBJECTS</th>
<th>TOTAL BYTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECOVERED</td>
<td>34 (1%)</td>
<td>68 MB (1%)</td>
</tr>
<tr>
<td>NOT RECOVERED</td>
<td>11317 (99%)</td>
<td>106960 MB (99%)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>INDEX PARTITIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>RECOVERED</td>
</tr>
<tr>
<td>NOT REC/REBUILT</td>
</tr>
</tbody>
</table>

### Tablespace Partitions

<table>
<thead>
<tr>
<th>TABLESPACE PARTITIONS</th>
<th>OBJECTS</th>
<th>TOTAL BYTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANGED</td>
<td>68 (1%)</td>
<td>390 MB (1%)</td>
</tr>
<tr>
<td>UNCHANGED</td>
<td>11283 (99%)</td>
<td>106638 MB (99%)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>INDEX PARTITIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANGED</td>
</tr>
<tr>
<td>UNCHANGED</td>
</tr>
</tbody>
</table>

### Total Partitions

<table>
<thead>
<tr>
<th>TABLESPACE PARTITIONS</th>
<th>OBJECTS</th>
<th>TOTAL BYTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANGED</td>
<td>68 (1%)</td>
<td>390 MB (1%)</td>
</tr>
<tr>
<td>UNCHANGED</td>
<td>11283 (99%)</td>
<td>106638 MB (99%)</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>TABLESPACE PARTITIONS</th>
<th>OBJECTS</th>
<th>TOTAL BYTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>COPIED</td>
<td>160 (1%)</td>
<td>1241 MB (1%)</td>
</tr>
<tr>
<td>NOT COPIED</td>
<td>11191 (99%)</td>
<td>105787 MB (99%)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>INDEX PARTITIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>COPIED</td>
</tr>
<tr>
<td>NOT COPIED</td>
</tr>
</tbody>
</table>

### Total Partitions

<table>
<thead>
<tr>
<th>TABLESPACE PARTITIONS</th>
<th>OBJECTS</th>
<th>TOTAL BYTES</th>
</tr>
</thead>
<tbody>
<tr>
<td>COPIED</td>
<td>160 (1%)</td>
<td>1241 MB (1%)</td>
</tr>
<tr>
<td>NOT COPIED</td>
<td>11191 (99%)</td>
<td>105787 MB (99%)</td>
</tr>
</tbody>
</table>

---

**ESTIMATION NOTES**

The approximate I/O factor can be found in BMC COPY output DD ACPPRTnn or SYSPRINT.

Recovery time estimates require some DB2 catalog statistics.
IF ANY REQUIRED STATISTICS ARE ZERO, YOU WILL SEE MESSAGE
<NOS ESTIMATE | NO CATALOG STATS AVAILABLE>. IN THIS CASE,
YOU SHOULD UPDATE THE CATALOG STATISTICS FOR YOUR OBJECTS.

THE OLDEST STATS TIME FOUND IS 0001-01-01 00:00:00.000000
THE MOST RECENT STATS TIME FOUND IS 2016-06-21 20:12:05.671219

Figure 160: ARMBOERR sample output—Backout Recovery Exceptions

** RECOVERY MANAGER FOR DB2 V11.2.00 - PROGRESS REPORT 02/14/2014 12:57:09 **
** BMC802201 RECOVERY MANAGEMENT FOR DB2 V11.2.00**
** BMC802201 BMC RECOVERY FOR DB2 V11.2.00**

START TIME 2014-01-31 11:00:00.000000
GROUP RDAJTR.BY_TS2

TABLESPACES AND INDEXES BACKOUT RECOVERY EXCEPTIONS

<table>
<thead>
<tr>
<th>OBJECT NAME</th>
<th>DSNUM</th>
<th>ICTYPE</th>
<th>STYPE</th>
<th>JOBNAME</th>
<th>TIMESTAMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARMDBJTR.TS40P1</td>
<td>0</td>
<td>W</td>
<td>JTRREORG</td>
<td>2014-01-30 14:20:51.93773</td>
<td></td>
</tr>
</tbody>
</table>

Figure 161: ARMFWERR sample output—Forward Recovery Exceptions

** RECOVERY MANAGER FOR DB2 V11.2.00 - PROGRESS REPORT 02/14/2014 12:57:09 **
** BMC802201 RECOVERY MANAGEMENT FOR DB2 V11.2.00**
** BMC802201 BMC RECOVERY FOR DB2 V11.2.00**

START TIME 2014-01-31 11:00:00.000000
GROUP RDAJTR.BY_TS2

TABLESPACES AND INDEXES FORWARD RECOVERY EXCEPTIONS

<table>
<thead>
<tr>
<th>OBJECT NAME</th>
<th>DSNUM</th>
<th>ICTYPE</th>
<th>STYPE</th>
<th>JOBNAME</th>
<th>TIMESTAMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMDDB41.LS41G1</td>
<td>0</td>
<td></td>
<td>LOB OBJ</td>
<td>2014-01-30 14:24:46.06069</td>
<td></td>
</tr>
<tr>
<td>RMDDB41.XT410000</td>
<td>0</td>
<td></td>
<td>XML OBJ</td>
<td>2014-01-30 14:24:33.89135</td>
<td></td>
</tr>
</tbody>
</table>

Figure 162: ARMAUERR sample output—Backout Auto Estimation Exceptions

*********************BACKUP TIME ESTIMATES*********************

CABINET 00:15:20 SNAPSHOT **:**:** HYBRID TOTAL **:**:**
STANDARD 00:43:45 SNAPSHOT **:**:** HYBRID TOTAL **:**:**
CABINET 00:15:20 STANDARD **:**:** HYBRID TOTAL **:**:**

*********************RECOVER TIME ESTIMATES*********************

FORWARD 00:00:00 NO ESTIMATE - NO USABLE COPY FOR ONE OR MORE OBJECTS
BACKOUT 00:00:00 NO ESTIMATE - BACKOUT NOT VALID FOR ONE OR MORE OBJECTS
AUTO 00:00:00 NO ESTIMATE - OBJECTS HAVE BACKOUT AND FORWARD EXCEPTIONS

3) Sample for ARMAUERR:
** RECOVERY MANAGER FOR DB2 V12.1.00 - PROGRESS REPORT 06/22/2016 08:06:53 **
** BMC802201 RECOVERY MANAGEMENT FOR DB2 V12.1.00**
** BMC802201 BMC RECOVERY FOR DB2 V12.1.00**
Executing the JCL

This section describes special instructions or information required to run the ARMBRPR JCL.

- Ensure that you have the appropriate authorizations. See “Authorizations” on page 630 for required authorizations.

- No restart is available for ARMBRPR. You must resubmit the job after correcting any error conditions.
ARMBSDR—Extend recovery point at disaster recovery site

This chapter describes ARMBSDR—Extend recovery point at disaster recovery site.

About ARMBSDR

ARMBSRR generates the JCL for the ARMBSDR program if you specify the ARMBSRR option DREXTEND YES (“DB2 Version 10 and later” on page 681) to extend the recovery point at the disaster site. ARMBSDR finds the most recent BSDS and archive logs at the disaster recovery site (for each member if data sharing) and updates the BSDS. ARMBSDR also adds a new conditional restart control record to the BSDS.

If you specify DREXTEND YES, ARMBSRR generates a single step for ARMBSDR in the Phase 1 JCL if the all of the following conditions exist:

- Simulate is No.
- JCL Type is DR.
- MAXLOGJOBS is 1.
- Initialize Actives is not used.
- Mirroring is not used.

For data sharing mode only, ARMBSRR generates JCL for the ARMBCOR program at the beginning of all Phase 1 jobs except the Phase 1 job that contains ARMBSDR. ARMBCOR manipulates the value of the ARMBSDR member in the CNTL data set that drives a process to coordinate all Phase 1 jobs so that ARMBSDR processes all members. (The ARMBSDR member in the CNTL data set is for internal use only.) If ARMBSDR fails, correct the problem and resubmit that Phase 1 job. The rest of the Phase 1 jobs wait for ARMBSDR to complete successfully.
You should not modify the JCL or syntax generated for ARMBCOR and ARMBSDR.

The archive prefix is determined by scanning the LASTARC and DUAL data set names. The format of the data set names is determined by the ARCTSTMP and TSTAMP options in ARMSOPTS. Those options must be correct or the process of finding additional logs will fail.

### Authorizations

The following authorizations are required to execute the ARMBSDR program:

- APF authorization for the ARMBSDR program and the RECOVERY MANAGER load library
- EXECUTE authority on the RECOVERY MANAGER plan

### Specifying the ARMBSDR data set DD statements

This topic describes the data sets ARMBSDR uses.

Each data set is specified by a ddname (data definition name).

- *(required)* **ARMPRINT**
  The output for messages that are returned from RECOVERY MANAGER. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

- *(optional)* **ARMOPTS**
  The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARMSOPTS is used as the default option set name.

  You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

  ```
  //ARMOPTS  DD *
  ssid.configurationOption=value
  /*
  ```
- **(required) ARMMSG**

  The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of `hilvl.RMGR.ARMCNTL(ARMMSG)`. The data set must be allocated with `DISP=SHR`.

- **(optional) ARME**

  The output for compiler run time errors. If compiler errors are detected and ARME is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

- **(optional) ARMJU003**

  RECOVERY MANAGER writes the SYSPRINT from the IBM DB2 Change Log Inventory (DSNJU003) utility to this file if the DD statement exists. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

- **(required) ARMCOM**

  ARMBCOR and ARMBSDR batch programs communicate with each other by using the ARMBSDR member in the CNTL data set during Phase 1 execution of DR JCL for data sharing. (The ARMBSDR member in the CNTL data set is for internal use only.)

---

### Sample JCL

The following figure shows sample JCL for ARMBSDR.

**Figure 163: Sample ARMBSDR JCL**

```plaintext
//** *************************************************************** */
//** *************************************************************** */
//**                    EXTEND RECOVERY POINT AT THE DR SITE               */
//** *************************************************************** */
//** *************************************************************** */
//ARM00018 EXEC PGM=ARMBSDR,REGION=4M,COND=(4,LT)
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//    DD DISP=SHR,DSN=DSNEXIT
//    DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSG)
//ARMPRT DD SYSTOUT=*<--- optional file to capture all DSNJU003 output
//ARMCOM DD DISP=SHR,
//    DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMBSDR)
//ARMJU003 DD SYSTOUT=*<--- optional file to capture all DSNJU003 output
//ARMIN DD *
//DATASHARING YES
//CRESTART C3D6E87F80CC
//ARCHIVE1 SSID DHA1 LASTARC DSNHDA.DHA1.ARCHLOG1.A0014847
//                  DSNHDA.DHA1.ARCHLOG2.A0014847
```
Sample output

The following figure shows sample output from ARMBSDR with OFFSITE NO.

Figure 164: ARMBSDR Sample ARMPRINT for OFFSITE NO.
BMC80799I REPRO TO DSNDHA.DHA1.BSDS02J
BMC80799I IDCAMS REPRO RC: 0

BMC80799I CHANGE LOG INVENTORY COMMANDS:
DELETE DSNAME=DSNDHA.DHA1.LOGCOPY1.DS02
NEWLOG DSNAME=DSNDHA.DHA1.LOGCOPY1.DS02,COPY1
DELETE DSNAME=DSNDHA.DHA1.LOGCOPY1.DS03
NEWLOG DSNAME=DSNDHA.DHA1.LOGCOPY1.DS03,COPY1
DELETE DSNAME=DSNDHA.DHA1.LOGCOPY1.DS01
NEWLOG DSNAME=DSNDHA.DHA1.LOGCOPY1.DS01,COPY1
DELETE DSNAME=DSNDHA.DHA1.LOGCOPY2.DS02
NEWLOG DSNAME=DSNDHA.DHA1.LOGCOPY2.DS02,COPY2
DELETE DSNAME=DSNDHA.DHA1.LOGCOPY2.DS03
NEWLOG DSNAME=DSNDHA.DHA1.LOGCOPY2.DS03,COPY2
DELETE DSNAME=DSNDHA.DHA1.LOGCOPY2.DS01
NEWLOG DSNAME=DSNDHA.DHA1.LOGCOPY2.DS01,COPY2
CRESTART CREATE,ENDLRSN=C3F0D31BBF62
DELETE DSNAME=DSNDHA.DHA1.ARCHLOG1.A0013902
NEWLOG DSNAME=DSNDHA.DHA1.ARCHLOG1.A0013902,
STARTRBA=0076FE02E000,ENDRBA=0077001EDFFF,
STRTLRSN=C3F709C25707,ENDLRSN=C3F709CE4C68,
COPY1VOL=112021,UNIT=3490,
CATALOG=YES
DELETE DSNAME=DSNDHA.DHA1.ARCHLOG2.A0013902
NEWLOG DSNAME=DSNDHA.DHA1.ARCHLOG2.A0013902,
STARTRBA=0076FE02E000,ENDRBA=0077001EDFFF,
STRTLRSN=C3F709C25707,ENDLRSN=C3F709CE4C68,
COPY2VOL=111944,UNIT=3490,
CATALOG=YES
BMC80799I CHANGE LOG INVENTORY RC: 0

BMC80799I DATA SET DSNDHA.DHA2.ARCHLOG1.B0009489
BMC80799I REPRO TO DSNDHA.DHA2.BSDS01J
BMC80799I DATA SET DSNDHA.DHA2.ARCHLOG1.B0009489
BMC80799I REPRO TO DSNDHA.DHA2.BSDS02J
BMC80799I IDCAMS REPRO RC: 0

BMC80799I CHANGE LOG INVENTORY COMMANDS:
DELETE DSNAME=DSNDHA.DHA2.LOGCOPY1.DS02
NEWLOG DSNAME=DSNDHA.DHA2.LOGCOPY1.DS02,COPY1
DELETE DSNAME=DSNDHA.DHA2.LOGCOPY1.DS03
NEWLOG DSNAME=DSNDHA.DHA2.LOGCOPY1.DS03,COPY1
DELETE DSNAME=DSNDHA.DHA2.LOGCOPY1.DS01
NEWLOG DSNAME=DSNDHA.DHA2.LOGCOPY1.DS01,COPY1
DELETE DSNAME=DSNDHA.DHA2.LOGCOPY2.DS02
NEWLOG DSNAME=DSNDHA.DHA2.LOGCOPY2.DS02,COPY2
DELETE DSNAME=DSNDHA.DHA2.LOGCOPY2.DS03
NEWLOG DSNAME=DSNDHA.DHA2.LOGCOPY2.DS03,COPY2
DELETE DSNAME=DSNDHA.DHA2.LOGCOPY2.DS01
NEWLOG DSNAME=DSNDHA.DHA2.LOGCOPY2.DS01,COPY2
CRESTART CREATE,ENDLRSN=C3F0D31BBF62
DELETE DSNAME=DSNDHA.DHA2.ARCHLOG1.A0008489
NEWLOG DSNAME=DSNDHA.DHA2.ARCHLOG1.A0008489,
STARTRBA=004AF5AD2000,ENDRBA=004AF7C91FFF,
STRTLRSN=C3E131DDF412,ENDLRSN=C3F0D31BBF63,
COPY1VOL=144623,UNIT=3490,
CATALOG=YES
DELETE DSNAME=DSNDHA.DHA2.ARCHLOG2.A0008489
NEWLOG DSNAME=DSNDHA.DHA2.ARCHLOG2.A0008489,
STARTRBA=004AF5AD2000,ENDRBA=004AF7C91FFF,
STRTLRSN=C3E131DDF412,ENDLRSN=C3F0D31BBF63,
COPY2VOL=101898,UNIT=3490,
CATALOG=YES
BMC80799I CHANGE LOG INVENTORY RC: 0

BMC80571I PROGRAM COMPLETE RC = 0
Figure 165 on page 656 shows sample output from ARMBSDR with OFFSITE YES.

**Figure 165: ARMBSDR Sample ARMPRINT for OFFSITE YES**

```
DATASHARING NO
CRESTART 00769E7AA000
ARCHIVE3
SSID DEC2 LASTARC BMCARM.DEC2LOG3.D09085.T1029522.A0016927
    DUAL    BMCARM.DEC2LOG4.D09085.T1029522.A0016927
BMC80799I DETERMINE ARCHIVE PREFIX
BMC80799I SSID = DEC2, TSTAMP = Y, PREFIX = BMCARM.DEC2LOG3
BMC80799I SSID = DEC2, TSTAMP = Y, PREFIX = BMCARM.DEC2LOG4
BMC80799I LAST ARCHIVE FOUND = BMCARM.DEC2LOG3.D09086.T1025106.A0016928
BMC80799I LAST BSDS FOUND = BMCARM.DEC2LOG3.D09086.T1025106.B0016928
BMC80799I LAST HIST FOUND = BMCARM.DEC2LOG3.D09086.T1025106.H0016928
BMC80799I BMCARM.DEC2LOG3.D09086.T1025106.A0016928 STARTRBA = 00769E7AAC0B
BMC80799I BMCARM.DEC2LOG3.D09086.T1025106.A0016928 ENDRBA = 00769F7F144A
BMC80799I LAST ARCHIVE FOUND = BMCARM.DEC2LOG4.D09086.T1025106.A0016928
BMC80799I BMCARM.DEC2LOG4.D09086.T1025106.A0016928 STARTRBA = 00769E7AAC0B
BMC80799I BMCARM.DEC2LOG4.D09086.T1025106.A0016928 ENDRBA = 00769F7F144A
BMC80799I ORIGINAL CRESTART = 00769E7AA000
BMC80799I NEW CRESTART = 00769F7F2000
BMC80799I CHANGE LOG INVENTORY COMMANDS:
CRESTART CREATE,ENDRBA=00769F7F2000
DELETE DSNAME=DEC2CAT.ARCLOG1.D06356.T1029124.A0008538
    STARTRBA=00769E7AA000,ENDRBA=00769F7F1FFF,
    COPY1VOL=114245,UNIT=3490,
    CATALOG=YES
DELETE DSNAME=DEC2CAT.ARCLOG2.D07202.T1349414.A0011334
    STARTRBA=00769E7AA000,ENDRBA=00769F7F1FFF,
    COPY2VOL=106805,UNIT=3490,
    CATALOG=YES
BMC80799I CHANGE LOG INVENTORY RC: 0
BMC80571I PROGRAM COMPLETE RC = 0
```

Figure 166 on page 656 shows how the ARMBSDR program ends if additional archive logs are not found at the disaster recovery site.

**Figure 166: ARMBSDR sample output if no additional archive logs are found**

```
BMC80799I ORIGINAL CRESTART = 00767C1C5000
BMC80799I NEW CRESTART = 00767C1C5000
BMC80799W CRESTART NOT EXTENDED - NO NEED TO CONTINUE
BMC80571I PROGRAM COMPLETE RC = 4
```
Executing the JCL

This section describes special instructions or information required to run the ARMBSDR JCL.

- Ensure that you have the appropriate authorizations. See “Authorizations” on page 652 for required authorizations.

- No restart is available for ARMBSDR. You must resubmit the job after correcting any error conditions.
ARMBSET— OBJECTSET processing

This chapter describes ARMBSET— OBJECTSET processing.

About ARMBSET

ARMBSET uses stored information, pulls objects based on the specified OBJECTSET, and issues CHECK and REPAIR commands.

You can use ARMBSET to start, stop, quiesce, run statistics (RUNSTATS), or reset GRECP/LPL status.

The ARMBSET program is generated after a NGT Recover step to issue a CHECK or REPAIR for each object (table space and index) in an OBJECTSET. With the implementation of OBJECTSET for NGT Recover, ARMBSET is needed for the following reasons:

- The Check Pend Action option ("General recovery options" on page 847) does not generate a step for CHECK or REPAIR when OBJECTSET syntax is generated.

- You cannot use classic CHECK or REPAIR JCL, which would represent a static set of objects, because it may not match the contents of the OBJECTSET, which is dynamic.

Note

- With the ARMBSET program, if you choose a CHECK PEND ACTION of CHECK, in order for ARMBSET to generate the correct CHECK statements, you must run RUNSTATS for the table spaces in your group. If you do not, RECOVERY MANAGER might not generate all of the CHECK statements that are needed.

- If you have selected the BMC NGT Check for DB2 for z/OS product as the check utility in the general recovery options of an object set, ARMBSET invokes NGT Check to process CHECK DATA TABLESPACE for the supported objects in the object set.
Table spaces that have parents or hash tables will be selected to have CHECK DATA after they are recovered and when the group option Check Pend Action is CHECK.

**WARNING**

Do not run CHECK DATA on encrypted data. Because CHECK DATA does not decrypt the data, the utility might produce unpredictable results.

### Authorizations

The following authorizations are required to execute the ARMBSET program:

- APF authorization for the ARMBSET program and the RECOVERY MANAGER load library
- EXECUTE authority on the RECOVERY MANAGER plan

### Building the JCL

The generation of the NGT Recover job JCL completes with the automatic addition of ARMBSET and without any input or changes necessary on your part. The ARMBSET syntax is generated from information that is already available.

You will need to code only the LEVELID action. A sample job is located in the ARMBSET$.SAMP member in the .ARMSAMP data set that was created during installation. The syntax consists of a STOP, a START, and the action requested and the OBJECTSET name to which the action is to apply. All other syntax is pulled from existing information.

The ARMBSET JCL includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Output data sets
Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends that you specify REGION=0M, which makes the amount of virtual storage that is needed to run the job automatically available when the ARMBSET job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

Specifying the EXEC statement

The EXEC statement has the following format:

```
//stepname EXEC PGM=ARMBSET,PARM='ssid,ARMOPTS=optionSet, REGION=0M'
```

The variable `ssid` is the DB2 subsystem on which the program is executing. If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

**Note**

The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

The variable `optionSet` is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBSET to use. For example:

```
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// DD DISP=SHR,DSN=DSNLOAD
```
Specifying the ARMBSET data set DD statements

This subsection describes the data sets ARMBSET uses.

Each data set is specified by a *ddname* (data definition name). You must specify all required data sets in the JCL.

- *(required)* ARMPRINT
  
The output for messages that are returned from RECOVERY MANAGER. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

- *(optional)* ARMOPTS
  
The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.
  
You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

```
//ARMOPTS  DD *
 ssid.configurationOption=value
/*
```

- *(required)* ARMMSGS
  
The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of *hilvl.RMGR.ARM_CNTL(ARMMSGS)*. The data set must be allocated with DISP=SHR.

- *(optional)* ARMERROR
  
The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

Syntax diagrams

The ARMBSET syntax (the following figures) is generated by RECOVERY MANAGER. You will need to code the action for CHECK_PEND_ACTION only. Syntax descriptions can be found on the pages that are shown in parentheses.
Figure 167: ARMBSET control statement—SET CURRENT SQLID

SET CURRENT SQLID = sqlid ;

Figure 168: ARMBSET syntax

```
START
OBJECTSET objectSetName

STOP
OBJECTSET objectSetName

WAIT n

RUNSTATS
REPORT
SHRLEVEL
YES
NO

UPDATE
ALL
NONE
ACCESSPATH
SPACE

QUIESCE WRITE
YES
NO

RW
RO
UT
FORCE
```
Figure 169: Syntax for ARMBSET

**Syntax diagrams**

[Diagram showing syntax for ARMBSET]

**OBJECTSET objectSetName**

- **CHECK_PEND_ACTION**
  - 0 or UNSET
  - 1 or NONE
  - 2 or CHECK
  - 3 or REPAIR
  - 4 or LEVELID

- **RESET_GRECEP_LPL**

**OBJECT_PATTERN** databaseNamePattern.tableSpaceNamePattern *

* OBJECT_PATTERN is valid only for CHECK_PEND_ACTION LEVELID.
Figure 170: Syntax for ARMBSET — Options specification

Option descriptions

This section contains descriptions of syntax options.

**SET CURRENT SQLID**

In the ARMIN input data set, optionally provide the control statement SET CURRENT SQLID = sqlid to set the SQLID to be used for SQL execution.

The SQLID defaults to your user ID.

You can have multiple SET CURRENT statements in the control data set. The SET CURRENT SQLID statement is in effect for all statements that follow it until another SET CURRENT SQLID statement is issued.
STOP

Indicates the objects to be stopped.

START

Indicates the objects to be started.

WAIT

WAIT n specifies the number of seconds to wait. Valid values for n are 1 to 86400 seconds. If you provide no value, the default is 60 seconds.

QUIESCE WRITE

QUIESCE indicates objects to be quiesced.

This option, which defaults to YES, instructs DB2 to finish writing any pending transactions for the target spaces before applying the quiesce.

QUIESCE WRITE NO is ignored for objects having the attribute NOT LOGGED.

RUNSTATS

RUNSTATS allows you to collect DB2 statistics for the objects defined in the object set. The subordinate RUNSTATS options, REPORT, SHRLEVEL, and UPDATE, allow you to report the statistics in SYSPRINT and/or update the DB2 catalog with the statistics.

See the DB2 for z/OS Utility Guide and Reference for a description of the statistics produced by RUNSTATS.

REPORT

REPORT specifies whether RUNSTATS is to generate a set of messages that report the collected statistics.

REPORT YES

REPORT YES, the default value, indicates that collected statistics is sent as output via messages to SYSPRINT. The messages generated are dependent upon the combination of keywords specified with RUNSTATS. REPORT YES always generates a report of SPACE and ACCESSPATH statistics regardless of what UPDATE option specifies.

REPORT NO

REPORT NO indicates that collected statistics should not be output via messages to SYSPRINT.
**SHRLEVEL**

SHRLEVEL indicates whether other programs that access the table space while RUNSTATS is running must use read-only access or can change the table space.

**SHRLEVEL REFERENCE**

SHRLEVEL REFERENCE, the default value, allows only read-only access by other programs.

**SHRLEVEL CHANGE**

SHRLEVEL CHANGE allows other programs to change objects defined in the object set. With SHRLEVEL CHANGE, RUNSTATS might collect statistics on uncommitted data.

**UPDATE**

UPDATE indicates which collected statistics are to be inserted into the catalog tables.

**UPDATE ALL**

UPDATE ALL, the default value, indicates that all collected statistics in the DB2 catalog will be updated.

**UPDATE NONE**

UPDATE NONE indicates that no catalog tables are to be updated with the collected statistics.

Executing RUNSTATS always invalidates the dynamic cache; however, when you specify UPDATE NONE REPORT NO, RUNSTATS invalidates statements in the dynamic statement cache without collecting statistics, updating catalogs tables, or generating reports.

**UPDATE ACCESSPATH**

UPDATE ACCESSPATH indicates that DB2 is to update the catalog with only those statistics that are used for access path selection.

**UPDATE SPACE**

UPDATE SPACE indicates that DB2 is to update the catalog with only space-related statistics.
CHECK_PEND_ACTION

Indicates the action to take against the specified objects.

Use LEVELID (which is 4) to accept the use of a down-level data set. No other options are valid when LEVELID is requested.

RECOVERY MANAGER does no checking to see if the object is in logical page list status (LPL) or has outstanding INDOUBT transactions. RECOVERY MANAGER handles the error, prints out the results, and continues with the next object.

--- WARNING ---
Accepting the use of a down-level data set might cause data inconsistencies.

RESET_GRECP_LPL

Indicates to search for GRECP and LPL statuses and fix them at remote site for data sharing environments.

OBJECTSET objectSetName

Specifies the objects by using the object set name.

Only the table spaces from the object set are processed. If you want to generate the command for indexes as well, use an object set name where the indexes are included in the object set.

OBJECT_PATTERN
databaseNamePattern.tableSpaceNamePattern

Specifies the objects by using the database name pattern and the table space name pattern.

The patterns cannot be delimited. RECOVERY MANAGER gets table spaces from the catalog.

OBJECT_PATTERN is valid only for CHECK_PEND_ACTION LEVELID.

RECOVERTYPE

This option is not valid when CHECK_PEND_ACTION is LEVELID.

You do not need to code this syntax. RECOVERTYPE is pulled from existing information.
BACKOUT

This option is not valid when CHECK_PEND_ACTION is LEVELID. You do not need to code this syntax. BACKOUT is pulled from existing information.

INDEX

This option is not valid when CHECK_PEND_ACTION is LEVELID. You do not need to code this syntax. INDEX is pulled from existing information.

CLONE

This option is not valid when CHECK_PEND_ACTION is LEVELID. You do not need to code this syntax. CLONE is pulled from existing information.

REPAIRTS

This option is not valid when CHECK_PEND_ACTION is LEVELID. You do not need to code this syntax. REPAIRTS is pulled from existing information.

Sample ARMBSET JCL

The following figure shows sample JCL for ARMBSET.

A sample job is located in the ARMBSET$.SAMP member in the .ARMSAMP data set that was created during installation.

Figure 171: Sample ARMBSET JCL

```sql
/* *************************************************************** */
/* *************************************************************** */
/*          RECOVERY MANAGER - V11.1.00 - BMC SOFTWARE, INC.        */
/* *************************************************************** */
/*             ARMBSET - CHECK/REPAIR VIA OBJECT SET               */
/*                FOR SELECTED TABLESPACES/INDEXES                 */
/* *************************************************************** */
/* *************************************************************** */
ARM0002 EXEC PGM=ARMBSET,
   PARM='DEDL,ARMOPTS=ARM$OPTS',
   REGION=(4,LT)
STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
   DD DISP=SHR,DSN=DSNEXIT
   DD DISP=SHR,DSN=DSNLOAD
ARMSGS DD DISP=SEQ,DSN=RMD.TEST1110.UDBCNTL(ARMSGS)
ARMPRINT DD SYSOUT=*
ARMERROR DD SYSOUT=*  
SYSPRINT DD DISP=(NEW,DELETE,DELETE),DSN=&amp;TEMP2,
   UNIT=SYSALLDA,SPACE=(TRK,(1,1))
SYSIN DD DISP=(NEW,DELETE,DELETE),DSN=&amp;TEMP1,
   UNIT=SYSALLDA,SPACE=(TRK,(1,1))
ARMIN DD *
SET CURRENT SQLID = RDATQG
STOP
OBJECTSET "RMD"."SET1D"
```
Sample ARMBSET output

The following figure shows sample output for ARMBSET.

** Figure 172: Sample ARMBSET output **

```
** RECOVERY MANAGER FOR DB2 V11.1.00 - BATCH OBJECTSET PROCESSING 01/30/2013 09:12:54 **
** BMCB02201 RECOVERY MANAGEMENT FOR DB2 V11.1.00 **

(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884

BMCB0223I MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMCB0223I MAINT: NO SCC PTFS APPLIED

BMCB0309I CONNECTED TO DB2 SSID = DEDL VERSION 0910 MODE = NFM

BMCB0799I OPTION SET ARM$OPTS LAST UPDATED 01/25/2013 09:06:04 BY RDAJBW3
BMCB0799I DB2 SSID = DC2A  STARTED TASK = DBCRI101  JOBID = 27327

SET CURRENT SQLID = RDATQG
;
BMCB05701 COMMAND COMPLETE RC = 0

CHECK_PEND_ACTION CHECK
RECOVERTYPE TOQUIESCE
BACKOUT AUTO
INDEX RECOVER
REPAIRS YES
OBJECTSET "RMD"."SET1D"
;
BMCB0799I PROCESSING OBJECTSET RMD.SET1D

REPAIR SET TABLESPACE RMDB48.BS48S03 NOAUXCHKP
Execute RC=0 Output:
    DSNU0001 030 09:12:55.68 DSNUGUTC - OUTPUT START FOR UTILITY, UTILID = RDATQG2.RMDSET1P
    DSNU1044I 030 09:12:55.78 DSNUGTIS - PROCESSING SYSIN AS EBCDIC
    DSNU050I 030 09:12:55.78 DSNUGUTC - REPAIR
    DSNU650I *DEDL 030 09:12:55.78 DSNUCBRS - SET TABLESPACE RMDB48.BS48S03 NOAUXCHKP
    DSNU684I *DEDL 030 09:12:55.78 DSNUCBRS - SET NOAUXCHKP OPERATION SUCCESSFUL
```
Executing the JCL

This section describes special instructions or information required to run the ARMBSET JCL.

- Ensure that you have the appropriate authorizations. See “Authorizations” on page 660 for required authorizations.

- No restart is available for ARMBSET. You must resubmit the job after correcting any error conditions.
ARMBSRR—System resource recovery

This chapter describes how to use the ARMBSRR program to create batch jobs locally for restoring DB2 system resources at a recovery site.

About ARMBSRR

The ARMBSRR program enables you to create batch jobs at the local site to restore DB2 system resources at the recovery site before you recover applications. The jobs execute in two phases at the recovery site, an initialization phase (Phase 1) followed by a recovery phase (Phase 2). The jobs do not require any ISPF facilities at either site.

In a non-data-sharing environment, you must run ARMBSRR for each DB2 subsystem that is included in your disaster recovery planning. BMC recommends that DB2 be active when you run the program. Otherwise, some steps in the process are not generated and others may not be optimized.

In a data sharing environment, you must run ARMBSRR once for each DB2 data sharing group that is included in your disaster recovery planning. DB2 must be active on the member on which you run ARMBSRR, but other members do not need to be active.

ARMBSRR supports data sharing when the data sharing group includes members with different IBM DB2 version levels. In these cases, you should execute ARMBSRR to connect to the member with the higher version level. For example, if a data sharing group has both DB2 Version 11 members and DB2 Version 12 members, you should execute ARMBSRR to connect to a DB2 Version 12 member.
Note

- ARMBSRR does not support Catalog/Directory copies by part.
- If you have a BMC Recovery for DB2 solution password, RECOVERY MANAGER writes the validation report to the new RECOVERY_EXCEPTIONS table for later reporting to the BMC Workbench for DB2 product. For more information, see “Job history: BMCARM.BMCRMD_RECOVERY_EXCEPTIONS table” on page 807.

Recovery simulation

The ARMBSRR program can generate JCL to simulate recovery of system resources for a disaster recovery.

Recovery simulation is a feature of the BMC BMC Recovery Management for DB2 solution and requires the solution password. The recovery simulation feature simulates all aspects of recovery up to, but not including, the actual I/O. You may find disaster recovery simulation useful in reducing your disaster recovery testing costs.

Note

- For recovery simulation, you must specify an output data set with a GDG specification with .SIM suffix or a suffix ending in SIM to ensure that the JCL is not mistaken for actual disaster recovery JCL. ARMBSRR issues an error message if you specify a data set without the .SIM or xxxxSIM suffix.
- BMC recommends that you review the volume specifications (especially the VOLSERS) for data sets (such as the alternative BSDS and active logs) that are generated by ARMBSRR to verify they are appropriate for your organizational standards. For example, you may not want the simulation to run on your production volumes.

The ARMBGEN program can simulate the recovery of application resources. For more information, see “ARMBGEN—Backup and recovery JCL” on page 431. Online support for both system and application recovery simulation is also available. For more information about simulation, see the Recovery Management for DB2 User Guide.
Recovery estimation

The ARMBSRR program can generate JCL to estimate recovery of system resources for a disaster recovery. Estimation is only available with the Recovery Management for DB2 solution. The recovery estimation feature provides an estimate in hours and minutes for the recovery time.

Note
For recovery estimation, you must specify an output data set with a GDG specification with .EST suffix or a suffix ending in EST to ensure that the JCL is not mistaken for actual disaster recovery JCL. ARMBSRR issues an error message if you specify a data set without the .EST or xxxxEST suffix.

Hardware mirroring support

RECOVERY MANAGER supports systems that include DASD hardware mirroring technology as part of their remote site recovery planning.

ARMBSRR supports two levels of hardware mirroring. The first level is for systems that mirror the DB2 BSDS and active logs. The second level is for systems that mirror the catalog and directory data sets in addition to the BSDS and active logs. The JCL generated by ARMBSRR bypasses those steps made unnecessary by the hardware mirroring.

Note
As a component of the Recovery Management for DB2 solution, RECOVERY MANAGER provides additional support for hardware mirroring in disaster recoveries. For more information, see the Recovery Management for DB2 User Guide.

You indicate the level of hardware mirroring using the online interface or by specifying the HWLEVEL option in the ARMBSRR JCL.

In addition, if your system is mirroring only one copy of the BSDS and active logs, you specify which offsite data set copy is being used by using the online interface or by specifying the option in the ARMBSRR JCL.

For systems using hardware mirroring, preparations for disaster recovery are somewhat different than those used for standard systems because updates are being made to the remote site in near real time. For Level 1 systems, you can run ARMBSRR after making backups of the catalog and directory data sets. For Level 2 systems, you can run ARMBSRR at a user-defined frequency.
Extending the recovery point at the disaster recovery site

RECOVERY MANAGER supports disaster recovery scenarios where the target application objects have been copied by methods other than DB2 (such as full volume dumps or XRC) and log only recovery is desired.

In these scenarios, you continue to ship archive logs to the disaster recovery site after running ARMBSRR at the local site. This type of recovery recognizes the additional logs and modifies the BSDS and the conditional restart point. To extend the recovery point at the disaster recovery site, RECOVERY MANAGER uses the following options and programs:

- **DREXTEND NO | YES option for ARMBSRR**
- **LOGONLY NO | YES option for ARMBGEN**

*Note*
DREXTEND YES does not require LOGONLY. But you should use LOGONLY YES with DREXTEND YES.

- **ARMBCOR program**—ARMBCOR manipulates the value of the ARMBSDR member in the CNTL data set to ensure that all data sharing members are processed. The JCL generated for ARMBCOR should not be modified.

- **ARMBSDR program**—The ARMBSDR program finds the most recent BSDS and archive logs at the disaster recovery site (for each member if data sharing) and updates the BSDS. ARMBSDR also adds a new conditional restart control record to the BSDS. For more information about ARMBSDR, see “ARMBSDR—Extend recovery point at disaster recovery site” on page 651.

About JES support

ARMBSRR supports both JES2 and JES3 systems by generating JCL that is optimized to use the job routing features of each.

**To enable JES support**

RECOVERY MANAGER assumes that each subsystem is running with JES2 and that the JES2 IDs are the same as the operating system IDs. If this is not true for your system, you must do one of the following steps:

- For data sharing JES3 systems, you must add the JES3NAME= option to the ARM $OPTS member of the .CNTL file for each DB2 subsystem.

- For data sharing JES2 systems, if the JES2 ID is different than the operating system ID, you must add the JES2NAME= option to the ARM$OPTS member of
the .CNTL file. (If the JES ID is the same as the operating system ID, you do not need to add this option.)

**Job routing cards**

ARMBSRR generates appropriate routing cards, as follows:

- For JES2 data sharing systems, the following is generated with the JESID:

  ```
  /*JOBPARM SYSAFF=ssid
  ```

- For JES3 data sharing systems, the following is generated with the JESID:

  ```
  //*MAIN SYSTEM=ssid
  ```

  **Note**
  
  For non-data-sharing systems, the /*JOBPARM and //*MAIN cards are not required.

---

**How ARMBSRR selects a subsystem recovery point**

The value of the default subsystem recovery point selected by ARMBSRR depends on the following items:

- Which version of DB2 is used and whether the mode is data sharing or non-data-sharing

- Which of the following types of archive log are sent to the recovery site:
  
  -- Recovery site log copy generated by ARMBARC (or PACLOG)
  
  -- One of the local site copies (as specified by OFFSITE NO ARCHIVE1 or ARCHIVE2)

- The parameters of ARMBSRR, which can specify a recovery point

  The default value of the recovery point determined by ARMBSRR is shown in Table 46 on page 677 for different DB2 and archive log scenarios.

**Table 46: Default subsystem recovery point selection**

<table>
<thead>
<tr>
<th>System Configuration</th>
<th>When OFFSITE YES is specified, ARMBSSR selects</th>
<th>When OFFSITE NO is specified, ARMBSSR selects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-data-sharing</td>
<td>The ENDRBA of the last archive log found in the archive history file</td>
<td>The ENDRBA of the last archive log found in the BSDS</td>
</tr>
</tbody>
</table>
You can override the default by specifying the parameter LASTRBA= or LASTLRSN= in your job EXEC statement, as follows:

- For a non-data-sharing environment, use LASTRBA to specify the hexadecimal value of the starting RBA of the archive log that you want RECOVERY MANAGER to use as the last archive log at the recovery site.

- For a data sharing environment, use LASTLRSN to specify the hexadecimal value of the starting LRSN of the archive log you want RECOVERY MANAGER to use as the last archive log at the recovery site.

- For coordinated recovery, set LASTRBA or LASTLRSN to the keyword CRRPOINT. ARMBSRR will locate the RBA or LRSN of the last CRRPOINT contained in the repository table. Optionally, you can use CRRPOINT (value) to give a specific point. The value is the RBA or LRSN in the table in hexadecimal format. ARMBCRC must have been run to update the table.

### ARMBSRR jobs

ARMBSRR generates the following sets of jobs to perform a conditional restart recovery of a DB2 subsystem or data sharing group:

- Phase 1 jobs—run while DB2 is down
- Phase 2 jobs—run after DB2 is restarted in MAINT mode.
- data collection jobs—after application data recovery (Recovery Management solution only)

The number of jobs in each phase varies depending on the number of members in the data sharing group and the options selected. Jobs also vary depending on whether you are using hardware mirroring. See “Functions accomplished during the initialization phase (Phase 1)” on page 683 and “Functions accomplished during the recovery phase (Phase 2)” on page 685.

For information about restarting failed jobs, see “Restarting failed recovery jobs” on page 222.
Phase 1 jobs—Initialization

For Phase 1, ARMBSRR generates at least one job per member of a data sharing group in order to perform the initialization for the members.

For non-data-sharing, it generates at least one job. If you are using the Recovery Management for DB2 solution, a data collection job is also placed on hold. Also, data collection information is written to a flat file during Phase 1 processing.

If you specify RESTORE ARCHIVE n to move the archive logs from tape to disk, ARMBSRR uses the value that you specify for MAXLOGJOBS to generate multiple ARMBARC jobs. This action increases the concurrency of the process and reduces the amount of time required to move the logs from tape to disk. ARMBSRR generates a maximum of 32 jobs for Phase 1.

A simplified formula for the number of Phase 1 jobs is as follows:

- For a data sharing group
  \[
  \text{number of members} \times \text{MAXLOGJOBS}
  \]

- For a non-data-sharing group
  \[
  \text{MAXLOGJOBS}
  \]
Note

- RECOVERY MANAGER performs stacked tape analysis prior to creating the Phase 1 JCL. The number of log copy jobs may vary based on this analysis and may possibly be less than you requested with the MAXLOGJOBS option.

- For simulation mode, the generated JCL acts on copies of the BSDS and logs, not on production files, and only those steps that process the BSDS or log files are executed. Any steps that might update other aspects of the system are automatically commented out and do not execute.

- For estimation mode, only the data collection steps are performed.

- If you use the INITIALIZE ACTIVES option, you can set the MAXLOGJOBS value to specify the number of jobs to use per member or per subsystem for active log initialization. If MAXLOGJOBS is set to one (the default), the initialization of the active logs is included in the initialization job of each member of the subsystem. For values greater than one, ARMBSRR generates at least one log data set initialization step per job, so the number of jobs is also limited by the number of active log data sets to be initialized.

- To extend the recovery point at the disaster recovery site, DREXTEND YES causes ARMBSRR to generate JCL to run the ARMBSDR program at the disaster recovery site as part of the Phase 1 job. (For more information, see “ARMBSRR syntax and option descriptions” on page 694.)

Phase 2 jobs—Recovery JCL generation

During Phase 2, ARMBSRR generates jobs as follows:

- Multiple jobs for data sharing
  For data sharing environments, ARMBSRR generates one job per member to perform recovery for the members.

- Multiple jobs, for DB2 9 and earlier, when you specify a MAXCATJOBS value greater than one
  If more than one job is generated, synchronization steps are embedded within each job at the necessary points.

- One job with multiple tasks, for DB2 Version 10 and later, when you specify a value for MAXCATJOBS greater than one
  RECOVERY MANAGER uses the value of MAXCATJOBS for PARALLEL and TAPEUNITS to perform multiple tasks in one job.
If you are using the BMC Recovery Management for DB2 solution, data collection is performed throughout Phase 2.

**DB2 Version 10 and later**

Because DB2 Version 10 does not allow multiple simultaneous catalog recoveries, ARMBSRR does not generate multiple catalog recovery jobs even if MAXCATJOBS is greater than 1 when you are running under DB2 Version 10 or later. Instead, ARMBSRR generates a single catalog recovery job using the value that you specify for MAXCATJOBS in the PARALLEL and TAPEUNITS syntax. The result enables multitasking by DSNUTILB recovery instead of multiple jobs.

**Tasks performed for job generation**

To create the Phase 1 and Phase 2 jobs, the ARMBSRR program performs the following tasks:

- Captures information by examining the integrated catalog facility (ICF) catalog, the RECOVERY MANAGER option set, the last archive log data set, the copy of the Boot Strap Data Set (BSDS) on the most recent archive log data set, and the current BSDS

- Determines any INDOUBT transactions that may exist at the end of the last archive log data set

  ARMBSRR generates a step at the beginning of the Phase 2 job to process indoubt threads. This step invokes ARMBRID. For information on ARMBRID, see “ARMBRID—Recover indoubt threads” on page 623.

- Analyzes the ICF catalog to determine the allocations for the following items:
  - BSDSs
  - Active log data sets
  - Catalog and directory
  - Temporary work file databases
  - BMC Common DB2, RECOVERY MANAGER, and R+/CHANGE ACCUM repositories
  - Log Master repository

- Examines the DB2 catalog to determine
— Which objects compose the BMC Common DB2, RECOVERY MANAGER, and R+/CHANGE ACCUM repositories

— Which objects compose the Log Master repository

— Which objects compose the temporary work file database

— Which image copies will be used for recovery

Note
ARMBSRR can obtain this information only when the target DB2 subsystem is active. If the DB2 is not active when you run ARMBSRR, the corresponding steps are omitted from the generated JCL, and a warning message is generated.

■ Examines the archive history file to determine the names of the recovery site copies of the archive log data sets and the image copies available for the special spaces.

■ Performs stacked tape analysis of copies of the following items:
  — The catalog and directory spaces
  — The BMC Common DB2, RECOVERY MANAGER, and R+/CHANGE ACCUM repositories
  — The Log Master repository

■ Examines the repository (subsystem utility options) to determine the work unit and space allocation information for the recovery of the following items:
  — The DB2 catalog and directory
  — The BMC Common DB2, RECOVERY MANAGER, and R+/CHANGE ACCUM repositories
  — The Log Master repository

If DB2 is down when you execute the ARMBSRR program, the work unit is determined from the option set, and the space allocations default to CYL(10, 20).

■ Selects a recovery point for use by ARMBGEN with the TORESTARTRBA option for the recovery of applications at the recovery site.
  This recovery point is stored in the archive history file.

■ Performs multiple job optimization for the catalog and directory recovery

■ Generates an ARMBSET step to reset GRECP and LPL status to run after catalog/directory recovery at the DR site for data sharing systems.
Important recommendations:

BMC recommends the following items:

- Run ARMBSRR only when all target DB2 subsystems are active.

- For a data sharing group, the member that you are running on must be active before all information can be obtained. Otherwise, ARMBSRR is unable to identify all of the members of the group.

- The ARMBSRR program uses the table SYSIBM.SYSPLANDEP when executing.

- Use ARMBARC or PACLOG to identify image copies for SYSCOPY, SYSUTILX, and DSNDB01 in order to generate the most optimal JCL. See “HISTONLY” on page 415 for more information.

- ARMBSRR supports data sharing when the data sharing group includes members with different IBM DB2 version levels. In these cases, you should execute ARMBSRR to connect to the member with the higher version level. For example, if a data sharing group has both DB2 Version 11 members and DB2 Version 12 members, you should execute ARMBSRR to connect to a DB2 Version 12 member.

Functions performed by recovery jobs

The following lists show the major functions that the system resource recovery jobs accomplish at the recovery site.

**Note**

For simulation mode, the generated JCL acts on working copies of the BSDS and logs, not on production files, and only those steps that process the BSDS or log files are executed. Any steps that might update other aspects of the system are automatically commented out and do not execute.

Functions accomplished during the initialization phase (Phase 1)

During the initialization phase, the following functions are performed for each non-data-sharing subsystem.

Whether they are performed on only one or all members of a data sharing group is shown in parentheses. For simulation mode, only steps 1-6 and 8 are performed.
<table>
<thead>
<tr>
<th>Step</th>
<th>Systems without hardware mirroring</th>
<th>Systems with level 1 hardware mirroring</th>
<th>Systems with level 2 hardware mirroring</th>
<th>Systems with level 3, 4, or 5 hardware mirroring</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Delete and define each BSDS (all)</td>
<td>Verify mirroring of system resources (Recovery Management solution only) b</td>
<td>Verify mirroring of system resources (Recovery Management solution only) b</td>
<td>Verify mirroring of system resources (Recovery Management solution only) b</td>
</tr>
<tr>
<td>2</td>
<td>Delete and define each active log (all)</td>
<td>Delete and define each BSDS c (all)</td>
<td>Delete and define each BSDS c (all)</td>
<td>Delete and define each BSDS c (all)</td>
</tr>
<tr>
<td>3</td>
<td>Recover each BSDS from the last archive log data set (all)</td>
<td>REPRO new BSDS from the mirrored copy c (all)</td>
<td>REPRO new BSDS from the mirrored copy c (all)</td>
<td>REPRO new BSDS from the mirrored copy c (all)</td>
</tr>
<tr>
<td>4</td>
<td>Use <strong>Change Log Inventory</strong> to rename the archive log data sets (if required) and to add the last archive log data set to the BSDSs (all)</td>
<td>Delete and define each active log c (all)</td>
<td>Delete and define each active log b (all)</td>
<td>Delete and define each active log b (all)</td>
</tr>
<tr>
<td>5</td>
<td>Use <strong>Change Log Inventory</strong> to generate the CRCR (all)</td>
<td>REPRO new active logs from the mirrored copy (all)</td>
<td>REPRO new active logs from the mirrored copy (all)</td>
<td>REPRO new active logs from the mirrored copy (all)</td>
</tr>
<tr>
<td>6</td>
<td>Use <strong>Change Log Inventory</strong> to add the active logs to the BSDS. Optionally initialize all but one of the active log data sets (all)</td>
<td>Delete and define catalog and directory spaces (one)</td>
<td>Print Log Map (all)</td>
<td>Print Log Map (all)</td>
</tr>
<tr>
<td>7</td>
<td>Delete and define catalog and directory spaces (one) d</td>
<td>Print Log Map (all)</td>
<td>Delete and recreate the history file</td>
<td>Delete and recreate the history file</td>
</tr>
<tr>
<td>8</td>
<td>Print Log Map (all)</td>
<td>Delete and recreate the history file</td>
<td>Restore the archive logs to disk (optional) (all)</td>
<td>Restore the archive logs to disk (optional) (all)</td>
</tr>
<tr>
<td>9</td>
<td>Delete and recreate the history file</td>
<td>Restore the archive logs to disk (optional) (all)</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>10</td>
<td>Execute ARMBEOL for coordinated recovery (non-data-sharing only)</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>11</td>
<td>Execute ARMBSDR to extend the recovery point at the disaster recovery site (optional)</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>12</td>
<td>Restore the archive logs to disk (optional) (all)</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>
## Functions accomplished during the recovery phase (Phase 2)

During the recovery phase, the following functions are performed for each non-data-sharing subsystem.

Whether they are performed on only one or all members of a data sharing group is shown in parentheses. When running in simulation mode, this phase does not execute.

### Table 48: Functions accomplished during Phase 2

<table>
<thead>
<tr>
<th>Step</th>
<th>Systems without hardware mirroring</th>
<th>Systems with level 1 hardware mirroring</th>
<th>Systems with level 2 hardware mirroring</th>
<th>Systems with level 3, 4, or 5 hardware mirroring</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Recover INDOUBT transactions (all)</td>
<td>Recover SYSUTIL(X), DBD01, and SYSDBDXA using DB2 RECOVER (one)</td>
<td>Terminate utility jobs other than COPY, REORG, LOAD, RECOVER, and MODIFY (all)</td>
<td>Terminate utility jobs other than COPY, REORG, LOAD, RECOVER, and MODIFY (all)</td>
</tr>
<tr>
<td>2</td>
<td>Recover SYSUTIL(X), DBD01, and SYSDBDXA using DB2 RECOVER (one)</td>
<td>Terminate utility jobs other than COPY, REORG, LOAD, RECOVER, and MODIFY (all)</td>
<td>Start all catalog and directory spaces in read/write mode (one)</td>
<td>Start all catalog and directory spaces in read/write mode (one)</td>
</tr>
<tr>
<td>Step</td>
<td>Systems without hardware mirroring</td>
<td>Systems with level 1 hardware mirroring</td>
<td>Systems with level 2 hardware mirroring</td>
<td>Systems with level 3, 4, or 5 hardware mirroring</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------</td>
<td>---------------------------------------</td>
<td>---------------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>3</td>
<td>Terminate utility jobs other than COPY, REORG, LOAD, RECOVER, and MODIFY (all)</td>
<td>Recover the catalog and directory spaces (one)</td>
<td>Initialize the temporary work file database if DB2 is active (all)</td>
<td>Initialize the temporary work file database if DB2 is active (all)</td>
</tr>
<tr>
<td>4</td>
<td>Recover SYSCOPY, SYSLGRNIX, SYSTSSTG, SYSTSVOL (one)</td>
<td>Start all catalog and directory spaces in read/write mode (one)</td>
<td>Define the temporary database table spaces.</td>
<td>Define the temporary database table spaces.</td>
</tr>
<tr>
<td>5</td>
<td>Terminate all remaining utilities (all)</td>
<td>Initialize the temporary work file database if DB2 is active (all)</td>
<td>Display, then terminate all remaining utilities (all)</td>
<td>Display, then terminate all remaining utilities (all)</td>
</tr>
<tr>
<td>6</td>
<td>Recover remaining catalog and directory spaces (one)</td>
<td>Define the temporary database table spaces.</td>
<td>Recover the BMC Common DB2, RECOVERY MANAGER, and R+/CHANGE ACCUM repositories (optional) (one)</td>
<td>Recover the BMC Common DB2, RECOVERY MANAGER, and R+/CHANGE ACCUM repositories (Level 3 or 5) (optional) (one)</td>
</tr>
<tr>
<td>7</td>
<td>Start all catalog and directory spaces in read/write mode (one)</td>
<td>Display, then terminate all remaining utilities (all)</td>
<td>Recover the Log Master repository (optional) (one)</td>
<td>Recover the Log Master repository (Level 4 or 5) (optional) (one)</td>
</tr>
<tr>
<td>8</td>
<td>Initialize the temporary work file database if DB2 is active (all)</td>
<td>Recover the BMC Common DB2, RECOVERY MANAGER, and R+/CHANGE ACCUM repositories (optional) (one)</td>
<td>Stop DB2 (all)</td>
<td>Stop DB2 (all)</td>
</tr>
<tr>
<td>9</td>
<td>Define the temporary database table spaces.</td>
<td>Recover the Log Master repository (optional) (one)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Recover the BMC Common DB2, RECOVERY MANAGER, R+/CHANGE ACCUM, and Log Master repositories (optional) (one)</td>
<td>Clear BMCUTIL and BMCSYNC (one) Stop DB2 (all)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Terminate all remaining utilities (all)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Step</td>
<td>Systems without hardware mirroring</td>
<td>Systems with level 1 hardware mirroring</td>
<td>Systems with level 2 hardware mirroring</td>
<td>Systems with level 3, 4, or 5 hardware mirroring</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------</td>
<td>----------------------------------------</td>
<td>----------------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>12</td>
<td>Stop DB2 (all)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

a Levels 3, 4, and 5 are features available only with the BMC Recovery Management for DB2 solution. For more information, see the Recovery Management for DB2 User Guide.

b The BMC Common DB2 repository, RECOVERY MANAGER repository, R+/CHANGE ACCUM repository, and Log Master repository are not included in the mirrored JCL at levels 3 and 5. If nonmirrored JCL is generated, the repositories are included in the recovery.

**ARMBSRR-generated JCL**

ARMBSRR might generate JCL for the following RECOVERY MANAGER programs:

- **ARMBCOR** manipulates the value of the ARMBSDR member in the CNTL data set to ensure that all data sharing members are processed. The JCL generated for ARMBCOR should not be modified. ARMBCOR supports the feature to extend the recovery point at the disaster recovery site.

- **ARMBEOL** is used in non-data-sharing systems when ARMBSRR is executed with PARM LASTRBA=CRRPOINT or PARM LASTLRSN=CRRPOINT. ARMBEOL coordinates recoveries and truncates archive logs to the point specified in the CRRDRPT table.

- **ARMBSDR** finds the most recent BSDS and archive logs at the disaster recovery site (for each member if data sharing) and updates the BSDS. ARMBSDR also adds a new conditional restart control record to the BSDS. ARMBSDR supports the feature to extend the recovery point at the disaster recovery site.

- **ARMBSTP** stops and starts activity against table spaces where required.

- **ARMBTRM** terminates utilities running against the catalog and directory.

- **ARMBUTL** terminates BMC utilities.

- **ARMBWDC** — writes data collection information about recovery start and end times for actual, estimated, and simulated disaster recoveries of system resources. This program is only available with the BMC Recovery Management for DB2 solution.

- **ARMBRDC** — reads data collection information about the disaster recovery jobs and populates the UTILITY_RUN table. It also generates the data collection reports. This program is only available with the BMC Recovery Management for DB2 solution.
ARMBRID — recover indoubt threads.

If you are using the BMC Recovery Management for DB2 solution, ARMBSRR also generates JCL for Log Master (ALPMAIN) to create a data collection that you can use to populate the data collection tables at the local site.

## Authorizations

The following authorizations are required to execute the ARMBSRR program:

- APF authorization for the ARMBSRR program and the RECOVERY MANAGER load library
- EXECUTE authority on the RECOVERY MANAGER DB2 plan
- READ authority for the archive log data sets and BSDSs
- READ authority for the ICF catalog
- READ authority for the archive history file
- EXECUTE authority to run report recovery on DSNDB06.

**Note**

Installation SYSADM authority is required to run the output jobs created by ARMBSRR.

## Building the JCL

Building your own ARMBSRR job to generate JCL to recover the DB2 subsystem involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Input data sets
  - Output data sets
Specifying the JOB statement

This topic describes specifying the JOB statement.

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run (see “Job card (JCARD) specification” on page 52).

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends you specify REGION=0M, which makes the amount of virtual storage needed to run the job automatically available when the ARMBSRR job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

You can optionally include the following substitution variables in the job card in addition to the standard symbolic variables:

- &MBRNAME--the member name that is being processed for a data sharing group or the SSID for non-data sharing systems.
- &PHASE--the processing phase. Valid values are 1, 2, or 3.

Example

//J&MBRNAME&PHASE&& JOB

For subsystem DEDL, Phase 1, job 01, this job card generates the following job name: //JDEDL101 JOB

Specifying the EXEC statement

The EXEC statement has one of the following formats.

Note

For all formats, the variable ssid is the DB2 subsystem ID or data sharing group attach name where the RECOVERY MANAGER groups reside.

If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list. The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

For all formats, the variable optionSet is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.
If you want to use the last archive log to determine the recovery point, or if you are using hardware mirroring, use the following statement:

```//stepname EXEC PGM=ARMBSRR, // PARM='ssid,ARMOPTS=optionSet', // REGION=OM```

If you want to use a specified RBA as the disaster recovery point, use the following statement:

```//stepname EXEC PGM=ARMBSRR, // PARM='ssid,LASTRBA=startRba,ARMOPTS=optionSet' // REGION=OM```

LASTRBA tells ARMBSRR to find the log beginning with the specified `startRba` and use it to determine the disaster recovery point.

**Note**
If DB2 is in data sharing mode, you must use LASTLRSN instead of LASTRBA.

If you want to use a specified coordinated disaster recovery point or perform a conditional restart to a point in time (inserted by ARMBTSI and resolved by ARMBCRC), use the following statement:

```//stepname EXEC PGM=ARMBSRR, // PARM='ssid,LASTRBA=CRRPOINT,ARMOPTS=optionSet' // REGION=OM```

LASTRBA=CRRPOINT tells ARMBSRR to use the latest recovery point registered in the BMCARM_CRRDRPT table to determine the disaster recovery point.

If you want to specify the RBA to be used for the coordinated disaster recovery point or perform a conditional restart to a point in time (inserted by ARMBTSI and resolved by ARMBCRC), use the following statement:

```//stepname EXEC PGM=ARMBSRR, // PARM='ssid,LASTRBA=CRRPOINT(hexRba),ARMOPTS=optionSet' // REGION=OM```

The variable `hexRba` is an entry in the repository that is set by the ARMBCRC program.

**Note**
If DB2 is in data sharing mode, you must use LASTLRSN instead of LASTRBA, and you must use a `hexLrsn` entry instead of `hexRba`.

---

**Specifying the STEPLIB DD statement**

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBSRR to use. For example:

```//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS // DD DISP=SHR,DSN=DSNEXIT // DD DISP=SHR,DSN=DSNLOAD```
Specifying the ARMBSRR data set DD statements

This section describes the data sets ARMBSRR uses.

Each data set is specified by a `ddname` (data definition name). You must specify all required data sets in the JCL.

- **(required) ARMIN**
  The input data set that contains one or more control statements. Attributes for this data set must be fixed length records, with length of 80 (RECFM=F or FB, LRECL=80).

- **(required) ARMPRINT**
  The output for messages that are returned from RECOVERY MANAGER. RECOVERY MANAGER also echoes the contents of the ARMIN data set in the ARMPRINT output. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.

- **(optional) ARMOPTS**
  The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

  You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

  ```
  //ARMOPTS  DD *
  ssid.configurationOption=value
  /*
  ```

- **(required) ARMMSGS**
  The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of hilv/RMGR.ARMCNTL(ARMMSGS). The data set must be allocated with DISP=SHR.

- **(optional) ARMERROR**
  The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

- **(required) ARMJCIN**
  The data set that contains a job card for use in the generated JCL. The job name must contain the symbolic &##. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80). The user ID specified on the job card should have installation SYSADM authority.
(required) ARMJCL

The output data set that contains the JCL required to recover your DB2 subsystem. This data set must be sent offsite with your other recovery resources. BMC recommends that you use a generation data group (GDG). Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

Be aware of the following information about the ARMJCL data set:

— For systems with mirroring, this data set contains JCL as follows:

  — If all included objects can be verified as mirrored, the JCL generated into this data set omits mirrored objects (because they are assumed to be already available at the recovery site).

  — If any objects cannot be verified as mirrored, the JCL generated into this data set includes all objects, whether mirrored or not.

— For recovery simulation, you must specify a data set with a .SIM suffix or a suffix ending in SIM to ensure that the JCL is not mistaken for actual disaster recovery JCL. ARMBSRR issues an error message if you specify a data set without the .SIM or xxxSIM suffix.

— For recovery estimation (Recovery Management for DB2 solution only), you must specify a data set with a .EST suffix or a suffix ending in EST to ensure that the JCL is not mistaken for actual disaster recovery JCL. ARMBSRR issues an error message if you specify a data set without the .EST or xxxxEST suffix.

(optional) ARMJCL2

If this DD statement is included in the job, ARMBSRR writes a second copy of the generated recovery JCL in this data set. The data set can be sent offsite to provide a fallback in case the data set created in ARMJCL is unusable. BMC recommends allocating to a GDG data set and sending it offsite without your other recovery resources. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

(optional) ARMJCL3

This data set contains the alternate JCL that is generated if:

— Mirroring options are set at the subsystem level
— All objects were verified as successfully mirrored

The alternate JCL in this data set recovers all objects whether mirrored or not. It is used as a fallback in the event that mirroring fails and the mirrored JCL generated in the ARMJCL data set cannot be used.
Tip

BMC recommends allocating to a GDG data set and sending it offsite without your other recovery resources. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

- *(optional)* ARMJCL4

  If this DD statement is included in the job, ARMBSRR writes a second copy of the alternate recovery JCL data set (ARMJCL3). The data set can be sent offsite to provide a fallback in case the data set created in ARMJCL3 is unusable. BMC recommends allocating to a GDG data set and sending it offsite without your other recovery resources. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

- *(required)* ARMLOAD

  Specifies the load library that contains the RECOVERY MANAGER load modules. The data set must be allocated with DISP=SHR.

  **Note**

  The ALTLOAD syntax option can be used to specify an alias for the RECOVERY MANAGER load library. If you want to use an alias, delete or comment out the ARMLOAD DD statement and specify the ALTLOAD option in the syntax.

- *(optional)* ARMPICK

  The output data set that contains a report (pick list) of tape volumes that are required for recovery of the DB2 catalog and directory and the repository. This data set also finds the volumes on which the related log-data for the catalog and directory and repositories reside and adds any non-DASD volumes to the PICKLIST report. This report enables tape operators to retrieve the tapes required for recovery before the Phase 2 jobs execute. BMC recommends that you allocate to a GDG data set and send it offsite with your other recovery resources. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

- *(for local PIT recovery only)* ARMWPEND

  This statement instructs ARMBSRR to analyze for any objects in WRITE PENDING status so that they can be included in a local PIT recovery even if their status is UNCHANGED. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

- *(optional)* ARMVRPT

  This statement instructs ARMBSRR to generate a system validation report. This report lists the recovery point, the image copies required to recover to that point, and information about the BSDS and archive logs required to recovery the DB2 subsystem. If mirroring is enabled, the mirror status for the system data sets is also included. The ARMVRPT DD can be allocated to SYSOUT. If this DD is not...
present, the information is printed at the end of ARMPRINT. See “Sample output” on page 712 for a sample of the report.

■ (optional—Recovery Management solution only) ARMDDL
This statement points to the file built by Log Master that contains any DDL found between the specified recovery point and the current time current. If this DD statement is not present in the JCL, ARMBSRR sets a flag indicating that the catalog and directory must be recovered. For more information about conditional restart avoidance processing during local full subsystem recovery, see the Recovery Management for DB2 User Guide.

■ (optional—Recovery Management solution only) ARMQPT
This statement points to the file built by Log Master that contains quiet points found in the specified recovery range. If this DD statement is not present in the JCL, ARMBSRR sets a flag indicating that no quiet points were found. For more information about conditional restart avoidance processing during local full subsystem recovery, see the Recovery Management for DB2 User Guide.

ARMBSRR syntax and option descriptions

The ARMBSRR syntax and option descriptions in this section are the control statements that you use when you build ARMIN input.

For more information about online support, see “Creating a system resource recovery job (ARMBSRR)” on page 320.

Note
You must insert the SET OPTIONS statement before any other ARMBSRR syntax. For more information on syntax rules and wildcard support, see “Syntax rules” on page 102.
Figure 173 on page 695 shows the ARMBSRR Syntax.

Figure 173: ARMBSRR syntax

- **SET OPTIONS**
- **JCLTYPE**
- **DR**
- **LOCAL**
- **SINGLE LPAR**
- **OFFSITE**
  - **YES**
  - **NO**
  - **ARCHIVE2**
  - **ARCHIVE1**
- **RESTORE ARCHIVE1**
- **RESTORE ARCHIVE2**
- **INITIALIZE ACTIVES**
- **MAXLOGJOBS n**
- **CREATEHIST**
- **BSDS ARCHLIMIT n**
- **RECOVER REPOSITORY**
- **RECOVER LOGMASTER**
- **MAXCATJOBS n**
- **MAXLOGJOBS n**
- **COPYTYPE**
  - **LP**
  - **LB**
  - **RP**
  - **RB**
- **CONDRESTART**
  - **YES**
  - **NO**
  - **AUTO**
- **DREXTEND**
  - **NO**
  - **YES**
- **ALT dsName**
- **SYNCHNAME dsName**

*Recovery Management solution only*
Figure 174: ARMBSRR syntax

```
HWLEVEL 1
  ▼
  HWCOPY 1
  ▼
  HWCOPY 2
  ▼

HWLEVEL 2
  ▼
  HWCOPY 1
  ▼
  HWCOPY 2

EXCLUDE MEMBERS — mem1, mem2 —

BYPASS QUIESCED
  ▼
  VALIDATE_WARN — VALIDATE_WARN

ARCHIVE NOTCATLG
  ▼
  ESTIMATE
    ▼
    NO — YES

SIMULATE
  ▼
  NO — YES

MISSINGCOPIES
  ▼
  FAIL — WARN

PRIMEALLOC
  ▼
  ASIS — ALL

BYPASS DEACT

DCTOKEN
  ▼
  NO

DRECOVER

RECOVER WORKFILE
  ▼
  YES — NO

DSNTEP2LIB dataSetName

;```

RECOVERY MANAGER for DB2 User Guide
Figure 175: ARMBSRR syntax—Restore archive options

Table 49 on page 697 lists the syntax options in alphabetical order.

Table 49: ARMBSRR syntax

<table>
<thead>
<tr>
<th>Option</th>
<th>Batch Default</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTLOAD</td>
<td>None</td>
<td>“ALTLOAD” on page 707</td>
</tr>
<tr>
<td>ARCHIVE NOTCATLG</td>
<td>None</td>
<td>“ARCHIVE NOTCATLG” on page 708</td>
</tr>
<tr>
<td>BDS ARCHLIMIT</td>
<td>None</td>
<td>“BDS ARCHLIMIT” on page 703</td>
</tr>
<tr>
<td>BDS DAYSLIMIT</td>
<td>None</td>
<td>“BDS DAYSLIMIT” on page 703</td>
</tr>
<tr>
<td>BDS HOURS LIMIT</td>
<td>None</td>
<td>“BDS HOURS LIMIT” on page 704</td>
</tr>
<tr>
<td>BYPASS DEACT</td>
<td>None</td>
<td>“BYPASS DEACT” on page 708</td>
</tr>
<tr>
<td>BYPASS QUIESCED</td>
<td>VALIDATE_WARN</td>
<td>“BYPASS QUIESCED” on page 708</td>
</tr>
<tr>
<td>COMPRESS</td>
<td>None</td>
<td>“RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700</td>
</tr>
<tr>
<td>CONDRESTART</td>
<td>None</td>
<td>“CONDRESTART” on page 705</td>
</tr>
<tr>
<td>COPYTYPE</td>
<td>None</td>
<td>“COPYTYPE” on page 705</td>
</tr>
<tr>
<td>CREATEHIST</td>
<td>None</td>
<td>“CREATEHIST” on page 703</td>
</tr>
<tr>
<td>DATACLAS</td>
<td>None</td>
<td>“RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700</td>
</tr>
<tr>
<td>DCTOKEN</td>
<td>DRECOVER</td>
<td>“DCTOKEN” on page 710</td>
</tr>
<tr>
<td>Option</td>
<td>Batch Default</td>
<td>Reference</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DISK</td>
<td>None</td>
<td>“RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700</td>
</tr>
<tr>
<td>DREXTEND</td>
<td>NO</td>
<td>“DREXTEND” on page 706</td>
</tr>
<tr>
<td>DSNTEP2 <em>dataSetName</em></td>
<td>DSNLOAD</td>
<td>“DSNTEP2LIB dataSetName” on page 711</td>
</tr>
<tr>
<td>EXCLUDE MEMBERS</td>
<td>None</td>
<td>“EXCLUDE MEMBERS” on page 708</td>
</tr>
<tr>
<td>ESTIMATE</td>
<td>None</td>
<td>“ESTIMATE” on page 709</td>
</tr>
<tr>
<td>HWCOPY</td>
<td>None</td>
<td>“HWLEVEL 1/HWLEVEL 2” on page 707</td>
</tr>
<tr>
<td>HWLEVEL</td>
<td>None</td>
<td>“HWLEVEL 1/HWLEVEL 2” on page 707</td>
</tr>
<tr>
<td>INITIALIZE ACTIVES</td>
<td>None</td>
<td>“INITIALIZE ACTIVES” on page 702</td>
</tr>
<tr>
<td>JCLTYPE</td>
<td>None</td>
<td>“SET OPTIONS JCLTYPE” on page 699</td>
</tr>
<tr>
<td>LIMIT DAYS</td>
<td>None</td>
<td>“LIMIT DAYS” on page 710</td>
</tr>
<tr>
<td>LIMIT HOURS</td>
<td>None</td>
<td>“LIMIT HOURS” on page 710</td>
</tr>
<tr>
<td>LIMIT LOGS</td>
<td>None</td>
<td>“LIMIT LOGS” on page 710</td>
</tr>
<tr>
<td>MAXCATJOBS</td>
<td>None</td>
<td>“MAXCATJOBS” on page 704</td>
</tr>
<tr>
<td>MAXLOGJOBS</td>
<td>None</td>
<td>“MAXLOGJOBS” on page 705</td>
</tr>
<tr>
<td>MAXTAPEUNITS</td>
<td>0</td>
<td>“MAXTAPEUNITS” on page 710</td>
</tr>
<tr>
<td>MISSINGCOPIES</td>
<td>FAIL</td>
<td>“MISSINGCOPIES” on page 709</td>
</tr>
<tr>
<td>MGMTCLAS</td>
<td>None</td>
<td>“RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700</td>
</tr>
<tr>
<td>OFFSITE</td>
<td>OFFSITE NO ARCHIVE2</td>
<td>“OFFSITE” on page 700</td>
</tr>
<tr>
<td>PACLOG</td>
<td>None</td>
<td>“RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700</td>
</tr>
<tr>
<td>PRIMEALLOC</td>
<td>None</td>
<td>“PRIMEALLOC” on page 708</td>
</tr>
<tr>
<td>RECOVER LOGMASTER</td>
<td>None</td>
<td>“RECOVER LOGMASTER” on page 704</td>
</tr>
<tr>
<td>RECOVER REPOSITORY</td>
<td>None</td>
<td>“RECOVER REPOSITORY” on page 704</td>
</tr>
<tr>
<td>RECOVER WORKFILE</td>
<td>Yes</td>
<td>“RECOVER WORKFILE” on page 711</td>
</tr>
<tr>
<td>RESTORE ARCHIVE</td>
<td>None</td>
<td>“RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700</td>
</tr>
<tr>
<td>SIMULATE</td>
<td>None</td>
<td>“SIMULATE” on page 709</td>
</tr>
<tr>
<td>SINGLE LPAR</td>
<td>None</td>
<td>“SINGLE LPAR” on page 709</td>
</tr>
</tbody>
</table>
### SET OPTIONS JCLTYPE

This option specifies whether the JCL is for local point in time recoveries (LOCAL) or disaster recoveries (DR) (the default).

If the recovery is local, ARMBSRR does not generate IDCAMS DELETE/DEFINE statements for:

- DB2 catalog and directory data sets
- BMC Common DB2, RECOVERY MANAGER, and R+/CHANGE ACCUM repository data sets (if the RECOVER REPOSITORY keyword is specified)
- Log Master repository data sets (if the RECOVER LOGMASTER keyword is specified)

SET OPTIONS JCLTYPE DR is required for mirroring support. ARMBSRR verifies the mirroring status of the subsystem data sets if all of the following conditions are true:

- The BMC Recovery Management for DB2 solution password is applied
- Remote mirroring is specified in the subsystem recover options
- SET OPTIONS JCLTYPE DR is specified in the ARMBSRR syntax

If these conditions are met, ARMBSRR performs mirror verification and reports mirror information (including data set name, mirror status, VOLSER, and device address) to the ARMVRPT file. In addition, ARMBSRR prints a volume summary report for the mirror data sets to ARMVRPT.

---

<table>
<thead>
<tr>
<th>Option</th>
<th>Batch Default</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>STORCLAS</td>
<td>None</td>
<td>“RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700</td>
</tr>
<tr>
<td>SYNCNAME</td>
<td>None</td>
<td>“SYNCNAME” on page 707</td>
</tr>
<tr>
<td>UNIT</td>
<td>None</td>
<td>“RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700</td>
</tr>
<tr>
<td>UNITCNT</td>
<td>None</td>
<td>“RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700</td>
</tr>
<tr>
<td>VOLUMES</td>
<td>None</td>
<td>“RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700</td>
</tr>
<tr>
<td>ZIIP</td>
<td>ENABLED</td>
<td>“RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700</td>
</tr>
</tbody>
</table>
If these conditions are not met, ARMBSRR does not check the mirroring status even if the remote mirroring subsystem option is set.

**SINGLE LPAR**

This option generates JCL that does not include the SYSAFF cards designed to route different JCL steps to different members. Use this option if your system is data-sharing and all members are running on a single LPAR at the remote site. If you do not specify this option, ARMBSRR generates JCL for the remote site as if the remote site configuration matches the local configuration. If your system is non-data-sharing, ARMBSRR ignores this option.

**OFFSITE**

This option specifies which archive log data set is to be used at the recovery site, as follows:

- Specify OFFSITE YES if using the recovery site archive log data sets generated by ARMBARC or PACLOG.

- Specify OFFSITE NO ARCHIVE2 or OFFSITE NO ARCHIVE1 if sending one of the local copies offsite instead of using the recovery site archive log data sets generated by ARMBARC or PACLOG. To provide compatibility with the OFFSITE NO option of early releases, the default is OFFSITE NO ARCHIVE2.

**RESTORE ARCHIVE1 or RESTORE ARCHIVE2**

This option specifies that recovery-site archive log copies should be restored to disk. You can use either the PACLOG utility (PACLOG YES option) or the ARMBARC program to restore the logs. ARMBSRR can generate multiple ARMBARC jobs to restore the logs, based on the value that you specify for MAXLOGJOBS.

**Note**

RESTORE ARCHIVE1 or RESTORE ARCHIVE2 is mutually exclusive with ARCHIVE NOTCATLG (see “SET OPTIONS JCLTYPE” on page 699) and INITIALIZE ACTIVES (see “INITIALIZE ACTIVES” on page 702).

**ZIIP**

The ZIIP option specifies whether to attempt to use IBM System z Integrated Information Processors (zIIPs). RECOVERY MANAGER can use enclave service request blocks (SRBs) to enable zIIP processing automatically while running jobs. Using zIIP processing can reduce the overall CPU time for RECOVERY MANAGER jobs.

You can specify one of the following values:
ENABLED tells RECOVERY MANAGER to attempt to offload eligible processing to an available zIIP. If the zIIP is busy or not available, normal processing continues on a general-purpose processor.

DISABLED tells RECOVERY MANAGER to not attempt to use zIIP processing.

To enable and use zIIP processing with RECOVERY MANAGER, you must:

- Have an installed authorized version of XBM or SUF
- Start and maintain an XBM subsystem in your environment
- Have a zIIP available in your environment

You can specify a particular XBM subsystem to use by specifying a value for the XBMID option (with a length of up to 8 characters), or RECOVERY MANAGER will discover an XBM subsystem that meets the requirements for zIIP processing.

XBM and SUF are licensed, installed, and maintained separately from RECOVERY MANAGER. You can use either XBM or SUF, depending on the license that you have obtained:

- A license for the full version of the XBM product authorizes you to use all features of XBM.
- A license for SUF authorizes you to use only the snapshot and zIIP-processing features of XBM.

PACLOG

This option specifies whether to use PACLOG or the ARMBARC program to restore the archive logs. Use PACLOG YES if you have PACLOG installed.

DISK

The options under DISK specify functions related to disk usage.

Table 50: Disk Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIT</td>
<td>This option specifies the name of the recovery-site disk drive unit to be used when restoring the recovery site archive log data set copies to disk.</td>
</tr>
<tr>
<td>UNITCNT</td>
<td>This option specifies the number of units to be allocated for the output log copies. The default is to leave this option blank, which enables the unit count to be controlled by SMS. If you want to override the system value for this option, you can specify an integral number from 1 to 59.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>DATACLAS</td>
<td>This option specifies a valid Storage Management Subsystem (SMS) data class name for the log data sets to be restored to Direct Access Storage Device (DASD) at the recovery site (not to exceed 8 characters). Use this option only when you use SMS.</td>
</tr>
<tr>
<td>MGMTCLAS</td>
<td>This option specifies a valid SMS management class name for the log data sets to be restored to DASD at the recovery site (not to exceed 8 characters). Use this option only when you use SMS.</td>
</tr>
<tr>
<td>STORCLAS</td>
<td>This option specifies a valid SMS management class name for the log data sets to be restored to DASD at the recovery site (not to exceed 8 characters). Use this option only when you use SMS.</td>
</tr>
<tr>
<td>VOLUMES</td>
<td>This option specifies a valid volume for the log data sets to be restored to DASD at the recovery site (not to exceed 8 characters).</td>
</tr>
<tr>
<td>COMPRESS</td>
<td>This option specifies whether PACLOG will compress the data when you restore a log copy to disk at the recovery site. This prompt is valid only when you use PACLOG to restore the archive logs.</td>
</tr>
</tbody>
</table>

**Note**

You should specify COMPRESS NO when you have DASD hardware compression enabled.

### INITIALIZE ACTIVES

This option specifies whether the active logs are to be populated with the contents of the archive log data sets at the recovery site during resource recovery. When you use this option, RECOVERY MANAGER populates the first n-1 active logs with the contents of the last n-1 archive log data sets (where the variable n is the number of active logs that are defined in the BSDS at the recovery site). The nth active log is left empty. If you do not use this option, RECOVERY MANAGER initializes the logs to be empty.

You can specify the number of jobs to use for initialization per subsystem by specifying the MAXLOGJOBS option.

Be aware that INITIALIZE ACTIVES and HWLEVEL are mutually exclusive. Also, if you specify both INITIALIZE ACTIVES and RESTORE ARCHIVE, RECOVERY MANAGER ignores the INITIALIZE ACTIVES option.

### MAXLOGJOBS

This option specifies the number of jobs to use per member for initialization when INITIALIZE ACTIVES is chosen. Valid values are 1 through 10.

MAXLOGJOBS also specifies the number of ARMBARC jobs generated when you specify RESTORE ARCHIVES.
**WARNING**

Do not specify a value that is higher than the number of initiators available. Doing so may cause an unending wait situation.

Multiple jobs cannot be created if the archive logs are stacked on the same tape and have not been restored to disk. RECOVERY MANAGER performs stacked tape analysis prior to creating the Phase 1 JCL. The number of log copy jobs may vary based on this analysis and may possibly be less than you requested with the MAXLOGJOBS option.

**INITIALIZE DSNJLOGF**

This option generates JCL for ARMBACT to initialize active logs with DSNJLOGF. This is a stand-alone method to initialize all active logs for a subsystem in a single execution.

**CREATEHIST**

This option creates an archive history file even when RECOVERY MANAGER does not restore the archives to disk. (This archive history file is always created when the archives are restored to disk).

**BSDS ARCHLIMIT**

This option specifies the maximum number of logs that you want processed by ARMBSRR. The default is to process all logs in the BSDS. This value is from 1 to 9999.

This option is useful when you know that some logs are no longer cataloged and prevents RECOVERY MANAGER from trying to process missing or uncataloged archive logs.

**Note**

If you specify INITIALIZE ACTIVES, the limit you use should be greater than the number of active logs minus 1. Otherwise, not all of the active logs will be populated.

**BSDS DAYS LIMIT**

This option specifies the maximum number of days of logs that you want ARMBSRR to process. Valid values are from 1 to 99.

This option is useful when you do not know the number of logs that you want to process, but you know the number of days.
**BSDS HOURSLIMIT**

This option specifies the maximum number of hours of logs that you want ARMBSRR to process. Valid values are from 1 to 999.

This option is useful when you do not know the number of logs that you want to process, but you know the number of hours.

**RECOVER REPOSITORY**

This option recovers the BMC Common DB2 repository, RECOVERY MANAGER repository, and the R+/CHANGE ACCUM repository, if installed. If you do not include this option, the recovery of the repository is omitted from the recovery job.

Be aware of the following information:

- You must recover the repository at the disaster recovery site to use the data collection feature of the BMC Recovery Management for DB2 solution.

- The R+/CHANGE ACCUM repository objects are included with the RECOVERY MANAGER repository if R+/CHANGE ACCUM is installed.

- If mirroring is enabled and the repository is designated as mirrored, ARMBSRR does not include the repository in the mirrored JCL. However, if nonmirrored JCL is generated, ARMBSRR includes the repository even if you do not specify RECOVER REPOSITORY.

**RECOVER LOGMASTER**

This option recovers the Log Master repository. If you do not include this option, the recovery of the repository is omitted from the recovery job.

**Note**

If mirroring is enabled and the Log Master repository is designated as mirrored, ARMBSRR does not include the repository in the mirrored JCL. However, if nonmirrored JCL is generated, ARMBSRR includes the repository even if you do not specify RECOVER LOGMASTER.

**MAXCATJOBS**

**DB2 Version 10 and later**

MAXCATJOBS specifies a value that is used in the PARALLEL and TAPEUNITS syntax. The result enables multitasking by DSNUTILB recovery instead of multiple jobs. Because DB2 Version 10 does not allow multiple simultaneous catalog recoveries, ARMBSRR does not generate multiple catalog recovery jobs even if
MAXCATJOBS is greater than 1 when you are running under DB2 Version 10 or later. Instead, ARMBSRR generates a single catalog recovery job that uses multitasking.

**MAXLOGJOBS**

For more information, see “INITIALIZE ACTIVES” on page 702.

**COPYTYPE**

This option specifies the image copy type (as indicated in the ICBACKUP column of the SYSIBM.SYSCOPY table) to be used for recovery of catalog, directory, and repository table spaces at the recovery site. If unspecified, the default is RP. Valid values are LP (local primary), LB (local backup), RP (remote primary), or RB (remote backup).

This option is ignored for hardware mirroring level 2 (HWLEVEL 2).

---

**Note**

NGT Copy cannot make RP Flash copies of the DB2 catalog.

---

**CONDRESTART**

This option controls whether a conditional restart is included in the recovery JCL during a local full subsystem recovery.

---

**Note**

CONDRESTART AUTO and NO are only available when you are using RECOVERY MANAGER as a component of the Recovery Management for DB2 solution. If the solution password is not found, RECOVERY MANAGER automatically uses CONDRESTART YES. For more information about conditional restart avoidance during local full subsystem recovery, see the Recovery Management for DB2 User Guide.

Valid values are:

- AUTO—(the default if the Recovery Management solution password is found)
  This parameter instructs ARMBSRR to avoid a conditional restart if possible, but to include it if required. If no DDL activity occurred, recovery of the catalog and directory is not required and is not included in the recovery JCL. If quiet points are found within the specified time range, ARMBSRR generates JCL to perform the recovery to the most recent point without the conditional restart. If no quiet points are found, RECOVERY MANAGER generates a recovery to the end point of the specified time range and DSNUTILB handles any inflight transactions.
- **YES**—(the default if the Recovery Management solution password is not found)
  This parameter instructs ARMBSRR to generate conditional restart JCL for the recovery, regardless of whether quiet points are available within the specified time range.

- **NO**—This parameter instructs ARMBSRR to avoid a conditional restart. If quiet points are found within the specified time range, the recovery is performed to the most recent point. When no quiet points are found, RECOVERY MANAGER generates a recovery to the end point of the specified time range and DSNUTILB handles any inflight transactions.

### DREXTEND

This option enables you to extend the recovery point at the disaster recovery site by generating JCL to run the ARMBSDR program at the disaster recovery site as part of the Phase 1 job.

ARMBSDR finds the most recent BSDS and archive log (for each member if data sharing) at the recovery site and updates the BSDS. ARMBSDR also adds a new conditional restart control record to the BSDS.

ARMBSRR will not generate the ARMBSDR JCL if any of the following subsystem options are specified:

- **DREXTEND** is **NO** or **DREXTEND** is not specified.
- **SIMULATE** is **YES**.
- **JCLTYPE** is **LOCAL**.
- **INITIALIZE ACTIVES** is specified.
- **MAXLOGJOBS** is greater than **1**.
- **MIRROR** is **YES**.

Valid values for **DREXTEND** are:

- **NO**—**NO** is the default value. **DREXTEND NO** instructs ARMBSRR to not generate the ARMBSDR JCL.

- **YES**—**DREXTEND YES** instructs ARMBSRR to generate JCL to run the ARMBSDR program at the disaster recovery site as part of the Phase 1 job.

For data sharing environments, a process is in place that coordinates all Phase 1 jobs so that ARMBSDR processes all members. If ARMBSDR fails, correct the problem and resubmit the Phase 1 job. The rest of the Phase 1 jobs wait for ARMBSDR to complete successfully.

For more information about the ARMBSDR program, see “**ARMBSDR—Extend recovery point at disaster recovery site**” on page 651.
**ALTLOAD**

This option enables you to specify a load library that contains RECOVERY MANAGER. This load library is generated in steplibs.

**SYNCNAME**

This option specifies the name of the synchronization file to be used to synchronize the generated jobs that run in parallel.

The default name is `userid.D date.T time.BMCSYNC`, where:

- Date is in the format `ymmd`  
- Time is in the format `hhmmss`

If you use a different name than the default, do not enclose it in quotation marks.

**HWLEVEL 1/HWLEVEL 2**

These options specify the level of hardware mirroring implemented on the system, as follows:

**HWLEVEL 1**—indicates hardware mirroring Level 1, which means that only the BSDS data sets and active logs are mirrored.

**HWLEVEL 2**—indicates hardware mirroring Level 2, which means that the catalog and directory data sets as well as the BSDS and active logs are mirrored.

The default is no mirroring at all.

ARMBSRR does not allow the use of `OFFSITE YES` or `INITIALIZE ACTIVES` in conjunction with `HWLEVEL 1` or `2`.

If you are using the Recovery Management solution and you have specified hardware mirroring at the subsystem level, the subsystem mirroring settings override the HWLEVEL options.

**HWCOPY 1/HWCOPY 2**

These options indicate that only one copy of the BSDS and active logs is being mirrored, as follows:

**HWCOPY 1**—indicates that Copy 1 is the mirrored copy and therefore Copy 2 must be rebuilt by ARMBSRR

**HWCOPY 2**—indicates that Copy 2 is the mirrored copy and Copy 1 must be rebuilt
EXCLUDE MEMBERS

This option excludes permanently quiesced members from a disaster recovery. You should use this option if the permanently quiesced members are no longer in use and do not need to be recovered even in the event of a system-wide disaster. Enter one or more DB2 member names of 1-8 characters in length.

**Note**
In version 11.1 and later, EXCLUDE MEMBERS replaces QUIESCED MEMBER, which is available in RECOVERY MANAGER versions earlier than 11.1.

BYPASS QUIESCED

This option ignores members that are in QUIESCED status when calculating the recovery point. If any log ranges exist after the end of the last archive log, data could be lost by recovering to the end of the last archive log.

**VALIDATE_WARN**—writes a warning message and continues processing. VALIDATE WARN is the default.

**VALIDATE_FAIL**—writes a message and stops processing with return code 8.

BYPASS DEACT

This option excludes members that are in DEACT status from disaster recovery.

PRIMEALLOC

This option specifies the allocations to be used when allocating VSAM files for the DB2 catalog and directory, as follows:

**PRIMEALLOC ASIS**—RECOVERY MANAGER uses the same primary and secondary space allocations that are used at the local site

**PRIMEALLOC ALL**—RECOVERY MANAGER uses all required space in the primary extent

ARCHIVE NOTCATLG

This option indicates that the archive log data sets are not cataloged at the recovery site. If the operating system and tape management catalogs will not be brought up to date at the recovery site, this option lets you make that situation known to RECOVERY MANAGER so that all references to the archive logs will be generated with the unit and volume specified.
**Note**  
ARCHIVE NOTCATLG is not valid if you specify that the archive logs are to be restored to disk (see “RESTORE ARCHIVE1 or RESTORE ARCHIVE2” on page 700). Restoration to disk requires that all archive logs be cataloged.

---

### ESTIMATE

This option estimates system resource recovery for a remote site. When you specify this option, ARMBSRR produces an estimate of the amount of time that the recovery of your subsystem resources will take, and takes into consideration the options that you set and the unique configuration of your subsystem, including number of table spaces, size of table spaces, and more. The default is NO.

This option is only available when you are using RECOVERY MANAGER as a component of the BMC Recovery Management for DB2 solution. For more information about estimation, see the *Recovery Management for DB2 User Guide*.

---

### SIMULATE

This option simulates system resource recovery for a remote site. When you specify this option, the simulation JCL copies the BSDS and creates the proposed conditional restart control record (CRCR) in the BSDS. It also makes other checks on your specifications for disaster recovery. However, no actual conditional restart or recovery of the catalog and directory is performed. The default is NO.

**Note**  
You must have NGT Recover to use this option.

---

### MISSINGCOPIES

This option indicates whether RECOVERY MANAGER is to FAIL, which is the default value, or WARN if copies are missing.

- When MISSINGCOPIES FAIL is specified in the JCL, RECOVERY MANAGER sets RC=8, issues message BMC80622W, and stops processing if missing copies are detected.

Following is an example of output when MISSINGCOPIES FAIL is specified and missing copies are detected:

```
BMC80622W NO COPIES FOUND FOR DSNDB01.SYSUTILX. CONTINUING WITHOUT COPY
ANALYSIS
BMC80622W NO COPIES FOUND FOR DSNDB01.DBD01. CONTINUING WITHOUT COPY
ANALYSIS
BMC80622W NO COPIES FOUND FOR DSNDB01.SYSDBDXA. CONTINUING WITHOUT COPY
ANALYSIS
BMC80622W NO COPIES FOUND FOR DSNDB06.SYSCOPY. CONTINUING WITHOUT COPY
ANALYSIS
BMC80622W COPY NOT CATALOGED FOR DSNDB06.SYSGRTNS. CONTINUING WITHOUT
```
If you specify MISSINGCOPIES WARN in the JCL, RECOVERY MANAGER sets RC=4, issues message BMC80622W, and continues processing if missing copies are detected.

**MAXTAPEUNITS**

MAXTAPEUNITS is not currently used for catalog recovery.

This option determines the maximum number of tape units to use at one time for repository recovery. MAXTAPEUNITS is generated in ARMBSRR syntax with a value of 0 that means that DSNUTILB RECOVER determines the maximum number of tape units. ARMBSRR accepts a range of 0 through 32 for MAXTAPEUNITS.

**LIMIT LOGS**

This option specifies how many log data sets are to be restored for each recovery-site archive log copy. The variable \( n \) is the number of logs and can be any value from 1 - 9999. The default is to process all logs in the archive history file.

**LIMIT DAYS**

This option specifies how many days' worth of log data sets are to be restored for each recovery-site archive log copy. Valid values are from 1 to 99.

This option is useful when you do not know the number of logs you want to process, but you know the number of days.

**LIMIT HOURS**

This option specifies how many hours' worth of log data sets are to be restored for each recovery-site archive log copy. Valid values are from 1 to 999.

This option is useful when you do not know the number of logs you want to process, but you know the number of hours.

**DCTOKEN**

DCTOKEN *token* is an identifier used by RECOVERY MANAGER to differentiate different types of data collection information. The default DCTOKEN value is DRECOVER, which turns data collection on. To turn data collection off, use DCTOKEN NO.
RECOVER WORKFILE

This option specifies that RECOVERY MANAGER should recover work file database (DSND807) objects during local or remote subsystem recovery.

Valid values for RECOVER WORKFILE are:

- **YES** (the default) instructs ARMBSSR to include work file database objects in the recovery job.
- **NO** instructs ARMBSSR to exclude work file database objects from the recovery job.

DSNTEP2LIB *dataSetName*

This option specifies the library that contains DSNTEP2.

If RECOVERY MANAGER does not find DSNTEP2LIB in the syntax, the library defaults to DSNLOAD.

Sample JCL

The following figure shows a sample of disaster recovery JCL for ARMBSSR.

**Figure 176: Sample ARMBSSR JCL**

```plaintext
//ARM0003 EXEC PGM=ARMBSSR,PARM='DEBN,ARMOPTS=ARM$OPTS',  
    REGION=4M,COND=(4,LT)  
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS  
// DD DISP=SHR,DSN=DSNEXIT  
// DD DISP=SHR,DSN=DSNLOAD  
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTRLIBS(ARMMSGS)  
//ARMJCLN DD DISP=SHR,DSN=ARM.V110QA.RECSIM.DEBN.TEST.JCL(JOBCARD)  
//ARMJCL DD DSN=ARM.DEBN.ARMBSSR.JCL(+1),  
    // DISP=(NEW,CATLG),UNIT=SYSALLDA,  
    // SPACE=(CYL,(1,1)),  
    // DCB=(RDASZS,PDS,OUTPUT,  
    // RECFM=FB,LRECL=80,BLKSIZE=3120)  
//ARMPRT DD SYSOUT=*  
//ARMERROR DD SYSOUT=*  
//ARMIPDD DD SYSOUT=*  
//ARMVRPT DD SYSOUT=*  
//ARMIN DD *  
SET OPTIONS  
JCLTYPE DR  
CREATEHIST  
RECOVER REPOSITORY  
OFFSITE NO ARCHIVE1  
MAXCATJOBS 1  
MAXLOGJOBS 1  
COPYTYPE LP  
DCTOKEN DRECOVER  
;```
The following figure shows sample system recovery job created by ARMBSRR.

**Figure 177: Sample ARMBSRR output (actual recovery)**

```
** RECOVERY MANAGER FOR DB2 V11.1.00 - SYSTEM RESOURCES RECOVERY 08/16/2012 17:05:12 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.1.00**

(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884

BMC80223I MAINT:  NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT:  BPJ0197  BPJ0215  BPJ0219

BMC80309I CONNECTED TO DB2 SSID = DEC2 VERSION 910

SET OPTIONS
    JCLTYPE DR
;
BMC80570I COMMAND COMPLETE RC = 0

CREATEHIST
RECOVER REPOSITORY
OFFSITE NO ARCHIVE1
MAXCATJOBS 1
MAXLOGJOBS 1
COPYTYPE LP
DCTOKEN DRECOVER
;
BMC80477I SSID=DEC2, RESTARTRBA=00652E143FFF
BMC80608W ARCHIVE LOG COPY 2 NOT FOUND IN THE BSDS
BMC80610I LOGNUM=A0008536  STARTRBA=003DC95F8000  ENDRBA=003DCA939FFF
BMC80570I COMMAND COMPLETE RC = 4

BMC805711 PROGRAM COMPLETE RC = 4
```

*Figure 178 on page 712* shows sample simulation recovery job created by ARMBSRR.

**Figure 178: Sample ARMBSRR output (SIMULATE YES)**

```
** RECOVERY MANAGER FOR DB2 V11.1.00 - SYSTEM RESOURCES RECOVERY 08/16/2012 17:05:12 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.1.00**

(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
RECOVERY MANAGEMENT TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBER 7133884

BMC80223I MAINT:  NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT:  BPJ0197  BPJ0215  BPJ0219
BMC80309I CONNECTED TO DB2 SSID = DEBN VERSION 910

SET OPTIONS
    JCLTYPE DR
;
BMC80570I COMMAND COMPLETE RC = 0

RESTORE
PACLOG NO
```
Figure 179 on page 713 shows sample estimation recovery job created by ARMBSRR.

Figure 179: Sample ARMBSRR output (ESTIMATE YES)
Figure 180 on page 714 shows sample system validation report created by ARMBSRR.

**Figure 180: Sample ARMBSRR System Validation Report**

```
** RECOVERY MANAGER FOR DB2 V11.1.00 - System Validation Report 08/16/2012 17:05:12 **
** BMC80220I RECOVERY MANAGEMENT FOR DB2 V11.1.00 **

User RDAJBM: Module 'ARMBSRR' invoked on 08/16/2012 17:05:12

Actual Recovery
Conditional Restart Generated

Requested Mirroring Level:
None

Non-DataSharing system

SSID = DEC2   Recovery Point = 00652E144000
Workfile Data Base Name: DSNDB07
History File: BMCARM.DEC2.HISTORY
Last Archive 3: NULL
Last Archive 4: NULL
BSDS: DEC2CAT.BSDS01
Restart RBA: 00652E143FFF

DSNDB01.DBD01: Copy Information
  DSN = RMD.SZ.DEC2.DBD01.LP00.D11.T090141
    ICType/SType(F/ ) ICBackup=LP Unit=3490     RBA=006372C5B8B0
    Taken on 08-11-10 at 09:01:43
  DSN = RMD.SZ.DEC2.DBD01.LB00.D11.T090141
    ICType/SType(F/ ) ICBackup=LB Unit=3490     RBA=006372C5B8B0
    Taken on 08-11-10 at 09:01:43
  DSN = RMD.SZ.DEC2.DBD01.RP00.D11.T090141
    ICType/SType(F/ ) ICBackup=RP Unit=3490     RBA=006372C5B8B0
    Taken on 08-11-10 at 09:01:43
.
.
DSN = RDAJBM.DSNDB01.DBD01.RP100814.T102435
    ICType/SType(F/ ) ICBackup=RP Unit=3390     RBA=006490411956
    Taken on 08-14-10 at 10:32:13

DSN = RDAJBM.DSNDB01.DBD01.LP100814.T104516
    ICType/SType(F/ ) ICBackup=LP Unit=3490     RBA=0064904D17B5
    Taken on 08-14-10 at 10:45:39

DSN = RDAJBM.DSNDB01.DBD01.RP100814.T104516
    ICType/SType(F/ ) ICBackup=RP Unit=3490     RBA=0064904D17B5
    Taken on 08-14-10 at 10:45:39

DSNDB06.SYSCOPY: Copy Information
  DSN = RMD.SZ.DEC2.SYSCOPY.LP00.D11.T090137
    ICType/SType(F/ ) ICBackup=LP Unit=3490     RBA=006372AB68B9
    Taken on 08-11-10 at 09:01:40
  DSN = RMD.SZ.DEC2.SYSCOPY.LB00.D11.T090137
    ICType/SType(F/ ) ICBackup=LB Unit=3490     RBA=006372AB68B9
    Taken on 08-11-10 at 09:01:40
  DSN = RMD.SZ.DEC2.SYSCOPY.RP00.D11.T090137
    ICType/SType(F/ ) ICBackup=RP Unit=3490     RBA=006372AB68B9
    Taken on 08-11-10 at 09:01:40
.
```
DSN = RMD.SZ.DEC2.SYSCOPY.RB00.D11.T132804
ICType/SType(F/ ) ICBackup=RB Unit=3490     RBA=0063C05FBA25
Taken on 08-11-10 at 13:28:10

DSN = RDAJBM.DSNDB06.SYSCOPY.LP100814.T104516
ICType/SType(F/ ) ICBackup=LP Unit=3490     RBA=0064906DE26D
Taken on 08-14-10 at 10:51:25

DSN = RDAJBM.DSNDB06.SYSCOPY.RP100814.T104516
ICType/SType(F/ ) ICBackup=RP Unit=3490     RBA=0064906DE26D
Taken on 08-14-10 at 10:51:25

DSN = RDAJBM.DSNDB01.SYSUTILX.RP100814.T102435
ICType/SType(F/ ) ICBackup=RP Unit=3390     RBA=0064904024D4
Taken on 08-14-10 at 10:32:03

DSN = RDAJBM.DSNDB01.SYSUTILX.LP100814.T104516
ICType/SType(F/ ) ICBackup=LP Unit=3490     RBA=0064904C20DB
Taken on 08-14-10 at 10:45:29

BSDS Information:
Archive1 Entries          = 5780
Archive1 Last Log         = DEC2CAT.ARCLOG1.D08229.T1700382.A0014317
Archive1 Last Log End RBA = 00652E143FFF

Archive2 Entries          = 2983
Archive2 Last Log         = DEC2CAT.ARCLOG2.D08229.T1700382.A0014317
Archive2 Last Log End RBA = 00652E143FFF

Active1 Entries           = 3
Active1 Last Log          = DEC2CAT.LOGCOPY1.DS02
Active1 Last Log End RBA  = 0065305D3FFF

Active2 Entries           = 3
Active2 Last Log          = DEC2CAT.LOGCOPY2.DS02
Active2 Last Log End RBA  = 0065305D3FFF

Primary Archive Log List
Last log used = DEC2CAT.ARCLOG1.D08229.T1700382.A0014317
No RBA information

Image Copy Usage for Catalog and Directory Recovery:
Object = DSNDB06.SYSEBCDC Dsnum = 0
DSN = BMCARM.SYSEBCDC.LP.D120815.T101918
ICType(F) SType( ) ICBackup=LP Taken on 08-15-10 at 10:19:21

Object = DSNDB06.SYSJAVA Dsnum = 0
DSN = BMCARM.SYSJAVA.LP.D120815.T101929
ICType(F) SType( ) ICBackup=LP Taken on 08-15-10 at 10:19:31

Object = DSNDB06.SYSJAUXA Dsnum = 0
DSN = BMCARM.SYSJAUXA.LP.D120815.T101932
Executing the JCL

This section describes special instructions or information required to run the ARMBSRR JCL.

- Ensure that the job owner has the appropriate authorizations. See “Authorizations” on page 688.

- Run this program after ARMBCRC (if used) and ARMBARC.

- Run this program once for an entire data sharing group.

- Run this program before ARMBGEN and ARMBGPV if they use the TORESTARTRBA option.

- No restart is available for ARMBSRR. You must resubmit the job after correcting any error conditions.

Troubleshooting an ARMBSRR job

When you execute the ARMBSRR job, you might receive error or warning messages.

You should always correct any error situations, and then rerun the job before transporting the system resource recovery job to the recovery site.

In general, you should take the following return code-dependent actions:

- If you receive a return code 4, good JCL has been generated but you should examine the warning messages and take appropriate action.

- If you receive a return code 8, JCL is generated but some values are missing. The JCL will not run until you correct the problems.
If you receive a return code 12, no JCL has been generated. The errors must be corrected before you run the ARMBSRR job again.

For more information, see “Running and restarting DB2 conditional restart recovery jobs” on page 337.

The following messages are the most important ARMBSRR messages. Full explanations and user actions are provided in “RECOVERY MANAGER repository” on page 773.

Table 51: ARMBSRR messages

<table>
<thead>
<tr>
<th>Message Number</th>
<th>Short explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC80300E</td>
<td>DB2 was not active when ARMBSRR executed.</td>
</tr>
<tr>
<td>BMC80403W</td>
<td>No JCL was generated for a specified operation.</td>
</tr>
<tr>
<td>BMC80424W</td>
<td>RECOVERY MANAGER could not find the required integrated catalog facility (ICF) data.</td>
</tr>
<tr>
<td>BMC80607E or W</td>
<td>RECOVERY MANAGER found an uncataloged archive log copy.</td>
</tr>
<tr>
<td>BMC80608E or W</td>
<td>RECOVERY MANAGER could not find a specified archive log copy.</td>
</tr>
<tr>
<td>BMC80609I</td>
<td>RECOVERY MANAGER increased the active log allocation.</td>
</tr>
<tr>
<td>BMC80614E</td>
<td>An archive-history-file read error occurred.</td>
</tr>
<tr>
<td>BMC80615E</td>
<td>RECOVERY MANAGER could not find an archive log with a specified start RBA.</td>
</tr>
<tr>
<td>BMC80618E</td>
<td>The OFFSITE NO option can be used only in dual logging environments.</td>
</tr>
<tr>
<td>BMC80620E</td>
<td>OFFSITE NO ARCHIVE1 is invalid when the recovery site archives will not be cataloged.</td>
</tr>
<tr>
<td>BMC80621W</td>
<td>No stacked tape analysis will occur due to previous errors.</td>
</tr>
<tr>
<td>BMC80622W</td>
<td>No copies were found for the indicated table space, so no stacked tape analysis will occur.</td>
</tr>
</tbody>
</table>
This chapter explains how to use the ARMBTSI program to insert a user-specified time stamp into the RECOVERY MANAGER CRRDRPT table. The time stamp can help facilitate a coordinated point-in-time recovery at a local site.

About ARMBTSI

The timestamp insertion program, ARMBTSI, inserts a row containing a user-specified timestamp into the RECOVERY MANAGER CRRDRPT table.

For data sharing systems, the program inserts a row for each member of the data sharing group. After the rows are inserted, you can run the ARMBCRC program to perform log analysis and generate a relative byte address (RBA) or log range sequence number (LRSN) associated with the inserted timestamp. You can use this RBA or LRSN to generate a coordinated point-in-time recovery at a local site using a DB2 conditional restart.

**Note**

- BMC strongly recommends using the ARMBTSI program to insert timestamps into the CRRDRPT table rather than using previously published procedures that utilize separate SQL INSERT statements.

- The timestamp recorded in the CRRDRPT table must be equal to or less than the current time. If you enter a timestamp greater than the current time (that is, a time in the future), ARMBCRC bypasses the entry and issues an informational message.

Authorizations

The following authorizations are required to execute the ARMBTSI program:
Building the JCL

Building your own ARMBTSI job involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Output data sets

Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends that you specify REGION=0M, which makes the amount of virtual storage that is needed to run the job automatically available when the ARMBTSI job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

Specifying the EXEC statement

The EXEC statement has the following format:

```
//stepname EXEC PGM=ARMBTSI,PARM='ssid,timestamp,ARMOPTS=optionSet',
   REGION=0M
```

where:
The variable *ssid* is the DB2 subsystem on which the program is executing. If you do not provide a subsystem ID, the program uses the subsystem ID indicated in the DSNHDECP module found in the STEPLIB or link list.

**Note**
The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDECP module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

The variable *timestamp* is entered in the format *yyyyMMdd-hh.mm.ss* and represents local time.

The variable *optionSet* is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is `ARM$OPTS`.

### Specifying the STEPLIB DD statement

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBTSI to use. For example:

```jcl
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
```

### Specifying the ARMBTSI data set DD statements

This subsection describes the data sets ARMBTSI uses.

Each data set is specified by a `ddname` (data definition name). You must specify all required data sets in the JCL.

- *(optional)* **ARMIN**
  The input data set that contains one or more control statements. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

- *(required)* **ARMPRINT**
  The output for messages that are returned from RECOVERY MANAGER. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.
(optional) ARMOPTS
The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

```
//ARMOPTS  DD *
ssid.configurationOption=value /*
```

(required) ARMMSGS
The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of `hilvl.RMGR.ARMCNTL(ARMMSGS)`. The data set must be allocated with DISP=SHR.

(optional) ARMERROR
The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. This data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

ARMBTSI syntax and option descriptions

The ARMBTSI syntax and option descriptions in this section are the control statements that you can use when you build ARMIN input.

Figure 181 on page 722 shows the ARMTSI Syntax.

Figure 181: ARMBTSI syntax
EXCLUDE MEMBERS

This option excludes permanently quiesced members when inserting timestamps into the CRRDRPT table. You should use this option if the permanently quiesced members are no longer in use. Enter one or more DB2 member names of 1-8 characters in length.

BYPASS QUIESCED

This option ignores members that are in QUIESCED status when inserting timestamps into the CRRDRPT table.

BYPASS DEACT

This option ignores deactivated members when inserting timestamps into the CRRDRPT table.

Sample JCL

The following figure shows sample JCL for ARMBTSI that is executed in a local subsystem recovery.

Figure 182: Sample ARMBTSI JCL

```
//ARMDO02 EXEC PGM=ARMBTSI,
PARM='DECI,2012-02-18-11.16.47,ARMOPTS=ARM$OPTS',
REGION=4M,COND=(4,LT)
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// DD DISP=SHR,DSN=DSNLOAD
//ARMMGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMGS)
//ARMPRINT DD SYSOUT=* 
//ARMERROR DD SYSOUT=* 
```

Sample output

The following figure shows sample output for ARMBTSI that is executed in a local subsystem recovery.

Figure 183: Sample ARMBTSI output

```
** RECOVERY MANAGER FOR DB2 V11.1.00 - TIMESTAMP INSERT 08/16/2012 15:56:43
**
(c) COPYRIGHT 1994-2013 BMC SOFTWARE, INC.
RECOVERY MANAGER TECHNOLOGY IS PROTECTED BY U.S. PATENT NUMBERS 5625817 AND 5761676
BMCB0223I MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMCB0223I SOLUTION COMMON CODE V11.1.00
BMCB0223I MAINT: BPJ0197  BPJ0215  BPJ0219
```
Executing the JCL

This section describes special instructions or information required to run the ARMBTSI JCL.

- Ensure that you have the appropriate authorizations. See “Authorizations” on page 719 for required authorizations.

- No restart is available for ARMBTSI. You must resubmit the job after correcting any error conditions.
ARMBWDC—System recovery data collection

This chapter describes how to use the ARMBWDC program to write data-collection information related to start and end times for potential or actual disaster recoveries.

About ARMBWDC

The ARMBWDC program writes data collection information about recovery start and end times for actual, estimated, and simulated disaster recoveries of system resources.

It is only available with the Recovery Management for DB2 solution. For more information about data collection, see the Recovery Management for DB2 User Guide. The data collection information includes the recovery start and end times for system resources and the start and end times for DSNUTILB calls for application recovery. It can optionally record information about the recovery phases, catalog and directory, catalog indexes, BMC Common DB2 repository, RECOVERY MANAGER repository, and CHANGE ACCUM repository.

Authorizations

The following authorizations are required to execute the ARMBWDC program:

- APF authorization for the ARMBWDC program and the RECOVERY MANAGER load library
- EXECUTE authority on the RECOVERY MANAGER DB2 plan
- READ authority for the archive log data sets and BSDSs
- READ authority for the ICF catalog
READ authority for the archive history file

Building the JCL

Building your own ARMBWDC job to generate JCL to recover the DB2 subsystem involves creating JCL that includes the following statements:

- A JOB statement
- An EXEC statement
- Data definition statements that specify the use of the following libraries and data sets:
  - RECOVERY MANAGER and DB2 load libraries
  - Input data sets
  - Output data sets

The descriptions that follow provide more details.

Specifying the JOB statement

The JOB statement starts with a job name and includes standard JOB statement parameters, such as accounting information and a name that identifies the run.

The JOB statement should include the REGION parameter, which specifies the amount of virtual storage that the job requires. If you omit the REGION parameter from the JOB statement, you can include it in the EXEC statement. BMC recommends you specify REGION=0M, which makes the amount of virtual storage needed to run the job automatically available when the ARMBWDC job is executed. If REGION=0M is not allowed at your company, specify REGION=4M.

Specifying the EXEC statement

The EXEC statement has the following format:

```
//stepname EXEC PGM=ARMBWDC,
//              PARM='ssid,ARMOPTS=optionSet',
//              REGION=0M
```

The variable `ssid` is the DB2 subsystem or group attach name where the RECOVERY MANAGER groups reside. If you do not provide a subsystem ID, the program uses
the subsystem ID indicated in the DSNHDEC module found in the STEPLIB or link list.

**Note**
The SSID parameter is positional and requires the comma even if you do not enter a specific subsystem ID. If the program cannot find the SSID that you specified or that is listed in the DSNHDEC module, it will issue message BMC80583E INVALID PARAMETER FOR SSID and set the return code to 8.

The variable `optionSet` is the name of an XML file that contains all of the product’s configuration option values. The default option set for RECOVERY MANAGER is ARM$OPTS.

**Specifying the STEPLIB DD statement**

The STEPLIB DD statement identifies the RECOVERY MANAGER load library and DB2 load libraries that you want ARMBWDC to use. For example:

```
//STEPLIB   DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
//          DD DISP=SHR,DSN=DSNEXIT
//          DD DISP=SHR,DSN=DSNLOAD
```

**Specifying the ARMBWDC data set DD statements**

This subsection describes the data sets that ARMBWDC uses.

Each data set is specified by a `ddname` (data definition name). You must specify all required data sets in the JCL.

- **(required) ARMIN**
  The input data set that contains one or more control statements. Attributes for this data set must be fixed length records, with a length of 80 (RECFM=F or FB, LRECL=80).

- **(required) ARMPRINT**
  The output for messages that are returned from RECOVERY MANAGER. RECOVERY MANAGER also echoes the contents of the ARMIN data set in the ARMPRINT output. ARMPRINT may be allocated to SYSOUT or to a data set with a data control block (DCB) of LRECL=121, RECFM=VB.
(optional) ARMOPTS

The configuration options are read from the option set named in the EXEC statement parameters (PARM=). If an option set name is not specified there, ARM $OPTS is used as the default option set name.

You can temporarily override one or more configuration options using the following ARMOPTS DD statement:

```plaintext
//ARMOPTS DD *
ssid.configurationOption=value
/*
```

(required) ARMMSGS

The RECOVERY MANAGER messages data set created during RECOVERY MANAGER installation with the default name of `hilvl/RMGR.ARMCTRL(ARMMSGS)`. The data set must be allocated with DISP=SHR.

(optional) ARMERROR

The output for compiler run time errors. If compiler errors are detected and ARMERROR is not present in the JCL, the errors are printed in the JES log. The data set may be allocated to SYSOUT or to a data set with a DCB of LRECL=121, RECFM=VB.

ARMBWDC syntax and option descriptions

The ARMBWDC syntax and option descriptions in this section are the control statements that you use when you build ARMIN input.

Note

See “Syntax rules” on page 102 for more information on syntax rules and wildcard support.
Figure 184 on page 729 shows the ARMBWDC syntax.

Figure 184: ARMBWDC control statement

**DRECOVER**

The data collection token name indicating the recovery is a disaster recovery.

**PHASE1**

Records the start or end time of Phase 1 of system recovery.

**PHASE2**

Records the start or end time of Phase 2 of system recovery.

**LOGREST**

Records the start or end time of the log restore step.
DSNDB01

Records the start or end time of the recovery of SYSUTIL, DBD01, and SYSDBDXA of DSNDB01.

CATDIR

Records the start or end time of the recovery of the catalog and directory.

UCATIX

Records the start or end time of the recovery of user-defined catalog indexes.

RMGRREPP

Records the start or end time of the recovery of the repository.

CAREP

Records the start or end time of the recovery of the CHANGE ACCUM repository.

DSNUTILB

Records the recovery of an application object by DSNUTILB. If you specify START, ARMBWDC inserts a new JOB row into the table using a system generated job name. If you specify END, ARMBWDC updates the end timestamp of the newly created JOB row.

START

Causes ARMBWDC to record the start time of the job step corresponding to the option that you specify. For example, the following syntax records the start time of the recovery of the catalog and directory:

```
DRECOVER CATDIR START
```

END

Causes ARMBWDC to record the end time of each option that you specify. For example, the following syntax records the end time of the recovery of the catalog and directory:

```
DRECOVER CATDIR END
```

TOLOGPOINT

The restart RBA or LRSN.

DCNAME

The data set into which the start and end data is stored.
**DB2WRITE**

Writes the collected data from a flat file specified in DCNAME to the RECOVERY MANAGER Data Collection tables. ARMBWDC inserts a row in the UTILITY_RUN table, as well as rows in the JOB table, depending on the number of invocations existing in the recovery run.

**UID n**

When ARMBSRR generates the JCL for the disaster recovery site and is generating data collection information as well, RECOVERY MANAGER gets the highest identity value from the UTILITY_RUN_ID column in the UTILITY_RUN table, increments that value by 1, and generates the UID n syntax. Doing this helps ensure that when the data collection information is pulled from the disaster recovery site for insertion at the local site that the UTILITY_RUN_ID value is inserted at the local site as well.

**ESTIMATE**

Generates estimated values for system recovery and writes those to the data collection tables.

**SIMULATE**

Generates simulated values for all system objects and writes those to the data collection tables.

**MIRROR LEVEL**

Indicates in the UTILITY_RUN entry whether mirroring was specified. For more information about mirroring, see the *Recovery Management for DB2 User Guide*.

---

**Sample JCL**

The following figure shows a sample of JCL for ARMBWDC.

*Figure 185: Sample ARMBWDC JCL*

```
//BWDCSIM EXEC PGM=ARMBWDC,PARM='&SSID,ARMOPTS=ARM$OPTS',
// REGION=0M,COND=(4,LT)
//STEPLIB DD DISP=SHR,DSN=PRODUCT.LOAD.LIBS
// DD DISP=SHR,DSN=DSNEXIT
// DD DISP=SHR,DSN=DSNLOAD
//ARMMSGS DD DISP=SHR,DSN=PRODUCT.CNTL.LIBS(ARMMSGS)
//ARMPRINT DD SYSOUT=* 
//ARMERROR DD SYSOUT=* 
//ARMIN DD *
DRECOVER SIMULATE
DRECOVER MIRROR LEVEL 0
```
Sample output

The following figure shows sample output for ARMBWDC.

Figure 186: Sample ARMBWDC output - Recovery History report

** RECOVERY MANAGER FOR DB2 V11.1.00 - WRITE DATA COLLECTION 02/20/2012 12:28:17
BMC80223I MAINT: NO RECOVERY MANAGER PTFS APPLIED
BMC80223I SOLUTION COMMON CODE V11.1.00
BMC80223I MAINT: BPJ0088
DRECOVER SIMULATE
DRECOVER MIRROR LEVEL 0
BMC80571I PROGRAM COMPLETE RC = 0

Executing the JCL

This section describes special instructions or information required to run the ARMBWDC JCL.

- Ensure that the job owner has the appropriate authorizations. See “Authorizations” on page 725.

- No restart is available for ARMBWDC. You must resubmit the job after correcting any error conditions.
Option sets and configuration options

This appendix discusses the configuration options that you can use to define and maintain option sets.

Overview of DB2 Product Configuration

This product uses the DB2 Product Configuration technology, which separates product installation from configuration, and allows you to manage option sets and configuration option values through its online interface.

DB2 Product Configuration technology simplifies the configuration process as follows:

- Saves time by
  - Setting default option values for you (but allowing you to change the values, if needed)
  - Consolidating the process into scrollable panels that feature a simple navigation model (expanding and contracting sections on the panel)
  - Retaining data for subsequent installations and upgrades
  - No longer needing to submit batch jobs
- Validates any option values that you enter
- Lets you link to the configuration interface from within a product or solution
- Captures option value changes that you make after installation and initial configuration
- Provides help through the Product Configuration interface for the configuration options
- Provides maintenance via PTFs
- Supports multiple screen sizes (for example, MOD 2, MOD 4, MOD 5, or custom sized terminals)

The DB2 Product Configuration technology uses XML files called *option sets*, which contain all of your product’s configuration option values. For products that allow multiple option sets, you can run the product using different configuration option values for different jobs by using different option sets.

**Note**

When migrating from releases prior to 11.1, you can use the Option Value Migration technology to convert an existing RECOVERY MANAGER control file to an XML option set. For more information about this technology, see the *Installation System User Guide*.

The BMC DB2 Product Configuration technology requires the BMC DB2 Component Services (DBC) technology. For more information about DBC, see the *BMC Global Infrastructure Components Administration Guide*.

---

**DB2 Product Configuration interface**

DB2 Product Configuration provides ISPF panels in which you can enter or change product or solution option set values.

**Figure 187 on page 734** shows the main menu for the DB2 Product Configuration technology. For RECOVERY MANAGER, this menu is accessed from the RECOVERY MANAGER Main Menu by selecting 5. **Product Option Sets - Set RECOVERY MANAGEMENT Product options**.

**Figure 187: DB2 Product Configuration - Main Menu**

<table>
<thead>
<tr>
<th>File</th>
<th>Help</th>
<th>DB2 Product Configuration - Main Menu</th>
</tr>
</thead>
<tbody>
<tr>
<td>Option</td>
<td>====&gt;</td>
<td></td>
</tr>
<tr>
<td>Select an option and press Enter.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Configuration Functions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Manage Shared Product Options</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Manage Product Options</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

Copyright (C) 2009-2013 BMC Software, Inc. as an unpublished licensed work. All rights reserved.

F1=Help  F3=Exit  F6=Actions  F7=Backward  F8=Forward  F12=Cancel

734  *RECOVERY MANAGER for DB2 User Guide*
Note
Option 1. **Manage Shared Product Options** is not currently available.
Option 2. **Manage Product Options** is the only selection currently available.

The **Product Option Sets** panel (Figure 188 on page 735) lists the products and solutions for which you can configure option sets.

**Figure 188: Product Option Sets panel**

Placing your cursor on the plus sign (+) next to a name and pressing **Enter** expands that section to show additional choices. For example, the + next to a solution name expands the list to show products in that solution (Figure 189 on page 736). Expanding the + next to a product shows the available option sets for that product (Figure 190 on page 736). Placing your cursor on the minus sign (-) and pressing **Enter** collapses the list, reversing the expansion.

Following are some terms that apply when you are working in the DB2 Product Configuration interface:

- The expandable and collapsible areas, denoted by the + and - to the left, that contain one or more options are called **sections**.

- A **section collection** contains one or more options and is always expandable and collapsible. The group of options can be repeated. Typically, one or more options are designated as **keys** to ensure that each occurrence in the collection is unique. Section collections are denoted by a + or - to the immediate left of the section title, and the section title is followed by **n** where **n** is the number of occurrences in the section collection.
Section defaults are an instance in the section collection that provides default values for the group of options. The default values are used unless you have updated an instance to contain a different value.

Press Enter or use the letter S to toggle the state of a section on the Product Option Sets panel. Using S allows multiple sections to be opened or closed with one press of the Enter key.

Figure 189: Product Option Sets panel (expanded to show products in a solution)

Figure 190: Product Option Sets panel (expanded to show option sets)

For steps to create a new option set, see “Defining an option set” on page 745.

Menu bar and drop-down menus

The top of the Product Option Sets panel provides the following drop-down menus:
From the File menu, you have the following options:

- Save changes the option set file using the current name
- Save the current option set with a new name to create a new option set
- Cancel work in the option set and return to the option set selection panel
- Save changes to the option set and return to the RECOVERY MANAGER Main Menu

From the Filter menu, you have the following options:

- Turn off filtering (display all options)
- Display only required options
- Display only modified options

**Note**
Modified options are those for which the current value differs from the default value that shipped with the product.

- Display invalid options
- Display default options
- Reset the filter

**Note**
Alternatively, you can enter the shortcuts `FILTDEF, FILTINV, FILTOFF, FILTREQ, FILRES`, or `FILTMOD` on the Command line to change filtering on the panel. For more information, see “Commands for the Command line” on page 738.

From the Help menu, you can access general Help about DB2 Product Configuration or product-specific Help. Additionally, the About option displays information about maintenance applied as well as connection information.
**Action bar**

The action bar at the bottom of the **Product Option Sets** panel includes the following allowable actions or line commands:

- B to browse an option set
- E to edit an option set
- C to copy an option set
- D to delete an option set
- I to insert, which is available only at the product level to create a new option set
- R to rename an option set

For example, you can insert, or create, a new option set by placing an I in front of the product (noted on the panel as product-level only in the action bar) for which you want to create a new option set.

**Commands for the Command line**

The following table lists the commands that you can enter in the Command line as an alternative to selecting them from the drop-down menu:

**Table 52: Command line commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABOUT</td>
<td>Displays information about DB2 Product Configuration including the DBC connected to and applied maintenance for DB2 Product Configuration and the components that it uses</td>
</tr>
<tr>
<td>CANCEL</td>
<td>Cancels any changes</td>
</tr>
<tr>
<td>FILTDEF</td>
<td>Filter defaults</td>
</tr>
<tr>
<td>FILTINV</td>
<td>Filters on invalid options (applied automatically when one or more options are updated but fail validation)</td>
</tr>
<tr>
<td>FILTMOD</td>
<td>Shows all modified options</td>
</tr>
<tr>
<td>FILTOFF</td>
<td>Turns off filtering and shows all options</td>
</tr>
<tr>
<td>FILTREQ</td>
<td>Shows all required options</td>
</tr>
<tr>
<td>FILRES</td>
<td>Resets the filter to the previous filter value</td>
</tr>
<tr>
<td>GENHELP</td>
<td>Opens the general Help panel for DB2 Product Configuration</td>
</tr>
<tr>
<td>HELP</td>
<td>Opens ISPF Help</td>
</tr>
</tbody>
</table>
Some of these commands require that you select an option set before you can use the command (for example, FILTMOD, FILTOFF, FILTREQ, PRODHELP, SAVE, and SAVEAS).

### Action Selection Menu

The Action Selection Menu is available in the following panels:

- Product Option Sets panel, place ? in front of a product name
- Product Options Sets panel, place ? in front of option set name
- After selecting an option set, place ? in front of a section that opens a set of configuration options

Press F3 to close the Action Selection Menu.

The menu provides different selections depending on where you are when you open it. Figure 191 on page 739 shows the Action Selection Menu opened by typing ? beside a product name on the Product Option Sets panel and pressing Enter.

**Figure 191: Action Selection Menu from product name in Product Option Sets panel**
If you open an Action Selection Menu by typing the ? beside an option set, the Action Selection Menu allows you to perform the actions shown in the menu in Figure 192 on page 740.

**Figure 192: Action Selection Menu from option set name in Product Option Sets panel**

![Action Selection Menu from option set name in Product Option Sets panel](image)

After opening an option set, typing the ? in front of a section that opens a set of configuration options, the Action Selection Menu allows you to perform the actions shown in the menu in Figure 193 on page 740.

**Figure 193: Action Selection Menu from within an opened option set**

![Action Selection Menu from within an opened option set](image)

### Interface tools

The following table describes the interface tools that help you configure your product and solution option sets.
Table 53: Interface tools

<table>
<thead>
<tr>
<th>Navigation aids or options</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plus sign (+)</td>
<td>Place the cursor on the + and press <strong>Enter</strong> to expand a section. <strong>Note:</strong> If you attempt to expand an empty section, the plus sign simply changes to a minus sign. Consider using I (described later in this table) to insert information.</td>
</tr>
<tr>
<td>Minus sign (-)</td>
<td>Place the cursor on the - and press <strong>Enter</strong> to collapse a section.</td>
</tr>
<tr>
<td>S</td>
<td>Place S on the + or - to expand or collapse sections. Place S in the blank before the name of a product or option set to select the product or option set.</td>
</tr>
<tr>
<td>Greater-than sign (&gt;)</td>
<td>&gt; is displayed beside an input field that is too large to display. Placing the cursor on &gt;, or placing S on the &gt;, and pressing <strong>Enter</strong> opens another panel, where you can enter or view larger values. Fields that use this tool are referred to as zoomable fields.</td>
</tr>
<tr>
<td>Scorable panels or repositioning</td>
<td>If you run out of available viewing space on the panel, you can reposition by placing the cursor at the top of the panel and pressing the <strong>Down</strong> key. Also, if you have set the scroll amount to CSR, you can place the cursor on any line and scroll. <strong>Note:</strong> Alternatively, use S to expand or collapse sections.</td>
</tr>
<tr>
<td>R - repeat section</td>
<td>Conditionally, you can place R in the blank to the left of an option section to duplicate the section with its current values. You can then edit the repeated section to create a unique section. a</td>
</tr>
<tr>
<td>I - insert section</td>
<td>Conditionally, you can place I in the blank to the left of an option section to insert a new section that uses all default values. a</td>
</tr>
</tbody>
</table>

1 Not all sections allow you to use the repeat section or insert section function. Some sections simply represent a collection of items, such as product names under a solution.

Input panels for option sets

When you are working with option sets, the input panels have multiple columns.

The column widths vary, based on product requirements. Table 54 on page 741 lists the columns that DB2 Product Configuration uses.

Table 54: Columns in input panels

<table>
<thead>
<tr>
<th>Column position</th>
<th>Column use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>Field prompts</td>
</tr>
<tr>
<td>Center</td>
<td>Input area</td>
</tr>
</tbody>
</table>

Appendix A  Option sets and configuration options 741
Option set names

A valid option set name follows these rules:

- One to eight characters in length
- First character must be uppercase A through Z or the special characters @, #, or $
- Remaining characters must be
  - Uppercase A through Z
  - 0 through 9
  - Special characters @, #, $, or {

DB2 Product Configuration automatically changes lowercase characters to uppercase.

Note
Some products might allow different naming standards. However, the information above is the default standard.

Section defaults

DB2 Product Configuration allows you to edit the defaults for a section (group of configuration options).

The default values are used to initialize option values in each section. If you update the values, you can apply them to existing groups of options at your discretion.

For example, for RECOVERY MANAGER, you can edit each subsystem entry. If the subsystem defaults have not been edited, the values from the first subsystem entry will be used when inserting a new subsystem. After the defaults are edited, any subsystem inserts would use the new default values.

To update the subsystem defaults, place an E to the left of Subsystem Options and press Enter as shown in Figure 194 on page 742.
You can then update the options values displayed, as shown in Figure 195 on page 743.

Figure 195: Subsystem Options - Section Defaults panel

<table>
<thead>
<tr>
<th>File</th>
<th>Filter</th>
<th>Confirm</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command:====&gt;</td>
<td>Subsystem Options - Section Defaults</td>
<td>Scroll:====&gt;CSR</td>
<td></td>
</tr>
<tr>
<td>Filter: Off</td>
<td>More: +</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSID . . . . . . . . . . . .</td>
<td>CJ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Plan . . . . . . . . . . . .</td>
<td>ARMB110T</td>
<td>Maximum Length of 8 Bytes</td>
<td></td>
</tr>
<tr>
<td>Bind Qualifier . . . . . .</td>
<td>BMCARM</td>
<td>Maximum Length of 8 Bytes</td>
<td></td>
</tr>
<tr>
<td>Change Accum On Tape . . . .</td>
<td>Y</td>
<td>(Y=Yes,N=No)</td>
<td></td>
</tr>
<tr>
<td>Check Plus Load . . . . . .</td>
<td>D2U.INST1010.BMCL</td>
<td>Load Library if Installed</td>
<td></td>
</tr>
<tr>
<td>NGT Copy Load . . . . . .</td>
<td>RMD.TEST1110.BMCL</td>
<td>Load Library if Installed</td>
<td></td>
</tr>
<tr>
<td>NGT Copy Optionset . . . .</td>
<td>ACP$OPTS</td>
<td>Maximum Length of 8 Bytes</td>
<td></td>
</tr>
<tr>
<td>NGT Recover Load . . . . .</td>
<td>RMD.TEST1110.BMCL</td>
<td>Load Library if Installed</td>
<td></td>
</tr>
<tr>
<td>NGT Recover Optionset . . .</td>
<td>AFR$OPTS</td>
<td>Maximum Length of 8 Bytes</td>
<td></td>
</tr>
<tr>
<td>Log Master Load . . . . . .</td>
<td>RMD.TEST1110.BMCL</td>
<td>Load Library if Installed</td>
<td></td>
</tr>
<tr>
<td>Log Master Optionset . . . .</td>
<td>ALP$OPTS</td>
<td>Maximum Length of 8 Bytes</td>
<td></td>
</tr>
<tr>
<td>Paclog Load . . . . . . . .</td>
<td>RMD.TEST1110.BMCL</td>
<td>Load Library if Installed</td>
<td></td>
</tr>
<tr>
<td>Paclog Optionset . . . . .</td>
<td>ALM$OPTS</td>
<td>Maximum Length of 8 Bytes</td>
<td></td>
</tr>
<tr>
<td>Paclog CNTL . . . . . . . .</td>
<td>RMD.TEST1110.UBDC</td>
<td>Paclog CNTL Library</td>
<td></td>
</tr>
<tr>
<td>Active Log 1 Prefix . . . .</td>
<td>DECJCAT.LOGCOPY1</td>
<td>Active Log 1 Prefix</td>
<td></td>
</tr>
<tr>
<td>Active Log 2 Prefix . . . .</td>
<td>DECJCAT.LOGCOPY2</td>
<td>Active Log 2 Prefix</td>
<td></td>
</tr>
<tr>
<td>Archive Log 1 Prefix . . .</td>
<td>DECJCAT.ARCHLOG1</td>
<td>Archive Log 1 Prefix</td>
<td></td>
</tr>
<tr>
<td>Archive Log 2 Prefix . . .</td>
<td>DECJCAT.ARCHLOG2</td>
<td>Archive Log 2 Prefix</td>
<td></td>
</tr>
<tr>
<td>Primary Arc On Tape . . . .</td>
<td>N</td>
<td>(Y=Yes,N=No)</td>
<td></td>
</tr>
<tr>
<td>Alternate Arc On Tape . . .</td>
<td>N</td>
<td>(Y=Yes,N=No)</td>
<td></td>
</tr>
<tr>
<td>Archive use timestamp in dsn N</td>
<td>Y</td>
<td>(Y=Yes,N=No)</td>
<td></td>
</tr>
<tr>
<td>BDS1D . . . . . . . . . . . .</td>
<td>DECJCAT.BSDS01</td>
<td>BDS1 data set name</td>
<td></td>
</tr>
<tr>
<td>BDS2 . . . . . . . . . . . .</td>
<td>DECJCAT.BSDS02</td>
<td>BDS2 data set name</td>
<td></td>
</tr>
<tr>
<td>Data Collection . . . . . .</td>
<td>Y</td>
<td>(Y=Yes,N=No)</td>
<td></td>
</tr>
<tr>
<td>DSNEXIT . . . . . . . . . . .</td>
<td>SYS3.DECJ.DSNEXIT</td>
<td>DSNEXIT data set name</td>
<td></td>
</tr>
<tr>
<td>DSNLOAD . . . . . . . . . . .</td>
<td>CSG1.DB2V91M.DSNL</td>
<td>DSNLOAD data set name</td>
<td></td>
</tr>
<tr>
<td>DS Member . . . . . . . . . .</td>
<td>Data sharing member name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>History File . . . . . . . .</td>
<td>BMCARM.DECJ.HISTV</td>
<td>History data set name</td>
<td></td>
</tr>
<tr>
<td>Use CATALOG parm . . . . . .</td>
<td>N</td>
<td>(Y=Yes,N=No)</td>
<td></td>
</tr>
<tr>
<td>BMC Infrastructure Load . . .</td>
<td>Infrastructure data set name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BMC Infrastructure Load 2.</td>
<td>Infrastructure 2 data set name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JCL Output . . . . . . . . .</td>
<td>JCL Output data set name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JES2 Name . . . . . . . . . .</td>
<td>JES2 ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JES3 Name . . . . . . . . . .</td>
<td>JES3 ID</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

+/- expandable section (enter ? for action menu), > zoomable field
After updating any values, press **F3**. You then have the choice of updating all subsystems with the new defaults or selecting specific subsystems to be updated, as shown in Figure 196 on page 744.

**Figure 196: Apply Section Defaults Updates panel**

![Apply Section Defaults Updates panel](image)

After you make your selection, press **Enter** and then press **F3**.

You will see your updates applied to the selected subsystems. When editing the subsystem defaults, changes are applied *only* to options whose value matches the subsystem default *before* the change was made.

### Accessing DB2 Product Configuration

To make changes and create additional options sets, you access DB2 Product Configuration in the following ways:

- When you are installing the product, you access DB2 Product Configuration through the Installation System.

- From the RECOVERY MANAGER Main Menu, you can access the DB2 Product Configuration menu by selecting **5. Product Option Sets - Set RECOVERY MANAGEMENT Product options**.

- You can use LGC$ISPF to access the DB2 Product Configuration menus. This REXX proc is produced by the installation process and requires some simple customization. Instructions for the customization are in the proc.
Defining an option set

This task describes how to create a new RECOVERY MANAGER option set from the DB2 Product Configuration interface.

To define an option set

1. On the DB2 Product Configuration - Main Menu, select **2. Manage Product Options**.

   **Figure 197: DB2 Product Configuration - Main Menu**

   
<table>
<thead>
<tr>
<th>File</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Option ===&gt; DB2 Product Configuration - Main Menu</td>
<td></td>
</tr>
<tr>
<td>Select an option and press Enter.</td>
<td></td>
</tr>
</tbody>
</table>

   Configuration Functions
   1. Manage Shared Product Options
   2. Manage Product Options

   Enter X or press PF3 to exit the application.

2. On the Product Option Sets panel, expand Recovery Management for DB2 by placing your cursor on the plus sign (+) and pressing **Enter**.

   **Figure 198: DB2 Product Configuration - Product Option Sets panel**

   
<table>
<thead>
<tr>
<th>File</th>
<th>Filter</th>
<th>Confirm</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command ===&gt; Product Option Sets</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scroll ===&gt; PAGE</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

   Solution/Product | Version Changed | More: - + |
   + RECOVERY MANAGEMENT for DB2 | V11.1.0 |
   + System and SQL Performance for DB2 | V11.1.0 |

   End of List

   B browse, E edit, C copy, D delete, I insert (product-level only), R rename

---

**Note**

If RECOVERY MANAGER is unable to connect to DB2 Product Configuration (or the DB2 Component Services), RECOVERY MANAGER issues the message BMC80287E followed by BMC80288I. RECOVERY MANAGER waits 20 seconds and tries again. RECOVERY MANAGER retries up to 9 times (total of 3 minutes).
3 Type I next to RECOVERY MANAGER and press Enter to create a new option set.

**Note**

I (insert) is available at the product level only as noted at the bottom of the panel. The panel also shows the other actions that are available, such as B (browse), E (edit), C (copy), D (delete), and R (rename).

**Figure 199: Selecting the RECOVERY MANAGER default option set from the Product Option Sets panel**

4 (Optional) Use the Filter menu at the top of the panel to turn off filtering so that all options are shown.

   The default filter is Required Options, which means that only options that require an entry from you are displayed. If necessary, you can change the default filter to one of the other options available on the Filter menu shown in Figure 200 on page 746.

**Figure 200: Filter menu - Turn off filtering**

5 On the option set panels, expand each section and review or change the configuration option values.

   **Tip**

   To expand sections on the option set panels, place the cursor on the plus sign (+) next to a section and press Enter.

   A detailed description of all option set values is provided in “RECOVERY MANAGER option sets, Product Configuration panels, and configuration options” on page 747 and in the online Help that you access by pressing F1. If
you press **F1** while the cursor is positioned on an input or output field on a panel, specific information about that field is displayed. To view general information or information about a panel, use the Help menu at the top of the panel.

6 Press **F3** when you finish.

7 When prompted, name the option set and provide a description.

**RECOVERY MANAGER option sets, Product Configuration panels, and configuration options**

The RECOVERY MANAGER option set, which is ARM$OPTS by default, and configuration options includes the following items:

- BMC utilities that are available to RECOVERY MANAGER
- TSO, DB2, and user step libraries
- DB2 subsystem resources and JCL destination information
- Operating system resources and job card specification

If DB2 is not available, RECOVERY MANAGER must obtain the following information from the ARM$OPTS member:

- Bootstrap data set prefixes
- Active and archive log data set prefixes
- Work file database name
- System VCAT name

It is important to ensure that this information is accurate and kept up-to-date.

**Tip**

If you use the CA ACF2 security system and your shop is restricting TSO commands, add ARMUMAN, ARMUSEL, ARMOPTM, and DSNJU004 to the list of commands in the TSOCMD$ module. If your site restricts the use of TSO commands through an option of the system security package (RACF, ACF2) or an add-on product such as PCF, be sure the ARMUMAN, ARMUSEL, ARMOPTM, and DSNJU004 command names are added to the appropriate command table. Otherwise, the message IKJ56500I command COMMAND NOT FOUND is issued when attempting to invoke the RECOVERY MANAGER CLIST or when using the logging environment modeling tool.
BMC utility configuration options

Through the RECOVERY MANAGER Main Menu, you can display and update configuration options for other BMC utilities that you have licensed and which use the DB2 Product Configuration technology.

This information is provided initially during installation. If you install a new version of a BMC utility after RECOVERY MANAGER is installed, you must update the utility configuration options.

**Note**

The infrastructure load library contains code used by multiple BMC products, including the security code that is required to run RECOVERY MANAGER. The installation default is the installation target library. If you are not using the target library for the infrastructure code, you must either specify the correct library in this field or the library must be link listed. RECOVERY MANAGER jobs cannot run without access to this library.

TSO DB2 and user step library information

You can display and update the names for the following libraries and data sets that are used by RECOVERY MANAGER for both test and production environments:

- DB2 libraries (DSNEXIT and DSNLOAD)
- Optional user job libraries (up to three)
- Optional steplib override data sets (up to five)
- Optional additional steplib data sets (up to five)

This information is initially supplied during RECOVERY MANAGER installation.

**Note**

The steplib override and additional steplib data sets must be APF-authorized. The steplib override data sets are generated prior to the generation of the BMC product step libraries. You can use the override data sets for any executable code that should be accessed prior to the BMC product load libraries. For example, an override data set could contain an options module that would override the product default options.

The additional steplib data sets are generated after the generation of the BMC product step libraries. You can use the additional data sets for any executable code that should be accessed after the BMC product load libraries. For example, an additional data set could contain an options module that would be used to specify default values.
DB2 subsystem resource information

You can browse and update the options for the current DB2 subsystem by using the DB2 Subsystem Resource Information panel.

To reach this panel:

1. From the RECOVERY MANAGER Main Menu, select **Product Option Sets**.
2. Expand Recovery Management and then expand RECOVERY MANAGER.
3. Select your option set.
4. Expand Subsystems Options
5. Select the subsystem who option you want to browse or update.

Operating system resource information

You can browse and update operating system information that is shared by the subsystems that are served by RECOVERY MANAGER.

- The name of the work unit to be used by RECOVERY MANAGER for temporary disk work space
- The percentage of the work data sets to be allocated as a primary extent. The range is 10 through 100 percent.
- The names of three pseudo-volumes used for archive data sets
  The pseudo-volumes are used by the ARMBGPV batch revalidation program when generating data set recall reports. If your company uses names other than MIGRAT and ARCIVE to specify migrated or archived DASD data sets, enter those names as the pseudo-volume names. See “Revalidating and reporting on groups in batch” on page 134 for information about ARMBGPV.
- Up to five lines of job statement
  These are used by default when you do not supply your own job statement when you request JCL generation by the ARMBGEN batch program. The job statement must contain the required symbolic variables in order to satisfy the multiple job name requirement for multiple job optimization.
Alphabetical list of configuration options

The following table shows the configuration options contained in the default ARM $OPTS option set.

For each option, the table provides the value that ships with this version of RECOVERY MANAGER (or No value if the option is shipped without a value), a brief description, and a reference to more details. For quick reference, the table presents the options in alphabetical order and a cross-reference to more information.

### Table 55: RECOVERY MANAGER configuration options

<table>
<thead>
<tr>
<th>Option</th>
<th>Default value</th>
<th>Brief description</th>
<th>Interface location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Log 1 Prefix</td>
<td>No</td>
<td>Prefix of the primary active log data set names</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>Active Log 2 Prefix</td>
<td>No value</td>
<td>Prefix of the alternate active log data set names if you are using dual actives</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>Alternate Archive On Tape</td>
<td>No</td>
<td>Specifies if the alternate copy of the archive logs is on tape</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>Archive Log 1 Prefix</td>
<td>No value</td>
<td>Prefix of the primary archive log data set names</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>Archive Log 2 Prefix</td>
<td>No value</td>
<td>Prefix of the alternate archive log data set names if you are using dual archive logs</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>Archive Use Timestamp in DSN</td>
<td>No</td>
<td>Specifies if the archive logs use a date/timestamp value in the data set names</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>Arcive1 through Arcive3</td>
<td>No value</td>
<td>Specifies the names that are used in your environment by DASD management software to identify data sets that have been migrated from DASD to tape</td>
<td>Common Options</td>
</tr>
<tr>
<td>Bind Qualifier</td>
<td>ARMvvr</td>
<td>Bind qualifier for the dynamic bind process</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>BMC Infrastructure Load</td>
<td>No value</td>
<td>SMP/E target library in which the BMC infrastructure common code is stored</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>Option</td>
<td>Default value</td>
<td>Brief description</td>
<td></td>
</tr>
<tr>
<td>--------------------------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>BMC Infrastructure Load 2</td>
<td>No value</td>
<td>SMP/E target library in which the BMC infrastructure common code is stored</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>BSDS 1</td>
<td>No value</td>
<td>Full data set name of the primary bootstrap data set</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>BSDS 2</td>
<td>No value</td>
<td>Full data set name of the alternate bootstrap data set</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Change Accum on Tape</td>
<td>No</td>
<td>Specifies if you have R+/CHANGE ACCUM installed and your change accum files are on tape</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Check Plus Load</td>
<td>No value</td>
<td>Fully qualified name of your CHECK PLUS load library (if installed)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Copy Archive Timestamp</td>
<td>Yes</td>
<td>Indicator that you want ARMBARC to copy the date and timestamp from the archive logs to the ARCHIVE3 and ARCHIVE4 copies</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>NGT Copy Load</td>
<td>No value</td>
<td>Fully qualified name of your NGT Copy load library (if installed)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>NGT Copy Optionset</td>
<td>ACP$OPTS</td>
<td>Name of your NGT Copy option set (if installed)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Data Collection</td>
<td>Yes</td>
<td>Turns data collection on or off</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>DSNEXIT</td>
<td>No value</td>
<td>Fully qualified name of the DB2 exit library</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>DSNLOAD</td>
<td>No value</td>
<td>Fully qualified name of the DB2 load library</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>DS Member</td>
<td>No value</td>
<td>Subsystem’s member name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Group Attach ID</td>
<td>No value</td>
<td>Indicates if the subsystem options are for group attach</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location:</strong> Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Option</td>
<td>Default value</td>
<td>Brief description</td>
<td>Interface location</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>History File</td>
<td>No value</td>
<td>BMC archive history file is used by ARMBARC to record copies of the archive logs (ARCHIVE3 and ARCHIVE4)</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>ISPMLIB</td>
<td>SYS1.PROD.ISPMENU</td>
<td>Name of the ISPF message library (in which the ISP* messages reside)</td>
<td>Common Options</td>
</tr>
<tr>
<td>ISPTLIB</td>
<td>SYS1.PROD.ISPTENU</td>
<td>Name of the ISPF table library (in which the ISPCMD5 and ISPPROF tables reside)</td>
<td>Common Options</td>
</tr>
<tr>
<td>JCL Output</td>
<td>No value</td>
<td>Output data set name for JCL generated online if no data set name is provided on the file tailoring panel</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>JES2 Name</td>
<td>operating system ID</td>
<td>JES2 ID for each subsystem only if the JES2 IDs are different than the operating system IDs and the system is a data sharing environment</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>JES3 Name</td>
<td>No value</td>
<td>JES3 ID to enable RECOVERY MANAGER to recognize JES3 systems</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>Job Card 1 through Job Card 5</td>
<td>No value</td>
<td>Specifies a default job card for online execution and for ARMBGEN execution (in the absence of an ARMJCIN control card)</td>
<td>Common Options</td>
</tr>
<tr>
<td>Log Master Load</td>
<td>No value</td>
<td>Fully qualified name of the Log Master load library (if installed)</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>Log Master Optionset</td>
<td>ALP$OPTS</td>
<td>Name of your Log Master option set (if installed)</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>Log Range File</td>
<td>No value</td>
<td>For local point-in-time subsystem recoveries, RECOVERY MANAGER uses this sequential file to determine which table spaces have changed since their last copy</td>
<td>Subsystem Options</td>
</tr>
<tr>
<td>Logrange for Common Points</td>
<td>Yes</td>
<td>Enables or disables log range processing for common point analysis and unchanged analysis</td>
<td>Common Options</td>
</tr>
<tr>
<td>Option</td>
<td>Default value</td>
<td>Brief description</td>
<td>Interface location</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------------</td>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>Paclog CNTL</td>
<td>No value</td>
<td>Fully qualified name of the PACLOG .CNTL library</td>
<td><strong>Interface location</strong>: Subsystem Options</td>
</tr>
<tr>
<td>Paclog Load</td>
<td>No value</td>
<td>Fully qualified name of the PACLOG load library (if installed)</td>
<td><strong>Interface location</strong>: Subsystem Options</td>
</tr>
<tr>
<td>Paclog Optionset</td>
<td>ALM$OPTS</td>
<td>Name of your PACLOG option set (if installed)</td>
<td><strong>Interface location</strong>: Subsystem Options</td>
</tr>
<tr>
<td>Percent Prime</td>
<td>50</td>
<td>Percentage (1 to 100) to use for allocation of the primary extent for DASD data sets (SYSUT and SORTWK)</td>
<td><strong>Interface location</strong>: Common Options</td>
</tr>
<tr>
<td>Plan</td>
<td>ARMB vvr</td>
<td>Name of the RECOVERY MANAGER plan for repository and DB2 catalog access</td>
<td><strong>Interface location</strong>: Subsystem Options</td>
</tr>
<tr>
<td>Primary Archive On Tape</td>
<td>No</td>
<td>Specifies if the primary copy of the archive logs is on tape</td>
<td><strong>Interface location</strong>: Subsystem Options</td>
</tr>
<tr>
<td>Public Plan</td>
<td>Yes</td>
<td>Specifies whether to grant EXECUTE authority to PUBLIC to enable RECOVERY MANAGER to dynamically bind or rebind the plan as needed</td>
<td><strong>Interface location</strong>: Subsystem Options</td>
</tr>
<tr>
<td>NGT Recover Load</td>
<td>No value</td>
<td>Fully qualified name of your NGT Recover load library (if installed)</td>
<td><strong>Interface location</strong>: Subsystem Options</td>
</tr>
<tr>
<td>NGT Recover Optionset</td>
<td>AFR$OPTS</td>
<td>ddname of your NGT Recover option set (if installed)</td>
<td><strong>Interface location</strong>: Subsystem Options</td>
</tr>
<tr>
<td>Recovery Site Del/Def</td>
<td>Yes</td>
<td>Affects processing for SITETYPE RECOVERY</td>
<td><strong>Interface location</strong>: Subsystem Options</td>
</tr>
<tr>
<td>SMS VCAT</td>
<td>No</td>
<td>Specifies if the IDCAMS DEFINE statements generated by the product use an asterisk (*) in the VOLUMES parameter instead of the VOLSER where the data set currently resides</td>
<td><strong>Interface location</strong>: Common Options</td>
</tr>
<tr>
<td>Steplib Addition 1 through Steplib Addition 5</td>
<td>No value</td>
<td>Up to five additional STEPLIB data sets</td>
<td><strong>Interface location</strong>: Subsystem Options</td>
</tr>
<tr>
<td>Option</td>
<td>Default value</td>
<td>Brief description</td>
<td></td>
</tr>
<tr>
<td>---------------------------------</td>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Steplib Override 1 through</td>
<td>No value</td>
<td>Up to five STEPLIB override data sets</td>
<td></td>
</tr>
<tr>
<td>Steplib Override 5</td>
<td></td>
<td><strong>Interface location</strong>: Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Truncation Position</td>
<td>E (End)</td>
<td>Portion of a DB2 long name truncated when displayed on one of the online panels</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location</strong>: Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Truncation Characters</td>
<td>&gt;&gt;</td>
<td>Characters used as the substitution string for the truncated part of a DB2 long name when displayed on one of the panels</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location</strong>: Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Use CATALOG Parm</td>
<td>No</td>
<td>Use to have RECOVERY MANAGER use the CATALOG parameter with the VCAT name on your IDCAMS DEFINE statements</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location</strong>: Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>User Joblib 1 through User</td>
<td>No value</td>
<td>The fully qualified name of any operating system library that you require as a JOBLIB in the execution JCL</td>
<td></td>
</tr>
<tr>
<td>Joblib 3</td>
<td></td>
<td><strong>Interface location</strong>: Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>VCAT</td>
<td>No value</td>
<td>Name of your VSAM catalog</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location</strong>: Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Work File Database</td>
<td>DSNDB07</td>
<td>The work file database name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location</strong>: Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Work File Prefix</td>
<td>No value</td>
<td>The high-level data set prefix to be used when RECOVERY MANAGER generates references to work data sets in the output JCL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location</strong>: Subsystem Options</td>
<td></td>
</tr>
<tr>
<td>Work Unit</td>
<td>SYSDA</td>
<td>A valid unit name for temporary DASD allocations</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location</strong>: Common Options</td>
<td></td>
</tr>
<tr>
<td>zIIP Redirection</td>
<td>ENABLED</td>
<td>Determines whether zIIP processing is enabled</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Interface location</strong>: Subsystem Options</td>
<td></td>
</tr>
</tbody>
</table>

Whenever you create a new option set using DB2 Product Configuration, the technology provides the default values. If you migrated your RECOVERY MANAGER control file to an option set prior to invoking DB2 Product Configuration, the option set will not contain defaults.
Configuration option categories

The following figure shows the categories that are available for the RECOVERY MANAGER configuration options in the ARM$OPTS option set.

This panel shows the default categories available when you select Off from the Filter menu at the top of the panel.

Figure 201: Top level of RECOVERY MANAGER default ARM$OPTS option set

Common options

The following figure shows the Common Options, which are common to all subsystems.

Figure 202: Common Options
These configuration options apply to all DB2 subsystems in your environment:

**Arciven (1, 2 or 3)**

This option enables you to specify up to three names that are used in your environment by DASD management software to identify data sets that have been migrated from DASD to tape. RECOVERY MANAGER automatically recognizes **MIGRAT** and **ARCIVE** as archive volume names. If your DASD management archival system uses different names than these, specify them here.

**ISPMLIB**

This option contains the name of the ISPF message library (in which the ISP* messages reside). The installation default value is SYS1.PROD.ISPMENU.

**ISPTLIB**

This option contains the name of the ISPF table library (in which the ISPCMDS and ISPPROF tables reside). The installation default value is SYS1.PROD.ISPTENU.

**Job Card Options (1 through 5)**

This option enables you to specify a default job card for online execution and for ARMBGEN execution (in the absence of an ARMJCIN control card). You can enter up to 5 lines.

**Logrange for Common Points**

This option is not prompted during installation. It enables or disables log range processing for common point analysis and unchanged analysis. If LOGRANGE=NO is set in the options file, common recovery points are located via SYSCOPY.

**Percent Prime**

This option specifies a percentage (1 to 100) to use for allocation of the primary extent for DASD data sets (SYSUT and SORTWK). The default value is 50 per cent

--- **Note**

RECOVERY MANAGER overrides this value if the MAXPRIME value of a group is invoked.
**SMS VCAT**

If SMS VCAT=Y is set in the option set, the IDCAMS DEFINE statements generated by the product use an asterisk (*) in the VOLUMES parameter instead of the VOLSER where the data set currently resides. The default value is N for No.

**Work Unit**

This option specifies a valid unit name for temporary DASD allocations. The default is SYSDA.

**Subsystem options**

RECOVERY MANAGER requires configuration options that are unique to each of your DB2 subsystems.

For each subsystem and data sharing member, you should set specific configuration options. When you expand Subsystem Options, a list of subsystem displays. Expand the subsystem name to change option values.

*Figure 203 on page 757 through Figure 205 on page 758 show the Subsystem Options.*

*Figure 203: Subsystem Options (Part 1 of 3)*

<table>
<thead>
<tr>
<th>Command</th>
<th>ARM$OPTS - DEFAULT - RMD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter: Off</td>
<td>More: +</td>
</tr>
<tr>
<td>+ Common Options</td>
<td>Options Common to All Subsystems</td>
</tr>
<tr>
<td>- Subsystem Options (76)</td>
<td>Options for each Subsystem</td>
</tr>
<tr>
<td>+ DEDK</td>
<td>DB2 SSID</td>
</tr>
<tr>
<td>- DEDL</td>
<td>DB2 SSID</td>
</tr>
<tr>
<td>Plan . . . . . . . . . . ARMB110T</td>
<td></td>
</tr>
<tr>
<td>Bind Qualifier . . . . . BMACM</td>
<td></td>
</tr>
<tr>
<td>Public Plan . . . . . . Y</td>
<td>(Y=Yes, N=No)</td>
</tr>
<tr>
<td>Change Accum On Tape . . N</td>
<td>(Y=Yes, N=No)</td>
</tr>
<tr>
<td>Check Plus Load . . . . D2U.TEST1110.BMCL</td>
<td></td>
</tr>
<tr>
<td>NGT Copy Load . . . . RMD.TEST1110.BMCL</td>
<td></td>
</tr>
<tr>
<td>NGT Copy Optionset . . ACP$OPTS</td>
<td></td>
</tr>
<tr>
<td>NGT Recover Load . . . RMD.TEST1110.BMCL</td>
<td></td>
</tr>
<tr>
<td>NGT Recover Optionset . AFR$OPTS</td>
<td></td>
</tr>
<tr>
<td>Log Master Load . . . RMD.TEST1110.BMCL</td>
<td></td>
</tr>
<tr>
<td>Log Master Optionset . ALP$OPTS</td>
<td></td>
</tr>
<tr>
<td>Paclog Load . . . . . RMD.TEST1110.BMCL</td>
<td></td>
</tr>
<tr>
<td>Paclog Optionset . . . ALM$OPTS</td>
<td></td>
</tr>
<tr>
<td>Paclog CNTL. . . . . RMD.TEST1110.UDBC</td>
<td></td>
</tr>
</tbody>
</table>

*Figure 204: Subsystem Options (Part 2 of 3)*

<table>
<thead>
<tr>
<th>Command</th>
<th>ARM$OPTS - DEFAULT - RMD</th>
</tr>
</thead>
</table>

Appendix A  Option sets and configuration options 757
These configuration options must be specified separately for each DB2 subsystem and data sharing member in your environment.

**Plan**

This is the name of the RECOVERY MANAGER plan for repository and DB2 catalog access. The default name is ARMB\(vvr\), where \(vvr\) is the version and release level of RECOVERY MANAGER. For example in version 11.1.00, the
default value would be ARMB111. If RECOVERY MANAGER is already installed at your site and is using this default name (which is recommended), the installation process automatically replaces the vvr with the release number of the RECOVERY MANAGER being installed to avoid reusing the active RECOVERY MANAGER plan.

**Bind Qualifier**

This is the bind qualifier for the dynamic bind process. The default name is ARM vvr, where vvr is the version and release level of RECOVERY MANAGER. For example in version 11.1.00, the default value would be ARM111.

**WARNING**

When you change the bind qualifier, you must run a FREE for the plan and packages or an error will occur.

**Public Plan**

This option specifies whether to grant EXECUTE authority to PUBLIC to enable RECOVERY MANAGER to dynamically bind or rebind the plan as needed. If you choose to set this option to N, you must grant EXECUTE authority to users as needed. The default is Y.

**Change Accum on Tape**

Specify Y if you have R+/CHANGE ACCUM installed and your change accum files are on tape. Otherwise specify N, which is the default value.

**Check Plus Load**

Specify the fully qualified name of your NGT Check load library (if installed).

**NGT Copy Load**

Specify the fully qualified name of the NGT Copy load library (if installed).

**NGT Copy Optionset**

Specify the name of your NGT Copy option set (if installed). The default is ACP$OPTS.

**NGT Recover Load**

Specify the fully qualified name of the NGT Recover load library (if installed).

**NGT Recover Optionset**

Specify the name of your NGT Recover option set (if installed). The default is AFR$OPTS.
Log Master Load

Specify the fully qualified name of the Log Master load library (if installed).

Log Master Optionset

Specify the name of your Log Master option set (if installed). The default is ALP$OPTS.

Paclog Load

Specify the fully qualified name of the PACLOG load library (if installed).

Paclog Optionset

Specify the name of your PACLOG option set (if installed). The default is ALM$OPTS.

Paclog CNTL

Specify the fully qualified name of the PACLOG .CNTL library.

Active Log 1 Prefix

Enter the prefix of the primary active log data set names. For example, if the first active log name is DBDFCAT.LOGCOPY1.DS01, the prefix for the active logs is DBDFCAT.LOGCOPY1.

Active Log 2 Prefix

Enter the prefix of the alternate active log data set names if you are using dual actives. (In DSNZPARM, the DSN6LOGP macro specifies TWOACTV=YES.)

Archive Log 1 Prefix

Enter the prefix of the primary archive log data set names. For example, if one of the archive logs is named DBDFCAT.ARCHLOG1.A0002266, the prefix is DBDFCAT.ARCHLOG1. This value is the same as the one specified in DSNZPARM on the DSN6ARVP macro, ARCPFX1=.

Archive Log 2 Prefix

Enter the prefix of the alternate archive log data set names if you are using dual archive logs. (In DSNZPARM, the DSN6LOGP macro specifies TWOARCH=YES.) This value is the same as the one specified in DSNZPARM on the DSN6ARVP macro, ARCPFX2=. 
Primary Archive On Tape

Specify Y if the primary copy of the archive logs is on tape. Otherwise specify N. N is the default value.

Alternate Archive On Tape

Specify Y if the alternate copy of the archive logs is on tape. Otherwise specify N. N is the default value.

Archive Use Timestamp in DSN

Specify Y if the archive logs use a date/timestamp value in the data set names. Otherwise specify N. If you are uncertain, you can look at the archive logs under ISPF 3.4 or look in DSNZPARM for TSTAMP on the DSN6ARVP macro. The default value is N.

BSDS 1

Enter the full data set name of the primary bootstrap data set.

BSDS 2

Enter the full data set name of the alternate bootstrap data set.

Data Collection

The default value is Y, which turns data collection on. Enter N to turn data collection off.

DSNEXIT

Specify the fully qualified name of the DB2 exit library. This option is necessary only if the DB2 exit library is not in the operating system link list. This data set name is added as a STEPLIB to any DB2 execution required by RECOVERY MANAGER.

DSNLOAD

Specify the fully qualified name of the DB2 load library. This option is necessary only if the DB2 load library is not in the operating system link list. This data set name is added as a STEPLIB to any DB2 execution required by RECOVERY MANAGER.

DS Member

For a data sharing subsystem, enter the subsystem's member name. If you are uncertain of the name, a DISPLAY GROUP will show the name, or you can look in DSNZPARM for MEMBNAME on the DSN6GRP macro.
History File

The BMC archive history file is used by ARMBARC to record copies of the archive logs (ARCHIVE3 and ARCHIVE4). It also records the copies of the DB2 catalog spaces that are not registered in SYSIBM.SYSCOPY (SYSCOPY, SYSUTILX, DBD01, SYSDBDXA). Because this file is used to record archive logs, a separate history file must exist for each data sharing member. If it does not already exist, the file is allocated and initialized by the ARMBARC program. If at any time you wish to re-create the history file, use the member ARMHSTEX in the .CNTL data set. New users of RECOVERY MANAGER must run ARMBARC or use the sample member ARMHSTEX in the .CNTL data set to create the history file for each DB2 subsystem. BMC recommends that you use the subsystem ID in the name of the history file.

Note

If you use the PACLOG for DB2 product to process archive logs, PACLOG must share the same history file used by RECOVERY MANAGER.

Use CATALOG Parm

Enter Y to have RECOVERY MANAGER use the CATALOG parameter with the VCAT name on your IDCAMS DEFINE statements. The default value is N.

BMC Infrastructure Load

Specify the SMP/E target library in which the BMC infrastructure common code is stored. This library contains code that is required to run RECOVERY MANAGER and other BMC products. The installation default is the installation target library. If you are not using the target library for the infrastructure code, you must either specify the correct library in this field or the library must be link listed. RECOVERY MANAGER jobs cannot run without access to this library.

BMC Infrastructure Load 2

Specify the SMP/E target library in which the BMC infrastructure common code is stored. This library contains code that is required to run RECOVERY MANAGER and other BMC products. The installation default is the installation target library. If you are not using the target library for the infrastructure code, you must either specify the correct library in this field or the library must be link listed. RECOVERY MANAGER jobs cannot run without access to this library.

JCL Output

The JCL Output option is only used on panel ARMFT001 (JCL Specification) when generating JCL online. If the data set name to write the JCL into is blank, the JCL Output option is retrieved to get the data set name. The name
is then validated and stored in an ISPF variable. The next time the panel is displayed, the data set name is retrieved from the ISPF variable pool.

**JES2 Name**

Enter the JES2 ID for each subsystem only if the JES2 IDs are different than the operating system IDs and the system is a data sharing environment. This value defaults to the operating system ID.

**JES3 Name**

Enter the JES3 ID to enable RECOVERY MANAGER to recognize JES3 systems.

**Log Range File**

For local point-in-time subsystem recoveries, RECOVERY MANAGER uses this sequential file to determine which table spaces have changed since their last copy. Only one file exists per DB2 subsystem and, on a data sharing subsystem, each subsystem uses the same file name. Enter a fully qualified data set name (RECOVERY MANAGER will allocate the file if it does not already exist).

**Note**

For data sharing environments, be sure the ARM$OPTS member has the same name for each member in the data sharing group.

**Recovery Site Del/Def**

Recovery Site Del/Def only affects processing for SITETYPE RECOVERY:

- Recovery Site Del/Def Y, the default value, generates DELETE/DEFINE statements regardless of the group Recover option Redefine VCAT objects.

- With Recovery Site Del/Def N, the group Recover option Redefine VCAT objects determines if DELETE/DEFINE statements are generated.

**Note**

For SITETYPE LOCAL, the group Recover option Redefine VCAT objects always determines if DELETE/DEFINE statements are generated.

**Steplib Addition 1 through Steplib Addition 5**

Optionally specify up to five additional STEPLIB data sets. These data sets are generated after the generation of the BMC product step libraries. ALL additional STEPLIB libraries must be APF-authorized.

You can use the additional data sets for any executable code that should be accessed after the BMC product load libraries. For example, an additional
data set could contain an options module that would be used to specify default values.

**Steplib Override 1 through Steplib Override 5**

Optionally specify up to five STEPLIB override data sets. These data sets are generated prior to the generation of the BMC product step libraries. ALL override libraries must be APF-authorized.

You can use the override data sets for any executable code that should be accessed prior to the BMC product load libraries. For example, an override data set could contain an options module that would override the product default options.

**Truncation Position**

Specify which portion of a DB2 long name is truncated when displayed on one of the online panels. Valid values are

- **E**—end (the default)
- **B**—beginning
- **M**—middle

**Example**

The long name
`RMD128CHARACTERCREATOR.IC15P21L128MAXIMUMCHARACTERIXN` will be truncated as follows:

- `ssid.TRUNCPOS=E, ssid.TRUNCCCHAR=>>
  RMD128=>>.IC15P21L128MAXIM>>`
- `ssid.TRUNCPOS=M, ssid.TRUNCCCHAR= >>
  RMD>>.IC15P21L>> ACTERIXN`
- `ssid.TRUNCPOS=B, ssid.TRUNCCCHAR= !`
  `!!REATOR.!!MUMCHARACTERIXN`

**Truncation Characters**

Specify which characters are used as the substitution string for the truncated part of a DB2 long name when displayed on one of the panels. The default is `>>`.

**Copy Archive Timestamp**

Specify `Y`, the default value, if you want ARMBARC to copy the date and timestamp from the archive logs to the ARCHIVE3 and ARCHIVE4 copies. Otherwise specify `N`. 
User Joblib 1 through User Joblib 3

Specify the fully qualified name of any operating system library that you require as a JOBLIB in the execution JCL.

VCAT

Enter the name of your VSAM catalog (VCAT) if the high level of your VSAM data sets is not defined as an alias in the operating system master catalog. This value enables the VCAT to be used on the IDCAMS DELETE/DEFINE statements when you delete and define new VSAM data sets.

Work File Database

For a non-data-sharing subsystem, the work file database name is (and must be) DSNDB07. For a data sharing subsystem, the work file database name is unique for each member of the subsystem.

Work File Prefix

This option enables you to specify the high-level data set prefix to be used when RECOVERY MANAGER generates references to work data sets in the output JCL. The prefix can be 1 to 17 characters and must conform to the operating system data set qualifier conventions. If not specified, RECOVERY MANAGER work data sets are qualified by the TSO prefix (if specified) or by the TSO USERID. A WORKPREFIX value can be specified for each DB2 subsystem.

zIIP Redirection

Specify whether you want to use zIIP redirection to run I/O completion Service Request Blocks (SRBs) on zIIP processors. This option requires EXTENDED BUFFER MANAGER (XBM) or SNAPSHOT UPGRADE FEATURE (SUF) from BMC. The default, ENABLED, uses the zIIP redirection. Specify DISABLED if you do not want to use zIIP redirection.
BMC Common DB2 repository tables

The BMC common DB2 repository is made up of several DB2 tables.

Naming conventions

The BMC common DB2 repository tables follow a naming convention. The following table provides the synonyms and local table names.

**Note**
The local table names might be different at your site, based on options selected during product installation.

<table>
<thead>
<tr>
<th>Synonym</th>
<th>Local table name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMCSCC_OBJSETS</td>
<td>BMCUTIL.CMN_OS</td>
</tr>
<tr>
<td>BMCSCC_OBJSET_DEF</td>
<td>BMCUTIL.CMN_OS_DEF</td>
</tr>
<tr>
<td>BMCSCC_OBJSET_SQL</td>
<td>BMCUTIL.CMN_OS_SQL</td>
</tr>
<tr>
<td>BMCSCC_GRPOPTS</td>
<td>BMCUTIL.CMN_OS_OPTS</td>
</tr>
<tr>
<td>BMCSCC_PRODREG</td>
<td>BMCUTIL.CMN_OS_PREG</td>
</tr>
<tr>
<td>BMCSCC_GROUPAUTH</td>
<td>BMCUTIL.CMN_OS_GAUTH</td>
</tr>
</tbody>
</table>

OBJSETS table

The following table describes the contents of the OBJSETS table. This table describes and provides information about object sets. This table contains one row for each object set defined in the repository.
### OBJSET_DEF table

The following table describes the contents of the OBJSET_DEF table. This table contains one row for each object set definition specification defined for an object set.

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSNAME</td>
<td>VARCHAR(27) NOT NULL</td>
<td>Name of the object set</td>
</tr>
<tr>
<td>SEQNO</td>
<td>SMALLINT NOT NULL WITH DEFAULT</td>
<td>Sequence number of definition</td>
</tr>
<tr>
<td>CHECKSUM</td>
<td>SMALLINT NOT NULL</td>
<td>Verification value from API updates</td>
</tr>
<tr>
<td>OSNAME_DELIMITED</td>
<td>CHAR(1) NOT NULL WITH DEFAULT 'N'</td>
<td>For use with delimited names Will be 'Y' if the related column is a delimited name (entered with double quotes around it) when entered</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>INCEXC_IND</td>
<td>CHAR(1) NOT NULL</td>
<td>Include or exclude indicator (+, -)</td>
</tr>
<tr>
<td>PATTERN_TYPE</td>
<td>CHAR(2) NOT NULL</td>
<td>Pattern for include or exclude:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- TS (table space name pattern)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- IX (index name pattern)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- TB (table name pattern)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- IS (index space name pattern)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- PL (plan name pattern)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- PG (package name pattern)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- SG (stogroup name pattern)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- OS (object set name pattern)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- SQ (dynamic SQL pattern)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- RP (repository plan)</td>
</tr>
<tr>
<td>INC_IX</td>
<td>CHAR(1) NOT NULL</td>
<td>Include related indexes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Y (Yes)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- N (No)</td>
</tr>
<tr>
<td>INC_RI</td>
<td>CHAR(1) NOT NULL</td>
<td>Include RI objects</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Y (Yes)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- N (No)</td>
</tr>
<tr>
<td>INC_LOBS</td>
<td>CHAR(1) NOT NULL</td>
<td>Include LOB objects</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Y (Yes)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- N (No)</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| INC_XML     | CHAR(1) NOT NULL | Include XML objects  
  ▪ Y (Yes)  
  ▪ N (No) |
| INC_CLONES  | CHAR(1) NOT NULL | Include clones only  
  ▪ Y (Yes)  
  ▪ N (No) |
| BY_PART     | CHAR(1) NOT NULL | Expand objects by partition  
  ▪ Y (Yes)  
  ▪ N (No) |
| PART_BEG    | SMALLINT NOT NULL | Beginning partition number (0-4096) |
| PART_END    | SMALLINT NOT NULL | Ending partition number (0-4096) |
| REF_SEQ_NBR | SMALLINT NOT NULL | For future use |
| DESC        | VARCHAR(60) NOT NULL | Description of the specification |
| OBJ_QUAL1   | VARCHAR(128) NOT NULL | Object qualifier 1 |
| OBJ_QUAL2   | VARCHAR(128) NOT NULL | Object qualifier 2 |
| OBJ_QUAL3   | VARCHAR(128) NOT NULL | Object qualifier 3 |
| UNIQUALS    | CHAR(1) NOT NULL | UNICODE indicator |
| UPDATE_UID  | CHAR(8) NOT NULL | ID of last updater of object set definitions |
| UPDATE_TSMP | TIMESTAMP NOT NULL WITH DEFAULT | Timestamp of last maintenance activity |
| PACKAGE_VERSION | SMALLINT NOT NULL | Package version |
| INC_HISTORY | CHAR(1) NOT NULL WITH DEFAULT 'N' | Include related history objects  
  ▪ Y (Yes)  
  ▪ N (No) |
<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUAL1_DELIMITED</td>
<td>CHAR(1) NOT NULL WITH DEFAULT 'N'</td>
<td>For use with delimited names Will be 'Y' if the related column is a delimited name (entered with double quotes around it) when entered</td>
</tr>
<tr>
<td>QUAL2_DELIMITED</td>
<td>CHAR(1) NOT NULL WITH DEFAULT 'N'</td>
<td>For use with delimited names Will be 'Y' if the related column is a delimited name (entered with double quotes around it) when entered</td>
</tr>
<tr>
<td>QUAL3_DELIMITED</td>
<td>CHAR(1) NOT NULL WITH DEFAULT 'N'</td>
<td>For use with delimited names Will be 'Y' if the related column is a delimited name (entered with double quotes around it) when entered</td>
</tr>
</tbody>
</table>

**OBJSET_SQL table**

The following table describes the contents of the OBJSET_SQL table. This table contains one row for each object set specification in dynamic SQL (type SQ).

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSNAME</td>
<td>VARCHAR(27) NOT NULL</td>
<td>Name of the object set</td>
</tr>
<tr>
<td>SPEC_SEQNO</td>
<td>SMALLINT NOT NULL</td>
<td>Sequence number from OBJSET_DEF table</td>
</tr>
<tr>
<td>SEQNO</td>
<td>SMALLINT NOT NULL</td>
<td>Sequence number to order multiple SQL entries</td>
</tr>
<tr>
<td>TEXT</td>
<td>VARCHAR(72) NOT NULL</td>
<td>Line of SQL text</td>
</tr>
</tbody>
</table>

**GRPOPTS table**

The following table describes the contents of the GRPOPTS table. This table contains one row for each option defined to either a defined group, or a subsystem level option.

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSNAME</td>
<td>VARCHAR(27) NOT NULL</td>
<td>Name of object set</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td>OPTION_TYPE</td>
<td>CHAR(10) NOT NULL</td>
<td>Option type</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Backup—ARMOPTBKUP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Recover—ARMOPTRCVR</td>
</tr>
<tr>
<td></td>
<td></td>
<td>These are the option types</td>
</tr>
<tr>
<td></td>
<td></td>
<td>currently used by RECOVERY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MANAGER. The option type is</td>
</tr>
<tr>
<td></td>
<td></td>
<td>defined by the product, so this</td>
</tr>
<tr>
<td></td>
<td></td>
<td>list is product-dependent.</td>
</tr>
<tr>
<td>OPTION</td>
<td>VARCHAR(200) NOT NULL</td>
<td>Option name</td>
</tr>
<tr>
<td>OPT_VALUE</td>
<td>VARCHAR(200) NOT NULL</td>
<td>Value for named option</td>
</tr>
</tbody>
</table>

### PRODREG table

The following table describes the contents of the PRODREG table. There should be one entry for each product and version that is registered.

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRODUCT_ID</td>
<td>CHAR(3) NOT NULL</td>
<td>Product ID</td>
</tr>
<tr>
<td>PLAN_NAME</td>
<td>VARCHAR(24) NOT NULL</td>
<td>Plan name</td>
</tr>
<tr>
<td>PRODUCT_VERSION</td>
<td>CHAR(4) NOT NULL</td>
<td>Product version</td>
</tr>
</tbody>
</table>

### GROUPAUTH table

The following table describes the contents of the GROUPAUTH table. This table optionally contains one row for each authority granted on a group. No rows exist if no authority has been granted.

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSNAME</td>
<td>VARCHAR(27) NOT NULL</td>
<td>Name of object set</td>
</tr>
<tr>
<td>GRANTEE</td>
<td>CHAR(8) NOT NULL</td>
<td>AUTHID to whom authorization was granted</td>
</tr>
<tr>
<td>TYPE</td>
<td>CHAR(1) NOT NULL</td>
<td>Type of authorization granted</td>
</tr>
<tr>
<td>GRANTOR</td>
<td>CHAR(8) NOT NULL</td>
<td>Grantor of authorization</td>
</tr>
<tr>
<td>DATE_GRANTED</td>
<td>TIMESTAMP NOT NULL</td>
<td>Timestamp of when authorization was granted</td>
</tr>
</tbody>
</table>
RECOVERY MANAGER repository

This appendix describes the tables associated with the RECOVERY MANAGER repository.

RECOVERY MANAGER repository tables

The RECOVERY MANAGER repository is stored in DB2 tables.

This storage requires that DB2 be operational when you save and retrieve a group. The DB2 tables that compose the RECOVERY MANAGER repository are described in the following sections.

The "short" name of each table is given in the table title. The full name is ARMvr_xxx, where xxx is the short name.

Coordinated disaster recovery information: CRRDRPT table

The following table describes the contents of the CRRDRPT table.

This table contains one row for each coordinated recovery point.

Table 56: CRRDRPT table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRRDR_POINT</td>
<td>TIMESTAMP NOT NULL</td>
<td>Timestamp of coordinated recovery point</td>
</tr>
<tr>
<td>MEMBER</td>
<td>CHAR(8)</td>
<td>DB2 member name or SSID</td>
</tr>
<tr>
<td>RBA</td>
<td>VARCHAR(10) BINARY DATA</td>
<td>RBA corresponding to recovery point</td>
</tr>
<tr>
<td>LRSN</td>
<td>VARCHAR(10) BINARY DATA</td>
<td>LRSN corresponding to recovery point</td>
</tr>
</tbody>
</table>
Recovery history: UTILITY_RUN table

The following table describes the contents of the UTILITY_RUN table.

This table is only used by the BMC Recovery Management for DB2 solution.

Table 57: UTILITY_RUN table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UTILITY_RUN_ID</td>
<td>INTEGER NOT NULL GENERATED BY DEFAULT AS IDENTITY</td>
<td>Identifier for the recovery run&lt;br&gt;One value is generated for each set of jobs that are generated by ARMBSRR and ARMBGEN. The value is stored in the JOB, PHASE, and TS tables to link all data for a run.</td>
</tr>
<tr>
<td>VERSION</td>
<td>INTEGER NOT NULL</td>
<td>Version value used to detect incompatibility between code and table structure&lt;br&gt;This is the version of the code which stored the row, and is checked when the row is retrieved.</td>
</tr>
<tr>
<td>SSID</td>
<td>CHAR(4) NOT NULL</td>
<td>Data sharing member name or SSID</td>
</tr>
<tr>
<td>DCTOKEN</td>
<td>CHAR(8) NOT NULL</td>
<td>Token used to link JCL steps for a common run and to allow concurrent runs&lt;br&gt;For this release, this value is always DRRUN.</td>
</tr>
<tr>
<td>START_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>Start of recovery run</td>
</tr>
<tr>
<td>END_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>End of recovery run&lt;br&gt;Must be NULL to allow data storage by NGT Recover. At completion of a DR or test, this value minus START_TIMESTAMP gives the overall elapsed time of the DR.</td>
</tr>
<tr>
<td>RUN_TYPE</td>
<td>CHAR(1) NOT NULL</td>
<td>Indicates the type of recovery, as follows:&lt;br&gt;1=actual recovery&lt;br&gt;2=simulated&lt;br&gt;3=estimated</td>
</tr>
<tr>
<td>MIRRORING_USED</td>
<td>CHAR(1) NOT NULL WITH DEFAULT</td>
<td>Indicates whether mirroring is used&lt;br&gt;0=mirroring is not used&lt;br&gt;1=mirroring is used</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>NUM_TS_TO_SAVE</td>
<td>SMALLINT NOT NULL</td>
<td>Number of worst-case table spaces to keep on record. Used to limit data storage. RECOVERY MANAGER currently stores a value of 10.</td>
</tr>
<tr>
<td>TOLOGPOINT</td>
<td>VARCHAR(10) FOR BIT DATA</td>
<td>Recovery point. Currently unused.</td>
</tr>
<tr>
<td>ELAP_SEC_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated elapsed time. <strong>Note:</strong> This value reflects the Recovery Management solution's estimate of END_TIMESTAMP minus START_TIMESTAMP. This value may not be accurate in this release, but will be consistent. By comparing the estimates to actual recovery times, you should be able to calculate a factor to apply to the estimates and so derive an accurate time estimate.</td>
</tr>
<tr>
<td>AVG_TS_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Average elapsed recovery time per table space in number of seconds (rounded down).</td>
</tr>
<tr>
<td>AVG_TS_SEC_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated elapsed recovery time per table space (rounded down)</td>
</tr>
<tr>
<td>NUM_JOBS</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Number of jobs in the recovery run (stored during an estimation run)</td>
</tr>
<tr>
<td>NUM_TS_RECOV</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Number of table spaces in the recovery run (stored during an estimation run)</td>
</tr>
<tr>
<td>NUM_IX_RECOV</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Number of index spaces in the recovery run (stored during an estimation run) This value reflects indexes recovered from copies and log, but does not include rebuilds.</td>
</tr>
<tr>
<td>NUM_BYTES_RECOV</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of bytes recovered during the recovery run (stored during an estimation run) This value reflects table spaces and indexes recovered from copies and log, but does not include indexes that are rebuilt.</td>
</tr>
</tbody>
</table>
Recovery history: JOB table

The following table describes the contents of the JOB table.

This table is only used by the BMC Recovery Management for DB2 solution.

Table 58: JOB table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB_ID</td>
<td>INTEGER NOT NULL GENERATED BY DEFAULT AS IDENTITY</td>
<td>Identifier for each job in the recovery run</td>
</tr>
<tr>
<td>UTILITY_RUN_ID</td>
<td>INTEGER NOT NULL</td>
<td>Identifier for the recovery run (matches the value in the UTILITY_RUN table)</td>
</tr>
<tr>
<td>JOBNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Name of the job</td>
</tr>
<tr>
<td>START_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>Starting time of the job</td>
</tr>
<tr>
<td>END_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>Ending time of the job The elapsed time to run a recovery job is represented by the difference between START_TIMESTAMP and END_TIMESTAMP.</td>
</tr>
<tr>
<td>JOB_TYPE</td>
<td>CHAR(1) NOT NULL</td>
<td>Indicates the type of job, as follows: 1= NGT Recover job 2= Phase 1 of system recovery 3= Phase 2 of system recovery 4= SYSUTIL recovery 5= log file restore 6= DBD01 recovery 7= remainder of catalog and directory recovery 8= recovery of user indexes on the catalog 9= recovery of the RECOVERY MANAGER repository A= recovery of the CHANGE ACCUM repository B= system recovery using DSNUTILB C= application recovery using DSNUTILB D= estimation of system recovery E= simulation of system recovery</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>INCOMPLETE_FLAG</td>
<td>CHAR(1) NOT NULL WITH DEFAULT</td>
<td>Flag to indicate run completion 0— indicates that this row completely records all data for this job. 1— indicates the record is from a run that either was restarted or experienced automatic fallback; therefore the data is stored in more than one row. If more than one row is recorded for a job, the rows will have the same JOBNAME value but JOB_ID values will be different.</td>
</tr>
<tr>
<td>DC_RET_CODE</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>A nonzero integer indicates a problem occurred with data collection. The programs attempt to save as much data as possible, even when an error occurs.</td>
</tr>
<tr>
<td>ELAP_SEC_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated elapsed time in seconds (rounded down)</td>
</tr>
<tr>
<td>NUM_TS_RECOV</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Number of table spaces recovered in the job</td>
</tr>
<tr>
<td>NUM_IX_RECOV</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Number of index spaces recovered in the job. This value reflects indexes recovered from copies and log, but not rebuilt.</td>
</tr>
<tr>
<td>NUM_BYTES_EST</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Estimated number of bytes recovered in the job. This value reflects the sum of NACTIVE(F) for each table space, and a calculated number of pages for each index, based on CARD(F) for each table.</td>
</tr>
<tr>
<td>NUM_BYTES_RECOV</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Actual number of bytes recovered in the job. This value does not reflect bytes written during index rebuilds.</td>
</tr>
<tr>
<td>TAF_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Estimated tape allocation factor. This value is used to estimate tape allocation time (currently always set to 30 seconds).</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| DAF_EST     | REAL NOT NULL WITH DEFAULT | Disk allocation factor
This value is used to estimate disk allocation time (currently always set to 1 second). |
| LRF_EST     | REAL NOT NULL WITH DEFAULT | Log read factor
This value is used to estimate time to read log files (currently set to 1.4E-7 seconds per byte, which corresponds to about 400M per minute). |
| MLT_EST     | SMALLINT NOT NULL WITH DEFAULT | MAXLOGS threshold
This value is used to reflect the fact that increasing MAXLOGS past a point does not speed up log reading (currently set to 6). |
| LSC_EST     | REAL NOT NULL WITH DEFAULT | Log sort coefficient
This value is used to estimate log sort time; the time is calculated as:
\[ t = \text{bytes} \times \log(\text{bytes}) \times LSC\_EST. \]
It is currently set to 6.0E-8. |
| RF_EST      | REAL NOT NULL WITH DEFAULT | Recall factor
The number of seconds estimated for each data set to be recalled. Currently set to 0.1 second. |
| SCT_EST     | REAL NOT NULL WITH DEFAULT | Snap copy term
The number of seconds to snap a data set. Currently set to 3 seconds. |
| OMF_EST     | REAL NOT NULL WITH DEFAULT | Optimized merge factor
The fraction of pages read during a LOGONLY or BACKOUT run. Currently set to .2 (20%). |
| SRF_EST     | REAL NOT NULL WITH DEFAULT | Space read factor
Value used to estimate time to read VSAM data sets. Currently set to 1.8E-7 seconds per byte (about 80,000 4K pages per minute). |
| CRF_EST     | REAL NOT NULL WITH DEFAULT | Copy read factor
Value used to estimate time to read input image copies. Currently set to 1.4E-7 seconds per byte (about 100,000 4K pages per minute). |
<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARF_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Accum read factor &lt;br&gt;Value used to estimate time to read input change accumulation files. Currently set to 1.4E-7 seconds per byte (about 400M per minute).</td>
</tr>
<tr>
<td>SWF_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Space write factor &lt;br&gt;Value used to estimate time to write VSAM data sets. Currently set to 2.4E-7 seconds per byte (about 60,000 4K pages per minute).</td>
</tr>
<tr>
<td>CWF_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Copy write factor &lt;br&gt;Value used to estimate time to write output copies. Currently set to 1.8E-7 seconds per byte (about 80,000 4K pages per minute).</td>
</tr>
<tr>
<td>AWF_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Accum write factor &lt;br&gt;Value used to estimate time to write accum files. Currently set to 1.8E-7 seconds per byte (about 300M per minute).</td>
</tr>
<tr>
<td>KSC_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Key sort coefficient &lt;br&gt;Used to estimate key sort time; the time is calculated as: &lt;br&gt;$t = \text{bytes} \times \log(\text{bytes}) \times \text{KSC_EST}$. &lt;br&gt;Currently set to 6.0E-8.</td>
</tr>
<tr>
<td>BSWF_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Build space write factor &lt;br&gt;Value used to estimate time to rebuild an index. Currently set to 3.7E-7 seconds per byte (about 40,000 pages per minute).</td>
</tr>
<tr>
<td>WWF_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Writekeys write factor &lt;br&gt;Value used to estimate time to write UNLOADKEYS file. Currently set to 1.8E-7 seconds per byte (about 300M per minute).</td>
</tr>
<tr>
<td>NUM_LOGS_RESTORED</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>The number of logs restored during recovery &lt;br&gt;This value refers to the log files copied to DASD during Phase 1 of system recovery.</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>LOG_RECS_SORTED</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of log records sorted during recovery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This value reflects the amount of log selected for all spaces in the recover job.</td>
</tr>
<tr>
<td>LOG_RECS_PROCESSED</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of log records processed during recovery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This reflects the log records (that are sorted and processed by the merge process) that are to be written to an accum file or to be considered for application to a page image.</td>
</tr>
<tr>
<td>LOG_RECS_APPLIED</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of log records applied during recovery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This reflects log records actually applied to page images. A log record can be included in the LOG_RECS_PROCESSED count and not in LOG_RECS_APPLIED because a comparison of its LRSN or RBA to the PGLOGRBA value in the corresponding page shows that the page already reflects this update. This difference can be significant if SHRLEVEL CHANGE image copies are used.</td>
</tr>
<tr>
<td>LOG_BYTES_SORTED</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of bytes of log records sorted during recovery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Corresponds to LOG_RECS_SORTED, and with that value, enables calculation of the average log record size.</td>
</tr>
<tr>
<td>NUM_E35_LOG_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of times that the merge process waited for a buffer of log records from the log sort</td>
</tr>
<tr>
<td>E35_WAIT_LOG_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds that the merge process waited for log records from the log sort</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If this value is unacceptably high relative to the elapsed time of the job, you should consider reducing the size of the log sort (by splitting the job, copying more often, or running R+/CHANGE ACCUM).</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MAX_E35_LOG_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in one wait for the log sort. This value is saved to show anomalies. A high value is typical in the first merge in the job because it is probably waiting for the output phase of the log sort to finish. Otherwise, if one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>SPI_BYTES_READ</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of space input bytes read during recovery. This is the number of bytes reads during BACKOUT and LOGONLY recoveries, and during merges following inline copy restores.</td>
</tr>
<tr>
<td>SPI_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Time in seconds required to allocate spaces for input. This is the total time spent in dynamic allocation of spaces for BACKOUT and LOGONLY recoveries. It is usually not a concern because this time is incurred by a subtask.</td>
</tr>
<tr>
<td>NUM_SPI_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for reads from input spaces.</td>
</tr>
<tr>
<td>SPI_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for reads from input spaces.</td>
</tr>
<tr>
<td>MAX_SPI_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in one wait for a read from an input space. If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>FCI_BYTES_READ</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of bytes read from full image copies during recovery.</td>
</tr>
<tr>
<td>FCI_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Total time in seconds required to allocate full image copies.</td>
</tr>
<tr>
<td>NUM_FCI_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for reads from full image copies.</td>
</tr>
<tr>
<td>FCI_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for reads from full image copies.</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MAX_FCI_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in one wait for a read from a full image copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>ICI_BYTES_READ</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of bytes read from incremental image copies during recovery</td>
</tr>
<tr>
<td>ICI_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Total time in seconds required to allocate incremental image copies</td>
</tr>
<tr>
<td>NUM_ICI_WAIT</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for reads from incremental image copies</td>
</tr>
<tr>
<td>ICI_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for reads from incremental image copies</td>
</tr>
<tr>
<td>MAX_ICI_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a read from an incremental copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>CAI_BYTES_READ</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of bytes read from change accumulation files during recovery</td>
</tr>
<tr>
<td>CAI_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Total time in seconds required to allocate input change accumulation files</td>
</tr>
<tr>
<td>NUM_CAI_WAIT</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for reads from change accumulation files</td>
</tr>
<tr>
<td>CAI_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for reads from change accumulation files</td>
</tr>
<tr>
<td>MAX_CAI_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a read from a change accumulation file</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>SPO_BYTES_WRITTEN</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of bytes written to table spaces and index spaces during recovery</td>
</tr>
<tr>
<td>SPO_SNAP_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds required to restore Instant Snapshot copies to the space output fields</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SPO_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds required to allocate table space and index space data sets for output.</td>
</tr>
<tr>
<td>NUM_SPO_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for writes to table spaces and indexes</td>
</tr>
<tr>
<td>SPO_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for writes to table spaces and indexes</td>
</tr>
<tr>
<td>MAX_SPO_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a write to a table space or index.  If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>FCO_BYTES_WRITTEN</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of bytes written to full copy output files during recovery.</td>
</tr>
<tr>
<td>FCO_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Amount of time in seconds required to allocate the full copy output files.</td>
</tr>
<tr>
<td>NUM_FCO_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for writes to full copy output files.</td>
</tr>
<tr>
<td>FCO_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for writes to full copy output files.</td>
</tr>
<tr>
<td>MAX_FCO_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a write to an output full image copy. If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>CAO_BYTES_WRITTEN</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of bytes written to change accumulation output files during recovery.</td>
</tr>
<tr>
<td>CAO_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Amount of time in seconds required to allocate the change accumulation output files. Currently unused (set to 0).</td>
</tr>
<tr>
<td>NUM_CAO_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for writes to the change accumulation output files. Currently unused (set to 0).</td>
</tr>
<tr>
<td>CAO_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting while attempting to write to the change accumulation output files. Currently unused (set to 0).</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MAX_CAO_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a write to a change accumulation output file. Currently unused (set to 0).</td>
</tr>
<tr>
<td>NUM_KEY_BYTES</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of bytes sorted for index rebuilds</td>
</tr>
<tr>
<td>NUM_E15_KEY_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of times the key extraction process (merge, unload, or the readkeys step of a build) waited for the E15 (input) key sort exit to pass a buffer to sort</td>
</tr>
<tr>
<td>E15_WAIT_KEY_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds in key sort input waits</td>
</tr>
<tr>
<td>MAX_E15_KEY_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for the E15 key sort exit to pass a buffer to sort. If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>NUM_E35_KEY_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of times the index build process waited for a buffer of keys from the E35 (output) key sort index</td>
</tr>
<tr>
<td>E35_WAIT_KEY_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds in key sort output waits</td>
</tr>
<tr>
<td>MAX_E35_KEY_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for the E35 key sort exit to return a buffer of keys from sort. If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>INCOMPLETE_FLAG</td>
<td>CHAR(1) NOT NULL WITH DEFAULT</td>
<td>Indicates the record is from a run that either was restarted or experienced automatic fallback; therefore the data is stored in more than one row.</td>
</tr>
<tr>
<td>LSF_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Log sort factor. This value is multiplied by the number of bytes of log to be read and is used to calculate the number of bytes to sort. It is currently set to .125 (if there are 32K spaces in the run) .19 (if there are 16K spaces) .25 (if there are no 16K or 32K spaces)</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>NBS_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number bytes space The default to be used if NGT Recover cannot find any data for the number of bytes in a space. Currently set to 82 megabytes (20,000 4K pages).</td>
</tr>
<tr>
<td>OOMF_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Output optimized merge factor Default fraction of pages written during a LOGONLY or BACKOUT run. Currently set to 0.05 (5%).</td>
</tr>
<tr>
<td>ONBA_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Output number bytes accum Default value for number of bytes in an output accum file (used if there is no input). Currently set to 100,000.</td>
</tr>
<tr>
<td>ONBAF_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Output accum bytes factor Used to estimate number of bytes in an output accum file. Currently set to 1.2 (120% of input number of bytes).</td>
</tr>
<tr>
<td>TC_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Table cardinality Default value for number of rows in a table (used if CARD and CARDF in SYSTABLES are set to -1). Currently set to 1 million.</td>
</tr>
<tr>
<td>RPK_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Default value for number of rows per key in a nonunique index Used to calculate the number of bytes in an index rebuild. Currently set to 5.</td>
</tr>
<tr>
<td>ATR_EST</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Default value for average truncation ratio (the ratio of the length of a truncated nonleaf key to the index key length) Used to calculate the number of bytes in an index rebuild. Currently set to 0.6.</td>
</tr>
</tbody>
</table>

**Recovery history: PHASE table**

The following table describes the contents of the PHASE table.

This table is only used by the RECOVERY MANAGER for DB2 solution.
Table 59: PHASE table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB_ID</td>
<td>INTEGER NOT NULL</td>
<td>Identifier for each job in the recovery run</td>
</tr>
<tr>
<td>STEPNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Name of the JCL job step in which the recovery utility executed</td>
</tr>
<tr>
<td>PHASENO</td>
<td>INTEGER NOT NULL</td>
<td>Unique identifier for each phase in a NGT Recover execution</td>
</tr>
<tr>
<td>UTILITY_RUN_ID</td>
<td>INTEGER NOT NULL</td>
<td>Identifier for the recovery run</td>
</tr>
<tr>
<td>PHASE_TYPE</td>
<td>CHAR(1) NOT NULL</td>
<td>Records components of recovery time that cannot be allocated to any one space</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Current possible values are</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1—analysis phase</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2—log input</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3—sequential data set recall</td>
</tr>
<tr>
<td>PHASE_RC</td>
<td>INTEGER NOT NULL</td>
<td>Return code for the phase</td>
</tr>
<tr>
<td>START_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>Start time of the phase</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Start and end times are not set during estimation (ANALYZE ONLY) runs.</td>
</tr>
<tr>
<td>END_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>End time of the phase</td>
</tr>
<tr>
<td>ELAP_SEC_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated elapsed time in seconds</td>
</tr>
<tr>
<td>CPU_SEC</td>
<td>REAL NOT NULL</td>
<td>CPU time in seconds used during the phase</td>
</tr>
<tr>
<td>WAIT_SEC</td>
<td>REAL NOT NULL</td>
<td>Currently unused (set to 0).</td>
</tr>
<tr>
<td>CAT_READ_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time spent reading the catalog during the phase (for analysis phase only)</td>
</tr>
<tr>
<td>SYSCOPY_READ_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time spent reading SYSCOPY during the phase (for analysis phase only)</td>
</tr>
<tr>
<td>SYSTP_READ_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time spent reading SYSTABLEPART during the phase (for analysis phase only)</td>
</tr>
<tr>
<td>SYSIP_READ_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time spent reading SYSINDEXPART during the phase (for analysis phase only)</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| SYNC_SEC            | INTEGER NOT NULL WITH DEFAULT | Elapsed time spent in job synchronization  
This time includes issuing DB2 commands and updating the BMCUTIL and BMCSYNC tables (for analysis phase only). |
| DDSCAN_SEC          | INTEGER NOT NULL WITH DEFAULT | Elapsed time spent in the DDSCAN process  
This is the time necessary to scan operating system control blocks and access operating system catalog(s) to validate the JCL and input commands (for analysis phase only). |
| SYSLGRNX_READ_SEC   | INTEGER NOT NULL WITH DEFAULT | Time spent reading SYSLGRNX during the phase (for analysis phase only)                                                                                                                                                                                                 |
| PLANPHS_SEC         | INTEGER NOT NULL WITH DEFAULT | Elapsed time spent in the PLANPHS process  
This is time NGT Recover spends building an execution plan (for analysis phase only). If this number is more than a few seconds, contact your BMC Customer Support representative. |
| PLANSHD_SEC         | INTEGER NOT NULL WITH DEFAULT | Elapsed time spent in the PLANSHD process  
This is time NGT Recover spends scheduling phases of the execution plan (for analysis phase only). If this number is more than a few seconds, contact your BMC Customer Support representative. |
| LOG_ALLOC_SEC_TP    | REAL NOT NULL WITH DEFAULT     | Number of seconds spent allocating log files on tape (for log input phase only)  
Currently unused (set to 0).                                                                 |
| LOG_ALLOC_SEC_DS    | REAL NOT NULL WITH DEFAULT     | Number of seconds spent allocating log files on disk (for log input phase only)  
Currently unused (set to 0).                                                                 |
| LOG_FILES_READ_TP   | INTEGER NOT NULL WITH DEFAULT  | Number of log files read from tape (for log input phase only)                                                                                                                                           |
| LOG_FILES_READ_DS   | INTEGER NOT NULL WITH DEFAULT  | Number of log files read from disk (for log input phase only)                                                                                                                                           |
| LOG_RECS_READ       | DOUBLE NOT NULL WITH DEFAULT   | Number of log records read during the phase (for log input phase only)                                                                                                                                  |
| LOG_BYTES_READ      | DOUBLE NOT NULL WITH DEFAULT   | Number of bytes of log read during the phase (for log input phase only)                                                                                                                                  |
### Recovery history: TS table

The following table describes the contents of the TS table.

Each row in this table summarizes all recovery activity for one table space and its indexes during a recovery. A table space could have more than one row, if partitions or indexes are recovered or rebuilt in more than one recovery job, or if a recovery job is restarted. Only the table spaces requiring the most recovery time and their indexes are represented in this table. This table is only used by the Recovery Management for DB2 solution.

#### Table 60: TS table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB_ID</td>
<td>INTEGER NOT NULL</td>
<td>Identifier for each job in the recovery run</td>
</tr>
<tr>
<td>DBNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Database in which the table space is located</td>
</tr>
<tr>
<td>TSNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Table space name</td>
</tr>
<tr>
<td>UTILITY_RUN_ID</td>
<td>INTEGER NOT NULL</td>
<td>Identifier for the recovery run</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>START_TIMESTAMP</td>
<td>TIMESTAMP NOT NULL</td>
<td>Time at which the first recover operation was started for this table space or one of its indexes</td>
</tr>
<tr>
<td>END_TIMESTAMP</td>
<td>TIMESTAMP NOT NULL</td>
<td>Time at which recovery of this table space and all its indexes was completed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This value can be the same as that of START_TIMESTAMP if recovery did not complete (INCOMPLETE_FLAG in the JOB table is set to 1).</td>
</tr>
<tr>
<td>OPERATION</td>
<td>CHAR(1) NOT NULL</td>
<td>The operation used to recover the table space and all its indexes, as follows: 1—MERGE recovery from image copy 2—MERGE recovery without image copy 3—MERGE recovery using LOGONLY 4—SNAPSHOT recovery 5—BACKOUT recovery 6—REBUILD indexes only 7—UNLOADKEYS/BUILDINDEX recovery 8—DSNUTILB execution</td>
</tr>
<tr>
<td>SUCCESS</td>
<td>CHAR(1) NOT NULL</td>
<td>Indicates whether recovery of the space was successful, as follows: 0—recovery failed 1—recovery succeeded</td>
</tr>
<tr>
<td>TOTAL_ELAP_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated elapsed time in seconds necessary to recover and/or rebuild the table space and all its indexes</td>
</tr>
<tr>
<td>RUNSTATS_RUN</td>
<td>CHAR(1) NOT NULL</td>
<td>Indicates whether RUNSTATS has been run, as follows: 0—some estimates for this table space or its indexes are inaccurate because RUNSTATS has not been run against one or more of the spaces 1—all estimates are based on catalog statistics updated by RUNSTATS</td>
</tr>
</tbody>
</table>
Recovery history: TS_PART table

The following table describes the contents of the TS_PART table.

This table contains one row for each partition recovered or one row for a nonpartitioned table space for each table space in one recovery job (represented by a row in the TS table). This table is only used by the BMC Recovery Management for DB2 solution.

Table 61: TS_PART table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB_ID</td>
<td>INTEGER NOT NULL</td>
<td>Identifier for each job in the recovery run</td>
</tr>
<tr>
<td>DBNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Database in which the table space is located</td>
</tr>
<tr>
<td>TSNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Table space name</td>
</tr>
<tr>
<td>DSNUM</td>
<td>SMALLINT NOT NULL</td>
<td>Partition number Zero indicates a nonpartitioned space.</td>
</tr>
<tr>
<td>START_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>Start of recovery of the partition</td>
</tr>
<tr>
<td>END_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>End of recovery of the partition. This value can be equal to that of START_TIMESTAMP if the recovery did not complete (INCOMPLETE_FLAG in the JOB table is set to 1).</td>
</tr>
<tr>
<td>OPERATION</td>
<td>CHAR(1) NOT NULL</td>
<td>The operation used to recover the partition and all its indexes, as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1—MERGE recovery from image copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2—MERGE recovery without image copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3—MERGE recovery using LOGONLY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4—SNAPSHOT recovery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5—BACKOUT recovery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6—REBUILD indexes only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7—UNLOADKEYS/BUILDINDEX recovery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8—DSNUTILB execution</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>NUM_BYTES_EST</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Estimated number of bytes to be recovered in this partition or table space</td>
</tr>
<tr>
<td>ELAP_SEC_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated number of seconds to recover this partition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The following estimate columns are details of this total.</td>
</tr>
<tr>
<td>MERGE_ELAP_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Time estimated to merge this partition</td>
</tr>
<tr>
<td>MERGE_ELAP_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time in seconds spent in merge for this partition</td>
</tr>
<tr>
<td>SNAP_ELAP_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated time to restore an Instant Snapshot copy for this partition</td>
</tr>
<tr>
<td>SNAP_ELAP_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time spent restoring an Instant Snapshot for this partition</td>
</tr>
<tr>
<td>UNLOAD_ELAP_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated time to unload keys from this partition</td>
</tr>
<tr>
<td>UNLOAD_ELAP_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time spent unloading keys from this partition</td>
</tr>
<tr>
<td>RESTORE_ELAP_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated time to restore an inline copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This row is recorded for the first partition, but it applies to the entire space.</td>
</tr>
<tr>
<td>RESTORE_ELAP_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time spent restoring an inline copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This row is recorded for the first partition, but it applies to the entire space.</td>
</tr>
<tr>
<td>MERGE_CPU_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds of CPU in the merge phase</td>
</tr>
<tr>
<td>SNAP_CPU_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds of CPU in the Instant Snapshot restore phase</td>
</tr>
<tr>
<td>UNLOAD_CPU_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds of CPU in the unload phase</td>
</tr>
<tr>
<td>RESTORE_CPU_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds of CPU in the inline copy restore</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This row is recorded for the first partition, but it applies to the entire space.</td>
</tr>
</tbody>
</table>

Appendix C RECOVERY MANAGER repository 791
<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
</table>
| OV_LOGRNG_START             | VARCHAR(10) FOR BIT DATA         | LRSN of the start of the overall range of log for this partition  
Typically, this is the START_RBA of the last image copy used.                                                                                                                                                    |
| OV_LOGRNG_END               | VARCHAR(10) FOR BIT DATA         | LRSN of the end of the overall range of log for this partition  
Typically, this is the TOLOGPOINT specified in the RECOVER command or the current LRSN at the time of recovery.                                                                                                         |
| OV_LOGRNG_BYTES             | DOUBLE NOT NULL WITH DEFAULT     | Total number of log bytes represented by all log ranges for the partition  
With LOG_BYTES_SORTED, this can be used to calculate log density for this partition. This value may be higher than expected if an unterminated log range exists for the partition.                                                                                       |
| LOG_RECS_SORTED             | DOUBLE NOT NULL WITH DEFAULT     | Number of log records selected for this partition and passed to the log sort                                                                                                                                                                                                 |
| LOG_RECS_PROCESSED          | DOUBLE NOT NULL WITH DEFAULT     | Log records sorted and processed by the merge process to be written to an accum file or to be considered for application to a page image  
If recovery terminates normally, this should be equal to LOG_RECS_SORTED.                                                                                                                                                                                                 |
| LOG_RECS_APPLIED            | DOUBLE NOT NULL WITH DEFAULT     | Log records actually applied to page images  
A log record may be included in the LOG_RECS_PROCESSED count and not in LOG_RECS_APPLIED because a comparison of its LRSN or RBA to the PGLOGRBA value in the corresponding page shows that the page already reflects this update. This difference can be significant if SHRLEVEL CHANGE image copies are used. |
<p>| LOG_BYTES_SORTED            | DOUBLE NOT NULL WITH DEFAULT     | Corresponds to LOG_RECS_SORTED, and with that value, enables calculation of the average log record size for this partition                                                                                                                                                     |</p>
<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUM_E35_LOG_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of times the merge process waited for a buffer of log records from the log sort</td>
</tr>
<tr>
<td>E35_WAIT_LOG_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds the merge process waited for log records from the log sort</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If this value is unacceptably high relative to the elapsed time to recover the partition, you should consider copying more often, or running R+/CHANGE ACCUM.</td>
</tr>
<tr>
<td>MAX_E35_LOG_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in one wait for the log sort</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This value is saved to show anomalies. A high value here is typical in the first merge in the job because it is probably waiting for the output phase of the log sort to finish. Otherwise, if one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>SPI_BYTES_READ</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of space input bytes read during recovery.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This is the number of bytes reads during BACKOUT and LOGONLY recoveries, and during merges following inline copy restores.</td>
</tr>
<tr>
<td>SPI_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Time in seconds required to allocate spaces for input</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This is the total time spent in dynamic allocation of spaces for BACKOUT and LOGONLY recoveries. It is usually not a concern because this time is incurred by a subtask.</td>
</tr>
<tr>
<td>NUM_SPI_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for reads from input spaces</td>
</tr>
<tr>
<td>SPI_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for reads from input spaces.</td>
</tr>
<tr>
<td>MAX_SPI_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in one wait for a read from an input space</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>FCI_BYTES_READ</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of bytes read from full image copies during recovery</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>FCI_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Total time in seconds required to allocate full image copies</td>
</tr>
<tr>
<td>NUM_FCI_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT,</td>
<td>Total number of waits for reads from full image copies</td>
</tr>
<tr>
<td>FCI_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for reads from full image copies.</td>
</tr>
<tr>
<td>MAX_FCI_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in one wait for a read from a full image copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>ICI_BYTES_READ</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of bytes read from incremental image copies during recovery</td>
</tr>
<tr>
<td>ICI_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Total time in seconds required to allocate incremental image copies</td>
</tr>
<tr>
<td>NUM_ICI_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for reads from incremental image copies</td>
</tr>
<tr>
<td>ICI_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for reads from incremental image copies</td>
</tr>
<tr>
<td>MAX_ICI_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a read from an incremental copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>CAI_BYTES_READ</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of bytes read from change accumulation files during recovery</td>
</tr>
<tr>
<td>CAI_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Total time in seconds required to allocate input change accumulation files</td>
</tr>
<tr>
<td>NUM_CAI_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for read from change accumulation files</td>
</tr>
<tr>
<td>CAI_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for reads from change accumulation files</td>
</tr>
<tr>
<td>MAX_CAI_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a read from a change accumulation file</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>--------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SPO_BYTES_WRITTEN</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of bytes written to table spaces and index spaces during recovery</td>
</tr>
<tr>
<td>SPO_SNAP_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds required to restore Instant Snapshot copies to the space output fields</td>
</tr>
<tr>
<td>SPO_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds required to allocate table space and index space data sets for output.</td>
</tr>
<tr>
<td>NUM_SPO_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for writes to table spaces and indexes</td>
</tr>
<tr>
<td>SPO_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for writes to table spaces and indexes</td>
</tr>
<tr>
<td>MAX_SPO_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a write to a table space or index</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>FCO_BYTES_WRITTEN</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of bytes written to full copy output files during recovery</td>
</tr>
<tr>
<td>FCO_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Amount of time in seconds required to allocate the full copy output files.</td>
</tr>
<tr>
<td>NUM_FCO_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for writes to full copy output files</td>
</tr>
<tr>
<td>FCO_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for writes to full copy output files</td>
</tr>
<tr>
<td>MAX_FCO_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a write to an output full image copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>CAO_BYTES_WRITTEN</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of bytes written to change accumulation output files during recovery</td>
</tr>
<tr>
<td>CAO_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Amount of time in seconds required to allocate the change accumulation output files</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Currently unused (set to 0).</td>
</tr>
<tr>
<td>NUM_CAO_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for writes to the change accumulation output files</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Currently unused (set to 0).</td>
</tr>
</tbody>
</table>
### Recovery history: KEYSORT table

The following table describes the contents of the KEYSORT table.

One row is entered in this table for each sort run during recovery. For example, if a sort is run for the keys in each of 10 partitions, and a sort is planned for the keys in each of 2 NPIs, this table would contain 12 rows. The data for key sorts for estimations differs from the data for simulations and actual recoveries.

- For simulations and actual recoveries, a key sort:
  - Begins at the start of the merge or unload phase of the first partition from which keys are to be unloaded
  - Terminates input after the last keys is extracted and before the first build starts
  - Ends at the end of the last build to which it passes keys

- For estimations, a key sort
  - Begins after the last merge or unload
  - Ends before the first build

This method separates the estimated sort time from merge/unload and build times.

This table is only used by the Recovery Management for DB2 solution.

#### Table 62: KEYSORT table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB_ID</td>
<td>INTEGER NOT NULL</td>
<td>Identifier for each job in the recovery run</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>DBNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Database in which the table space is located</td>
</tr>
<tr>
<td>TSNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Table space name</td>
</tr>
<tr>
<td>SORTID</td>
<td>INTEGER NOT NULL</td>
<td>unique identifier assigned to each keysort. SORTID enables you to link each keysort with the table space and index partitions that it runs against because</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- One sort may run against one or more table space partitions</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Several sorts may run against one table space partition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- One sort may run against several partitions or indexes</td>
</tr>
<tr>
<td>IXGROUP</td>
<td>SMALLINT NOT NULL</td>
<td>The index group number. This value is the same as the INDEX GROUP reported in NGT Recover messages. You can use it to relate data stored in the tables to NGT Recover messages.</td>
</tr>
<tr>
<td>START_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>Start of keysort</td>
</tr>
<tr>
<td>TERM_INPUT_TIME</td>
<td>TIMESTAMP</td>
<td>Time at which all data is finished being input and the output phase is started</td>
</tr>
<tr>
<td>END_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>End of keysort</td>
</tr>
<tr>
<td>REL_START_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Time in seconds at which this sort is estimated to start, relative to recovery of the table space and all its indexes. If unloads are to be executed in parallel, this represents the longest estimated time for a group of unloads which will pass keys to this sort.</td>
</tr>
<tr>
<td>ELAP_SEC_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated sort time, in seconds</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>TOTAL_ELAP_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated time, in seconds, between REL_START_EST and the REL_START_EST value of the earliest dependent build. If sorts are executing in parallel, the difference between this value and ELAP_SEC_EST represents time this sort will wait for other parallel sorts to complete.</td>
</tr>
</tbody>
</table>

**Recovery history: TSPSORT table**

The following table describes the contents of the TSPSORT table.

Each row in this table represents data associated with one key sort running against one table space partition. This table is only used by the BMC Recovery Management for DB2 solution.

**Table 63: TSPSORT table**

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB_ID</td>
<td>INTEGER NOT NULL</td>
<td>Identifier for each job in the recovery run</td>
</tr>
<tr>
<td>DBNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Database in which the table space is located</td>
</tr>
<tr>
<td>TSNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Table space name</td>
</tr>
<tr>
<td>DSNUM</td>
<td>SMALLINT NOT NULL</td>
<td>Partition number</td>
</tr>
<tr>
<td>SORTID</td>
<td>INTEGER NOT NULL</td>
<td>Unique identifier assigned to each keysort. SORTID enables you to link each keysort with the table space and index partitions that it runs against because</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ One sort may run against one or more table space partitions</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Several sorts may run against one table space partition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ One sort may run against several partitions or indexes</td>
</tr>
<tr>
<td>E15_BYTES_WRITTEN</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of bytes of key data passed to the E15 (input) sort exit for this sort and this partition</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>NUM_E15_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of times the merge or unload for this partition waited for this sort's E15 (input) exit to pass a buffer of keys to the sort</td>
</tr>
<tr>
<td>E15_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds during spent waiting for the E15 (input) key sort exit</td>
</tr>
<tr>
<td>MAX_E15_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds spent in any one input key sort wait If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
</tbody>
</table>

**Recovery history: IX_PART table**

The following table describes the contents of the IX_PART table.

This table is only used by the Recovery Management for DB2 solution.

**Table 64: IX_PART table**

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB_ID</td>
<td>INTEGER NOT NULL</td>
<td>Identifier for each job in the recovery run</td>
</tr>
<tr>
<td>DBNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Database in which the table space is located</td>
</tr>
<tr>
<td>TSNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Table space name</td>
</tr>
<tr>
<td>IXSPNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Index space name</td>
</tr>
<tr>
<td>PART</td>
<td>SMALLINT NOT NULL</td>
<td>Partition number of index (value of 0 if nonpartitioned)</td>
</tr>
<tr>
<td>START_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>Start of recovery of the index partition</td>
</tr>
<tr>
<td>END_TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>End of recovery of the index partition</td>
</tr>
</tbody>
</table>

This value can be equal to that of START_TIMESTAMP if the recovery did not complete (INCOMPLETE_FLAG in the JOB table is set to 1).
<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REL_START_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Time in seconds, relative to recovery of the table space and all its indexes, at which the build or writekeys phase for this index is estimated to start. If sorts are to be executed in parallel, this value represents the latest estimated time at which a sort passing data to this build or writekeys phase will end.</td>
</tr>
<tr>
<td>OPERATION</td>
<td>CHAR(1) NOT NULL</td>
<td>The operation used to recover the partition and all its indexes, as follows: 1—MERGE recovery from image copy 2—MERGE recovery without image copy 3—MERGE recovery using LOGONLY 4—SNAPSHOT recovery 5—BACKOUT recovery 6—REBUILD indexes only 7—UNLOADKEYS/BUILDINDEX recovery 8—DSNUTILB execution</td>
</tr>
<tr>
<td>SUCCESS</td>
<td>CHAR(1) NOT NULL</td>
<td>Indicates whether recovery of the partition was successful, as follows: 0—recovery failed 1—recovery succeeded</td>
</tr>
<tr>
<td>NUM_BYTES_EST</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Estimated number of bytes to be recovered or rebuilt</td>
</tr>
<tr>
<td>ELAP_SEC_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated time, in seconds, to rebuild or recover the index or partition</td>
</tr>
<tr>
<td>BUILD_ELAP_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated time to be spent in the build phase</td>
</tr>
<tr>
<td>BUILD_ELAP_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time, in seconds, spent in the build phase</td>
</tr>
<tr>
<td>BUILD_CPU_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds of CPU in the build phase</td>
</tr>
<tr>
<td>WRKEYS_ELAP_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated time to be spent in the writekeys phase</td>
</tr>
<tr>
<td>WRKEYS_ELAP_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time, in seconds, spent in the writekeys phase</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>WRKEYS_CPU_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds of CPU in the writekeys phase</td>
</tr>
<tr>
<td>MERGE_ELAP_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated time to be spent in the merge phase</td>
</tr>
<tr>
<td>MERGE_ELAP_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time spent in the merge phase</td>
</tr>
<tr>
<td>SNAP_ELAP_EST</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Estimated time to be spent in the Instant Snapshot restore phase</td>
</tr>
<tr>
<td>SNAP_ELAP_SEC</td>
<td>INTEGER NOT NULL WITH DEFAULT</td>
<td>Elapsed time spent in the Instant Snapshot restore phase</td>
</tr>
<tr>
<td>MERGE_CPU_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds of CPU spent in the merge phase</td>
</tr>
<tr>
<td>SNAP_CPU_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds of CPU spent in the instant snap restore phase</td>
</tr>
<tr>
<td>OV_LOGRNG_START</td>
<td>VARCHAR(10) FOR BIT DATA</td>
<td>LRSN of the start of the overall range of log for this partition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Typically, this is the START_RBA of the last image copy used.</td>
</tr>
<tr>
<td>OV_LOGRNG_END</td>
<td>VARCHAR(10) FOR BIT DATA</td>
<td>LRSN of the end of the overall range of log for this partition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Typically, this is the TOLOGPOINT specified in the RECOVER command or the current LRSN at the time of recovery.</td>
</tr>
<tr>
<td>OV_LOGRNG_BYTES</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of log bytes represented by all log ranges for the partition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With LOG_BYTES_SORTED, this can be used to calculate log density for this partition. This value may be higher than expected if an unterminated log range exists for the partition.</td>
</tr>
<tr>
<td>LOG_RECS_SORTED</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of log records selected for this partition and passed to the log sort</td>
</tr>
<tr>
<td>LOG_RECS_PROCESSED</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Log records sorted and processed by the merge process to be written to an accum file or to be considered for application to a page image</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If recovery terminates normally, this should be equal to LOG_RECS_SORTED.</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>LOG_RECS_APPLIED</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Log records actually applied to page images</td>
</tr>
<tr>
<td></td>
<td></td>
<td>A log record may be included in the LOG_RECS_PROCESSED count and not in LOG_RECS_APPLIED because a comparison of its LRSN or RBA to the PGLOGRBA value in the corresponding page shows that the page already reflects this update. This difference can be significant if SHRLEVEL CHANGE image copies are used.</td>
</tr>
<tr>
<td>LOG_BYTES_SORTED</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Corresponds to LOG_RECS_SORTED, and with that value, enables calculation of the average log record size for this partition</td>
</tr>
<tr>
<td>NUM_E35_LOG_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of times the merge process waited for a buffer of log records from the log sort</td>
</tr>
<tr>
<td>E35_WAIT_LOG_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds the merge process waited for log records from the log sort</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If this value is unacceptably high relative to the elapsed time to recover the partition, you should consider copying more often, or running R+/CHANGE ACCUM.</td>
</tr>
<tr>
<td>MAX_E35_LOG_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in one wait for the log sort</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This value is saved to show anomalies. A high value here is typical in the first merge in the job because it is probably waiting for the output phase of the log sort to finish. Otherwise, if one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>SPI_BYTES_READ</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of space input bytes read during recovery.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This is the number of bytes reads during BACKOUT and LOGONLY recoveries, and during merges following inline copy restores.</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SPI_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Time in seconds required to allocate spaces for input. This is the total time spent in dynamic allocation of spaces for BACKOUT and LOGONLY recoveries. It is usually not a concern because this time is incurred by a subtask.</td>
</tr>
<tr>
<td>NUM_SPI_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for reads from input spaces.</td>
</tr>
<tr>
<td>SPI_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for reads from input spaces.</td>
</tr>
<tr>
<td>MAX_SPI_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in one wait for a read from an input space. If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>FCI_BYTES_READ</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of bytes read from full image copies during recovery.</td>
</tr>
<tr>
<td>FCI_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Total time in seconds required to allocate full image copies.</td>
</tr>
<tr>
<td>NUM_FCI_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for reads from full image copies.</td>
</tr>
<tr>
<td>FCI_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for reads from full image copies.</td>
</tr>
<tr>
<td>MAX_FCI_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in one wait for a read from a full image copy. If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>CAI_BYTES_READ</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of bytes read from change accumulation files during recovery.</td>
</tr>
<tr>
<td>CAI_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Total time in seconds required to allocate input change accumulation files.</td>
</tr>
<tr>
<td>NUM_CAI_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for read from change accumulation files.</td>
</tr>
<tr>
<td>CAI_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for reads from change accumulation files.</td>
</tr>
<tr>
<td>MAX_CAI_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a read from a change accumulation file. If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SPO_BYTES_WRITTEN</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of bytes written to table spaces and index spaces during recovery</td>
</tr>
<tr>
<td>SPO_SNAP_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds required to restore Instant Snapshot copies to the space output fields</td>
</tr>
<tr>
<td>SPO_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds required to allocate table space and index space data sets for output.</td>
</tr>
<tr>
<td>NUM_SPO_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for writes to table spaces and indexes</td>
</tr>
<tr>
<td>SPO_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for writes to table spaces and indexes</td>
</tr>
<tr>
<td>MAX_SPO_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a write to a table space or index. If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>FCO_BYTES_WRITTEN</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of bytes written to full copy output files during recovery</td>
</tr>
<tr>
<td>FCO_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Amount of time in seconds required to allocate the full copy output files.</td>
</tr>
<tr>
<td>NUM_FCO_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for writes to full copy output files</td>
</tr>
<tr>
<td>FCO_WAIT_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Number of seconds spent waiting for writes to full copy output files</td>
</tr>
<tr>
<td>MAX_FCO_WAIT</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Maximum number of seconds in a wait for a write to an output full image copy. If one wait accounts for most of the wait time, it could be an anomaly.</td>
</tr>
<tr>
<td>CAO_BYTES_WRITTEN</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Number of bytes written to change accumulation output files during recovery</td>
</tr>
<tr>
<td>CAO_ALLOC_SEC</td>
<td>REAL NOT NULL WITH DEFAULT</td>
<td>Amount of time in seconds required to allocate the change accumulation output files. Currently unused (set to 0).</td>
</tr>
<tr>
<td>NUM_CAO_WAITS</td>
<td>DOUBLE NOT NULL WITH DEFAULT</td>
<td>Total number of waits for writes to the change accumulation output files. Currently unused (set to 0).</td>
</tr>
</tbody>
</table>
### Recovery history: IXPSORT table

The following table describes the contents of the IXPSORT table.

This table is only used by the BMC Recovery Management for DB2 solution.

#### Table 65: IXPSORT table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JOB_ID</td>
<td>INTEGER NOT NULL</td>
<td>Identifier for each job in the recovery run</td>
</tr>
<tr>
<td>DBNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Database in which the index space is located</td>
</tr>
<tr>
<td>TSNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Table space name</td>
</tr>
<tr>
<td>IXSPNAME</td>
<td>CHAR(8) NOT NULL</td>
<td>Index space name</td>
</tr>
<tr>
<td>PART</td>
<td>SMALLINT NOT NULL</td>
<td>Partition number</td>
</tr>
</tbody>
</table>
| SORTID              | INTEGER NOT NULL                 | Unique identifier assigned to each keysort SORTID enables you to link each keysort with the table space and index partitions that it runs against because
  - One sort may run against one or more table space partitions
  - Several sorts may run against one table space partition
  - One sort may run against several partitions or indexes
| E35_BYTES_WRITTEN   | DOUBLE NOT NULL WITH DEFAULT     | Number of bytes of key data received from the E35 (output) sort exit for this sort and this index or partition                                                                                      |
**Table 66: JOB_RESTART table**

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDENTIFIER</td>
<td>CHAR(45) NOT NULL</td>
<td>Unique set identifier in the following format: userIDDssid.DyymmddThhmmsss</td>
</tr>
<tr>
<td>JOBNAME</td>
<td>CHAR(8)</td>
<td>Name of the job</td>
</tr>
<tr>
<td>STEPNAME</td>
<td>CHAR(8)</td>
<td>Name of the job step</td>
</tr>
<tr>
<td>STATUS</td>
<td>CHAR(8) NOT NULL</td>
<td>Status of a set of executing jobs down to the step level</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Possible status values are:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• COMPLETE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• EXECUTE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NEW</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• RESTART</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• WAIT</td>
</tr>
</tbody>
</table>
### Column name | Data type | Description
--- | --- | ---
EVENT | CHAR(8) | Type of event
 |  | Possible events are:
 |  | ■ BUILDIX
 |  | ■ CHECKDAT
 |  | ■ CHECKLOB
 |  | ■ CHECKLPL
 |  | ■ COPYAFT
 |  | ■ DEL/DEF
 |  | ■ JOB
 |  | ■ INDEXALL
 |  | ■ REBUILDX
 |  | ■ RECOVER
 |  | ■ RECOVERX
 |  | ■ REPAIR
 |  | ■ STARTTS
 |  | ■ UNLOADKY
OBJECTSET | VARCHAR(27) | Object set (group) name
UTILID | CHAR(16) | Utility identifier
RC | INTEGER | Return code
START_TIME | TIMESTAMP | Starting time of the step/job/set
END_TIME | TIMESTAMP | Ending time of the step/job/set

### Job history: BMCARM.BMCRMD_RECOVERYExceptions table

The following table describes the contents of the BMCARM.BMCRMD_RECOVERYExceptions table.

This table is used only if you have the BMC Recovery for DB2 solution password.

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RUN_TIME</td>
<td>VARCHAR(27) NOT NULL</td>
<td>The run time</td>
</tr>
<tr>
<td>OSNAME</td>
<td>VARCHAR(27) NOT NULL</td>
<td>Name of the object set</td>
</tr>
<tr>
<td>RECOVERY_POINT</td>
<td>VARCHAR(60) NOT NULL</td>
<td>The end point of the latest archive log data set available at the recovery site</td>
</tr>
<tr>
<td>LOGMARK_NAME</td>
<td>VARCHAR(40)</td>
<td>Name of the log mark</td>
</tr>
</tbody>
</table>
### Data collection SQL statements

This section contains sample SQL statements that you can use to view the data that is collected during recoveries if you are using the Recovery Management for DB2 solution.

```sql
SET CURRENT SQLID = 'BMCARM';
SELECT UTILITY_RUN_ID, VERSION, SSID, DCTOKEN,
   START_TIMESTAMP, END_TIMESTAMP,
   RUN_TYPE,
   MIRRORING_USED,
   NUM_TS_TO_SAVE,
   ...,
   VOLSER,
   DSNNAME
FROM db2_rmg_collected_data;
```

### Table: Sample SQL Statements

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REC_POINT_VERSION</td>
<td>CHAR(8)</td>
<td>Version of the recovery point</td>
</tr>
</tbody>
</table>
| BACKOUT           | CHAR(4) NOT NULL | Values that correspond to the previous ARMBGPV syntax
|                   |               | The values can be YES, NO, or AUTO.                                       |
| EXCEPTION_TITLE   | VARCHAR(70) NOT NULL | The exception title                                                         |
| OBJ_TYPE          | CHAR(1)       | Set of DB2 objects with which you want to work                              |
| OBJ_QUAL1         | VARCHAR(128)  | Object qualifier 1                                                          |
| OBJ_QUAL2         | VARCHAR(128)  | Object qualifier 2                                                          |
| INSTANCE          | CHAR(1)       | (RECOVERY MANAGER and NGT Recover) Instance number of the current base objects (table and index)
|                   |               | The default value is 1. The other utilities do not use this column.         |
| PARTITION         | CHAR(4)       | Name of the partition                                                       |
| RESOURCE          | VARCHAR(80)   | Log file resources                                                          |
| DSNUM             | CHAR(4)       | Partition number                                                             |
|                   |               | Zero indicates a nonpartitioned space.                                      |
| DATE              | CHAR(8)       | Date that the utility completed                                              |
| TIME              | CHAR(8)       | Time that the utility completed                                              |
| VOLSER            | CHAR(6)       | Volume serial numbers of the data set
|                   |               | Commas separate items in a list of 6-byte numbers. This column is blank if the data set is cataloged. |
| DSNNAME           | VARCHAR(44)   | Name of the data set
|                   |               | For STYPE V, DSNNAME is the name of the VSAM data component.                 |
HEX(TOLOGPOINT),
ELAP_SEC_EST,
AVG_TS_SEC,
AVG_TS_SEC_EST,
NUM_JOBS,
NUM_TS_RECOV,
NUM_IX_RECOV,
DECIMAL(NUM_BYTES_RECOV)
FROM BMCARM_UTILITY_RUN WHERE DCTOKEN = 'DRECOVER'
ORDER BY RUN_TYPE, START_TIMESTAMP;
SELECT UTILITY_RUN_ID, JOB_ID, JOBNAME, START_TIMESTAMP,
END_TIMESTAMP, JOB_TYPE, ELAP_SEC_EST, NUM_TS_RECOV,
NUM_IX_RECOV, DECIMAL(NUM_BYTES_EST),
DECIMAL(NUM_BYTES_RECOV),
NUM_LOGS_RESTORED
FROM BMCARM_JOB
WHERE UTILITY_RUN_ID IN
(SELECT UTILITY_RUN_ID FROM BMCARM_UTILITY_RUN
WHERE DCTOKEN = 'DRECOVER')
ORDER BY UTILITY_RUN_ID, JOB_ID, ELAP_SEC_EST;
SELECT UTILITY_RUN_ID, JOB_ID, START_TIMESTAMP, END_TIMESTAMP,
DBNAME, TSNAME, OPERATION,
TOTAL_ELAP_EST
FROM BMCARM_TS
WHERE JOB_ID IN
(SELECT JOB_ID FROM BMCARM_JOB
WHERE UTILITY_RUN_ID IN
(SELECT UTILITY_RUN_ID FROM BMCARM_UTILITY_RUN
WHERE DCTOKEN = 'DRECOVER')
ORDER BY UTILITY_RUN_ID, JOB_ID, ELAP_SEC_EST;
ORDER BY UTILITY_RUN_ID, JOB_ID, TOTAL_ELAP_EST;
SELECT UTILITY_RUN_ID, JOB_ID, STEPNAME, PHASENO, PHASE_TYPE,
ELAP_SEC_EST
FROM BMCARM_PHASE
WHERE JOB_ID IN
(SELECT JOB_ID FROM BMCARM_JOB
WHERE UTILITY_RUN_ID IN
(SELECT UTILITY_RUN_ID FROM BMCARM_UTILITY_RUN
WHERE DCTOKEN = 'DRECOVER')
ORDER BY UTILITY_RUN_ID, JOB_ID, ELAP_SEC_EST;
ORDER BY UTILITY_RUN_ID, JOB_ID, TOTAL_ELAP_EST;
SELECT UTILITY_RUN_ID, JOB_ID, DBNAME, TSNAME, SORTID, REL_START_EST,
ELAP_SEC_EST, TOTAL_ELAP_EST
FROM BMCARM_KEYSORT
WHERE JOB_ID IN
(SELECT JOB_ID FROM BMCARM_JOB
WHERE UTILITY_RUN_ID IN
(SELECT UTILITY_RUN_ID FROM BMCARM_UTILITY_RUN
WHERE DCTOKEN = 'DRECOVER')
ORDER BY UTILITY_RUN_ID, JOB_ID, TOTAL_ELAP_EST;
SELECT UTILITY_RUN_ID, JOB_ID, DBNAME, TSNAME, DSNUM, OPERATION,
DECIMAL(NUM_BYTES_EST), ELAP_SEC_EST
FROM BMCARM_TS_PART
WHERE JOB_ID IN
(SELECT JOB_ID FROM BMCARM_JOB
WHERE UTILITY_RUN_ID IN
(SELECT UTILITY_RUN_ID FROM BMCARM_UTILITY_RUN
WHERE DCTOKEN = 'DRECOVER')
ORDER BY UTILITY_RUN_ID, JOB_ID, TOTAL_ELAP_EST;
SELECT JOB_ID, DBNAME, TSNAME, DSNUM, SORTID
FROM BMCARM_TSP_SORT
WHERE JOB_ID IN
(SELECT JOB_ID FROM BMCARM_JOB
WHERE UTILITY_RUN_ID IN
(SELECT UTILITY_RUN_ID FROM BMCARM_UTILITY_RUN
WHERE DCTOKEN = 'DRECOVER')
ORDER BY UTILITY_RUN_ID, JOB_ID, TOTAL_ELAP_EST;
SELECT JOB_ID, DBNAME, TSNAME, IXSPNAME, PART, REL_START_EST,
OPERATION, DECIMAL(NUM_BYTES_EST), ELAP_SEC_EST
FROM BMCARM_IXSP
WHERE JOB_ID IN
(SELECT JOB_ID FROM BMCARM_JOB
WHERE UTILITY_RUN_ID IN
(SELECT UTILITY_RUN_ID FROM BMCARM_UTILITY_RUN
WHERE DCTOKEN = 'DRECOVER')
ORDER BY UTILITY_RUN_ID, JOB_ID, TOTAL_ELAP_EST;
FROM BMCARM_IX_PART
WHERE JOB_ID IN
  (SELECT JOB_ID FROM BMCARM_JOB
   WHERE UTILITY_RUN_ID IN
     (SELECT UTILITY_RUN_ID FROM BMCARM_UTILITY_RUN
      WHERE DCTOKEN = 'DRECOVER'));

SELECT JOB_ID, DBNAME, TSNAME, IXSPNAME, PART, SORTID
FROM BMCARM_IXP_SORT
WHERE JOB_ID IN
  (SELECT JOB_ID FROM BMCARM_JOB
   WHERE UTILITY_RUN_ID IN
     (SELECT UTILITY_RUN_ID FROM BMCARM_UTILITY_RUN
      WHERE DCTOKEN = 'DRECOVER'));
Common utility tables

This chapter describes the contents of the common utility tables, considerations for these tables, and how to maintain them if necessary.

Overview of common utility tables

The BMC common utility tables contain information about the BMC utilities that you generate and submit through a BMC utility product.

Table 67 on page 811 lists the tables that each utility uses and each table’s default name and alias.

Note
CHECK PLUS, LOADPLUS, REORG PLUS, and UNLOAD PLUS (available only as version 11.2) use synonyms instead of aliases.

<table>
<thead>
<tr>
<th>Table</th>
<th>Default name</th>
<th>Alias</th>
<th>Utilities that use this table</th>
</tr>
</thead>
</table>
| BMCDICT  | CMN_BMCDICT      | BMC_BMCDICT    | ■ LOADPLUS  
|          |                  |                | ■ REORG PLUS  
| BMCHIST  | CMN_BMCHIST      | BMC_BMCHIST    | ■ CHECK PLUS  
|          |                  |                | ■ NGT Copy  
|          |                  |                | ■ LOADPLUS  
|          |                  |                | ■ NGT Recover  
|          |                  |                | ■ REORG PLUS  
|          |                  |                | ■ UNLOAD PLUS  

<table>
<thead>
<tr>
<th>Table</th>
<th>Default name</th>
<th>Alias</th>
<th>Utilities that use this table</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMCSYNC</td>
<td>CMN_BMCSYNC</td>
<td>BMC_BMCSYNC</td>
<td>• CHECK PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Check</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DASD MANAGER PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• LOADPLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Load</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Recover</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• RECOVERY MANAGER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• REORG PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Reorg</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• UNLOAD PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Unload</td>
</tr>
<tr>
<td>BMCTRANS</td>
<td>CMN_BMCTRANS</td>
<td>BMC_BMCTRANS</td>
<td>• Log Master</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• RECOVERY MANAGER</td>
</tr>
<tr>
<td>BMCUTIL</td>
<td>CMN_BMCUTIL</td>
<td>BMC_BMCUTIL</td>
<td>• CHECK PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Check</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DASD MANAGER PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• LOADPLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Load</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Recover</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• RECOVERY MANAGER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• REORG PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Reorg</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• UNLOAD PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Unload</td>
</tr>
<tr>
<td>BMCXCOPY</td>
<td>CMN_BMCXCOPY</td>
<td>BMC_BMCXCOPY</td>
<td>• NGT Copy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Log Master</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Recover</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• RECOVERY MANAGER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• REORG PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• UNLOAD PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• NGT Unload</td>
</tr>
</tbody>
</table>
Warnings and considerations for common utility tables

This topic describes important information that you need to know when using the common utility tables.

**WARNING**

The following warnings apply to the common utility tables:

- To prevent unpredictable results, do not run any of the following products against the BMC common utility tables or table spaces:
  - LOADPLUS
  - NGT Load
  - REORG PLUS
  - NGT Reorg
  - UNLOAD PLUS
  - NGT Unload

- Because NGT Recover uses BMC tables during the recovery process, you cannot use NGT Recover to recover any BMC table except the BMCHIST table.

- Do not run the RUNSTATS utility against the BMC common utility tables. Doing so can negatively impact utility performance.

- BMC strongly recommends that you use the ISOLATION (UR) bind option and issue SQL COMMIT statements when querying the tables in the BMC database. If objects in the BMC database are restricted for UPDATE, the executing BMC utilities might not be able to complete successfully.

Note the following considerations:

- Some columns in the tables are present for compatibility with specific BMC utilities and are not used by all of the utilities.

- If you have applications that depend on the structure or content of these tables, be aware that these tables are subject to change.

- In general, the utility tables should not require maintenance, with the exception of BMCHIST.

- You should back up the BMC table spaces on a regular basis to enable recoveries. If you use NGT Copy as the copy utility, you must use SHRLEVEL CHANGE for the following spaces:
  - BMCUTIL
  - BMCHIST
  - BMCSYNC
  - BMCXCOPY
Supported versions of the following BMC products support the LOCKROW installation option:

- CHECK PLUS
- LOADPLUS
- REORG PLUS
- UNLOAD PLUS

If the value of the option is YES (which is the default value), the products use MVS enqueues instead of SQL LOCK TABLE statements to serialize updates to the BMCSYNC and BMCUTIL tables.

The following BMC products always use MVS enqueues for serialization when updating the BMCSYNC and BMCUTIL tables:

- NGT Copy
- DASD MANAGER PLUS
- NGT Recover

Managing common utility tables

This topic provides basic procedures for working with the common utility tables.

To determine your site’s table names

The names of the common utility tables can be changed during installation.

1. To determine the names that your site uses, perform one of the following actions:

   - Use your utility to run a job with restart parameters of MAINT and MSGLEVEL(1).
     Specifying MSGLEVEL(1) with MAINT prints the names of the BMC tables that your utility uses and identifies the applied maintenance. The utility does not perform any other processing, and the job ends without affecting any utility that is running.

   - Run the following SQL statement, replacing tableName with a BMC common utility table name (listed in “Overview of common utility tables” on page 811):

     ```sql
     SELECT CREATOR,NAME FROM SYSIBM.SYSTABLES
     WHERE TSNAME='tableName';
     ```

   - Get the names from your DB2 system administrator.

To query the tables

1. Run SQL statements similar to the following examples.
**Example**

This example queries the BMCXCOPY table to access information about the rows in an index space:

```sql
SELECT * 
FROM creatorName.CMN_BMCXCOPY 
WHERE DBNAME = 'databaseName' 
AND IXNAME = 'indexSpaceName' 
ORDER BY START_RBA;
```

This example identifies (from the BMCHIST table) the database name, table space name, elapsed time, and when the utility completed:

```sql
SELECT DBNAME,SPNAME,CHAR(ELAPSED,ISO),CHAR(TIME,ISO) 
FROM creatorName.CMN_BMCHIST 
WHERE UTILID='utilityID';
```

**To display BMC utility status**

1. Use one of the following methods to display the status of BMC utilities:

   - To display the status of all BMC utilities that are executing or awaiting restart for a given table space or index space, use the following SQL statements:

     ```sql
     SELECT * FROM creatorName.CMN_BMCUTIL 
     WHERE DBNAME='databaseName' 
     AND SPNAME='tableSpaceName' 
     SELECT * FROM creatorName.CMN_BMCSYNC 
     WHERE NAME1='databaseName' 
     AND NAME2='spaceName';
     ```

   - If you have a license for the NGT Check, NGT Load, NGT Reorg, NGT Stats, or NGT Unload product, specify NGTDISP BMCUTIL or NGTDISP BMCSYNC to display information about the utilities that are executing or awaiting restart. You can optionally filter this information by utility ID.

     For more information, see the *BMC Next Generation Technology General User Guide*.

**To terminate a BMC utility**

1. To terminate a BMC utility, perform one of the following actions:

   - To terminate a BMC utility that is executing, use the following SQL statements:

     ```sql
     DELETE FROM creatorName.CMN_BMCUTIL 
     WHERE UTILID='utilityID';
     DELETE FROM creatorName.CMN_BMCSYNC 
     WHERE UTILID='utilityID';
     DELETE FROM creatorName.CMN_BMCDICT -- for LOADPLUS and REORG PLUS 
     WHERE UTILID='utilityID';
     ```

     The utility terminates with return code 8 when the next checkpoint is taken.

   - To clean up a BMC utility that is not executing, run the utility with the correct utility ID and specify the TERM restart parameter.
BMCDICT table

The BMCDICT table stores the compression dictionary during load or reorganization processing.

Table 68 on page 816 describes the contents of the BMCDICT table.

Table 68: Contents of the BMCDICT table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UTILID</td>
<td>CHAR(16)</td>
<td>Utility identifier</td>
</tr>
<tr>
<td>DBNAME</td>
<td>CHAR(8)</td>
<td>Database name</td>
</tr>
<tr>
<td>TSNAME</td>
<td>CHAR(8)</td>
<td>Table space name</td>
</tr>
<tr>
<td>PARTITION</td>
<td>SMALLINT</td>
<td>Partition number</td>
</tr>
<tr>
<td>SEQNO</td>
<td>SMALLINT</td>
<td>Sequence number</td>
</tr>
<tr>
<td>DICTDATA</td>
<td>VARCHAR(4000)</td>
<td>Dictionary data</td>
</tr>
</tbody>
</table>

BMCDICT table considerations

This topic describes important information that you need to know about the BMCDICT table:

- If you are processing a large number of compressed partitions, you might need to increase the size of the BMCDICT table space significantly from the standard size that was allocated during installation. To estimate the allocation, multiply 64 KB by the number of compressed partitions that you are processing concurrently (loading with LOADPLUS or reorganizing with REORG PLUS).

- LOADPLUS inserts rows into the BMCDICT table during the PRELOAD phase and deletes those rows following compression processing in the LOAD phase.

- REORG PLUS inserts rows into the BMCDICT table during the UNLOAD phase and deletes those rows following compression processing in the RELOAD phase.

Maintaining the BMCDICT table

If LOADPLUS or REORG PLUS abends during the time between building the compression dictionary and completing compression, rows might remain in the BMCDICT table.
On rare occasions, you might need to take action to control expansion of the BMCDICT table.

**To control expansion of the BMCDICT table**

1. Delete any rows in the BMCUTIL table that you know are no longer valid.
   
   Do not delete any rows for instances of utilities that are awaiting restart.

2. Use the following SQL statement to delete rows from the BMCDICT table:

   ```sql
   DELETE
   FROM creatorName.CMN_BMCDICT
   WHERE UTILID NOT IN
   (SELECT UTILID FROM creatorName.CMN_BMCUTIL);
   ```

   **Note**
   The names of the BMCUTIL and BMCDICT tables might have been changed at your site during installation.

---

**BMCHIST table**

The BMCHIST table contains information about completed executions of the BMC utilities for DB2.

The following configuration or installation options control use of the BMCHIST table:

- HISTORY (for NGT Copy, NGT Recover, and UNLOAD PLUS)
- BMCHIST (for REORG PLUS)

If the option value is NO, the utility bypasses any updates to the BMCHIST table. If the value is YES (or the utility does not use a configuration or installation option), the utility inserts rows into the BMCHIST table during the UTILTERM phase.

For NGT Copy, if the value is SUMMARY, the utility inserts only summary information about the NGT Copy execution into the BMCHIST table. This option provides less information than the YES option.

*Table 69 on page 817* describes the contents of the BMCHIST table.

Table 69: Contents of the BMCHIST table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBNAME</td>
<td>CHAR(8)</td>
<td>Name of the database that contains the table or index space</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>SPNAME</td>
<td>CHAR(8)</td>
<td>Name of the table or index space</td>
</tr>
</tbody>
</table>
| UTILNAME    | CHAR(8)        | Name of the utility:  
- CHECK  
- COPY  
- LOAD  
- RECOVER  
- REORG  
- UNLOAD |
| UTILID      | CHAR(16)       | Utility identifier |
| AUTHID      | CHAR(8)        | User ID that ran the utility |
| DATE        | DATE           | Date that the utility completed |
| TIME        | TIME           | Time that the utility completed |
| ELAPSED     | TIME           | Elapsed time of the utility |
| PARTITION   | LONG VARCHAR   | ALL, or the partition numbers as specified by the DSNUM option (NGT Copy) or the PART option  
Note the following conditions:  
- This column lists only three-digit partitions (any loaded partitions 1 through 999). Four-digit partitions (any loaded partitions from 1000 through 4096) are not stored in this column. For jobs that load only four-digit partitions, this column is empty.  
- If the list of partitions exceeds 1011 bytes, the utility truncates the value that is stored in this column.  
- For UNLOAD PLUS, if you specified LOGICAL PART, these partitions are the physical partitions that correspond to the logical partitions that you specified. |
| OBJNAME     | VARCHAR(27)    | Fully qualified object name |
| PHASE_1     | CHAR(8)        | Name of utility phase 1 |
| ELAPSED_1   | TIME           | Elapsed time of phase 1 |
| PHASE_2     | CHAR(8)        | Name of utility phase 2 |
| ELAPSED_2   | TIME           | Elapsed time of phase 2 |
| PHASE_3     | CHAR(8)        | Name of utility phase 3 |
| ELAPSED_3   | TIME           | Elapsed time of phase 3 |
| PHASE_4     | CHAR(8)        | Name of utility phase 4 |
BMCHIST table considerations for NGT Copy

NGT Copy uses the BMCHIST table to record completed COPY and COPY IMAGECOPY command executions.

HISTRETN is available as an NGT Copy installation option or as an option on the OPTIONS command. HISTRETN tells NGT Copy the number of days to keep entries in the BMCHIST table.

**WARNING**

If you want to use BMCHIST, allocate adequate space for the table. NGT Copy makes an entry in the table for every copied space. If you are copying a large number of partitions, you might need to increase the size of the BMCHIST table space from the standard size that was allocated during installation.

BMCHIST table considerations for NGT Recover

For each execution of AFRMAIN, NGT Recover writes a single row to the BMCHIST table.

DBNAME, SPNAME, and OBJNAME columns will always be blank.

NGT Recover accumulates elapsed time for each of the following phases using the NGT Recover phase shown:

- **PHASE_1**: LOGSORT
- **PHASE_2**: MERGE (includes RESTORE phase)
- **PHASE_3**: SNAP
- **PHASE_4**: REBUILD (includes UNLOAD phase)
- **PHASE_5**: DB2UTIL (the time spent in DSNUTILB)

The elapsed time for each of the phases is a sum for all objects. The utility elapsed time, ELAPSED, is the duration from the start of the utility to until it finishes. Because NGT Recover multitasks, the sum of the phases might be greater than the total elapsed time of the utility. The elapsed time columns have a limit of 24 hours.

---

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELAPSED_4</td>
<td>TIME</td>
<td>Elapsed time of phase 4</td>
</tr>
<tr>
<td>PHASE_5</td>
<td>CHAR(8)</td>
<td>Name of utility phase 5</td>
</tr>
<tr>
<td>ELAPSED_5</td>
<td>TIME</td>
<td>Elapsed time of phase 5</td>
</tr>
</tbody>
</table>
Maintaining the BMCHIST table

When a utility completes successfully, it inserts a row into the BMCHIST table. You can control expansion of this table by deleting old rows. If you use REORG PLUS, you can also control inserts into the BMCHIST table.

To delete old rows from the BMCHIST table

1. To delete selected rows from the BMCHIST table based on the date that the utility completed, use the following sample SQL statement:

   ```sql
   DELETE
   FROM creatorName.CMN_BMCHIST
   WHERE DATE < 'yyyy-mm-dd';
   ```

To control inserts into the BMCHIST table (REORG PLUS only)

1. Use the TERMEXIT option to specify a user exit that controls inserts into the BMCHIST table.

BMCSYNC table

The BMCSYNC table contains information about the status of the objects that the currently executing utilities are accessing.

Table 70 on page 821 describes the contents of the BMCSYNC table. The BMCSYNC table synchronizes and controls access to DB2 spaces by concurrently executing BMC utility products. If you have more than one BMC utility installed, all of these utilities should share the same BMCSYNC table.

The following NGT utilities insert rows into BMCSYNC during the BEFOREACC phase and delete rows during the AFTERACC phase:

- NGT Check
- NGT Load
- NGT Reorg
- NGT Stats
- NGT Unload

All other utilities insert rows into BMCSYNC during the UTILINIT phase and delete rows during the UTILTERM phase; while the job executes, the utilities update the table when the status of the object changes.
### Table 70: Contents of the BMCSYNC table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UTILID</td>
<td>CHAR(16)</td>
<td>Utility identifier. This column is blank when a RECOVER UNLOADKEYS command creates the row and then a RECOVER BUILDINDEX command reads and deletes the row.</td>
</tr>
<tr>
<td>NAME1</td>
<td>CHAR(8)</td>
<td>Database name or creator name. (DASD MANAGER PLUS) This value is the database name. (CHECK PLUS, LOADPLUS, REORG PLUS, and UNLOAD PLUS) If the value for NAME1 would exceed 8 bytes or the value for NAME2 would exceed 18 bytes, NAME1 contains the DBID for the object. (NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload) If the value for NAME1 would exceed 8 bytes, NAME1 contains the OBID in hexadecimal format.</td>
</tr>
<tr>
<td>NAME2</td>
<td>CHAR(18)</td>
<td>Space, table, or index name. (DASD MANAGER PLUS) The BMCSTATS utility always inserts the space name (limited to a maximum of 8 characters). (CHECK PLUS, LOADPLUS, REORG PLUS, and UNLOAD PLUS) If the value for NAME1 would exceed 8 bytes or the value for NAME2 would exceed 18 bytes, NAME2 contains the table OBID or index ISOBID of the object in hexadecimal format. (NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload) If the value for NAME2 would exceed 18 bytes, NAME2 contains the OBID in hexadecimal format.</td>
</tr>
<tr>
<td>KIND</td>
<td>CHAR(2)</td>
<td>Type of object:</td>
</tr>
<tr>
<td>PARTITION</td>
<td>SMALLINT</td>
<td>Physical partition number:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Null or 0 for a single data set nonpartitioned space</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Data set number for a multi-data-set, nonpartitioned space</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Partition number for a partitioned space</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(all products except NGT Recover and RECOVERY MANAGER) The value is null or 0 for any nonpartitioned space.</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>BMCID</td>
<td>SMALLINT</td>
<td>Internal identifier of the object. DASD MANAGER PLUS, NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload do not use this column.</td>
</tr>
<tr>
<td>UTILNAME</td>
<td>CHAR(8)</td>
<td>Name of the executing utility: CHECK, CHECKIX, COPY, STATS, LOAD, NGTSTATS, REBUILD, RECOVER, REORG, UNLOAD</td>
</tr>
<tr>
<td>SHRLEVEL</td>
<td>CHAR(1)</td>
<td>Degree to which utilities can share this object: Blank means that no status is requested, and any other utility can obtain any status. S allows sharing among any number of SHRLEVEL S utilities. X indicates that exclusive control is required. No other utility can run with SHRLEVEL X. For more information, see “Shared access levels of BMC utilities” on page 825.</td>
</tr>
<tr>
<td>STATUS</td>
<td>CHAR(1)</td>
<td>Status of the utility or object: Blank (indicates no processing has been done) C (for CHECK PLUS, indicates checked) L (for LOADPLUS, indicates loaded) U (for UNLOAD PLUS, indicates unloaded) R (for REORG PLUS, indicates reloaded) DASD MANAGER PLUS does not use this column.</td>
</tr>
<tr>
<td>XCOUNT</td>
<td>INTEGER</td>
<td>Number of rows or keys processed in the current phase. DASD MANAGER PLUS, NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload do not use this column.</td>
</tr>
<tr>
<td>DDNAME</td>
<td>CHAR(8)</td>
<td>Check, load, unload, or work ddname. DASD MANAGER PLUS, NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload do not use this column.</td>
</tr>
<tr>
<td>BLOCKS</td>
<td>INTEGER</td>
<td>Number of blocks for the check, load, unload, or work data set. DASD MANAGER PLUS, NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload do not use this column.</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>ORIG_STATUS</td>
<td>CHAR(8)</td>
<td>Encoded representation of the original DB2 status of the space (NGT Recover) This column restores the DB2 status of a space after recovery, if necessary. DASD MANAGER PLUS, NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload do not use this column.</td>
</tr>
<tr>
<td>EXTRBA</td>
<td>CHAR(10)</td>
<td>(NGT Recover) Log point at which this space was externalized. NGT Recover serialization logic uses this column. The other utilities do not use this column. Note: NGT Recover no longer uses EXTRBA.</td>
</tr>
<tr>
<td>STATE</td>
<td>LONG VARCHAR</td>
<td>Restart information for the space. For example, the STATE indicates the object state and sync information. DASD MANAGER PLUS, NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload do not use this column.</td>
</tr>
<tr>
<td>INSTANCE</td>
<td>SMALLINT</td>
<td>(RECOVERY MANAGER and NGT Recover) Instance number of the current base objects (table and index). The default value is 1. The other utilities do not use this column.</td>
</tr>
</tbody>
</table>

**BMCSYNC table considerations**

This topic contains important information that you need to know about the BMCSYNC table:

- By default, DASD MANAGER PLUS uses the BMCSYNC table to synchronize access to DB2 spaces. However, if you want to turn this feature off, you may do so by specifying No for the BMCSYNC installation option. If you specify No for this option, DASD MANAGER PLUS does not use the BMCSYNC table and the product bypasses BMCUTIL table access, UTILID enqueue logic, and object name enqueue logic used for BMC utility concurrency control. Turning this feature off can lead to VSAM data set access failures in BMCSTATS or other utilities due to utility conflicts that are no longer detected.

- You might need to increase the size of the BMCSYNC table space from the standard size that was allocated during installation when any of the following conditions exists:
  
  — You are processing a large number of partitions.
  
  Estimate this allocation based on the following factors:
  
  — Number of utilities that you are executing concurrently
  
  — Number of partitions that you are processing concurrently
  
  — Number of files that you are allocating dynamically
— You are loading a partition-by-growth table space.  
   Estimate this allocation based on the following factors:  
   — Number of utilities that you are running concurrently  
   — Value of MAXPARTITIONS  
   — Number of files that you are allocating dynamically  

— You are loading or unloading XML data and the XML table space is partition-
   by-growth.  
   Estimate this allocation based on the following factors:  
   — Number of utilities that you are executing concurrently  
   — Number of XML columns that you are loading or unloading  
   — Value of MAXPARTITIONS (a minimum of 256 partitions in this case)  
   — Number of files that you are allocating dynamically  

— You are loading or unloading LOB data.  
   Estimate this allocation based on the following factors:  
   — Number of utilities that you are executing concurrently  
   — Number of LOB columns that you are loading or unloading  
   — Number of partitions in the base table space  
   — Number of files that you are allocating dynamically  

Maintaining the BMCSYNC table

When a utility abends, rows might remain in the BMCSYNC table. On rare
occasions, you might need to take action to control expansion of the BMCSYNC
table.

To control expansion of the BMCSYNC table

1 Use one of the following methods to delete rows in the BMCSYNC table:

   ■ Use the TERM restart parameter on the EXEC statement to delete rows from
     both the BMCUTIL and BMCSYNC tables. Do not delete any rows for instances
     of utilities that are awaiting restart.

   ■ Delete invalid rows from the BMCUTIL table. Do not delete any rows for
     instances of utilities that are awaiting restart.

Then use the following SQL statement to delete rows from the BMCSYNC
table:

```sql
DELETE
FROM creatorName.CMN_BMCSYNC
WHERE UTILID NOT IN
  (SELECT UTILID FROM creatorName.CMN_BMCUTIL);
```
Note
The names of the BMCUTIL and BMCSYNC tables might have been changed at your site during installation.

Cleaning up RECOVER UNLOADKEYS entries

Successful completion of a RECOVER UNLOADKEYS job leaves rows in BMCSYNC with blank utility IDs for table space partitions and indexes related to the unloaded keys. The table space rows prevent other BMC utilities from obtaining exclusive control of the table space.

To clean up RECOVER UNLOADKEYS entries

1. Use one of the following methods to remove the invalid BMCSYNC rows:
   - Run a RECOVER BUILDINDEX job.
   - Run a job that uses the following statement for the table space and each index:

```
DELETE FROM creatorName.CMN_BMCSYNC
WHERE UTILID = ' '
AND NAME1 = 'databaseName'
AND NAME2 = 'spaceName'
AND UTILNAME = 'RECOVER';
```

Shared access levels of BMC utilities

BMC utility jobs register DB2 objects in the BMCSYNC table.

The registering utility assigns a sharing level to each registered object. The sharing level controls access to that object from other BMC utilities. For partitioned DB2 spaces, registration is performed at the partition level.

Note
All BMC utility products use the BMCUTIL table to control the use of utility IDs, which identify executions of BMC utilities. Each BMC utility product must have a unique ID for restart purposes. This unique ID is stored in the BMCUTIL table. For more information about this table, see “BMCUTIL table” on page 829.

The BMCSYNC table allows multiple BMC utilities (or multiple instances of a utility) to operate concurrently on different partitions of a DB2 space if no nonpartitioning indexes are involved. In addition, some BMC utilities can operate concurrently on the same object or partition. For information about which products can operate
concurrently, see the following table. For additional serialization and concurrency issues for each utility, see that utility's reference manual.

The "Access level" column in the following table refers to the value of the **SHRLEVEL** column name in the **BMCSYNC** table (“**BMCSYNC table**” on page 820). The level can be one of the following values:

- S indicates shared access. Any other utility that registers with shared access (S) can run against the object.
- X indicates exclusive access. No other utility can run against the object.
- A blank value indicates that no status is requested and any other utility can run against the object.

### Table 71: Shared access levels of BMC utilities

<table>
<thead>
<tr>
<th>Product</th>
<th>Access level</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHECK PLUS</td>
<td>S</td>
<td>None</td>
</tr>
</tbody>
</table>
| DASD MANAGER PLUS      | S            | - If BMCSTATS is processing multiple objects and encounters an object that is held by another utility, the BMCSTATS job issues a warning. The warning identifies the object and the utility that is using it. BMCSTATS continues processing the next object.  
- If BMCSTATS is processing an object and another utility requires exclusive control of that object, the other utility stops execution at initialization time. |
<p>| (BMCSTATS)             |              |                                                                                        |
| LOADPLUS               | X            | If you specify PART, LOADPLUS registers only the specified partitions with exclusive access (X). If no nonpartitioned indexes exist on the table space, you can run other utilities on different partitions while running this job. |
| NGT Check              | S            | None                                                                                   |
| NGT Copy               | S or blank   | If you specify COPY IMAGECOPY, NGT Copy registers the object with no access status (blank). Otherwise, NGT Copy registers the object with shared access (S). |
| NGT Load               | X            | If you are loading specific partitions, NGT Load registers only the specified partitions with exclusive access (X). |</p>
<table>
<thead>
<tr>
<th>Product</th>
<th>Access level</th>
<th>Additional information</th>
</tr>
</thead>
<tbody>
<tr>
<td>NGT Recover</td>
<td>X, S, or blank</td>
<td>NGT Recover registers an object with shared access (S) under the following conditions:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ The table space for an index is registered with shared access if the index is being rebuilt and its table space is not recovered in the same job.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ A table space partition is registered with shared access if the keys for that partition are unloaded with a RECOVER UNLOADKEYS operation.</td>
</tr>
<tr>
<td>NGT Recover</td>
<td></td>
<td>NGT Recover registers an object with no access status (blank) if you specify the following commands or options:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ The ACCUM command</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ OUTCOPY ONLY</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ INDEP OUTSPACE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NGT Recover registers the object with exclusive access (X) in all other cases.</td>
</tr>
<tr>
<td>NGT Reorg</td>
<td>X</td>
<td>If you are loading specific partitions, NGT Reorg registers only the specified partitions with exclusive access (X).</td>
</tr>
<tr>
<td>NGT Stats</td>
<td>S</td>
<td>None</td>
</tr>
<tr>
<td>NGT Unload</td>
<td>S</td>
<td>None</td>
</tr>
<tr>
<td>RECOVERY MANAGER</td>
<td>S</td>
<td>None</td>
</tr>
<tr>
<td>REORG PLUS</td>
<td>X</td>
<td>If you specify PART, REORG PLUS registers only the specified partitions with exclusive access (X). If no nonpartitioned indexes exist on the table space, you can run other utilities on different partitions while running this job.</td>
</tr>
<tr>
<td>UNLOAD PLUS</td>
<td>S</td>
<td>None</td>
</tr>
</tbody>
</table>

**WARNING**

Do not run an IBM utility, command, or SQL statement that attempts to manipulate the structure, data, or status of an object that a BMC utility is currently processing. For example, commands and SQL statements such as -STOP, -START, EXCHANGE, and ALTER will produce unpredictable results.
BMCTRANS table

The BMCTRANS table contains information that RECOVERY MANAGER and Log Master use for transaction recovery.

Table 72 on page 828 describes the contents of the BMCTRANS table. The table contains one row for each execution of Log Master (that is, one row for each log scan performed).

Table 72: Contents of the BMCTRANS table

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>USERID</td>
<td>CHAR(8) NOT NULL</td>
<td>Transaction creator</td>
</tr>
<tr>
<td>TRANID</td>
<td>VARCHAR(18) NOT NULL</td>
<td>Transaction ID</td>
</tr>
<tr>
<td>STARTTIME</td>
<td>TIMESTAMP NOT NULL WITH DEFAULT</td>
<td>Transaction start time</td>
</tr>
<tr>
<td>PITRBA</td>
<td>CHAR(6) NOT NULL FORBIT DATA</td>
<td>RBA for point-in-time recovery</td>
</tr>
<tr>
<td>OUTDSNAME</td>
<td>VARCHAR(35) NOT NULL</td>
<td>Output data set prefix for SQL statements or the logical log</td>
</tr>
<tr>
<td>STATE</td>
<td>SMALLINT NOT NULL</td>
<td>Level of recovery analysis performed:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 0 (only UNDO analysis has been performed)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 1 through 9999 (UNDO and PIT analysis have been performed)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Greater than 10000 (UNDO, PIT, and REDO analysis have been performed)</td>
</tr>
<tr>
<td>PITTIME</td>
<td>TIMESTAMP NOT NULL WITH DEFAULT</td>
<td>Timestamp for the PIT RBA</td>
</tr>
<tr>
<td>SEQNO</td>
<td>SMALLINT NOT NULL</td>
<td>Sequence number of the filter text</td>
</tr>
<tr>
<td>PITWKEST</td>
<td>FLOAT NOT NULL</td>
<td>Work estimate</td>
</tr>
<tr>
<td>FILTERLINE</td>
<td>VARCHAR(1040) NOT NULL</td>
<td>Text of the filter (may span more than one row)</td>
</tr>
<tr>
<td>UNDONUMROWSUPD</td>
<td>FLOAT</td>
<td>Number of unique rows (RIDs) that are selected by the filter of the log scan</td>
</tr>
<tr>
<td>UNDOSEQUPDPDROWS</td>
<td>FLOAT</td>
<td>Total number of anomaly log records relating to one of the rows (RIDs) selected by the log scan</td>
</tr>
<tr>
<td>UNDOLOGRECROWS</td>
<td>FLOAT</td>
<td>Number of unique rows (RIDs) that are affected by an anomaly log record</td>
</tr>
<tr>
<td>Column Name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UNDOJOBSTATUS</td>
<td>SMALLINT</td>
<td>Code indicating the status of an UNDO log scan:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 0 (no action taken)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 1 (Log Master execution started)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 2 (Log Master execution completed successfully with return code 0,4)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 3 (Log Master execution completed unsuccessfully with return code 8,12)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 4 (Log Master execution abnormally ended)</td>
</tr>
<tr>
<td>REDOJOBSTATUS</td>
<td>SMALLINT</td>
<td>Code indicating the status of a REDO log scan:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 0 (no action taken)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 1 (Log Master execution started)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 2 (Log Master execution completed successfully with return code 0,4)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 3 (Log Master execution completed unsuccessfully with return code 8,12)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ 4 (Log Master execution abnormally ended)</td>
</tr>
<tr>
<td>ENDTIME</td>
<td>TIMESTAMP NOT NULL WITH DEFAULT</td>
<td>Transaction end time</td>
</tr>
<tr>
<td>ACTION</td>
<td>SMALLINT</td>
<td>Code indicating what recovery, if any, has been performed on the transaction</td>
</tr>
</tbody>
</table>

**BMCUTIL table**

The BMCUTIL table contains information about utilities that are currently running or started.

*Table 73 on page 830* describes the contents of the BMCUTIL table. The utilities use the table to control the use of utility IDs. Each BMC utility must have a unique ID for
restart purposes. If you have more than one BMC utility installed, all of these utilities should share the same BMCUTIL table.

The utilities insert rows into the BMCUTIL table during the UTILINIT phase and update the table as the job status changes. The utilities delete rows from the BMCUTIL table during the UTILTERM phase.

Table 73: Contents of the BMCUTIL table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UTILID</td>
<td>CHAR(16)</td>
<td>Utility identifier</td>
</tr>
</tbody>
</table>
| STATUS      | CHAR(1)   | Execution status of the utility:  
  - A (active, not executing command)  
  - I (initializing)  
  - P (pausing or pause-stopped)  
  - S (stopped)  
  - T (terminating)  
  - X (executing command) |
| UTILNAME    | CHAR(8)   | Name of the executing utility:  
  - CHECK  
  - COPY  
  - STATS  
  - LOAD  
  - NGT Load  
  - RECOVER  
  - REORG  
  - UNLOAD |
| PHASE       | CHAR(8)   | Current phase of the utility  
  NGT Copy does not use this column.  
  (NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload) The value for this column is always UTILINIT. |
<p>| USERID      | CHAR(8)   | User ID executing the utility |
| SSID        | CHAR(4)   | DB2 subsystem where the utility is running |</p>
<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
</table>
| RESTART       | CHAR(1)    | Restart option:  
  ■ N (not restart)  
  ■ P (RESTART(PHASE))  
  ■ Y (RESTART)  
  DASD MANAGER PLUS does not use this column.  
  *(NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload)* The value for this column is always N. |
| NOTEID        | CHAR(8)    | TSO user ID to be notified  
  DASD MANAGER PLUS, NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload do not use this column. |
| DBNAME        | CHAR(8)    | *(NGT Recover and REORG PLUS)* Name of the database containing the table or index space for which the last checkpoint was taken  
  This value can be blank.  
  The other utilities do not use this column. |
| SPNAME        | CHAR(8)    | *(NGT Recover and REORG PLUS)* Name of the table or index space for which the last checkpoint was taken  
  This value can be blank.  
  The other utilities do not use this column. |
| SPSTATUS      | CHAR(5)    | *(REORG PLUS)* Space status before the utility stopped  
  The other utilities do not use this column. |
| COMMANDNO     | SMALLINT   | *(NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload)* Always 1  
  For all other utilities, this column is always 0. |
| COMMAND       | VARCHAR(256) | First 256 characters of the utility command text  
  NGT Recover, DASD MANAGER PLUS, NGT Copy, NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload do not use this column. |
| STATE         | LONG VARCHAR | Utility state and sync information  
  DASD MANAGER PLUS, NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload do not use this column. |
| START_TIMESTAMP | TIMESTAMP | Starting timestamp of the utility  
  NGT Check, NGT Load, NGT Reorg, NGT Stats, and NGT Unload do not use this column. |
Maintaining the BMCUTIL table

When a utility abends, rows might remain in the BMCUTIL table.

On rare occasions, you might need to take action to control expansion of the BMCUTIL table.

To control expansion of the BMCUTIL table

1. Use one of the following methods to delete rows from the BMCUTIL table:

   ■ Use the TERM restart parameter on the EXEC statement to delete rows from both the BMCUTIL and BMCSYNC tables. Do not delete any rows for instances of utilities that are awaiting restart.

   ■ Delete invalid rows in the BMCUTIL table. Do not delete any rows for instances of utilities that are awaiting restart.

Then use the following SQL statement to delete rows from the BMCSYNC table:

```sql
DELETE FROM creatorName.CMN_BMCSYNC
WHERE UTILID NOT IN
  (SELECT UTILID FROM creatorName.CMN_BMCUTIL);
```

Note
The names of the BMCUTIL and BMCSYNC tables might have been changed at your site during installation.

BMCXCOPY table

The BMC utilities use the BMCXCOPY table to track registered copies.

Table 74 on page 833 describes the contents of the BMCXCOPY table, which contains information about the following types of registered copies:

■ Indexes that NGT Copy has copied:
  — COPY NO index copies
  — DSNUM n index (nonpartitioned) copies
  — Incremental index copies
  — Index copies that are made at data set level

■ Instant Snapshots made by NGT Copy that are not registered as Flash Copies in SYSCOPY with the BMC EXTENDED BUFFER MANAGER (XBM) product or
BMC SNAPSHOT UPGRADE FEATURE (SUF) technology, and any standard copies made in association with the Instant Snapshot

- Online consistent copies
- Cabinet copies
- Encrypted copies

The BMCXCOPY table functions like SYSIBM.SYSCOPY except that IXNAME replaces TSNAME in BMCXCOPY. You must control authorization and access to users for BMCXCOPY through standard DB2 authorization.

If you have more than one BMC utility installed, all of these utilities should share the same BMCXCOPY table.

### Table 74: Contents of the BMCXCOPY table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBNAME</td>
<td>CHAR(8)</td>
<td>Name of the database</td>
</tr>
<tr>
<td>IXNAME</td>
<td>CHAR(8)</td>
<td>Name of the index space or table space for Instant Snapshots and associated copies</td>
</tr>
<tr>
<td>DSNUM</td>
<td>INTEGER</td>
<td>Data set number within the index or table space</td>
</tr>
<tr>
<td>ICTYPE</td>
<td>CHAR(1)</td>
<td>Operation type:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• F (COPY FULL YES; for NGT Copy, online consistent copies)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• I (COPY FULL NO)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• W (REORG LOG NO)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• B (REBUILD INDEX)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• P (POINT-IN-TIME RECOVERY)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• C (for NGT Copy version 7.3 and earlier, online consistent copies)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• s (used by NGT Copy to track system pages)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• m (indicates that the table space was exported by the NGT Copy EXPORT command or migrated by the NGT Recover IMPORT command)</td>
</tr>
<tr>
<td>ICDATE</td>
<td>CHAR(6)</td>
<td>Date of the entry (yymmd)</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>START_RBA</td>
<td>VARCHAR(10)</td>
<td>The relative byte location of a point in the DB2 recovery log&lt;br&gt;The indicated point as follows:&lt;br&gt;■ For ICTYPE F, the starting point for all updates since the image copy was taken&lt;br&gt;■ For COPY_TYPE O, the minimum of the consistent point and the oldest inflight URID&lt;br&gt;■ (RECOVERY MANAGER) For ICTYPE C, the consistent log point for the copy — RBA for non-data-sharing systems — LRSN for data sharing systems</td>
</tr>
<tr>
<td>FILESEQNO</td>
<td>INTEGER</td>
<td>Tape file sequence number of the copy</td>
</tr>
<tr>
<td>DEVTYPE</td>
<td>CHAR(8)</td>
<td>Type of device on which the copy resides</td>
</tr>
<tr>
<td>IBMREQD</td>
<td>CHAR(1)</td>
<td>Whether the row came from the basic machine-readable material (MRM) tape:&lt;br&gt;■ N (NO)&lt;br&gt;■ Y (YES)</td>
</tr>
<tr>
<td>DSNAME</td>
<td>CHAR(44)</td>
<td>Name of the data set&lt;br&gt;If STYPE V, DSNAME is the name of the VSAM data component</td>
</tr>
<tr>
<td>ICTIME</td>
<td>CHAR(6)</td>
<td>Time at which this row was inserted (hh:mm:ss)&lt;br&gt;The insertion takes place after the completion of the operation that the row represents.</td>
</tr>
<tr>
<td>SHRLEVEL</td>
<td>CHAR(1)</td>
<td>SHRLEVEL parameter on COPY if ICTYPE F:&lt;br&gt;■ C (change)&lt;br&gt;■ R (reference)</td>
</tr>
<tr>
<td>DSVOLSER</td>
<td>VARCHAR(1784)</td>
<td>Volume serial numbers of the data set&lt;br&gt;Commas separate items in a list of 6-byte numbers. This column is blank if the data set is cataloged.</td>
</tr>
<tr>
<td>TIMESTAMP</td>
<td>TIMESTAMP</td>
<td>Date and time when the row was inserted&lt;br&gt;This column contains the date and time that are recorded in ICDATE and ICTIME. The use of TIMESTAMP over ICDATE and ICTIME is recommended, because later DB2 releases might not support the latter two columns.</td>
</tr>
<tr>
<td>Column name</td>
<td>Data type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| ICBACKUP         | CHAR(2)   | Type of image copy contained in the data set:  
|                  |           | ■ LB (data set contains local backup data)  
|                  |           | ■ RP (data set contains recovery system main data)  
|                  |           | ■ RB (data set contains recovery system backup data)  
|                  |           | ■ Blank (data set contains local system main data or is not one of multiple copies)                                                       |
| ICUNIT           | CHAR(1)   | Media on which the image copy data set is stored:  
|                  |           | ■ D (DASD)  
|                  |           | ■ T (tape)  
|                  |           | ■ Blank (medium is neither tape nor DASD)                                                   |
| STYPE            | CHAR(1)   | Type of copy:  
|                  |           | ■ Blank (for ICTYPE=F)  
|                  |           | ■ V (Instant Snapshot or a VSAM data set)  
|                  |           | ■ e (encrypted copy)                                                                       |
| PIT_RBA          | VARCHAR(10) | Point-in-time recovery:  
|                  |           | ■ X'000000000000' (for ICTYPE=F)  
|                  |           | ■ Consistent point (for COPY_TYPE=O)                                                        |
| GROUP_MEMBER     | CHAR(8)   | Data-sharing group member (the name of the SSID where the copy was made)  
|                  |           | This column is blank if you are not using data sharing.                                      |
| OTYPE            | CHAR(1)   | Type of object:  
|                  |           | ■ T (table)  
|                  |           | ■ I (index)  
|                  |           | ■ i (compressed index)                                                                       |
| LOWDSNUM         | INTEGER   | Not used                                                                                                                                  |
| HIGHDSNUM        | INTEGER   | Not used                                                                                                                                  |
| COPYPAGESF       | FLOAT(53) | Number of pages written to the copy data set                                                                                             |
| NPAGESF          | FLOAT(53) | High-used RBA divided by the page size                                                                                                     |
| CPAGESF          | FLOAT(53) | Total number of changed pages                                                                                                             |
| JOBNAME          | CHAR(8)   | Job name                                                                                                                                  |
| AUTHID           | CHAR(8)   | Authorization ID                                                                                                                           |
| OLDEST_VERSION   | SMALLINT  | When ICTYPE= B, F, I, S, W, or X, the version number of the oldest format of data for an object  
<p>|                  |           | For other values of ICTYPE, the value is -1.                                                                                             |
| LOGICAL_PART     | INTEGER   | Logical partition number                                                                                                                 |</p>
<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
</table>
| LOGGED       | CHAR(1)   | Logging attribute of the table space:  
- Y (logged)  
- N (not logged)  
- Blank (row inserted prior to DB2 version 9)  
  For a non-LOB table space or index space, blank indicates that the logging attribute is logged. |
| TTYPE        | CHAR(8)   | Row format for the table space or partition:  
- RRF (reordered row format)  
- BRF (basic row format)  
RBA/LRSN format for the space or partition:  
- B (basic 6-byte format)  
- E (extended 10-byte format) |
| INSTANCE     | SMALLINT  | Instance number of the current base objects (table and index)  
The default value is 1. |
| RELCREATED   | CHAR(1)   | DB2 release that created the object  
If the release is earlier than Version 9, the value is blank. |
| COPY_TYPE    | CHAR(1)   | Type of copy:  
- C (cabinet copy)  
- O (online consistent copy)  
- X (export copy)  
- I (import copy)  
- Blank (default value) |
| NOTE_VALUE   | CHAR(4)   | Encoded value that quickly locates data for a specific space in a cabinet copy  
The default value is blank. |
| NOTE_TYPE    | CHAR(1)   | Type of NOTE (issued by NGT Copy):  
- A (ABS - tape)  
- R (REL - disk)  
- F (frame)  
- Blank (default value) |
| OCC_COPY_RBA | VARCHAR(10) | Original START_RBA of an online consistent copy  
The default value is blank. |
| OCC_LOCKRULE | CHAR(1)   | Locking rule for a table space (not used for indexes):  
- A (for page level)  
- R (for row level)  
- Blank (default value) |
### BMCXCOPY table

<table>
<thead>
<tr>
<th>Column name</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OCC_SPACE_ALTERED</td>
<td>CHAR(1)</td>
<td>Whether the space was altered:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Y (altered)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ N (not altered)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Blank (default value)</td>
</tr>
<tr>
<td>CAB_BLOCKS</td>
<td>INTEGER</td>
<td>Total number of frames written for a cabinet copy</td>
</tr>
<tr>
<td>EXPSSID</td>
<td>VARCHAR(8)</td>
<td>Source location SSID of the migration file (valid with COPY_TYPE = I)</td>
</tr>
<tr>
<td>EXPSLRSN</td>
<td>VARCHAR(10)</td>
<td>Indicates the SYNC AUTO point on the source (valid with COPY_TYPE = I and</td>
</tr>
<tr>
<td></td>
<td></td>
<td>COPY_TYPE = X)</td>
</tr>
<tr>
<td>EXPTLRSN</td>
<td>VARCHAR(10)</td>
<td>Indicates the SYNC AUTO point on the target (valid with COPY_TYPE = I)</td>
</tr>
</tbody>
</table>

## Maintaining the BMCXCOPY table

Periodically, you should review BMCXCOPY and delete old rows to control its expansion.

**To control expansion of the BMCXCOPY table**

1. To delete all rows from the BMCXCOPY table that are older than 30 days, run an SQL DELETE statement, using the following statement as an example:

   ```sql
   DELETE
   FROM creatorName.CMN_BMCXCOPY
   WHERE DAYS(CURRENT_TIMESTAMP) - DAYS(TIMESTAMP) > 30;
   ```
BMCXCOPY table
RECOVERY MANAGER object exception status

This appendix describes the RECOVERY MANAGER object exception status.

RECOVERY MANAGER object exception status

The following table lists the various types of object status that can be returned by RECOVERY MANAGER and provides corrective actions you can take.

An object must have a status of OK for RECOVERY MANAGER to include it in recovery JCL.

For backup JCL, objects with certain non-OK statuses are included in addition to those with an OK status. These are also indicated in the following table.

Table 75: Object exception status

<table>
<thead>
<tr>
<th>Status</th>
<th>Included in Backup JCL</th>
<th>Included in Recovery JCL</th>
<th>Explanation</th>
<th>User Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER IX</td>
<td>Yes</td>
<td>No</td>
<td>The partitioned index has been altered and requires that all affected partitions be included in the recovery.</td>
<td>Include all affected partitions in the group. Use action code L and then D to view the SYSCOPY detail.</td>
</tr>
<tr>
<td>BADDSNUM</td>
<td>No</td>
<td>No</td>
<td>According to the DB2 catalog, the space is a nonpartitioned space and the data set number that is provided is not in the range of 1 to 32.</td>
<td>Return to the Object List Generation panel and correct the data set number.</td>
</tr>
<tr>
<td>BAD PART</td>
<td>No</td>
<td>No</td>
<td>According to the DB2 catalog, the object is partitioned and the partition number is not valid.</td>
<td>Return to the Object List Generation panel and correct the partition number.</td>
</tr>
<tr>
<td>Status</td>
<td>Included in Backup JCL</td>
<td>Included in Recovery JCL</td>
<td>Explanation</td>
<td>User Response</td>
</tr>
<tr>
<td>-------------</td>
<td>------------------------</td>
<td>--------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>BADSHRL</td>
<td>No</td>
<td>No</td>
<td>A SHRLEVEL CHANGE copy cannot be used to recover a NOT LOGGED object that is not a LOB. The object is not included in the recovery</td>
<td>You can select an alternate recovery point for the object (action code L) or select Recover again and choose a new recovery point for the entire group. It is possible that there is no valid recovery point for the object.</td>
</tr>
<tr>
<td>BAD TYPE</td>
<td>Yes</td>
<td>No</td>
<td>The object is not recoverable to the selected recovery point. An entry in SYSIBM.SYSCOPY indicates that an event that prevents recovery occurred between the recovery point and the prior full image copy. Events precluding a recovery are REORG LOG NO, LOAD LOG NO, LOAD REPLACE LOG NO, or a table being altered to rotate partitions.</td>
<td>You can select an alternate recovery point for the object (action code L) or select Recover again and choose a new recovery point for the entire group. It is possible that there is no valid recovery point for the object.</td>
</tr>
<tr>
<td>DEFER</td>
<td>No</td>
<td>No</td>
<td>The object was created with DEFINE NO. The underlying data set does not exist.</td>
<td>No action is required.</td>
</tr>
<tr>
<td>DS LEVEL</td>
<td>Yes</td>
<td>No</td>
<td>Copies are needed for recovery by data set, but the object is not expanded by partition and either DSNUTILB is specified or NGT Recover and a partial recovery is specified.</td>
<td>Expand the object into its component partitions in the Object List Generation or List Generation Options panel, as appropriate to your method of object list generation.</td>
</tr>
<tr>
<td>INFLIGHT</td>
<td>Yes</td>
<td>No</td>
<td>This status applies to DB2 V8 and higher. It indicates the object cannot be recovered because the selected recovery utility is DSNUTILB (inflight recovery is only possible when NGT Recover is the recovery utility)</td>
<td>You must manually recover the object.</td>
</tr>
</tbody>
</table>
## RECOVERY MANAGER object exception status

<table>
<thead>
<tr>
<th>Status</th>
<th>Included in Backup JCL</th>
<th>Included in Recovery JCL</th>
<th>Explanation</th>
<th>User Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVRECPT</td>
<td>No</td>
<td>Yes</td>
<td>The recovery point is invalid for the specified NOT LOGGED object. A NOT LOGGED object requires recovery to a copy, to an ALTER NOT LOGGED point, or to a CREATE NOT LOGGED point.</td>
<td>You can select an alternate recovery point for the object (action code L) or select Recover again and choose a new recovery point for the entire group. It is possible that there is no valid recovery point for the object.</td>
</tr>
<tr>
<td>NOCOPIES</td>
<td>Yes</td>
<td>No</td>
<td>No image copies were found in SYSIBM.SYSCOPY for the object, and there were no LOAD REPLACE LOG (YES), LOAD LOG (YES), or REORG LOG (YES) copies. This status applies only to table spaces.</td>
<td>Perform a separate recovery on such objects; select Recover and specify Log Only. This task requires that the space has already been restored outside of DB2 and RECOVERY MANAGER (for example, a volume restore).</td>
</tr>
<tr>
<td>NOTAVAIL</td>
<td>Yes</td>
<td>No</td>
<td>A recovery to a specified copy or quiesce point was requested but the copy or quiesce point could not be found. The specified copy or quiesce recovery point could not be found within the specified SYSCOPY limit.</td>
<td>Select an alternate recovery point for the object (action code L) or select Recover again and choose a new group recovery point. There may be no valid recovery point for the object.</td>
</tr>
<tr>
<td>NOTCLONED</td>
<td>Yes</td>
<td>Yes</td>
<td>A backup or recovery for clones only was requested. The specified object is not a clone and will not be included in the operation.</td>
<td>No action is required.</td>
</tr>
<tr>
<td>NOTDEFND</td>
<td>No</td>
<td>No</td>
<td>The index, table space, or partition was not found in the DB2 catalog.</td>
<td>Return to the Object List Generation panel and correct the entry.</td>
</tr>
<tr>
<td>OK</td>
<td>Yes</td>
<td>Yes</td>
<td>Object status is satisfactory.</td>
<td>No action is required.</td>
</tr>
<tr>
<td>PEND DDL</td>
<td>Yes</td>
<td>No</td>
<td>An entry in SYSIBM.SYSPENDINGDDL prevents the object from being recovered.</td>
<td>No action is required.</td>
</tr>
<tr>
<td>Status</td>
<td>Included in Backup JCL</td>
<td>Included in Recovery JCL</td>
<td>Explanation</td>
<td>User Response</td>
</tr>
<tr>
<td>----------</td>
<td>------------------------</td>
<td>--------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SYS OBJ</td>
<td>No</td>
<td>No</td>
<td>The object is a table space or index from the DB2 catalog and directory or from a temporary database.</td>
<td>Back up or recover the object by using System Resources on the Main Menu.</td>
</tr>
<tr>
<td>TBLPART</td>
<td>No</td>
<td>No</td>
<td>This status applies to DB2 V8 and higher. It indicates that the table space was defined with table-based partitioning. RECOVERY MANAGER does not back up or recover such table spaces or any of the associated indexes when using RECOVER PLUS version 5.1 or earlier.</td>
<td>You must manually back up and recover tables defined with table-based partitioning and their associated indexes.</td>
</tr>
<tr>
<td>TEMPDB</td>
<td>No</td>
<td>No</td>
<td>The object is a table space or index from a declared temporary database.</td>
<td>No action is required, although an object from a temporary database cannot be copied or recovered. BMC recommends that you remove these objects from any RECOVERY MANAGER groups to increase product efficiency.</td>
</tr>
<tr>
<td>TSREORP</td>
<td>No</td>
<td>No</td>
<td>Applies only to indexes. Following a PIT recovery, the table space on which the index is based is placed in REORG PENDING status, and the index cannot be recovered or rebuilt.</td>
<td>REORG the table space and its indexes or select an alternate recovery point.</td>
</tr>
<tr>
<td>TS STAT</td>
<td>No</td>
<td>No</td>
<td>Applies only to indexes. The table space on which the index is based has a status other than OK.</td>
<td>Correct the table space problem indicated by the table space status if possible, or proceed to generate JCL without those objects.</td>
</tr>
<tr>
<td>UNCHANGE</td>
<td>No</td>
<td>No</td>
<td>After a partial verification, this denotes that the table space has had no updates and will be excluded from the backup or recovery JCL.</td>
<td>No action is required.</td>
</tr>
<tr>
<td>Status</td>
<td>Included in Backup JCL</td>
<td>Included in Recovery JCL</td>
<td>Explanation</td>
<td>User Response</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------</td>
<td>--------------------------</td>
<td>-------------</td>
<td>---------------</td>
</tr>
<tr>
<td>VARBIN</td>
<td>Yes</td>
<td>No</td>
<td>Applies only to indexes. DSNUTILB does not support Rebuild Index if the index has a VARBINARY column.</td>
<td>No action is required.</td>
</tr>
</tbody>
</table>
Obtaining trace and maintenance information

When you contact customer support with a problem, you might be asked to provide a trace so that the events leading up to the problem can be analyzed. Also, trace entries from stacked tape analysis and multi-job are written to the ARMTRACE file. This appendix describes how to obtain trace and maintenance information.

Obtaining a trace for a batch job

Use one of the following methods to obtain a trace for a batch job:

- To send the trace to a user-defined data set, add the following DD statement to the batch job:

  ```
  //ARMTRACE DD DSN=dsName,DISP=(,CATLG)
  // UNIT=SYSDA,SPACE=(CYL,(10,10))
  ```

  **Note**
  The DCB characteristics are RECFM=FB,LRECL=80

- To send the trace to SYSOUT, add the following DD statement to the batch job:

  ```
  //ARMTRACE DD SYSOUT=*  
  ```

  **WARNING**
  The trace produces a large amount of output. For that reason, you may want to consider sending the output to a data set.

Obtaining a trace for online functions

This topic provides the steps needed to obtain a trace for online functions.
To obtain a trace for online functions, perform the following steps:

1. On the Main Menu, type **TRACE** on the command line.

2. Select the option that caused the problem.
   
   At the top right corner of the next panel, a short message is displayed to tell you that the trace data set was allocated.

3. Press the **F1** key to see the full data set name that was allocated.
   
   All data from the current RECOVERY MANAGER session is then written to that data set, which is closed when you exit RECOVERY MANAGER.

   **Note**
   
   If you do not exit RECOVERY MANAGER after you have recreated the problem, records will continue to be written to the trace data set. You must exit RECOVERY MANAGER to turn the trace off.

---

**Determining applied maintenance**

Use one of the following methods to determine what maintenance zaps and fixes have been applied to RECOVERY MANAGER:

- From the RECOVERY MANAGER Main Menu, select About to display the About panel. To display RECOVERY MANAGER and solution common code (SCC) fixes, type 1 at the prompt and press Enter.

- Run a trace (see “Obtaining a trace for a batch job” on page 845) to view the fix history that is included at the beginning of the ARMTRACE file.

- The fixes that have been applied are printed after the report heading of the ARMPRINT file in most batch jobs.
Copy and recover utility options

You can specify copy and recover options in both the ARMBGRP and ARMBGEN batch programs. Syntax for these programs is described in their respective chapters. This appendix provides descriptions of all of the copy and recover options available in batch mode.

General recovery options

The general recovery options enable you to specify the utilities to be used during a recovery. Most apply to all supported recovery utilities.

RECOVER_UTILITY

Choose either the NGT Recover utility or IBM’s DB2 RECOVER (DSNUTILB) utility for recovering the current group or object. Specify the recover utility to use for recovery, as follows:
- AFRMAIN—NGT Recover
- DSNUTILB—DSNUTILB RECOVER

CHECK_UTILITY

Choose either BMC’s CHECK PLUS utility or IBM’s DB2 CHECK (DSNUTILB) utility as the utility to be used for performing integrity checks on the current group or object. Specify the check utility as follows:
- ACKMAIN—CHECK PLUS
- DSNUTILB—DSNUTILB CHECK

COPY_UTILITY

Choose either the NGT Copy utility or IBM’s DB2 COPY (DSNUTILB) utility as the utility to be used for making post-recovery image copies immediately after the current group or object has been recovered. Specify the copy utility as follows:
- ACPMAIN—NGT Copy
- DSNUTILB—DSNUTILB COPY
REGION_SIZE

Specify the amount of virtual storage used by the recover utility. The valid range is -1 through 2047 MB.

The default value is 0 MB, in which case the amount of virtual storage needed to run the job is automatically made available when the recover utility runs. Some data centers do not allow a region size of 0 MB.

A value of -1 specifies that RECOVERY MANAGER will not generate region size at the step level. RECOVERY MANAGER JCL generation recognizes the -1 value and does not generate REGION=.

Note
For best performance, BMC recommends a region size of 0 MB.
A typical NGT Recover step requires between 5 MB and 8 MB of virtual storage for code, control blocks, and I/O buffers.

CHECK_PEND_ACTION

Specify the action that you want to take to correct check pending status on recovered spaces after a point-in-time recovery. You can specify one of the following commands:

- CHECK—run the specified check utility to correct check pending status
- REPAIR—run the REPAIR utility to turn off check pending status
- NONE—take no action
- RESET—causes the option to default to the value set at the subsystem level. If no subsystem value exists, the option defaults to the product level.

Table spaces that have parents or hash tables will be selected to have CHECK DATA after they are recovered and when the group option Check Pend Action is CHECK.

WARNING
Do not run CHECK DATA on encrypted data. Because CHECK DATA does not decrypt the data, the utility might produce unpredictable results.

REDEFINE_VCAT_OBJ

Include IDCAMS delete and define steps for VCAT spaces before the recover utility executes. To execute recovery JCL that includes this step, you must have DB2 STOP and DISPLAY authority and control authority on the physical data sets.
Be aware of the following items:

- If you specify SITETYPE RECOVERY, the product forces the option to YES.

- RECOVERY MANAGER cannot perform delete and define steps for VCAT-defined spaces that do not have ICF catalog data (for example, objects that have been deleted or migrated). For those objects, RECOVERY MANAGER performs one of the following actions:
  
  — For user-defined VCAT objects, ARMBGEN issues a warning message and generates JCL with the delete and define steps commented out. You can manually retrieve the object, then alter the JCL to include the delete and define statements.
  
  — For system VCAT objects (such as the DB2 catalog and directory, the repository, and the CHANGE ACCUM repository), ARMBSRR issues an error message and fails.

**COPY_AFTER_copyType**

Make a primary image copy for the local site after the group or object has been recovered.

The variable `copyType` is one of the following values:

- LP—local site primary
- LB—local site backup
- RP—recovery site primary
- RB—recovery site backup

You must specify a primary copy in order to specify a backup copy. For example, if you want a recovery site backup copy, you must specify a recovery site primary copy.

**DELETE_STOGROUP_OBJ**

Delete STOGROUP spaces before the recover utility executes. To execute recovery JCL that includes this step, you must have DB2 STOP and DISPLAY authority and control authority on the physical data sets. Specifying `DELETE_STOGROUP_OBJ YES` causes the data set to be deleted if the object is STOGROUP-defined, regardless of the setting of the REUSE parameter.

**REUSE**

Reuse target spaces. Specify one of the following options:

- NO—delete and/or redefine the target spaces
- YES—reuse those spaces without deleting/redefining them
- NOSCR—(NOSCRACTH) to avoid running IDCAMS
**MAX_CONCURRENT_JOBS**

Specify the maximum number of concurrent jobs that you want RECOVERY MANAGER to use when processing the current group. The range is 0-99.

*Note*
If you are using SIMULATE YES, you cannot set a value greater than 1.

*WARNING*
Do not specify a value that is higher than the number of initiators available at your site.

**ALWAYS_REBUILD_INDEXES**

Choose to either rebuild indexes from table data or to recover them from image copies and log data when possible.

- **YES**—Rebuilds all indexes from table data, even if an image copy and log data are available.

- **NO**—(The default) Attempts to recover indexes from image copies and log data when possible. Any index that cannot be recovered is automatically rebuilt.
  
  For BACKOUT recoveries, RECOVERY MANAGER changes this option to NO and issues a warning message.

**INDEX_ALL**

Recover all indexes for the table spaces in the group by using INDEX ALL syntax. This option is intended for applications having a large number of indexes (for example, ERP applications such as SAP/R3).

When you select the INDEX ALL option, the ARMBGEN program does not search for related indexes for objects in an group during JCL generation because their inclusion is implied.

*WARNING*

- Do not use this option for groups that explicitly include indexes or for groups created by partition.

- If you use INDEX ALL, you must verify that the primary and secondary allocations in the work file options of the group are large enough to accommodate the group.
**MIRROR**

Specify whether the objects in the group are mirrored. Mirrored groups are excluded from the primary disaster recovery JCL that you generate, although they can optionally be included in alternate JCL. This feature is only available with the RECOVERY MANAGER for DB2 solution and you must activate at least level 2 mirroring at the subsystem level. For more information, see the Recovery Management for DB2 User Guide.

---

**WARNING**

RECOVERY MANAGER checks the mirroring status only for objects explicitly included in the group. If you are mirroring both table spaces and indexes, you must explicitly include the indexes as well as the table spaces in the group. Do not use the Use INDEX ALL recover option as a means of including the indexes in the group. Doing so causes the indexes to be rebuilt during recovery whether they are mirrored or not.

---

**BLKALLOC**

Use this option to convert cylinder or track specifications to kilobytes when you redefine VCAT objects.

**LIMIT_SYSCOPY_SEARCH**

Limit the time range for which RECOVERY MANAGER searches the SYSIBM.SYSCOPY table for a requested copy or quiesce point. Type 0 to search all rows in the SYSCOPY catalog table. To limit the search (for performance reasons), type a value from 1 through 99 to indicate the number of days of SYSCOPY entries to include in the search.

---

**Tip**

Wherever possible, always limit SYSIBM.SYSCOPY searches to avoid unneeded I/O operations and excessive memory use.

---

**NGT Recover recover options**

The following options are valid when using the NGT Recover product as the recovery utility.

**ALTERNATE_ACT1**

Rank the active log copy 1 in the order that you would like it to be used for recovery, as follows:

- 1—log copy that you want as first choice
- 2—second choice (if any)
- 3—third choice (if any)
4—fourth choice (if any)
0—do not want to use the copy at all

ALTERNATE_ACT2

Rank the active log copy 2 in the order that you would like it to be used for recovery, as follows:

■ 1—log copy that you want as first choice
■ 2—second choice (if any)
■ 3—third choice (if any)
■ 4—fourth choice (if any)
■ 0—do not want to use the copy at all

ALTERNATE_ARC1

Rank the archive log copy 1 in the order that you would like it to be used for recovery, as follows:

■ 1—log copy that you want as first choice
■ 2—second choice (if any)
■ 3—third choice (if any)
■ 4—fourth choice (if any)
■ 0—do not want to use the copy at all

ALTERNATE_ARC2

Rank the archive log copy 2 in the order that you would like it to be used for recovery, as follows:

■ 1—log copy that you want as first choice
■ 2—second choice (if any)
■ 3—third choice (if any)
■ 4—fourth choice (if any)
■ 0—do not want to use the copy at all

ALTERNATE_CHANGE_ACCUM_copyType

Rank the change accumulation copy in the order that you would like it to be used for recovery, as follows:

■ 1—change accumulation copy that you want as first choice
■ 2—second choice (if any)
■ 3—third choice (if any)
■ 4—fourth choice (if any)
■ 0—do not want to use the copy at all
ALTERNATE_COPY_copyType

Rank the local primary copy in the order that you would like it to be used when performing a recovery, as follows:
- 1—image copy that you want as first choice
- 2—second choice (if any)
- 3—third choice (if any)
- 4—fourth choice (if any)
- 5—fifth choice (if any)
- 6—sixth choice (if any)
- 0—do not want to use the copy at all

The variable copyType is one of the following values:
- FC—DSNUTILB FlashCopy image copy (for DB2 Version 10 and later)
- LP—local site primary
- LB—local site backup
- RP—recovery site primary
- RB—recovery site backup
- SB—system backup

ALTERNATE_RESOURCES

Indicate a preferred order in the selection of image copies, logs, and change accumulation files. Choose whether to specify the recovery resources to be used in the recovery and the order in which they should be used. You can specify particular image copies, log copies, or copies of R+/CHANGE ACCUM groups. For DB2 Version 10 and later, you can specify DSNUTILB FlashCopy image copies.

For example, if your practices include taking a local site backup image copy and a secondary copy of the archive log to a recovery site (instead of offsite copies), you can select those resources when you use this procedure at the recovery site.

- YES—Always use the alternate recovery resources as specified in the group options.
- NO—Do not use the alternate recovery resources specified in the group options.
- AUTO—Use the alternate recovery resources specified in the group options, unless those resources are unavailable. If unavailable, use the default recovery resources.

If you select the RP or RB copy as your first choice for the image copy, RECOVERY MANAGER considers the site type to be RECOVER. This value
overrides any other site type setting you make, including the site type specified in ARMBGEN and ARMBGPV syntax.

CHECKPOINT

This option provides a means of controlling the overhead that is associated with taking checkpoints. The default is the configuration option value.

- NO—causes no checkpoints to be taken, except those necessary to synchronize NGT Recover execution with the execution of other BMC utilities and the MERGE checkpoints that are necessary to guarantee the integrity of output copy registration. This option is recommended for short NGT Recover jobs in which you do not want to incur checkpoint overhead and which you do not mind rerunning if necessary.

- SYNC—causes a checkpoint to be taken at the end of each processing phase and also at the completion of each log data set in the LOG APPLY phase. This allows either phase restart or sync restart in the LOG APPLY phase. Specify this option for recoveries that require the reading of many log data sets using the RESTORE/LOGAPPLY, LOGAPPLY ONLY, or LOGONLY strategies.

- PHASE—causes a checkpoint to be taken at the end of each processing phase if a set amount of time has passed. Choose this option for longer jobs when it would be costly to rerun the entire job.

DIAGNOSTIC_MESSAGES

Use this option if you want NGT Recover to provide diagnostic messages regarding the sort functions it performs.

DYNAMIC_SORTWORKS

Specify the type of sort work allocation you want to use. This option is only valid when NGT Recover is the recovery utility.

- YES causes NGT Recover to dynamically allocate sort works.

- NO causes the sort works to be allocated via DD statements.

EARLYRECALL

Specifies the early retrieval (during the ANALYZE phase) of any archived image copies and log data sets that are required during recovery.

EARLYCAT

This option causes NGT Recover to verify (during the ANALYZE phase) that all cataloged data sets that are required for recovery exist in the operating system catalog.
KSORTSHARE

Specify whether key sorts are shared among the execution queues.

- **YES**—the keys sorts are shared and value specified in MAXKSORT determines the maximum total number of active key sorts at any given time.
- **NO**—the key sorts are not shared among execution queues. The number of active key sorts at any given time could be MAXLSORT multiplied by MAXKSORT.

KSORTSHARE is available when you use NGT Recover as the recover utility.

The total number of key sorts in your system are affected by the KSORTSHARE, MAXLSORT, and MAXKSORT options. For more information, see the *BMC Next Generation Technology Recover for DB2 for z/OS Reference Manual*.

LOGSCAN

Scans the log and provides a report on the number and size of log records required for recovery. Specify **YES** to scan the log. Specify **NO** not to do so.

Be aware of the following information:

- This option is only valid when NGT Recover is the recovery utility.
- LOGSCAN **YES** and the disaster recovery simulation feature are mutually exclusive. Groups defined with LOGSCAN **YES** are excluded from simulation.
- A recovery to a copy overrides the LOGSCAN option and creates a normal recover job.
- If you specify **YES**, you cannot perform a BACKOUT AUTO recovery for the group. RECOVERY MANAGER changes BACKOUT AUTO to BACKOUT NO and issues a warning message.

MAXKSORT

Specify the maximum number of index key sorts that can be run concurrently. Valid values are from 1 to 999.

For each table space, index keys for all indexes being rebuilt are distributed over the number of sorts that you specify for this option and these sorts can then run in parallel. For a partitioned table space, if the partitioning index is being rebuilt, the rebuild of each partition is done at the completion of the MERGE or UNLOAD for each partition of the table space. (The rebuild can
run concurrently with the MERGE or UNLOAD for the next partition if the MAXKSORT number is not exceeded). Running concurrent index key sorts can increase the speed of the recovery. MAXKSORT is available when you use NGT Recover as the recover utility.

MAXKSORT overrides any value that you specify for WORKFILE_WORKDDN. It also cannot be used in conjunction with UNLOADKEYS/BUILDINDEX. For more information about maximizing the concurrency of key sorts, see “Maximizing concurrency of key sorts” on page 230.

The total number of key sorts in your system are affected by the KSORTSHARE, MAXLSORT, and MAXKSORT options. For more information, see the BMC Next Generation Technology Recover for DB2 for z/OS Reference Manual.

MAXLSORT

Specify the maximum number of log sorts and the maximum number of objects that can be recovered in parallel using subtasks.

Valid values are from 1 to 999. If you specify 1, the product creates only one log sort and the recovery of objects is performed serially in the main task. The configuration option default value is two times the number of CPUs in the system or 12, whichever is lower. MAXLSORT is available when you use NGT Recover as the recover utility.

The total number of key sorts in your system are affected by the KSORTSHARE, MAXLSORT, and MAXKSORT options. For more information, see the BMC Next Generation Technology Recover for DB2 for z/OS Reference Manual.

MAXLOGS

Use this option to specify the maximum number of log files that RECOVERY MANAGER allocates concurrently during a log input phase. Zero (0) indicates no limit.

This option:
- Controls the amount of memory used during the recovery
- Reduces the contention caused by reading many log files in parallel
- Controls the number of tape drives used for the log files

MSGLEVEL

This option specifies which output files and messages NGT Recover returns. Valid values for MSGLEVEL are STANDARD, OBJECT_SUMMARY, and PLAN_SUMMARY. See the BMC Next Generation Technology Recover for
Use this option to determine how NGT Recover is to proceed when errors are encountered. The default value is 10. The valid range is 0 to 2,147,483,646.

ON_ERROR_CONTINUE nnnnnnnnnnn allows nnnnnnnnnnn + 1 errors before NGT Recover terminates. If nnnnnnnnnnn is 0, NGT Recover stops processing immediately when the first recognized severe error occurs. If you specify ON_ERROR_CONTINUE 0, the subtask to preallocate VSAM data sets is disabled, which could increase the execution time by several seconds for each object recovered.
Note
If you are using the Recovery Management solution and specify BACKOUT AUTO, the number of errors allowed is not limited.

OPTIMIZE_FOR

If you are using NGT Recover and BMCSORT, use this option to control the relative importance of the system resources that are consumed by the sort when you use BMCSORT.

- I/O—minimize the I/O activities that a sort performs.
- CPU—minimize the central processing unit (CPU) time of each sort at the expense of sort elapsed time and I/O activity.
- BALANCE—provide the best overall balance between CPU time, elapsed time, and I/O activity.
- ELAPSED—minimize the elapsed time (wall clock time) for each sort.

OUTCOPY_AUTOSIZE

This option is the AUTOSIZE option of NGT Recover and turns dynamic sizing for output image copies or change accumulation output files on or off. Valid values are YES are NO. If you specify a value for AUTOSIZE, you override the AUTOSIZE installation option, which defaults to YES.

- YES—specifies dynamic sizing for output image copies or change accumulation output files allocated to DASD.
- NO—specifies that output image copies or change accumulation output files are allocated to DASD using the primary and secondary quantities that are specified in the R+/CHANGE ACCUM repository.

OUTCOPY_BY_RECOVER

Specify how you want the output copies to be made for partitioned table spaces after a successful recovery. The copy choices are as follows:

- ASCODED—tells RECOVERY MANAGER to use the OUTCOPY option of NGT Recover to make copies with the same DSNUM designation as is used for the recovery. For example, if DSNUM ALL is used for recovery, the copies are made by table space. If DSNUM n is used for recovery (n>0), the copy is for partition n.
- BYPART—tells RECOVERY MANAGER to use the OUTCOPY option of NGT Recover to make all copies of partitioned table spaces by partition whether or not the recovery is by table space (DSNUM ALL).
- **NO**—tells RECOVERY MANAGER not to use NGT Recover OUTCOPY and to use the selected copy utility instead. Copies are made with the same DSNUM designation as is used for recovery.

**OUTCOPY_MAXPRIM**

Specifying MAXPRIM allows you to set a maximum amount of disk space (in the units specified by SPACE) to allocate as primary space. Valid values are 0 through 65535. A nonzero value for integer establishes an upper limit for primary space allocation; 0 specifies no limit.

**REBUILD_INDEX_SHRLEVEL**

Specify the level of access that concurrently operating DB2 applications and utilities should have to the target table space or index space. You can specify read-only access or read-write access. If you do not specify this option, NGT Recover allows read-only access to the target table space during the rebuild process.

- **REFERENCE** tells RECOVERY MANAGER to allow read-only access by other programs to the target space during the rebuild process.
- **CHANGE** tells RECOVERY MANAGER to allow read-write application access to DB2 table space and index space objects during the rebuild process.

**Note**

You can use this option in RBDP and PSRBD statuses.

**SNAP**

The SNAP option indicates if you want NGT Recover to read VSAM copies, even if the data set is not on a snappable disk:

- **SNAP=HW** (the default) tells NGT Recover to use a hardware data set snapshot to restore an Instant Snapshot or VSAM data set.
- **SNAP=VSAM** tells NGT Recover to use conventional VSAM I/O to restore a VSAM data set if it is not on a snappable disk.

To read a VSAM copy with SNAP=VSAM, you specify the name of the VSAM data set in an INCOPY statement, just as you would if it was an Instant Snapshot copy, using the INCOPY FULL SNAPSHOT DSNAME dataSetName syntax.

SNAP=VSAM also allows you to recover using a VSAM copy registered in BMCXCOPY or SYSCOPY if that copy is not on snappable disk.

When you use SNAP=VSAM, RECOVERY MANAGER adds the DATAMVR option to the JCL and always sets DATAMVR to DFDSS in this case.
UNLOADKEYS_BUILDINDEX

This option facilitates the rebuild of large nonpartitioned indexes on partitioned table spaces by providing concurrency in the extraction of keys from multiple partitions. Using this option can dramatically reduce the elapsed time required to rebuild a nonpartitioned index. This option is not valid with compressed indexes.

UNLOADKEYS_BUILDINDEX should be used in conjunction with MAX_CONCURRENT_JOBS, which should have a setting greater than 1. If you choose INDEX_ALL, then RECOVERY MANAGER does not generate the UNLOADKEYS_BUILDINDEX syntax. UNLOADKEYS_BUILDINDEX also cannot be used in conjunction with MAXKSORT. For more information about maximizing the concurrency of key sorts, see “Maximizing concurrency of key sorts” on page 230.

XBMID

Specify the 1-8 character ID of the EXTENDED BUFFER MANAGER (XBM) subsystem (which is required for use with Instant Snapshot copies).

**DSNUTILB recover options**

The following options are valid when using the DB2 DSNUTILB Recover utility.

**DSNUTILB_SITE_TYPE**

Specify the site type for recoveries when using DSNUTILB as the recover utility.

- LOCAL—recover the local site image copy (DSNUTILB keyword LOCALSITE)
- RECOVERY—recover the remote site image copy (DSNUTILB keyword RECOVERYSITE).

**DSNUTILB_SORTKEYS**

Specifies that index keys are sorted in parallel with the reload and build phases to improve performance. BMC recommends using this option if you need to recover more than one index. Specify YES to sort the index keys in parallel. Specify NO not to do so. Any WORKDDN specifications are ignored when you specify YES.
DSNUTILB_STATISTICS

Use this option to gather index statistics from the DB2 catalog. Specify YES to gather statistics, specify NO not to do so.

DSNUTILB_REPORT

Use this option to print the statistics collected (the ACCESSPSATH and SPACE statistics reports).

Note

This option is valid only with STATISTICS YES.

DSNUTILB_UPDATE

Use the option to update the catalog tables, as follows. This option is valid only with REPORT YES.

- NONE—no update to the catalog tables.
- ALL—insert all of the collected statistics in the DB2 catalog tables.
- ACCESSPATH—update only those columns used for access path selection.
- SPACE—update only those columns that provide statistics about the status of the target indexes.

DSNUTILB_KEYCARD

Use this option to collect the values in all of the key column combinations for the target indexes. This option is valid only with STATISTICS YES.

DATASET_SIZING

This option specifies the method by which object sizes are determined. Statistics in the BMCSTATS table are collected by DASD MANAGER PLUS and optionally by NGT Copy.

- CATALOG—use the DB2 and integrated catalog facility (ICF) catalog information for sizing purposes at the time of JCL generation. Doing so requires sizing calculations at the time of JCL generation.
- DEFAULTS—use existing default sizing information from the Work File options established in the Recovery options.
- BMCSTATS—use statistics from the BMCSTATS tables.
Work file recover options

The following options enable you to specify values for the work files required during recovery.

**WORKFILE_MAX_PRIMARY**

Limits the amount of primary allocation space to be used for the sort work space. It also applies to copies made to DASD. Valid values are 0 - 9999. Zero (0) indicates no limit.

RECOVERY MANAGER compares the maximum primary allocation to the value of the calculated primary allocation value and selects the smaller of the two. If the maximum primary value is selected, the primary space allocation is set to that value, the secondary space allocation is set to 1/15 of that value, and the value set for percent prime (if any) is ignored.

RECOVERY MANAGER calculates the number of units based on the maximum primary value. The unit parameter is UNIT=( workUnit, n) where n is the calculated number of units up to a maximum of 59.

**Note**

You must ensure that you use a maximum primary value that fits on your DASD devices. If the maximum primary value exceeds the capacity of a volume, the job will fail. The following gives examples of the capacity of some typical DASD devices:

<table>
<thead>
<tr>
<th>Physical Data for 3380 (per device)</th>
<th>Physical Data for 3390 (per device)</th>
<th>Physical Data for 9345 (per device)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Single Density (Models D &amp; J)</strong></td>
<td><strong>Model 1 (Single)</strong></td>
<td><strong>Model 1</strong></td>
</tr>
<tr>
<td>tracks: 13,275</td>
<td>tracks: 16,695</td>
<td>tracks: 21,600</td>
</tr>
<tr>
<td>cyls: 885</td>
<td>cyls: 1,113</td>
<td>cyls: 1,440</td>
</tr>
<tr>
<td><strong>Double Density (Model E)</strong></td>
<td><strong>Model 2 (Double)</strong></td>
<td><strong>Model 2</strong></td>
</tr>
<tr>
<td>tracks: 26,550</td>
<td>tracks: 33,390</td>
<td>tracks: 32,340</td>
</tr>
<tr>
<td>cyls: 1,770</td>
<td>cyls: 2,226</td>
<td>cyls: 2,156</td>
</tr>
<tr>
<td><strong>Triple Density (Model K)</strong></td>
<td><strong>Model 3 (Triple)</strong></td>
<td><strong>Not applicable</strong></td>
</tr>
<tr>
<td>tracks: 39,825</td>
<td>tracks: 50,085</td>
<td><strong>Not applicable</strong></td>
</tr>
<tr>
<td>cyls: 2,655</td>
<td>cyls: 3,339</td>
<td><strong>Not applicable</strong></td>
</tr>
<tr>
<td><strong>Not applicable</strong></td>
<td><strong>Not applicable</strong></td>
<td><strong>Not applicable</strong></td>
</tr>
<tr>
<td>Model 9 (Mod 9)</td>
<td>tracks: 150,255</td>
<td><strong>Not applicable</strong></td>
</tr>
<tr>
<td>cyls: 10,017</td>
<td><strong>Not applicable</strong></td>
<td><strong>Not applicable</strong></td>
</tr>
</tbody>
</table>

**WORKFILE_ALLOCATION_TYPE**

Specifies whether the work file allocations quantities are expressed in cylinders (CYL) or tracks (TRACK) for the work files.
WORKFILE_WORK_UNIT

Use this option to specify a disk file for use when dynamically allocating work files.

Tip

To determine the size of the work file, run DSN1LOGP with SUMMARY(ONLY) and allocate the output to a disk file.

WORKFILE_PRIMARY_ALLOC

Use this option to specify the primary allocation quantity for work files. The default is 10, 20. This value is used when RECOVERY MANAGER is unable to estimate the quantity due to problems or when the DATASET_SIZING option is set to DEFAULTS.

WORKFILE_SECONDARY_ALLOC

Use this option to specify the secondary allocation quantity for work files. The default is 10, 20. This value is used when RECOVERY MANAGER is unable to estimate the quantity due to problems or when the DATASET_SIZING option is set to DEFAULTS.

WORKFILE_WORKDDN

Use this option to direct the sorting of extracted index keys in index recoveries.

- **NO** (the default) sorts the keys without writing them to SYSUT1 and omits that file from the recovery JCL.

- **YES** writes extracted index keys to a work file (SYSUT1) before sorting.

RECOVERY MANAGER ignores this option if you specify a value greater than 1 for MAXKSORT.

WORKFILE_WORKPREFIX

Use this option to specify the high-level name to use when allocating sort and work files.

- If you specify a work prefix in the group options, RECOVERY MANAGER uses that work prefix.

- If you do not specify work prefix in the group options, RECOVERY MANAGER uses the value of **Work File Prefix** in ARMSOPTS (see “Subsystem options” on page 757).

- If **Work File Prefix** has no value in ARMSOPTS, RECOVERY MANAGER uses the user ID.
Output recover options

The following options establish values for the output data sets required during recovery.

The variable `copyType` in the output recover options is one of the following values:

- LP— local site primary copy
- LB—local site backup copy
- RP—recovery site primary copy
- RB—recovery site backup copy

An LP copy must be specified in order to specify an LB copy. An RP copy must be specified in order to specify an RB copy.

**RECOVER_OUTPUT_copyType_dsn**

Use this option to specify the name of the disk or tape data set used for output for each copy type when making copies after a recovery.

The variable `dsn` is the data set name.

You can use symbolic variables to construct this name. Generation data groups are not allowed.

The following is a sample data set name:

```
&USERID.&DB.&TS.&TYPE&DATE.T&TIME.
```

**RECOVER_OUTPUT_copyType_dsn_FOR_REC_PLUS**

Use this option to specify the name of the disk or tape data set used for output for each copy type when using the NGT Recover OUTCOPY feature to make copies after a recovery.

The variable `dsn` is the data set name.

The specified data set name is used as a prefix to which is appended the partition number in the form `A nn` (the number at the end of the data set name in the Virtual Storage Access Method (VSAM) catalog).

The following is a sample data set name:

```
&USERID.&DB.&TS.&TYPE&DATE.
```
This data set is only used when making copies by partition after a recovery using the NGT Recover OUTCOPY feature when the group is defined as DSNUM=0. All other output is sent to the data set specified in the RECOVER_OUTPUT_copyType_DSN dsn field.

**RECOVER_OUTPUT_copyType_UNIT**

This option specifies the name of the disk or tape unit to which the image copy data sets will be written (for example, SYSALLDA).

**RECOVER_OUTPUT_copyType_TAPE**

This option specifies whether the output unit is a tape. YES indicates the output unit is a tape. NO indicates the output unit is disk.

**RECOVER_OUTPUT_copyType_VOL_COUNT**

Specify the largest number of volumes that you expect NGT Recover to process when copying a single data set. For both tape and disk data sets, `nnn` must be an integer equal to or greater than the number of volumes produced for the single largest output copy, whether or not you use stacked output. To use the operating system default, set the value to 0.

If you are using SMS in your system, BMC recommends that you use the operating system default.

**RECOVER_OUTPUT_copyType_CATALOG**

It indicates whether or not to redefine the operating system catalog directive for the named descriptor. If any SMS option (STORCLAS, DATAACLAS, or MGMTCLAS) is used, NGT Recover forces CATLG YES.

**RECOVER_OUTPUT_copyType_STACK**

This option specifies whether to stack the output copies from multiple RECOVER executions contiguously on the same tape volumes. Valid values are YES, NO, CABINET, and RESET.

If you are copying to a disk unit, specify NO for this value (unless making cabinet copies); otherwise you will receive an INVALID COMBINATION message.

For Recovery Management solution only - you can specify CABINET to create cabinet copies. Cabinet copies can be made to either disk or tape. For more information, see the Recovery Management for DB2 User Guide.

**RECOVER_OUTPUT_copyType_MODEL_DSN**

This is the MODELDCB option of NGT Recover. Use this option to redefine the model DCB for the named descriptor. To specify that no model DCB be
used, use NONE as the data set name (dsn). The specified model data set must be allocated on a mounted direct access volume. NGT Recover copies the DCB information from the data set label. Symbolic variables are not allowed.

**RECOVER_OUTPUT_copyType_MAX_PRIMARY**

This option allows you to limit the amount of primary allocation space to be used for the output copy data sets made to DASD. Valid values are 0 - 9999. Zero indicates no limit.

RECOVERY MANAGER compares the maximum primary allocation to the calculated primary allocation value and selects the smaller of the two. If the maximum primary value is selected, the primary space allocation is set to that value and the secondary space allocation is set to 1/15 of that value.

RECOVERY MANAGER calculates the number of units based on the maximum primary value. The unit parameter is UNIT=(workUnit, n) where n is the calculated number of units up to a maximum of 59. If the maximum primary value is 0, the unit count defaults to 1.

You must ensure that you use a maximum primary value that fits on your DASD devices. If the maximum primary value exceeds the capacity of a volume, the job will fail. See “Work file recover options” on page 862 for the capacity of typical DASD devices.

**RECOVER_OUTPUT_copyType_ALLOC_TYPE**

For disk units, this option specifies whether the primary and secondary allocation quantities are expressed in cylinders (CYL) or tracks (TRACK). This option does not apply to tape units.

**RECOVER_OUTPUT_copyType_PRIMARY_ALLOC**

This option specifies the primary allocation quantity for output to disk. Use this option only when RECOVERY MANAGER is unable to estimate the quantity or when the DATASET_SIZING option is set to DEFAULTS.

**RECOVER_OUTPUT_copyType_SECONDARY_ALLOC**

This option specifies the secondary allocation quantity for output to disk. Use this option only when RECOVERY MANAGER is unable to estimate the quantity or when the DATASET_SIZING option is set to DEFAULTS.

**RECOVER_OUTPUT_copyType_SMS_STORAGE**

This option specifies a valid Storage Management Subsystem (SMS) storage class name for disk data sets. The name must not exceed 8 characters. RECOVERY MANAGER forces RECOVER_OUTPUT_copyType_CATALOG YES when this option is specified.
RECOVER_OUTPUT_copyType_SMS_DATA

This option specifies a valid SMS data class name for disk data sets. The name must not exceed 8 characters. RECOVERY MANAGER forces RECOVER_OUTPUT_copyType_CATALOG YES when this option is specified.

RECOVER_OUTPUT_copyType_SMS_MGMT

This option specifies a valid SMS management class name for disk data sets. The name must not exceed 8 characters. RECOVERY MANAGER forces RECOVER_OUTPUT_copyType_CATALOG YES when this option is specified.

RECOVER_OUTPUT_copyType_RETENTION

This option specifies the tape copy data set retention period in days. The valid range is 1 through 999. Retention period and expiration date are mutually exclusive.

RECOVER_OUTPUT_copyType_EXPIRATION

This option specifies the expiration date for a tape copy data set. The date must be in the format yyyy/ddd. The value 99/000 indicates no expiration.

RECOVER_OUTPUT_copyType_EATTR

This option specifies whether a data set supports extended attributes or not. Specifying no value for EATTR allows the value for EATTR to be set by an SMS DATACLAS.

Note
IBM z/OS Versions 1.11 or later support the EATTR option.
You can set EATTR to OPT or NO in the JCL.
If an image copy was written to the cylinder-managed portion of an EAV under z/OS Version 1.11, you cannot use that image copy on z/OS Version 1.10; Version 1.10 does not support sequential data sets in the cylinder-managed portion of an EAV.

Valid values for EATTR are:

- OPT specifies that extended attributes are optional for the data set.
  You must set OPT to allocate an extended format sequential data set. By using OPT, NGT Recover supports sequential data sets in the cylinder-managed portion of EAVs.
  Extended format sequential data sets must be allocated on SMS-managed volumes and the size of the data set must be greater than the EAV break point, which is typically 10 cylinders.
NO specifies that the data set cannot have extended attributes.

### General copy options

The general copy options enable you to specify the utilities to be used during a backup. Most apply to all supported copy utilities.

**COPYUTILITY**

Select the backup utility to be used, as follows:
- ACPMAIN—NGT Copy
- AFRMAIN—NGT Recover - OUTCOPY
- DSNUTILB—DB2 COPY

**REGION_SIZE**

Specify the amount of virtual storage used by the copy utility. The default value is 4 MB. The valid range is -1 through 2047 MB.

A value of -1 specifies that RECOVERY MANAGER will not generate region size at the step level. RECOVERY MANAGER JCL generation recognizes the -1 value and does not generate REGION=.

**Note**

For best performance, BMC recommends a region size of 0 MB, in which case the amount of virtual storage needed to run the job is automatically made available when the recover utility runs. Some data centers do not allow a region size of 0 MB, so 4 MB usually ensures adequate storage.

**QUIESCE_BEFORE**

Establish a quiesce point for each table space in the group before the copy process starts.

**QUIESCE_AFTER**

Establish a quiesce point for each space in the group immediately after the copy process completes.

**QUIESCE_GROUP**

Establish a common quiesce point for all table spaces within the current group. This option is ignored if QUIESCE_BEFORE and QUIESCE_AFTER are set to NO.
QUIESCE_WRITE

Instruct DB2 to finish writing any pending transactions for the target spaces before applying the quiesce. This option is ignored if QUIESCE_BEFORE and QUIESCE_AFTER are set to NO.

COPY_INDEX_SPACES

Specify whether to back up eligible index spaces. YES specifies to back up all eligible indexes. AUTO specifies to back up indexes as large or larger than the size specified by the INDEX_SIZE_THRESHOLD option.

Index spaces are eligible for backup if the following conditions exist:

- For NGT Copy, the DB2 release is Version 5.1 or greater and FULL YES is specified.
- For DB2 COPY, the DB2 release is Version 6.1 or greater, FULL YES is specified, and the index has the COPY YES attribute.

INDEX_SIZE_THRESHOLD_TYPE

Specify the unit of measure for the threshold size, as follows:

- K - Kilobytes
- M - Megabytes
- G - Gigabytes

INDEX_SIZE_THRESHOLD

Specify the size threshold at which you want indexes backed up rather than rebuilt. This option is used in conjunction with COPY_INDEX_SPACE AUTO. You can enter the size as follows:

- 0-4194303 if using M as the index size threshold type
- 0-4294967295 if using K as the index size threshold type
- 0-4095 if using G as the index size threshold type

OUTPUT_TYPE_copyType

Specify the types of image copies to be made for the spaces in the group, where copyType is

- LP—local site primary
- LB—local site backup
- RP—recovery site primary
- RB—recovery site backup

You cannot make a backup copy unless you make a corresponding primary copy. Similarly, you cannot make a recovery site copy without making a local copy.
**Note**

Online Consistent Copy does not support LB, RP, or RB copies. If you select NGT Copy as the copy utility with Shrlevel Change Consistent, RECOVERY MANAGER overrides requests for those copies, which will not be made. You can make LB, RP, and RB copies using COPY IMAGECOPY in the NGT Copy options. RECOVERY MANAGER will use NGT Copy to make the copies from the copy created by Online Consistent Copy.

**SCOPE**

Specify the scope of the copy operation for the specified objects. This option is only displayed when working with DB2 Version 10 or later, and when you use DSNUTILB as the backup utility.

- **ALL** indicates that all specified objects should be copied. This is the default.
- **PENDING** indicates that only objects in DB2 status of COPY-pending (COPY) or information COPY-pending (ICOPY) will be copied.

**NGT Recover copy options**

The following options are valid when using the NGT Copy product as the copy utility.

**ACPGDG_DATA_SET**

Specify the name of a data set that is used to define a generation data group (GDG) base. The data set must contain the control cards that are necessary to perform an IDCAMS DEFINE as well as the symbolic variable &BASE, which NGT Copy replaces with the GDG base name.

**AFTER_INIT PHASE**

Specifies whether NGT Copy should end (PAUSE) or continue (CONTINUE) after completing XBM registration of SHRLEVEL CONCURRENT copies. PAUSE causes the job to halt after all INIT processing for a group of table spaces completes. At that time, all of the affected table spaces are under control of XBM. See the NGT Copy for DB2 Reference Manual for more information.

**CHECKERROR**

This option is the CHECKERROR option of NGT Copy and lets you control the severity of page checking errors. You must provide an integral number
from 1 through 254 for use as a condition code. A code greater than 4 causes job termination at the point of error in the event of a page checking error.

CHECKTSLEVEL

This option is the CHECKTSLEVEL option of NGT Copy and is used to identify damaged pages that are found during copying. It lets you control the level of table space checking.

Valid values are
- 0—standard minimal checking
- 1—intrapage integrity checks for all pages
- 2—both intrapage and interpage checks for all pages

COPY_ALL_INDEX[ES]

Copy all indexes for the table spaces that are included in your group by using the NGT Copy INDEX[ES][YES] syntax. The brackets [ ] indicate that this part of the syntax is optional. This option is available with NGT Copy only and is ignored if the release of NGT Copy used does not support index backup or if the index is ineligible for backup. RECOVERY MANAGER does not generate the JCL for INDEXES(YES) if you specify the table spaces in your group by DSNUM.

Note

You can make incremental index copies if you set COPY_ALL_INDEX[ES] YES with FULL_COPY NO|AUTO, and you have BMC-supported versions of both NGT Copy and NGT Recover. For more information, see “About incremental index copies” on page 193.

COPY_IMAGECOPY_copyType

Use the NGT Copy COPY IMAGECOPY command after a copy job to make additional image copies from a local site primary copy that is already registered in SYSCOPY. Successful copies are automatically registered in SYSCOPY. You cannot make a copy of a type that is already registered in SYSCOPY. COPY IMAGECOPY is incompatible with compressed indexes.

Specify the copies that you want to make and register using COPY_IMAGECOPY_copyType, where copyType is:

- LB—local site backup copy.
- RP—for a recovery site primary copy.
- RB—for a recovery site backup copy.
DSSNAP

If you use BMC’s NGT Copy and XBM or SUF, you can use this option to make a hardware-based Instant Snapshot copy of DB2 data.

- **YES**—indicates that a hardware copy should be made.
- **AUTO**—indicates that a hardware copy should be made if possible, but a standard copy should be made if the hardware copy fails (for example if XBM, SUF, or the required hardware is not in place).
- **NO**—indicates that a standard copy should be made.

NGT Copy and XBM are required to use this option. If you specify DSSNAP YES or AUTO, then RESETMOD must be NO. If you specify DSSNAP YES or AUTO, then FULL_COPY must be YES.

NGT Copy and XBM are required to use this option. If you specify DSSNAP YES or AUTO, then RESETMOD must be NO. If you specify DSSNAP YES or AUTO, then FULL_COPY must be YES.

DSSNAP is incompatible with the ENCIPHER option.

FULL_COPY

Specify whether to make full or incremental image copies. (This option is the FULL option of NGT Copy.) Specify YES to make a full image copy. Specify NO to make an incremental copy. Specify AUTO to tell NGT Copy to automatically escalate to a full copy when it encounters one of the following conditions:

- An entry in SYSCOPY prohibits an incremental copy.
- The target object or partition is in copy pending status.
- The target is a "special case" catalog or directory table space.
- A specified number of incremental copies is reached.
- A specified percentage of changed pages is reached.
- A specified day of the week occurs.

"Special case" table spaces are certain spaces in DSNDB01 and DSNB06. For more information, see the *BMC Next Generation Technology Copy for DB2 for z/OS Reference Manual*. 
Note
You can make incremental index copies if you set COPY_ALL_INDEX[ES] YES with FULL_COPY NO | AUTO, and you have BMC-supported versions of both NGT Copy and NGT Recover. For more information about incremental index copies, see the *BMC Next Generation Technology Copy for DB2 for z/OS Reference Manual* manual.

FULL_AUTO_READ_PERCENT

This option specifies the percentage of changed pages that must occur to allow escalation from random I/O to full table space scan.

This option is only valid when used in conjunction with FULL_READTYPE AUTO.

FULL_CUMULATIVE

This option is the CUMULATIVE option of NGT Copy and allows you to merge the requested incremental RESETMOD NO copy with the most recent prior incremental RESETMOD NO copy. Specify NO when you do not want to perform such a merge. Specify YES to perform the merge.

This option is only valid when used in conjunction with FULL_COPY NO or FULL_COPY AUTO.

FULL_DAY_OF_WEEK

Specifies the day of the week on which a full copy should always be made.

Valid values are:
- SUN
- MON
- TUES
- WED
- THU
- FRI
- SAT

This option is only valid when used in conjunction with FULL_COPY AUTO. It takes precedence over all other FULL_COPY AUTO options (regardless of the changed pages percentages).

FULL_EMPTY

This option is the EMPTY option of NGT Copy and allows you to make and register a copy when no changed pages are found. Select NO to make and register a copy even though no pages changed since the last incremental copy
was made. Select **YES** (the NGT Copy default) to *not* make or register such a copy.

This option is only valid when used in conjunction with FULL_COPY NO or FULL_COPY AUTO.

**FULL_FULL_PERCENT**

Specifies a percent of changed pages used to determine whether to make a full copy instead of an incremental copy. You can enter the full percentage value in either decimal or integer format. Valid integer values range from 0 to 100. Valid decimal values are 00.0 to 99.9. A decimal value can only be specified to the tenth’s place (1/10 of a percent).

This option is only valid when used in conjunction with FULL_COPY AUTO.

**FULL_INCREMENTAL_PERCENT**

Specifies a percent of changed pages used to determine whether to make an incremental copy or no copy. You can enter the incremental percentage value in either decimal or integer format. Valid integer values range from 0 to 100. Valid decimal values are 00.0 to 99.9. A decimal value can only be specified to the tenth’s place (1/10 of a percent).

This option is only valid when used in conjunction with FULL_COPY AUTO.

**FULL_KEEP_PREVIOUS**

This option is the KEEP option of NGT Copy and allows you to retain the entry for the most recent prior (merged) incremental copy in the SYSCOPY table. Select NO to delete that entry from SYSCOPY. Select YES (the NGT Copy default) to retain the entry.

This option is only valid when used in conjunction with FULL_COPY NO or FULL_COPY AUTO.

**FULL_MAX_INCREMENTALS**

This option is the MAXINCRS option of NGT Copy. Provide an integral number from 1 through 100 to tell NGT Copy to escalate to a full image copy when the number of incremental copies that are registered in SYSCOPY since the last full copy reaches this number.

This option is only valid when used in conjunction with FULL_COPY AUTO.

**FULL_MIN_PAGES**

Specifies the minimum number of pages that must exist in a space or partition before an incremental copy is considered. If the space or partition has less than the specified number of pages, a full copy is made.
This option is only valid when used in conjunction with FULL_COPY AUTO.

FULL_NACTIVE

This option allows you to specify that you want NGT Copy to update and collect statistics for only the NACTIVE column of SYSIBM.SYSTABLESPACE. This is done in combination with the production of image copies. The following rules apply to NACTIVE:

- NACTIVE is ignored if you specify RUNSTATS YES.
- NACTIVE is valid for full, DSNUM ALL copies only; otherwise, NGT Copy issues an error.
- NACTIVE is ignored for indexes and also if the copy is passed to the DB2 COPY utility.
- NACTIVE is not valid for DSNDB06.SYSCOPY or any spaces in DSNDB01.
- NACTIVE is not valid for spaces in REORP status.
- NACTIVE is ignored for Instant Snapshots.
  This option is only valid when used in conjunction with FULL_COPY AUTO.

FULL_READTYPE

This option is the READTYPE option of NGT Copy and allows you to specify the method that is used to make incremental copies.

- RANDOM—use the conventional (random I/O) method
- FULLSCAN—use full table space scan techniques to identify changed pages
- AUTO—NGT Copy determines the I/O method based on the number of changed pages. This number is specified using the FULL_AUTO_READ_PERCENT option.
  This option is only valid when used in conjunction with FULL_COPY NO or FULL_COPY AUTO.

GROUP

This option is the GROUP option of NGT Copy and tells NGT Copy whether the spaces specified in the Object List should be treated as a group and, when you specify SHRLEVEL REFERENCE or SHRLEVEL CONCURRENT, share a common consistent point. When you specify GROUP YES and SHRLEVEL CONCURRENT to make copies using the SNAPSHOT UPGRADE
FEATURE, you can also use the START_MESSAGE option to issue a text message when NGT Copy/Snapshot initialization is complete.

**MAX_TASKS and MAXTASKS2**

Specify the maximum number of tape subtasks, MAX_TASKS, and optionally the maximum number of all subtasks, MAX_TASKS2, that are used by NGT Copy when making image copies. (This option is the MAXTASKS option of NGT Copy.) The following rules apply:

- Valid values for maximum tape tasks are 1 through 32.
- Valid values for maximum total tasks are \(\text{maximum tape tasks} - 32\).
- Maximum tape tasks is required if maximum total tasks is specified.
- Maximum total tasks cannot be less than maximum tape tasks.
- If maximum tape tasks is set but maximum total tasks is not set, maximum total tasks defaults to AUTO, which allows NGT Copy to determine the value for maximum total tasks.

**NUMBER_READ/WRITE_BUFFERS**

This option is the NBRBUFS option of NGT Copy and tells NGT Copy how many read/write buffers to use. Specify an integer from 2 through 16.

**ON_DUPLICATEDS**

Specify what action NGT Copy should take if it encounters a copy data set that is already registered in SYSCOPY or BMCXCOPY.

- ERROR — terminate processing if the data set is already registered.
- DELETE — continue processing when a data set is already registered. When NGT Copy registers the new copy, it will delete the row containing the duplicate data set in SYSCOPY or BMCXCOPY. Only rows with the same DSNAME, TSNAME (IXNAME), or DSNUM as the new copy are deleted.

**ON_ERROR_BADSTATUS**

Specify the action NGT Copy should take if it encounters a table space or partition that has an unacceptable status or has a BMC or DB2 utility running against it.

- END—terminate processing with an RC=12
- SKIP—issue a message, skip over the space, and continue processing other spaces as specified.
ON_ERROR ICEXISTS

Specifies what action NGT Copy should take if it encounters a table space or partition for which an image copy already exists.

- END—terminate processing with an RC=12
- SKIP—issue a message, skip over the space, and continue processing other spaces as specified. This option is effective for the COPY IMAGECOPY command.

ON_ERROR NOTSUPPORTED

Specifies what action NGT Copy should take if it encounters a table space or partition that is of a type that is not supported by NGT Copy.

- END—terminate processing with an RC=12
- SKIP—issue a message, skip over the space, and continue processing other spaces as specified. (This option is effective for both the COPY and COPY IMAGECOPY commands.)

OUTSIZE_THRESHOLD

This option specifies a size threshold for making copies to an alternate DD or output descriptor and can be used to escalate output to tape rather than DASD. OUTSIZE_THRESHOLD is specified as number of pages. Valid values are 0 through 1073741823. Zero specifies no effect.

REPORT_STATISTICS

Report statistics via SYSPRINT.

RESETMOD

Specify whether to reset the modified page indicators in the table spaces and space maps after you make an image copy. Select YES to reset those indicators. Select NO not to do so.

RESYNC

Causes XBM or SUF to resynchronize hardware mirroring activities after an Instant Snapshot copy. Specify NO to leave the mirrors unsynchronized during subsequent user processing. If you specify NO, you must reestablish the mirroring synchronization through XBM or SUF.

This option is available with NGT Copy version 6.1 or later. It is ignored if the Instant Snapshot copy is made without hardware mirroring in place.
**RUNSTATS**

Update the DB2 catalog or the BMCSTATS table with statistics that are collected concurrently with the creation of full image copies. If you collect statistics, you can also choose to report statistics and specify how the DB2 catalog tables or the BMCSTATS table should be updated.

To use this option, you must also specify at least one of the following options: REPORT_STATISTICS, UPDATE_DB2_CATALOG, or UPDATE_BMCSTATS.

**SEPARATE_BY_PARTITION**

This option tells NGT Copy whether to make and register copies by partition when you back up partitioned table spaces. Specify **YES** to make copies by partition. Specify **NO** to make copies by table space. This option is applicable when the objects have not been listed by partition.

*Note*

This option is valid in NGT Copy syntax only when you have specified OBJECTSET NO in ARMBGEN. This option is invalid for the COPY TABLESPACE OBJECTSET group name.

**SHRLEVEL**

Specify to NGT Copy the level of access to the target spaces to be allowed to concurrently executing DB2 applications and utilities during the copy process. You can specify one of the following levels:

- **Reference**—allow only read-only access by other programs during the copy process.
- **Change**—maintain the initial status of the spaces.
- **Any**—use Shrlevel Change unless it encounters any conditions that require more restrictive access. If NGT Copy encounters such conditions, it uses Shrlevel Reference.
- **None**—stop all access by other programs to the target spaces during the copy process.
- **Concurrent**—utilize the SNAPSHOT UPGRADE FEATURE (SUF) feature that allows NGT Copy to make consistent copies of the table spaces while updates to those table spaces are in progress. You can specify a value of required or preferred for this option. For full support, you must have either the SNAPSHOT UPGRADE FEATURE (SUF) or EXTENDED BUFFER MANAGER (XBM) installed. If you do not specify the XBMID in the options statement, the value defaults to the NGT Copy configuration option value. The copy job will fail if the XBMID is not present.
■ Concurrent required—terminate the copy with a return code of 12 if a consistent point cannot be obtained or maintained.

■ Concurrent preferred—use Shrlevel Change when a consistent copy cannot be obtained or maintained or if initialization of the SUF or XBM fails.

■ Change Consistent Yes—use for Online Consistent Copy. This option requires a solution password and must be set at the group level.

**SNAP**

The SNAP option indicates if you want NGT Copy to make VSAM copies, even if the data set is not on a snappable disk:

■ SNAP=HW (the default) tells NGT Copy to use a hardware data set snapshot to make an Instant Snapshot. NGT Copy uses SNAP=HW if the source data set is not SMS-managed or if you did not specify an SMS STORCLAS on the NGT Copy OUTPUT command.

■ SNAP=VSAM tells NGT Copy to use conventional VSAM I/O to copy a VSAM data set if it is not on a snappable disk.

SNAP=VSAM is only supported when the source data set is SMS-managed or you specified an SMS STORCLAS on the NGT Copy OUTPUT command.

When you use SNAP=VSAM, RECOVERY MANAGER adds the DATAMVR option to the JCL and always sets DATAMVR to DFDSS in this case.

**SQUEEZE**

This option is the SQUEEZE option of NGT Copy and lets you specify whether to consolidate the rows on each target table space page so that all free space is contiguous.

**START_MESSAGE**

When you use SHRLEVEL CONCURRENT, use START_MESSAGE to write a message (BMC47497) to the system log of the operating system when NGT Copy/Snapshot initialization has successfully completed.

You can use this feature to trigger the submission of jobs that you want to run concurrently (SHRLEVEL CONCURRENT) with the NGT Copy job.

The message has the following format:

BMC47497 SNAPSHOT STARTED: ‘string’
string is a character string of your choice of up to 50 characters and must be enclosed in single quotes.

Be aware of the following restrictions:

- START_MESSAGE is valid only with GROUP YES.
- Strings over 50 characters are truncated
- No quotes must appear within the text string.
- Words within the message cannot be separated by spaces. Use underscore marks instead.

For example,

`'INITIALIZATION_COMPLETED_SUCCESSFULLY'`

**SUPPORT_FULL_COPY_DDS**

Specifies to use an alternative descriptor for COPYDDN when the FULL AUTO FULLPCT settings cause a full copy to be made. If YES is specified and a full copy is made, the image copy output goes to an alternate set of DDs that are specified with the following keywords:

- FULLDDN
- FULLDSN
- FULLRECDATA
- FULLRECDSN

**SUPPORT_OUTSIZE**

Specify YES to use an alternative descriptor for COPYDDN when the number of pages to copy is equal to or greater than the OUTSIZE_THRESHOLD value. If YES is specified and the threshold condition is met or exceeded, the image copy output will go to an alternative set of DDs that are specified with the following keywords:

- BIGDDN
- BIGDSN
- BIGRECDATA
- BIGRECDSN

**UNIT_COUNT**

This option is the UNITCNT option of NGT Copy and specifies the number of units to be allocated for the output image copy data sets. Provide an integral number from 0 to 59. Zero (0) means no unit count is specified and so can be controlled with SMS if required. The default is no unit count at all.
UPDATE_BMCSTATS

Collect statistics and update the BMCSTATS table.

UPDATE_DB2_CATALOG

Specify one of the following options to update the catalog:

- ALL—update all statistics.
- NONE—make no update to any statistics.
- ACCESSPATH—update only statistics that are used for access path selection.
- SPACE—update only those statistics that are used to assess table space status.

XBMID

Specify the XBM subsystem ID (1-8 characters) to be used when you are making Shrlevel Concurrent copies.

**DSNUTILB copy options**

The following options are valid when using the DB2 DSNUTILB Copy utility.

**DSNUTILB_FULL_COPY**

Specify the type of copy to make when using DB2 COPY (DSNUTILB) as your copy utility.

- YES - make a full image copy.
- NO - make an incremental image copy.
- AUTO - use the CHANGELIMIT option.

In the following situations, an incremental copy request escalates to a full image copy request:

- No full image copies exist for the spaces that are being copied.
- This is the first image copy request after a REORG or LOAD.
- This is the first copy since a partial recovery was performed.
- The most recent copy job for this space was terminated.
- The space is a "special case" table space.

- The most recent full image copy was a DFSMS Concurrent Copy.

"Special case" table spaces are certain spaces that are located in DSNDB01 and DSNDB06. See the IBM command and utility reference for more information.

**DSNUTILB_SHRLEVEL**

Specify the level of access that is allowed to DB2 applications and utilities that are executing concurrently during the copy process.

- REFERENCE -- read-only access by other programs to the spaces in the group during the copy process.

- CHANGE -- write access to the spaces in the group.

  This option is not valid for table spaces having a page size of 32 kilobytes (KB) when you specify Concurrent/DFSMS Yes.

**DSNUTILB_FULL_CONCURRENT**

Specifies whether to make concurrent copies by using the DFSMS Concurrent Copy utility. It is valid only when you are making a full image copy.

**DSNUTILB_FULL_INCREMENTAL_PERCENT**

Specifies a percent of changed pages used to determine whether to make an incremental copy or no copy. You can enter the incremental percentage value in either decimal or integer format. Valid integer values range from 0 to 100. Valid decimal values are 00.0 to 99.9. A decimal value can only be specified to the tenth’s place (1/10 of a percent).

  This option is only valid when used in conjunction with DSNUTILB_FULL_COPY AUTO.

**DSNUTILB_FULL_FULL_PERCENT**

Specifies a percent of changed pages used to determine whether to make a full copy instead of an incremental copy. You can enter the full percentage value in either decimal or integer format. Valid integer values range from 0 to 100. Valid decimal values are 00.0 to 99.9. A decimal value can only be specified to the tenth’s place (1/10 of a percent).

  This option is only valid when used in conjunction with DSNUTILB_FULL_COPY AUTO.
DSNUTILB_GROUP

Specifies whether the DB2 COPY utility ensures that all target objects share a common point of consistency. This is valid for table spaces, index spaces, and indexes. Only full image copies are supported for indexes.

DSNUTILB_COPY_PARALLEL

Specifies whether to copy objects in parallel.

Be aware of the following information:

- This option and DSNUTILB_FULL_CONCURRENT are mutually exclusive options.
- You must specify DSNUTILB_GROUP YES to be able to copy objects in parallel.
- This option is not available when copying objects to tape.

DSNUTILB_MAX_PARALLEL

Specifies the maximum number of objects that should be processed in parallel.

NGT Recover OUTCOPY copy options

The following options are valid when using the NGT Recover OUTCOPY feature to make backups:

OUTCOPY_EARLYRECALL

This option is the EARLYRECALL option of NGT Recover and allows you to retrieve archived image copies and log data sets during the ANALYZE phase. Specify YES to retrieve those items during the ANALYZE phase. Specify NO to delay recall until the data sets are accessed or allocated.

OUTCOPY_EARLYCAT

This option is the EARLYCAT option of NGT Recover and allows you to verify that data sets that are marked as cataloged during the ANALYZE phase actually exist in the operating system catalog. Specify YES to perform the verification. Specify NO to delay this check until the data sets are allocated.
OUTCOPY_USEACCUM

This option is the USEACCUM option of NGT Recover and allows you to access change accumulation files. This option is valid only when R+/CHANGE ACCUM is installed.

OUTCOPY_ANALYZE

This option is the ANALYZE option of NGT Recover and allows you to print information about objects that are targeted for recovery when you execute the recovery JCL.

- YES—print a recovery plan before performing the recovery
- NO—print only a subset of that information
- ONLY—terminate execution of the recovery JCL after the plan is printed.

OUTCOPY_MAXDRIVES

This option is the MAXDRIVES option of NGT Recover and specifies the maximum number of tape drives to be used during backup.

OUTCOPY_SORTDEVT

This option is the SORTDEVT option of NGT Recover and specifies the device type for temporary work data sets that are required for log sorts.

OUTCOPY_MAXPRIM

Specifying MAXPRIM allows you to set a maximum amount of disk space (in the units specified by SPACE) to allocate as primary space. Valid values are 0 through 65535. A nonzero value for integer establishes an upper limit for primary space allocation; 0 specifies no limit.

OUTCOPY_AUTOSIZE

This option is the AUTOSIZE option of NGT Recover and turns dynamic sizing for output image copies or change accumulation output files on or off. Valid values are YES are NO. If you specify a value for AUTOSIZE, you override the AUTOSIZE installation option, which defaults to YES.

- YES—specifies dynamic sizing for output image copies or change accumulation output files allocated to DASD.
- NO—specifies that output image copies or change accumulation output files are allocated to DASD using the primary and secondary quantities that are specified in the R+/CHANGE ACCUM repository.

TOLOGPOINT

For NGT Recover OUTCOPY only. Specify the point in the DB2 log to which the backup copies should be made. You can select one of the following:
- CURRENT—make backup copies of all recoverable objects in the group to the current time.

- LASTQUIESCE—make a backup of each recoverable object to the last quiesce that is registered in SYSCOPY for that object.

- SPECIFIC_RBA—make backup copies of all objects in the group to the same RBA. Specify the RBA by using the TORBA option.

- LASTARCHQUIESCE—make backup copies of each object in the group to the point in the log that is established by the most recent ARCHIVE LOG MODE(QUIESCE) for the DB2 subsystem.

- LASTSHUTDOWN—make backup copies of all objects in the group to the point in the log that is established by the most recent STOP DB2 command.

- LASTCOMMONQUIESCE—make backup copies of all objects in the group to the point in the log that is established by the most recent common point.

- TIMESTAMP—make backup copies of all objects in the group to the RBA associated with the timestamp in the format YYYY-MM-DD-HH.MM.SS.

- LOGMARK—make backup copies of all objects in the group to the RBA associated with the logmark.

TORBA

Use this option to specify an RBA to which you want to make a backup. This is used in conjunction with TOLOGPOINT SPECIFIC_RBA.

OUTCOPY_MSGLEVEL

This option specifies which output files and messages NGT Recover returns. Valid values for MSGLEVEL are STANDARD, OBJECT_SUMMARY, and PLAN_SUMMARY. For more information about the type of output produced by each option, see the BMC Next Generation Technology Recover for DB2 for z/OS Reference Manual.

STANDARD returns the following output files:

- AFRPRINT—execution messages

- AFRSUMRY—maintenance applied, phases completed, utility return codes

- AFRSTMT—input statements and options as specified in SYSIN, configuration option values, and log file resources
OBJECT_SUMMARY returns the following output files:

- AFRPRINT - execution messages
- AFRSUMRY - maintenance applied, phases completed, utility return codes
- AFRSTMT - input statements and options as specified in SYSIN, configuration option values, and log file resources
- AFROSUM - object summary for objects being recovered

PLAN_SUMMARY returns the following output files:

- AFRPRINT - execution messages
- AFRSUMRY maintenance applied, phases completed, utility return codes
- AFRSTMT - input statements and options as specified in SYSIN, configuration option values, and log file resources
- AFROSUM - object summary for objects being recovered
- AFRPLAN - execution plan

OUTCOPY_COPY_copyType

Rank each image copies in the order that you would like it to be used for recovery, as follows:

- 1 — image copy that you want as first choice.
- 2 — second choice (if any).
- 3 — third choice (if any).
- 4 — fourth choice (if any).
- 0 — do not want to use the copy at all.

The variable copyType is one of the following values:

- LP — local site primary
- LB — local site backup
- RP — recovery site primary
- RB — recovery site backup

OUTCOPY_ACT1

Rank the active log copy 1 in the order that you would like it to be used for recovery, as follows:

- 1 — log copy that you want as first choice.
- 2 — second choice (if any).
3 —third choice (if any).
4 —fourth choice (if any).
0 —do not want to use the copy at all.

OUTCOPY_ACT2

Rank the active log copy 2 in the order that you would like it to be used for recovery, as follows:
1 —log copy that you want as first choice.
2 —second choice (if any).
3 —third choice (if any).
4 —fourth choice (if any).
0 —do not want to use the copy at all.

OUTCOPY_ARC1

Rank the archive log copy 1 in the order that you would like it to be used for recovery, as follows:
1 —log copy that you want as first choice.
2 —second choice (if any).
3 —third choice (if any).
4 —fourth choice (if any).
0 —do not want to use the copy at all.

OUTCOPY_ARC2

Rank the archive log copy 2 in the order that you would like it to be used for recovery, as follows:
1 —log copy that you want as first choice.
2 —second choice (if any).
3 —third choice (if any).
4 —fourth choice (if any).
0 —do not want to use the copy at all.

OUTCOPY_CHANGE_ACCUM_copyType

Rank the change accumulation copy in the order that you would like it to be used for recovery, as follows:
1 —change accumulation copy that you want as first choice.
2 —second choice (if any).
3 —third choice (if any).
4 —fourth choice (if any).
0 —do not want to use the copy at all.
Output copy options

The following options establish values for the output data sets required during backup.

The variable *copyType* in the output copy options is one of the following values:

- **LP**—local site primary copy
- **LB**—local site backup copy
- **RP**—recovery site primary copy
- **RB**—recovery site backup copy

An LP copy must be specified in order to specify an LB copy. An RP copy must be specified in order to specify an RB copy.

**OUTPUT_copyType_DSN**

Specify the name of the disk or tape data set for the specified copy type. You can use symbolic variables to construct this name (see “Output data sets, job cards, and symbolic variables” on page 51).

**OUTPUT_copyType_ENCIPHER**

Specify whether to use the NGT Copy ENCIPHER syntax option when generating the JCL to make copies. COPY PLUS version 7.3 and later can make encrypted image copies to prevent unauthorized access to sensitive company information. (Encryption is a feature of the BMC Recovery Management for DB2 solution and requires a valid Recovery Management solution password.)

ENCIPHER is incompatible with DSSNAP and with compressed indexes.

**OUTPUT_copyType_UNIT**

Specify the name of the disk or tape unit to which the image copy data sets will be written.

**OUTPUT_copyType_TAPE**

Specify whether the output unit is tape or disk, as follows:

- **YES**—the unit is tape
- **NO**—the unit is disk.

**OUTPUT_copyType_STACK**

When you are using tape, this option specifies whether to stack image copies of the same type contiguously on the same tape. Valid values are **YES**, **NO**, **CABINET**, and **RESET**.
If you are copying to a disk unit, specify NO for this value (unless making cabinet copies); otherwise you will receive an **INVALID COMBINATION** message.

For Recovery Management solution only - you can specify CABINET to create cabinet copies. Cabinet copies can be made to either disk or tape. For more information, see the *Recovery Management for DB2 User Guide*.

**OUTPUT_copyType_CATALOG**

Catalog the data sets in the operating system catalog.

**OUTPUT_copyType_EATTR**

This option specifies whether a data set supports extended attributes or not. Specifying no value for EATTR allows the value for EATTR to be set by an SMS DATACLAS.

*Note*

IBM z/OS Versions 1.11 or later support the EATTR option.

You can also set EATTR to OPT or NO in the JCL.

If an image copy was written to the cylinder-managed portion of an EAV under z/OS Version 1.11, you cannot use that image copy on z/OS Version 1.10; Version 1.10 does not support sequential data sets in the cylinder-managed portion of an EAV.

Valid values for EATTR are:

- **OPT** specifies that extended attributes are optional for the data set.
  
  You must set OPT to allocate an extended format sequential data set. By using OPT, NGT Copy supports sequential data sets in the cylinder-managed portion of EAVs.
  
  Extended format sequential data sets must be allocated on SMS-managed volumes and the size of the data set must be greater than the EAV break point, which is typically 10 cylinders.

- **NO** specifies that the data set cannot have extended attributes.

**OUTPUT_copyType_MAX_PRIMARY**

Specify the maximum amount of disk space (in the units specified by OUTPUT_copyType_ALLOCATION_TYPE) that may be allocated as primary space when making copies with either DSNUTILB copy or NGT Copy. Zero indicates no limit, while a nonzero value establishes an upper limit on the value calculated by PCTPRIM. Valid values are 0 through 9999.
OUTPUT_copyType_ALLOCATION_TYPE

Specify the unit of disk space to be used with the OUTPUT_copyType_MAX_PRIMARY option. CYL indicates cylinders. TRACK indicates tracks.

OUTPUT_copyType_RETENTION

Specify the tape copy data set retention period in days. The valid range is 1 through 999. Retention period and expiration date are mutually exclusive.

OUTPUT_copyType_EXPIRATION

Specify the expiration date for a tape copy data set. Valid formats for the expiration date are as follows:

- yyyy/ddd
  where yyyy is a year in the range 1994 through 2155 and ddd is the Julian day in the range 0 through 366.

- yyddd
  where yy is the year in the range 94 through 99 only and ddd is the Julian day in the range 0 through 366.

OUTPUT_copyType_VOL_COUNT

This option specifies the largest number of tape volumes that are expected to be created. The valid range is 1 through 255. This option applies only to tape data sets.

OUTPUT_copyType_MODEL_DSN

Specify the fully qualified name of a cataloged data set to define the model data control block (DCB). Symbolic variables are not allowed.

OUTPUT_copyTypePRIMARY_ALLOC

This option specifies the primary allocation quantity (disk only). Use this option only when RECOVERY MANAGER is unable to estimate the quantity. This value is used when you make copies using DSNUTILB COPY or when you use NGT Copy to make CABINET copies to disk. It is ignored when you use NGT Copy to make any other type of copy because NGT Copy performs its own data set sizing.

OUTPUT_copyTypeSECONDARY_ALLOC

This option specifies the secondary allocation quantity (disk only). Use this option only when RECOVERY MANAGER is unable to estimate the quantity.
This value is used when you make copies using DSNUTILB COPY or when you use NGT Copy to make CABINET copies to disk. It is ignored when you use NGT Copy to make any other type of copy because NGT Copy performs its own data set sizing.

**OUTPUT_copyType_SMS_STORAGE**

This option specifies a valid SMS storage class name for disk data sets. The name must not exceed 8 characters. RECOVERY MANAGER forces OUTPUT_copyType_CATALOG YES when this option is specified.

**OUTPUT_copyType_SMS_DATA**

This option specifies a valid SMS data class name for disk data sets. The name must not exceed 8 characters. RECOVERY MANAGER forces OUTPUT_copyType_CATALOG YES when this option is specified.

**OUTPUT_copyType_SMS_MGMT**

This option specifies a valid SMS management class name for disk data sets. The name must not exceed 8 characters. RECOVERY MANAGER forces OUTPUT_copyType_CATALOG YES when this option is specified.

**OUTPUT_copyType_ACP_GDG**

Use this option to specify a fully qualified data set name to be used to define a GDG base for this copy. The named data set must contain the control cards that are needed to perform an IDCAMS DEFINE as well as the symbolic variable &base, which replaces the GDG base name.

**OUTPUT_copyType_DSSNAP**

If you use NGT Copy and XBM or SUF, you can use this option to make a hardware-based Instant Snapshot copy of DB2 data.

- **YES**—indicates that a hardware copy should be made.
- **AUTO**—indicates that a hardware copy should be made if possible, but a standard copy should be made if the hardware copy fails (for example if XBM, SUF, or the required hardware is not in place).
- **NO**—indicates that a standard copy should be made.

The following restrictions apply to this option:

- The NGT Copy and XBM or SUF products are required to use this option.
- DSSNAP is not allowed with Compressed Index.
- DSSNAP is not allowed with Copy Type INCR or AUTO.
- DSSNAP is not allowed with the ENCIPHER option.
- If you specify DSSNAP YES or AUTO, then RESETMOD must be NO.
- DSSNAP is not allowed with Online Consistent Copy.
- FULLDD is only valid with Copy Type Full Auto.
- If you specify DSSNAP YES or AUTO, then FULL_COPY must be YES.

**OUTPUT_copyType_MIGRATE**

Use this option to specify Hierarchical Storage Management (HSM) migration of copy data sets when NGT Copy is finished with them.

You must specify **OUTPUT_copyType_TAPE NO** and specify a disk unit for the **OUTPUT_copyType_UNIT option**.

Valid values are as follows:
- NO — (the default) suppresses migration. RECOVERY MANAGER generates no MIGRATE syntax in the JCL.
- HSM — specifies migration to compressed disk. Make sure that you have enough space available on the disk when using this value.
- HSM ML2 — causes immediate migration to a migration level 2 (MIGRATIONLEVEL2) volume.

**FULLDDN copy options**

If you want full copies to be allocated to a different output descriptor or data set name than incremental copies, use the FULLDDN copy options.

The variable *copyType* in the output copy options is one of the following values:
- LP— local site primary copy
- LB—local site backup copy
- RP—recovery site primary copy
- RP—recovery site backup copy

**FULLDDN_copyType_DSN**

Specify the name of the disk or tape data set for the specified copy type. You can use symbolic variables to construct this name (see “Output data sets, job cards, and symbolic variables” on page 51).
FULLDDN_copyType_ENCIPHER

Specify whether to use the NGT Copy ENCIPHER syntax option when generating the JCL to make copies. COPY PLUS version 7.3 and later can make encrypted image copies to prevent unauthorized access to sensitive company information. (Encryption is a feature of the BMC Recovery Management for DB2 solution and requires a valid Recovery Management solution password.)

ENCIPHER is incompatible with DSSNAP and with compressed indexes.

FULLDDN_copyType_UNIT

Specify the name of the disk or tape unit to which the image copy data sets will be written.

FULLDDN_copyType_TAPE

- YES—the unit is tape
- NO—the unit is disk.

Specify whether the output unit is tape or disk, as follows:

FULLDDN_copyType_CATALOG

Catalog the data sets in the operating system catalog.

FULLDDN_copyType_EATTR

This option specifies whether a data set supports extended attributes or not. Specifying no value for EATTR allows the value for EATTR to be set by an SMS DATACLAS.

Note
IBM z/OS Versions 1.11 or later support the EATTR option.
You can also set EATTR to OPT or NO in the JCL.
If an image copy was written to the cylinder-managed portion of an EAV under z/OS Version 1.11, you cannot use that image copy on z/OS Version 1.10; Version 1.10 does not support sequential data sets in the cylinder-managed portion of an EAV.

Valid values for EATTR are:

- OPT specifies that extended attributes are optional for the data set.
  You must set OPT to allocate an extended format sequential data set. By using OPT, NGT Copy supports sequential data sets in the cylinder-managed portion of EAVs.
  Extended format sequential data sets must be allocated on SMS-managed volumes and the size of the data set must be greater than the EAV break point, which is typically 10 cylinders.
NO specifies that the data set cannot have extended attributes.

**FULLDDN_copyType_STACK**

When you are using tape, this option specifies whether to stack image copies of the same type contiguously on the same tape. Valid values are YES, NO, CABINET, and RESET.

If you are copying to a disk unit, specify NO for this value (unless making cabinet copies); otherwise you will receive an **INVALID COMBINATION** message.

For Recovery Management solution only - you can specify CABINET to create cabinet copies. Cabinet copies can be made to either disk or tape. For more information, see the *Recovery Management for DB2 User Guide*.

**FULLDDN_copyType_MAX_PRIMARY**

Specify the maximum amount of disk space (in the units specified by **OUTPUT_copyType_ALLOCATION_TYPE**) that may be allocated as primary space. Zero indicates no limit, while a nonzero value establishes an upper limit on the value calculated by PCTPRIM. Valid values are 0 through 9999.

**FULLDDN_copyType_ALLOCATION_TYPE**

Specify the unit of disk space to be used with the **OUTPUT_copyType_MAX_PRIMARY** option. CYL indicates cylinders. TRACK indicates tracks.

**FULLDDN_copyType_RETENTION**

Specify the tape copy data set retention period in days. The valid range is 1 through 999. Retention period and expiration date are mutually exclusive.

**FULLDDN_copyType_EXPIRATION**

Specify the expiration date for a tape copy data set. The date must be in the format *yyyy/ddd*, where *yyyy* is the year and *ddd* is the Julian day.

**FULLDDN_copyType_VOL_COUNT**

This option specifies the largest number of tape volumes that are expected to be created for FULLDDN copies. The valid range is 1 through 255. This option applies only to tape data sets.

**FULLDDN_copyType_MODEL_DSN**

Specify the fully qualified name of a cataloged data set to define as the model data control block (DCB) for FULLDDN copies. Symbolic variables are not allowed.
FULLDDN_copyType_PRIMARY_ALLOC

This option specifies the primary allocation quantity (disk only) for FULLDDN copies. Use this option only when RECOVERY MANAGER is unable to estimate the quantity.

Primary allocation values are ignored if the selected backup utility is NGT Copy. NGT Copy calculates the correct allocations.

FULLDDN_copyType_SECONDARY_ALLOC

This option specifies the secondary allocation quantity (disk only) for FULLDDN copies. Use this option only when RECOVERY MANAGER is unable to estimate the quantity.

Secondary allocation values are ignored if the selected backup utility is NGT Copy. NGT Copy calculates the correct allocations.

FULLDDN_copyType_SMS_STORAGE

This option specifies a valid SMS storage class name for disk data sets used for FULLDDN copies. The name must not exceed 8 characters. RECOVERY MANAGER forces OUTPUT_ copyType_CATALOG YES when this option is specified.

FULLDDN_copyType_SMS_DATA

This option specifies a valid SMS data class name for disk data sets used for FULLDDN copies. The name must not exceed 8 characters. RECOVERY MANAGER forces OUTPUT_ copyType_CATALOG YES when this option is specified.

FULLDDN_copyType_SMS_MGMT

This option specifies a valid SMS management class name for disk data sets used for FULLDDN copies. The name must not exceed 8 characters. RECOVERY MANAGER forces OUTPUT_ copyType_CATALOG YES when this option is specified.

FULLDDN_copyType_ACP_GDG

Use this option to specify a fully qualified data set name to be used to define a GDG base for the FULLDDN copy. The named data set must contain the control cards that are needed to perform an IDCAMS DEFINE as well as the symbolic variable &base, which replaces the GDG base name.

FULLDDN_copyType_MIGRATE

Use this option to specify Hierarchical Storage Management (HSM) migration of copy data sets when NGT Copy is finished with them. MIGRATE is available only with COPY PLUS version 6.4 or later.
You must specify `FULLDDN_copyType_TAPE NO` and specify a disk unit for the `FULLDDN_copyType_UNIT` option.

Valid values are as follows:

- **NO**—(the default) suppresses migration. RECOVERY MANAGER generates no MIGRATE syntax in the JCL.
- **HSM**—specifies migration to compressed disk. Make sure that you have enough space available on the disk when using this value.
- **HSM ML2**—causes immediate migration to a migration level 2 (MIGRATIONLEVEL2) volume.

**FULLDDN_copyType_DSSNAP**

If you use NGT Copy and XBM or SUF, you can use this option to make a hardware-based Instant Snapshot copy of DB2 data.

- **YES**—indicates that a hardware copy should be made.
- **AUTO**—indicates that a hardware copy should be made if possible, but a standard copy should be made if the hardware copy fails (for example if XBM, SUF, or the required hardware is not in place).
- **NO**—indicates that a standard copy should be made.

The following restrictions apply to this option:

- The NGT Copy and XBM or SUF products are required to use this option.
- DSSNAP is not allowed with Compressed Index.
- DSSNAP is not allowed with Copy Type INCR or AUTO.
- DSSNAP is not allowed with the ENCIPHER option.
- If you specify DSSNAP YES or AUTO, then RESETMOD must be NO.
- DSSNAP is not allowed with Online Consistent Copy.
- FULLDD is only valid with Copy Type Full Auto.
- If you specify DSSNAP YES or AUTO, then FULL_COPY must be YES.
BIGDDN copy options

The use of the BIGDDN copy options (with OUTSIZE_THRESHOLD) provides a way to automatically copy large output copies to tape rather than DASD.

The variable copyType in the output copy options is one of the following values:

- LP—local site primary copy
- LB—local site backup copy
- RP—recovery site primary copy
- RP—recovery site backup copy

BIGDDN_copyType_DSN

Specify the name of the disk or tape data set for the specified copy type. You can use symbolic variables to construct this name. (For more information about symbolic variables, see “Output data sets, job cards, and symbolic variables” on page 51.)

BIGDDN_copyType_ENCIPHER

Specify whether to use the NGT Copy ENCIPHER syntax option when generating the JCL to make copies. COPY PLUS version 7.3 and later can make encrypted image copies to prevent unauthorized access to sensitive company information. (Encryption is a feature of the BMC Recovery Management for DB2 solution and requires a valid Recovery Management solution password.)

ENCIPHER is incompatible with DSSNAP and with compressed indexes.

BIGDDN_copyType_UNIT

Specify the name of the disk or tape unit to which the image copy data sets will be written.

BIGDDN_copyType_TAPE

Specify whether the output unit is tape or disk, as follows:
- YES—the unit is tape
- NO—the unit is disk.

BIGDDN_copyType_CATALOG

Catalog the data sets in the operating system catalog.
BIGDDN_copyType_STACK

When you are using tape, this option specifies whether to stack image copies of the same type contiguously on the same tape. Valid values are YES, NO, CABINET, and RESET.

If you are copying to a disk unit, specify NO for this value (unless making cabinet copies); otherwise you will receive an INVALID COMBINATION message.

For Recovery Management solution only - you can specify CABINET to create cabinet copies. Cabinet copies can be made to either disk or tape. For more information, see the Recovery Management for DB2 User Guide.

BIGDDN_copyType_MAX_PRIMARY

Specify the maximum amount of disk space (in the units specified by OUTPUT_copyType_ALLOCATION_TYPE) that may be allocated as primary space. Zero indicates no limit, while a nonzero value establishes an upper limit on the value calculated by PCTPRIM. Valid values are 0 through 9999.

BIGDDN_copyType_ALLOCATION_TYPE

Specify the unit of disk space to be used with the OUTPUT_copyType_MAX_PRIMARY option. CYL indicates cylinders. TRACK indicates tracks.

BIGDDN_copyType_RETENTION

Specify the tape copy data set retention period in days. The valid range is 1 through 999. Retention period and expiration date are mutually exclusive.

BIGDDN_copyType_EXPIRATION

Specify the expiration date for a tape copy data set. The date must be in the format yyyy/ddd, where yyyy is the year and ddd is the Julian day.

BIGDDN_copyType_VOL_COUNT

This option specifies the largest number of tape volumes that are expected to be created for BIGDDN copies. The valid range is 1 through 255. This option applies only to tape data sets.

BIGDDN_copyType_MODEL_DSN

Specify the fully qualified name of a cataloged data set to define as the model data control block (DCB) for BIGDDN copies. Symbolic variables are not allowed.
BIGDDN_copyType_PRIMARY_ALLOC

This option specifies the primary allocation quantity (disk only) for BIGDDN copies. Use this option only when RECOVERY MANAGER is unable to estimate the quantity.

Primary allocation values are ignored if the selected backup utility is Recovery Management. Recovery Management calculates the correct allocations.

BIGDDN_copyType_SECONDARY_ALLOC

This option specifies the secondary allocation quantity (disk only) for BIGDDN copies. Use this option only when RECOVERY MANAGER is unable to estimate the quantity.

Secondary allocation values are ignored if the selected backup utility is Recovery Management. Recovery Management calculates the correct allocations.

BIGDDN_copyType_SMS_STORAGE

This option specifies a valid SMS storage class name for disk data sets used for BIGDDN copies. The name must not exceed 8 characters. RECOVERY MANAGER forces OUTPUT_copyType_CATALOG YES when this option is specified.

BIGDDN_copyType_SMS_DATA

This option specifies a valid SMS data class name for disk data sets used for BIGDDN copies. The name must not exceed 8 characters. RECOVERY MANAGER forces OUTPUT_copyType_CATALOG YES when this option is specified.

BIGDDN_copyType_SMS_MGMT

This option specifies a valid SMS management class name for disk data sets used for BIGDDN copies. The name must not exceed 8 characters. RECOVERY MANAGER forces OUTPUT_copyType_CATALOG YES when this option is specified.

BIGDDN_copyType_ACP_GDG

Use this option to specify a fully qualified data set name to be used to define a GDG base for the BIGDDN copy. The named data set must contain the control cards that are needed to perform an IDCAMS DEFINE as well as the symbolic variable &base, which replaces the GDG base name.
**BIGDDN_copyType_MIGRATE**

Use this option to specify Hierarchical Storage Management (HSM) migration of copy data sets when Recovery Management is finished with them. MIGRATE is available only with COPY PLUS 6.4 or later.

You must specify BIGDDN_copyType_TAPE NO and specify a disk unit for the BIGDDN_copyType_UNIT option.

Valid values are as follows:

- **NO** — (the default) suppresses migration. RECOVERY MANAGER generates no MIGRATE syntax in the JCL.
- **HSM** — specifies migration to compressed disk. Make sure that you have enough space available on the disk when using this value.
- **HSM ML2** — causes immediate migration to a migration level 2 (MIGRATIONLEVEL2) volume.

**BIGDDN_copyType_EATTR**

This option specifies whether a data set supports extended attributes or not. Specifying no value for EATTR allows the value for EATTR to be set by an SMS DATACLAS.

*Note*

IBM z/OS Versions 1.11 or later support the EATTR option.

You can also set EATTR to OPT or NO in the JCL.

If an image copy was written to the cylinder-managed portion of an EAV under z/OS Version 1.11, you cannot use that image copy on z/OS Version 1.10; Version 1.10 does not support sequential data sets in the cylinder-managed portion of an EAV.

Valid values for EATTR are:

- **OPT** specifies that extended attributes are optional for the data set.
  
  You must set OPT to allocate an extended format sequential data set. By using OPT, NGT Copy supports sequential data sets in the cylinder-managed portion of EAVs.
  
  Extended format sequential data sets must be allocated on SMS-managed volumes and the size of the data set must be greater than the EAV break point, which is typically 10 cylinders.

- **NO** specifies that the data set cannot have extended attributes.
BIGDDN\_copyType\_DSSNAP

If you use NGT Copy and XBM or SUF, you can use this option to make a hardware-based Instant Snapshot copy of DB2 data.

- **YES**—indicates that a hardware copy should be made.

- **AUTO**—indicates that a hardware copy should be made if possible, but a standard copy should be made if the hardware copy fails (for example, if XBM, SUF, or the required hardware is not in place).

- **NO**—indicates that a standard copy should be made.

The following restrictions apply to this option:

- The NGT Copy and XBM or SUF products are required to use this option.

- DSSNAP is not allowed with Compressed Index.

- DSSNAP is not allowed with Copy Type INCR or AUTO.

- DSSNAP is not allowed with the ENCIPHER option.

- If you specify DSSNAP YES or AUTO, then RESETMOD must be NO.

- DSSNAP is not allowed with Online Consistent Copy.

- FULLDD is only valid with Copy Type Full Auto.

- If you specify DSSNAP YES or AUTO, then FULL\_COPY must be YES.
Index

? for subsystem ID 57, 86
#BEGINSQL 536
#ENDSQL 536

A

access, shared 820
accessing DB2 Product Configuration 744
ACF2 security 89, 747
ACKLOAD configuration option 757
ACPGDG_DATA_SET option 870
ACPLOAD configuration option 757
ACPMAIN option 868
ACPOPTIONSET configuration option 757
action codes, in object group creation 113, 289
active log
  discussion 269
  failure modes 269
  information 379
  log pairs 373
  record modification 279
  recovery procedure 270
  size 373
Active Log 1 Prefix configuration option 757
Active Log 2 Prefix configuration option 757
ACTLOG configuration option 757
ACTLOG2 configuration option 757
adding
  application group, new 113, 121
  group authorizations 128
  job card information 163, 208
  objects to an application group 122
  repository group 289
additional steplib data sets 748
AFRLOAD configuration option 757
AFRMAIN option 868
AFFOPTIONSET configuration option 757
AFTER_INIT_PHASE option 870

alias
  specifying in ARMBGEN 452
allocation type option 251, 254
allocation, max primary 251, 254
ALMCNTL configuration option 757
ALMLOAD configuration option 757
ALMOPTIONSET configuration option 757
ALPLOAD configuration option 757
ALPOPTIONSET configuration option 757
ALTER IX status 839
Alternate Archive On Tape configuration option 757
alternate recovery resources 238, 248
alternate resource selection 185, 248
ALTERNATE_ACT1 option 851
ALTERNATE_ACT2 option 852
ALTERNATE_ARC1 option 852
ALTERNATE_ARC2 option 852
ALTERNATE_CHANGE_ACCUM option 852
ALTERNATE_COPY_xx option 853
ALTERNATE_RESOURCES option 853
ALTLOAD option 452, 691, 707
ALWAYS_REBUILD_INDEXES option 847
AMRBRPR
  generating JCL for 389
ANALYZE option 183, 220
APF authorizations 90
application group
  creating 49
  repository 289
application ID 84
application recovery simulation 206, 213
applid variable 84
archive history file
  HISTONLY 357, 415
  limit 420
  limit logs 694
  shared with PACLOG 63
  used by ARMBARC program 409
  used by ARMBSRR program 681
archive log
  archive log command, ARMBLOG 312
  copies 315
  copy job 315
  discussion 273
  failure modes 273
  information 380
  password 279
  record modification 279
  recovery procedure 274
Archive Log 1 Prefix configuration option 757
Archive Log 2 Prefix configuration option 757
archive log, recovery site copies 603
archive logs, greater than 64K 65
Archive tables, grouping 477
Archive Use Timestamp in DSN configuration option 757
ARCHIVE1 option 415
ARCHIVE2 option 415
ARCHIVE3 option 415
ARCHIVE4 option 415
ARCHLOG1 option 373
ARCHLOG2 option 373
ARCIVE, pseudo-volume name 749
Arciven configuration option 755
ARCLOG1 configuration option 757
ARCLOG2 configuration option 757
ARCTSTMP configuration option 757
ARM$OPTS
  RECOVERY MANAGER option set 94
ARMAUTH
  authorization 90
  data set 518
ARMBACT program
  about 53, 403
  authorizations 403
  building the JCL 403
  data set DD statements 405
  EXEC statement 404
  executing the JCL 407
  JOB statement 404
  sample JCL 406
  sample output 406
  STEPLIB DD statement 405
ARMBARC program
  about 53, 409
  archive history file 409
  ARCHIVE1 option 415
  ARCHIVE2 option 415
  ARCHIVE3 option 415
  ARCHIVE4 option 415
  authorizations 410
  building the JCL 410
  creating archive log JCL 315
  data set DD statements 412
  DATAACLAS option 416
  DISK option 416
  DR authorizations 93
  establishing a DR recovery point 312
  EXEC statement 411
  executing the JCL 424
  EXPDT option 418
  filter options 419
  FILTERIX option 419
  FILTERRECTYPE option 419
  FILTERTS option 419
  HISTONLY option 415
  HOURS option 420
  JOB statement 411
  LIMIT option 420
  LOGS option 420
  MGMTCLAS option 416
  performance considerations 78
  PREFIX option 415
  RBARANGE option 420
  RETPD option 418
  sample JCL 421
  sample output 422
  scheduling jobs 315
  STACK option 418
  STEPLIB DD statement 412
  STORCLAS option 417
  symbolics 415
  syntax 413
  TAPE option 416
  TAPE UNIT option 418
  TRTCH option 418
  UNIT option 416
  UNITCNT option 417
  using in disaster recovery 335
  updating history file 315
  ZIIP option 418
ARMBCOR program
ARMBEOL program
  about 53
  ARMBSRR generated JCL 687
  ARMBCRC program
  about 53, 425
  authorizations 312, 426
  building the ARMBCRC JCL 426
  data set DD statements 428
  DR authorizations 93
  establishing a coordinated recovery point 426
  establishing a DR recovery point 312
  EXEC statement 427
  executing the JCL 430
  generating JCL 312
  in disaster recovery 335
  JOB statement 427
  sample JCL 429
  STEPLIB DD statement 428
  subsystem recovery point 312
  timestamp conversion 312
ARMBGEN program
  about 53, 431
  ANALYZE option 449
  application recovery JCL generation 302
  authorizations 435
  BACKOUT option 447
  batch backup JCL 164
  batch recovery JCL 208
  batch recovery job generation 210
  building the ARMBGEN JCL 436, 441
  building the ARMBGNR JCL 441
  data set DD statements 438
  disaster recovery options 302
  ESTIMATE option 447
  EXEC statement 436
  executing the JCL 463
  GDG base 296
  generating JCL 51
  in disaster recovery 211, 335, 433
  in full subsystem recovery 433
  in local recovery, full subsystem 355
  in subsystem recovery 349
  INDEX ALL option 210
  job statement 749
  JOB statement 436
  multiple job optimization 73
  online support 164, 211
  options, authorization 164, 211
  performance considerations 83
  recovering DB2 applications 309
  recovery point specification 210
  related objects 210
  sample JCL 457
  sample output 460
  SIMULATE option 446
  syntax 443
  UNCHANGED status 210
  updating group options 431, 452
  using IKJEFT1B 436
  XUNCHANGED 432, 452
ARMBGIM program 139
  about 53, 465
  authorizations 465
  building the JCL 466
  data set DD statements 467
  EXEC statement 466
  executing the JCL 471
  JOB statement 466
  sample JCL 470
  STEPLIB DD statement 467
  syntax 468
ARMBGLR program 357
ARMBGNR program
  about 53, 441
  data set DD statements 442
  EXEC statement 441
  MEMBER parameter 77
ARMBGPS program
  about 53, 347, 473
  authorizations 479
  backup strategy 349
  building groups 351
  building the JCL 479
  data set DD statements 481
  data set sizing 80
  exclusions 476
  EXEC statement 480
  executing the JCL 487
  full subsystem group creation 118
  group options 334
  in local recovery 355
  inclusions 476
  indexes 477
  JOB statement 480
  LOBs 476, 477
  STEPLIB DD statement 480
  syntax 481
ARMBGPV program
  about 53, 489
  ARMPICK 493
  ARMRCALL 493
  ARMRESRC 493
  ARMXCEPT 493
  audit recoverability 297
  authorizations 491
  BACKOUT 497
  batch revalidation program 134
  building the JCL 491
  data set DD statements 493
  disaster recovery options 302
  disaster recovery planning 490
  EXEC statement 492
  executing the JCL 505
  in disaster recovery 335
  JOB statement 492
  mirroring support 490
  object validation 50
  performance considerations 83
  pseudo-volumes 749
  sample JCL 502, 503
  STEPLIB DD statement 493
  syntax 494
ARMBGRP program
about 53, 609
authorizations 609
building the JCL 609
data set DD statements 611
EXEC statement 610
JOB statement 610
STEPLIB DD statement 611
ARMBMJO program
about 53
CLEAR_TABLE 224
report 224
restart recovery for concurrent jobs 223
syntax 224
ARMBRDRC program
about 53, 615
ARMBSRR generated JCL 687
authorizations 615
building the JCL 616
data set DD statements 617
EXEC statement 616
in disaster recovery 335
sample JCL 620
sample output 620
specifying a JOB statement 616
STEPLIB DD statement 617
syntax 618
ARMBRID program
and ARMBSRR 681
about 53, 623
ARMBSRR generated JCL 687
authorizations 623
building the JCL 623
data set DD statements 625
EXEC statement 624
executing the JCL 626
JOB statement 624
sample JCL 626
STEPLIB DD statement 625
ARMBRPR program
about 53
authorizations 630
Backout Auto Estimation Exceptions report 649
Backout Recovery Exceptions report 649
building the JCL 630
data set DD statements 632
EXEC statement 631
executing the JCL 650
Forward Recovery Exceptions report 649
JOB statement 631
Objects Changed report 642
Objects Copied report 644
Objects Not Copied report 647
Objects Not Recovered report 639
Objects Recovered report 638
Objects Unchanged report 640
overview 629
Progress Report 383
Progress reports 629
sample ARMPRINT 638
sample JCL 637
sample job member 637
STEPLIB DD statement 631
Summary report 648
syntax 634
syntax option descriptions 635
ARMBSDR program 694
sample JCL 653, 654
sample output 654
about 53
ARMBSRR generated JCL 687
authorizations 652
data set DD statements 652
executing the JCL 657
in disaster recovery 335
sample JCL 653, 654
ARMBSSET member 660, 669
ARMBSSET program
about 53, 659
authorizations 660
building the JCL 660
data set DD statements 662
EXEC statement 661
executing the JCL 671
JOB statement 661
overview 659
sample JCL 669, 670
STEPLIB DD statement 661
ARMBSRR program
about 53, 673
authorizations 320, 688
building the JCL 688
conditional restart recovery 337
creating JCL 320
CRRPOINT 677
data set DD statements 691
default allocations 97, 320
default recovery point 677
disaster recovery authorizations 93
editing JCL for DR 306
ESTIMATE option 709
executing the JCL 716
GDG base 296, 320, 332
generated programs 687
hardware mirroring 675
in disaster recovery 335
INDOUBT transactions 306
initialization phase 683
initializing DB2 subsystem 305
job failure 343, 344
job name 296
jobs 337, 678
LASTLRSN 677
LASTRBA 677
MAXCATJOBS 339
maximum catalog recovery jobs 329
MAXLOGJOBS 337, 679
MAXTAPEUNITS option 710
messages 716
MISSINGCOPIES option 709
performance considerations 78
permanently quiesced subsystems 293
Phase 1 306, 337, 683
Phase 1 jobs 679
phase 2 685
Phase 2 339
phase 2 jobs 680
recovery phase 685
restart 342
sample JCL 711, 712
sample output 712
SIMULATE option 709
specifying a JOB statement 689
specifying an EXEC statement 689
stacked tape analysis 306, 337, 679, 681
STEPLIB DD statement 690
subsystem recovery point 677
syntax 694, 722
SYSIBM.SYSPLANDEP 320
system resource recovery job 320
system validation report 712
troubleshooting 716
ZIIP option 700
ARMBSTPF program 222
about 53
ARMBSRR generated JCL 687
ARMBSYN program
synchronization steps 226
about 53
restarting Phase 2 jobs 344
ARMBTTRM program
about 53
ARMBSRR generated JCL 687
ARMBTSI program
about 53, 719
authorizations 719
building the JCL 720
data set DD statements 721
EXEC statement 720
executing the JCL 724
in disaster recovery 335
JOB statement 720
sample JCL 723
STEPLIB DD statement 721
timestamp 312
ARMBUTL program
about 53
ARMBSRR generated JCL 687
ARMBWDC program
about 53, 725
ARMBSRR generated JCL 687
authorizations 725
building the JCL 726
data set DD statements 727
EXEC statement 726
in disaster recovery 335
sample JCL 731, 732
sample output 732
specifying a JOB statement 726
STEPLIB DD statement 727
syntax 728
ARMCOPY data set 518
ARMDDL DD statement 691
ARMDEFN data set 518
ARMEXPRT data set 517
ARMIN data set 518
ARMLRNG file 356, 363, 438, 597, 599, 611
ARMMSGS data set 517
ARMOBS data set 518
ARMOPTM command 89, 747
ARMPICK output data set 493, 691
ARMPRINT data set 517
ARMQOPT DD statement 691
ARMQQUERY data set 518
ARMRCALL output data set 493
ARMRCVR data set 518
ARMRENAM data set 518
ARMRESET member 70
ARMRESRC output data set 493
ARMRSTOR DD 493
ARMSBGEN
program 227
restart macro 227
ARMSQL data set 509, 518
ARMTRACE 845
ARMUMAN command 89, 747
ARMUSEL command 89, 747
ARMVRPT DD 691
ARMVRPT file 699
ARMWPEND file 438, 609, 691
ARMXCEPT file 493
audit recoverability 297
authorization
adding 91
APF 90
ARMAUTH 90
ARMBARC 93
ARMBCRC 93
ARMBGPS 479
ARMBGRP 515
ARMBLGR 598
ARMBLRD 609
ARMBSRR 93
DB2 plan 90
delete/redefine options 93
disaster recovery 93
for ARMBGEN 164
for batch recovery JCL 211
group 90, 128
PUBLIC 91
RECOVERY MANAGER 89
retaining when copying groups 530
revoking 91
scenarios 92
subsystem options 93
system resource 92
to access system resources 266
to change, for a group 128
type A 91, 128
type O 91, 128
authorizations
mechanisms, description 98
AUTOSIZE 183
AUTOSIZE option 238

B
backing up BMC tables 813
backout recovery
BACKOUT AUTO 201
BACKOUT option 218, 447, 497
backout recovery, about 201
backout to forward recovery strategy 447
space status requirements 201, 434
backup
batch job generation 164
online job generation 163
strategy, subsystem recovery 349
strategy, using object sets 143
backup options
about 157
browse, update, delete 158
catalog and directory 266
DB2 COPY 180
discussion 94
DSNUMUTILB 180
general 167
NGT Copy 169
NGT Recover, OUTCOPY 183
output copy data set 186
symbolic variables 191
backup time estimates
saving 394
viewing 390, 394
BAD PART status 839
BAD TYPE status 220, 840
BADDSNUM status 839
BADSHRL status 840
batch backup job generation
using ARMBGEN 164
using online support 164
batch log range analysis 357, 597
batch recovery job generation
using ARMBGEN 210
using online support 210
batch revalidation reports 134
batch volume group creation 118
benefits, of RECOVERY MANAGER 45
BIGDDN options
ACP_GDG 897
ALLOCATION_TYPE 897
CATALOG 897
DSN 897
DSSNAP 897
EATTR 897
EXPIRATION 897
MAX_PRIMARY 897
MIGRATE 897
MODEL_DSN 897
PRIMARY_ALLOC 897
RETENTION 897
SECONDARY_ALLOC 897
SMS_DATA 897
SMS_MGMT 897
SMS_STORAGE 897
STACK 897
TAPE 897
UNIT 897
VOL_COUNT 897
BIGDDN syntax 186
BINDQUALIFIER configuration option 757
BLKALLOC option 847
BMC DB2 Component Services 57, 86
BMC Infrastructure Load 2 configuration option 757
BMC Infrastructure Load configuration option 757
BMC tables 767
BMC utilities
displaying status 814
running concurrently 825
terminating 814
BMCARM.BMCRMD_RECOVERY_EXCEPTIONS table 807
BMCDICT table
considerations 816
contents 816
maintaining 816
BMCHIST installation option, BMCHIST table 817
BMCHIST table
backing up 813
contents 817
maintenance 820
NGT Copy considerations 819
NGT Recover considerations 819
querying 814
BMCLGRNX
performance considerations 70
table 64
BMCSTATS table 80, 233
BMCSYNC
BMCSYNC table 64
BMCSYNC table
backing up 813
contents 817
maintenance 820
NGT Recover UNLOADKEYS 825
considerations 823
contents 820
LOB data considerations 823
maintaining 824
running utilities concurrently 825
XML data considerations 823
BMCTRANS table 828
BMCUTIL
table 64
BMCUTIL table
backing up 813
contents 829
maintaining 832
BMCXCOPY
table 64
BMCXCOPY table
backing up 813
contents 832
maintaining 837
querying 814
BSDS
days of log data in 373
log inventory JCL 281, 283
maintenance 279
number of entries in 373
reallocate 278
reallocation 278, 282
recovery 277, 278
recovery and maintenance 277
BSDS 1 configuration option 757
BSDS 2 configuration option 757
BSDS ARCHLIMIT option 703
BSDS DAYS LIMIT option 703
BSDS HOURS LIMIT option 704
building a new table space or index group 113, 121
BY PART option 543
BYPART 532, 540
BYPART option 542, 543
BYPART YES/NO 539
BYPASS DEACT option 708, 722
BYPASS QUIESCED option 708, 722

C
CA ACF2 security 89, 747
CA-ACF2 security product 98
CA-Top Secret security product 98
cabinet copies
  setting backup options 162
  specifying BIGDDN output 897
  specifying copy output 888
  specifying FULLDDN output 892
  specifying recover output 864
Calculation options, Progress Report 395
capacity of typical DASD devices 251
capacity, DASD devices 862
CAREP option 728
catalog & directory
  recovering 78
catalog and directory recovery
discussion 266
  options 267
  recovery procedure 267
catalog option 186, 254
catalog recovery 78
catalog search method, for ARMBGRP 118, 509
CATDIR option 728
Change Accum on Tape configuration option 757
to the product 25
CHECK command, ARMBSET program 659
check pending
  action 233
  status 107
check pending status 107
Check Plus Load configuration option 757
CHECK_PEND_ACTION option 847
CHECK_UTILITY option 847
CHECKERROR option 870
checkpoint maintenance 279
CHECKPOINT option 238, 854
CHECKTSLVERSION option 870
Ci column 122
CISIZE, for temporary workfiles 285
CLIST for product execution 84
clones
  Ci column on Object List panel 122
delta subsystem recovery 351
in ARMBGEN 451
in ARMBGPV 497
in ARMBGRP 530
local system recovery 357
SHOWCI command 122
specifying for batch group reports 136
specifying for recovery 220
specifying for subsystem backup 351
specifying for subsystem recovery 357
specifying for volume groups 119
CLONES ONLY 530
CLONES ONLY option 451, 497, 530
Command line commands in DB2 Product
  Configuration 738
common DB2 repository tables 767
common repository
  naming conventions 767
COMPARE ACTUAL ESTIMATE command 618
compressed indexes
  with cabinet copies 162
  with COPY IMAGECOPY 170, 870
  with DSSNAP 170
  with ENCIPHER 186
  with encryption 186
  with Instant Snapshots 196
compression
  BMCDICT table 816
  compression ratio 373
concurrent copy 213
CONCURRENT option 170
conditional restart
  ARMBTSI 719
  recovery 425
  timestamp insertion 719
conditional restart recovery jobs 337
CONDRESTART option 705
configuration option 96
  alphabetical table listing 750
configuration options 755

Public Plan 757
Active Log 1 Prefix 757
Active Log 2 Prefix 757
Alternate Archive On Tape 757
Archive Log 1 Prefix 757
Archive Log 2 Prefix 757
Archive Use Timestamp in DSN 757
Arcive 1, 2, or 3 755
BMC Infrastructure Load 757
BMC Infrastructure Load 2 757
BSDS 1 757
BSDS 2 757
Change Accum on Tape 757
Check Plus Load 757
Copy Archive Timestamp 757
Data Collection 757
data sharing member 757
DS Member 757
DSNEXIT 757
DSNLOAD 757
History File 757
ISPMLIB 755
ISPTLIB 755
JCL Output 757
JES2 Name 757
JES3 Name 757
Job Card Options 1 through 5 755
Log Master Optionset 757
Log Range File 757
Logrange for Common Points 755
NGT Copy Load 757
NGT Copy Optionset 757
NGT Recover Load 757
NGT Recover Optionset 757
Paclog CNTL 757
Paclog Optionset 757
Percent Prime 755
Recovery Site Del/Def 757
SMS VCAT 755
Steplib Addition 757
Steplib Override 757
subsystem 757
Truncation Characters 757
Truncation Position 757
Use CATALOG Parm 757
User Joblib 757
VCAT 757
Work File Database 757
Work File Prefix 757
Work Unit 755
zIIP Redirection 757
conventions, documentation 24
coordinated recovery 425
copy after recovery options 233
Copy All Indexes option 167
Copy Archive Timestamp configuration option 757
Copy Index Spaces option 167
COPY INDEXES option 477
COPY INDEXSPACES option 477
copy options 254
copy pending status 107
copy registration
  BMCXCOPY table 832
copy utility option 233
COPY_AFTER_xx option 847
COPY_IMAGECOPY option 870
COPY_INDEX_SPACES option 483, 868
COPY_UTILITY option 847, 868
COPYTYPE option 705
CRCR maintenance 279
CREATE GROUP command 507, 530
CREATE VOLUME GROUP command 544
CREATEHIST option 703
CRRDRPT table 719, 773
CRRDRPT, recovery point 312
CRRPOINT 677

DB2 long names
  displaying 87, 111
  setting options 87, 111
DB2 Product Configuration
  accessing 744
  Command line commands 738
  drop-down menus 737
  input panels 741
  interface tools 740
  interview overview 734
  menu bar 737
  option set names 742
  Product Option Sets panel 734
  sections 734
DB2 Product Configuration technology 63
DB2 RECOVER options 249
DB2 requirement 57
DB2 statistics 79
DB2 subsystem status 266
DB2WRITE option 728
DBC 57, 86, 733
DBC started task 89
DCNAME option 728
DCTOKEN token 618, 710
DDF maintenance 279
deactivate for data sharing members 278
DEFER status 840
definition types 126
DELETE GROUP option 562
delete STOGROUP option 93, 233
DELETE_STOGROUP_OBJ 847
deleting
  objects 112
DESCRIPTION 530
description change, group 129
destroy for data sharing members 278
Detail rows, Progress Report 395
Detail rows, Progress Report
  objects 395
DFSMS concurrent copy 213
diagnostic messages, recovery 238
DIAGNOSTIC_MESSAGES 854
dictionaries, compression
  BMCDICT table 816
disaster recovery

DASD capacity 862
data collection
  ARMBRDC program 615
  ARMBWDC program 725
Data Collection configuration option 757
data collection jobs 341
data set recall list 136
data set recall reports 136
data set sizing 80
  performance 80
  recovery option 233
data sharing considerations, disaster recovery 293
data sharing members
  deactivate, destroy, restore 278
Database, Progress Report 395
DATACLAS option 416
DATACOLLECTION configuration option 757
DATASET_SIZING option 860
DB2 Component Services 57, 86
DB2 COPY, backup options 180
archive copy to use offsite 326
archive log copies 315
Archives Cataloged 326
ARMBGEN program 211
ARMBSRR program 673
authorizations 93
BDS log processing limit 326
conditional restart jobs 337
coordinated 425
create archive log 311
data sharing considerations 293
establish a recovery point 311
estimation 295
group naming conventions 293
initialize active logs 329
jobs 683
local site preparations 335
MAXCAT Recovery Jobs 325
maximum log jobs per member 330
performing at the recovery site 305
permanently quiesced subsystems 293
preparing system resources 310
recommendations 320
recover ChgAccum repository 325
recover RM repository 330
recovery point 677
recovery simulation 674, 675
recovery site procedures 306
rerun Phase 2 jobs 343
restart Phase 2 jobs 344
restarting jobs 342
restarting Phase 1 343
restore archive copies to disk 331
simulation, application 206
simulation, system resources 294
steps for preparation 297
system resource recovery 320
system resource recovery options 317, 325
using ARMBGEN 433
DISK option 416
disk options, disaster recovery 317
displaying status of BMC utilities 814
documentation information 23
DREXTEND option 706
drop-down menus in DB2 Product Configuration 737
DS LEVEL status 840
DS Member configuration option 757
DSNDB01 option 728
DSNEXIT configuration option 757
DSNJU004 283
DSNJU004 command 89, 747
DSNLOAD configuration option 757
DSNTEP2 dataSetName option 711
DSNUM 533, 539, 540
DSNUTILB
backup options 180
recover options 249
site type 249
DSNUTILB options
COPY_PARALLEL 881
FULL_CONCURRENT 881
FULL_COPY 881
FULL_FULL_PERCENT 881
FULL_INCREMENTAL_PERCENT 881
GROUP 881
KEYCARD 860
MAX_PARALLEL 881
REPORT 860
SHRLEVEL 881
SITE_TYPE 860
SORTKEYS 860
STATISTICS 860
UPDATE 860
DSSNAP option 870
dynamic exception status
creating a group 113
Dynamic sortworks option 238
DYNAMIC SORTWORKS option 854

E
EARLYCAT option 238, 854
EARLYRECALL option 238, 854
EATTR option 186, 254
EBCDIC, translation from Unicode 111, 508
EDIT macro, ARMSBGEN 227
ENCIPHER option 186, 888, 892, 897
cryptography options 186, 888, 892, 897
COPY_ALL_INDEX 870
ESTIMATE option
ARMBGEN 447
ARMBSRR 709
estimation, system resource recovery 295
exception status
object 221
object specification 107
types of 839
EXCLUDE 534, 537, 539, 541, 543
EXCLUDE DSNUM 539, 541
EXCLUDE MEMBERS option 708, 722
EXCLUDE option 542
EXCLUDEDSNUM 534
EXCLUDEGROUP option 543
EXCLUDEIX 534, 538
EXCLUDEPARTS 533, 539
EXECUTE authority 90
execution CLIST 84
EXPDT option 418
expiration date 254
EXPORT GROUP command 261, 517
EXPORT GROUP option 568
extended address volumes 186, 254

F

filter options, ARMBARC 419
FILTERIX option 419
FILTERRECTYPE option 419
FILTERNS option 419
Full copy only option 136, 216
full subsystem
   ARMBGPS program 118
   group creation 118
   local recovery procedures 355
   recovery 347
full subsystem recovery 425
   using 433
FULL_AUTO_READ_PERCENT option 870
FULL_COPY option 870
FULL_CUMULATIVE option 870
FULL_DAY_OF_WEEK option 870
FULL_EMPTY option 870
FULL_FULL_PERCENT option 870
FULL_INCREMENTAL_PERCENT option 870
FULL_KEEP_PREVIOUS option 870
FULL_MAX_INCREMENTALS option 870
FULL_MIN_PAGES option 870
FULL_NACTIVE option 870
FULL_READTYPE option 870
FULLDDN 186
FULLDDN options

ACP_GDG 892
ALLOCATION_TYPE 892
CATALOG 892
DSN 892
DSSNAP 892
EATTR 892
EXPIRATION 892
MAX_PRIMARY 892
MODEL_DSN 892
PRIMARY_ALLOC 892
RETENTION 892
SECONDARY_ALLOC 892
SMS_DATA 892
SMS_MGMT 892
SMS_STORAGE 892
STACK 892
TAPE 892
UNIT 892
VOL_COUNT 892
function key display 87

G

GDG output data set, simulation 674, 675
general backup options
   Copy All Indexes 167
   Copy Index Spaces 167
   Copy utility 167
   Index size threshold 167
   QUIESCE WRITE 167
   Region Size 167, 233
   Scope 167
   table 167
general recovery options 233
generating JCL for ARMBRPR 389
GENJCL statement 452
global temporary tables 64
GPMEMBER configuration option 757
group
group attach names 64
authorizations 90, 128
backup and recovery JCL 431
backup authorizations 158
backup options 158
by volume 473
changing the description 129
changing the name 129
creating and managing 143
definitions 126
deleting groups in batch 513
excluding spaces by partition 539
exporting and migrating 261
group authorizations table 772
group creation 507
group maintenance 507
group renaming 511
impact analysis 139
including spaces by partition 541–543
interactive vs. batch creation 83
interactive vs. batch revalidation 83
naming 90
naming conventions
data sharing groups 293
XCF considerations 293
querying groups in batch 514
reporting on groups in batch 513
revalidation
batch 133
interactive 133
specification 107
updating groups in batch 512
Group name, Progress Report 395
GROUP option 870
Group owner, Progress Report 395
group revalidation reports 136
GROUPAUTH table 772
groups, large 122, 157, 229
GRPOPTS table 771

H

hardware compression
BMCDICT table 816
hardware mirroring 327, 675
Help panels 101
help, online 100
high RBA maintenance 279
HIST configuration option 757
HISTONLY 357
HISTONLY option 415
history (versioning) objects, inclusion in groups 132
History File configuration option 757
HISTORY installation option
BMCHIST table 817
History tables, grouping 477
history, recovery 615
HISTRETN installation option
BMCHIST table 819
HOURS option 420
HSM migration option 888, 892, 897
HSM ML2 option 186
HSM option 186
HWCOPY option 707
HWLEVEL option 707

I

I/O error
with both dual logs 271
with dual log 271
with single log 271
IDCAMS LISTCAT
about 286
generating a job 287
IDCAMSCAT configuration option 757
identifier types in name strings 112
IKJEFT1B 436
image copy, after recovery 233
impact analysis 465
impact analysis and reporting
background 139
current group 139
foreground 139
Include Clones field 357
Include indexes, Progress Report 395
INCLUDEARCHIVE 535
INCLUDEHISTORY 535, 537, 541, 543
INCLUDEHISTORY option 542
INCLUDEIX 534, 537, 541, 543
INCLUDEIX option 542
INCLUDELOB 535, 537, 541, 543
INCLUDELOB option 542
INCLUDER 534, 537
INCLUDERI 541, 543
INCLUDERI option 542
INCLUDEXML 535, 537, 541, 543
INCLUDEXML option 542
incremental index copies 193
INDEX ALL 82
INDEX ALL option 208, 210, 233
index copies, incremental 193
Index size threshold option 167
Index Term 261, 437, 442
INDEX ALL option 847
INDEX_SIZE_THRESHOLD option 483, 868
INDEX_SIZE_THRESHOLD_TYPE option 483, 868
indexes
   inclusion in object list 132
   INDEX ALL 208
   key sorts 230
   options in ARMBGPS groups 477
indexes, including in group 539, 540, 542
INDEXLOG AUTO option 477
indoubt threads, ARMBRID program 623
INDOUBT transactions, disaster recovery 306
inflight resolution option 497
INFLIGHT status 840
infights, option to resolve 450
INFRASTRUCT_LOAD configuration option 757
INFRASTRUCT_LOAD2 configuration option 757
infrastructure load library 748, 757
initialization phase 673
INITIALIZE ACTIVES option 702
input panels in DB2 Product Configuration 741
installation options
   BMCHIST 817
   HISTORY 817
   HISTRETN 819
Installation System 63
installing a subsystem 63
Instant Snapshots
   allocation 194
   registration 195
   restrictions 196
   with DSNUM 197
   with other BMC utilities 198
   with SHRLEVEL 197
interface tools in DB2 Product Configuration 740
INVRECPT status 841
ISPBKUP DD 438
ISPF requirement 57
ISPMLIB configuration option 755
ISPTLIB configuraitoin 755
IX_PART table 799
IXSORT table 805
IXSIZE option 477
IXSZET option 477

J

JCARD 52
JCL
   catalog/directory recovery 267
   generation 51, 163, 208
   optimization 206
   performance factors 206
JCL Output configuration option 757
JCL, interactive vs. batch generation 83
JCLOUT 52
JCLOUT configuration option 757
JCLTYPE option 450, 497, 699
JES support
   about 676
   enabling 676
   job routing cards 676
JES2 Name configuration option 757
JES3 Name configuration option 757
job card
   information 163, 208
   specification 52
Job Card configuration option 755
job restart 166
job set, about 226
job statement, ARMBGEN 749
JOB table 776
JOB_RESTART table 806

K

key sorts, maximizing concurrency 230
KEYCARD option 249
keys used by RECOVERY MANAGER 87
KEYSORT table 796
KSORTSHARE option 855

L

large groups 122, 157, 229
LAST ACTUAL command 618
LAST ESTIMATE command 618
LAST SIMULATE command 618
LASTLRSN 677
LASTRBA 677
level, utility option setting 229
LIKE 530, 532, 536, 538
LIMIT DAYS option 710
LIMIT HOURS option 710
LIMIT LOGS option 710
LIMIT option 420
LIMIT_SYSCOPY_SEARCH option 847
LOB data
   BMCSYNC table considerations 823
LOB spaces, grouping 476, 477
LOBs, inclusion in groups 132
local recovery procedures, full subsystem 355
local site recovery simulation 357
log
   active 379
   archive 380
   RBA maintenance 279
   recovery 269, 270, 273, 274
   synchronization 382
LOG GONE status 497
log inventory JCL 281, 283
log mark recovery 206
Log Master Optionset configuration option 757
log range analysis 357, 597
Log Range File configuration option 757
log range file, redefining 597
log range formatting 609
logging environment modeling tool
   about 369
   active log 379
   active log pairs 373
   active log size 373
   archive log 380
   audit synchronization 382
   compression ratio 373
   days of log data in BSDS 373
   entries in BSDS 373
   features 369
   hours of log on DASD 373
   logging rate 373
   model option 370
   modeling statistics 370
   optimize fields for DASD archives 373
   optimizing 370
   option descriptions 373
   source of highest logging rate 373
   viewing statistics 370
logging rate 373
logical page list 107
LOGONLY option 452
Logrange for Common Points configuration option 755
LOGREST option 728
LOGS option 420
LOGSCAN option 238, 855
   long IDs 112
   long names
      configuration option 757
      displaying 87, 111
      Log Master options 88
      setting options 88, 757
      truncation 88, 757
      truncation sample 88, 757
      zoom 87
LRNG configuration option 757
Lvl field 229

M
   maintaining common utility tables 813
   max concurrent jobs option 302
   MAX_CONCURRENT_JOBS option 847
   MAX_TASK2 option 870
   MAX_TASKS option 870
   MAXCATJOBS option 78, 680, 704
   MAXDRIVES 183
   maximizing concurrency of key sorts 230
   Maximum key sorts option 230
   Maximum Key Sorts option 238
   maximum primary allocation 254
   MAXKSORT option 238, 855
   MAXLOGJOBS
      in conditional restart recovery 337
      stacked tape analysis 306, 337
   MAXLOGJOBS option 679, 702
   MAXLOGS option 238, 856
   MAXLSORT option 856
   MAXPRIM 183
   MAXPRIM option 238
   MAXTAPEUNITS option
   ARMBSSR 710
   MAXTASKS option 170
   MEMBER parameter 77
   menu bar in DB2 Product Configuration 737
   merging
      groups 107
      object lists 112
   messages
      help for 101
      types 101
   MGMTCLAS option 416
MIGRAT pseudo-volume name 749
Migrate copy data sets option 186
MIGRATE option 186
MIGRATE_TO_SSID command 261
MIRROR option 847
mirroring support
   ARMBGPV 490
   JCLTYPE 699
mirroring, hardware 327, 675
MISSINGCOPIES option
   ARMBSSR 709
model data set name 186, 254
modeling tool. See logging environment modeling tool 369
MODEQ parameter 604
MSGLEVEL option 856
multiple job optimization 71
MVS
   resource information 749
   short IDs 112

name change, group 129
names of common utility tables, determining 814
naming a group 90
naming conventions
   common repository 767
NEWWNAME option 561
NGT Copy
   backup options 169
   BIGDDN 186
   FULLDDN 186
   SHRLEVEL 170
NGT Copy Load configuration option 757
NGT Copy Optionset configuration option 757
NGT Recover
   AUTOSIZE 238
   MAXKSORT 238
   MAXPRIM 238
   OUTCOPY 183
   recovery options 238
NGT Recover Load configuration option 757
NGT Recover Optionset configuration option 757
NGT Recover, multiple job optimization 73
NOCOPIES status 841
NOT LOGGED recovery 204
NOTAVAIL status 220, 841
NOTCLONED status 841
NOTCTLG status 497
NOTDEFND status 131, 841
NUMBER_READ/ WRITE_BUFFERS option 870

O
object
   exception status 221
   object group 112
   object list 132
   recoverability 132
   specifying by exception status 107
   status 131, 839
   status, unsatisfactory 839
   validation 130
object validation 50
objects
   retaining when copying groups 531, 532, 536, 538
objects set
   creating and managing 143
OBJECTSET option 169, 200, 452
OBJSET_DEF table 768
OBJSET_SQL table 771
OBJSETS table 767
OK status 839, 845
OMVS segment 89
ON DUPLICATEDS 170
ON ERROR BADSTATUS 170
ON ERROR CONTINUE 238
ON ERROR ICEXISTS 170
ON ERROR NOTSUPPORTED 170
ON_DUPLICATEDS option 870
ON_ERROR_BADSTATUS option 870
ON_ERROR_CONTINU 857
ON_ERROR_IEXISTS option 870
ON_ERROR_NOTSUPPORTED option 870
Online Consistent Copy, setting backup options 161
optimization
   logging environment 369, 370
   recovery JCL 206
OPTIMIZE_FOR option 858
option set
   BMC utilities 748
   data sharing systems, updating 94
   RECOVERY MANAGER 94
option set names in DB2 Product Configuration 742
option sets 63
options
ANALYZE 183, 220
catalog 254
CHECKPOINT 238
EARLYCAT 238
EARLYRECALL 238
group recovery 125
object recovery 125
OUTCOPY 238
OUTSIZE 170
recovery

about 229
alternate resources 238, 248
always rebuild indexes 233
archive copy to use offsite 326
archives cataloged 326
BSDS log processing limit 326
check pending action 233
check utility 233
checkpoint 238
copy after 233
copy utility 233
data set sizing 254
delete STOGROUP 233
deleting 231
diagnostic messages 238
DSNUTILB 249
DSNUTILB site type 249
dynamic sortworks 238
EARLYCAT 238
EARLYRECALL 238
general options 233
INDEX ALL 233
initialize active logs 329
KSORTSHARE 238
Library that contains DSNTEP2 330
limit SYSCOPY search 233
LOGSCAN 238
Lvl 229
max primary allocation 254
MAXCAT 325
maximum log jobs per member 330
MAXLOGS 238
MAXLSORT 238
OUTCOPY 238
output data set options 254
recover ChgAccum repository 325
recover RM repository 330
recover utility 233
Recover Work File Database 330
redefine VCAT objects 233
restore archive copies to disk 331
REUSE 233
secondary allocation 251
SMS data class 254
SMS management class 254
SMS storage class 254
SORTKEYS 249
STATISTICS 249
synchronization file name 331
UNLOADKEYS/BUILDINDEX 238
ACP_GDG 888
ALLOCATION_TYPE 888
CATALOG 888
DSN 888
DSSNAP 888
EATTR 888
ENCIPHER 888, 892, 897
EXPIRATION 888
MAX_PRIMARY 888
MIGRATE 888, 892
MODEL_DSN 888
PRIMARY_ALLOC 888
RETENTION 888
SECONDARY_ALLOC 888
SMS_DATA 888
SMS_MGMT 888
SMS_STORAGE 888
STACK 888
TAPE 888
UNIT 888
VOL_COUNT 888
OUTPUT_TYPE option 868
OUTSIZE option 170
OUTSIZE_THRESHOLD option 870
override data sets 748
overview, RMGR tasks 62

P

package impact analysis 139
PACLOG
  compression ratio 373
cylinder calculation 373
cylinders saved by 373
features shared with RECOVERY MANAGER 63
libraries 320
Paclog CNTL configuration option 757
PACLOG option, in ARMBFR 701
Paclog Optionset configuration option 757
panel Help 101
partition expansion 132
partition spaces, excluding from group 539
partition spaces, including in group 540–543
partition split 80
password maintenance
  archive log password 279
  password deletion 279
system data set password specification 279
PEND DDL status 841
Percent Prime configuration option 755
performance considerations
  BMCLGRNX and RUNSTATS 70
  factors affecting recovery time 70
INDEX ALL recovery 82
interactive vs. batch group creation 83
interactive vs. batch JCL generation 83
interactive vs. batch revalidation 83
multiple job optimization 71, 73, 74
repository 70
RUNSTATS 70
SYSIBM.SYSCOPY searches 70
performance, enhancing 79
PFSHOW 87
Phase 2 306
PHASE table 785
PHASE1 option 728
PHASE2 option 728
pick list 691
pick list report, ARMBGPV 493
plan and package impact analysis 465
plan authorization 90
plan impact analysis 139
plan specification 107
PLANA configuration option 757
PREFIX option 415
preparing for disaster recovery 335
Primary Archive On Tape configuration option 757
PRIMEALLOC option 708
printing
  batch revalidation reports 134
  log map 283
  population reports 134
  recovery plan 220, 446
procedure
active log recovery 270
archive log recovery 274
BSDS reallocation 278
BSDS recovery 278
building a new object group 113, 121
catalog and directory recovery 267
changing group authorizations 128
creating JCL for batch revalidation 135
generating an ARMBGEN job 211
generating an impact report 139
printing the log map 284
recovering a volume group 119
system resource recovery 320
updating group recovery options 231
using the DSNJU004 utility 284
Process option 98
PROCESS RECOVERY command 618
Processing mode, Progress Report 395
processing object lists 132
PRODREG table 772
product changes 25
Product Option Sets panel in DB2 Product
  Configuration 734
programs, in RMGR 53
Progress Report
accessing online 385
backup time estimates 390
Calculation options 395
Database 395
detail rows 395
entry field descriptions 395
general information 383
Group name 395
Group owner 395
include indexes 395
objects changed 642
objects copied 644
objects not copied 647
objects not recovered 639
objects recovered 638
objects unchanged 640
Processing mode 395
recover time estimates 392
recovered, changed, and copied objects 386
Recovery point 395
reported information 383
saving backup and recover time estimates 394
Start time 395
summary 648
tablespace 395
total bytes 395
viewing backup and recover time estimates 394
Pseudo-volumes 749
PUBLIC authorization 91
Public Plan configuration option 757
publications, related 23
PUBLICPLAN configuration option 757
QUERY command 514
QUIESCE WRITE option 167, 665
QUIESCE_AFTER option 868
QUIESCE_BEFORE option 868
QUIESCE_GROUP option 868
QUIESCE_WRITE option 868
quiesced subsystems 293

security exit 98
RBARANGE option 420
reallocate BSDS 278
REBUILD_SCOPE option 573
recall report, ARMBGPV 493
recall reports 136
RECOVER INDEXES ALL option 477
recover indoubt threads 623, 659
recover pending status 107
recover time estimates
  saving 394
  viewing 392, 394
RECOVER WORKFILE option 711
RECOVER_OUTPUT option
  ALLOC_TYPE 864
  CATALOG 864
  DSN 864
  DSN_FOR_REC_PLUS 864
  EATTR 864
  EXPIRATION 864
  MAX_PRIMARY 864
  MODEL_DSN 864
  PRIMARY_ALLOC 864
  RETENTION 864
  SECONDARY_ALLOC 864
  SMS_DATA 864
  SMS_MGMT 864
  SMS_STORAGE 864
  STACK 864
  TAPE 864
  UNIT 864
  VOL_COUNT 864
RECOVER_SCOPE option 571
RECOVER Utility option 847
recoverability report 136
recoverability report, ARMBGPV 493
recoverability reports 136
recovered, changed, and copied objects
  viewing 386
recovering/maintaining system resources 265
recovery

R

R+/CHANGE ACCUM repository 288
RACF
  OMVS segment 89
RACF (IBM Resource Access Control Facility)
about 199
actions on objects 220
active log 270
analyze 220
application objects 199
archive log 274
ARMBGEN 211
backout 218
batch JCL generation 211
batch job generation 210
BSDS 278
catalog and directory 267
catalog and directory 267
full copy only 136, 216
group 199
indexes 208
interactive job generation 208
job generation, batch 208
job generation, online 208
job preparation 199
job submission 221
key sorts 230
log only 219
object exception status 221
options
about 229
alternate resources 238, 248
always rebuild indexes 233
authorizations 93
check pending action 233
check utility 233
CHECKPOINT 238
copy after 233
copy utility 233
data set sizing 233
delete STOGROUP 233
deleting 231
diagnostic messages 238
discussion 94
DSNUTILB 249
DSNUTILB site type 249
dynamic sortworks 238
EARLYCAT 238
EARLYRECALL 238
general 233
group and object 125
hierarchy 94
INDEX ALL 233
Limit SYSCOPY search 233
LOGSCAN 238
Lvl setting 229
max primary allocation 254
MAXLOGS 238
NGT Recover 238
OUTCOPY 238
output data set options 254
primary allocation 251
Recover utility 233
Redefine vcat objects 233
REUSE 233
rules of precedence 95
SMS data class 254
SMS management class 254
SMS storage class 254
SORTKEYS 249
STATISTICS 249
subsystem 97
UNLOADKEYS/BUILDINDEX 238
updating 231
options for JCL 213
points 210
repository 288
resource selection 185, 238, 248
restarting a job set 223
restarting failed jobs 222
restarting single job 222
restarting synchronized jobs 227
strategy, using object sets 143
system resources 50, 68
to a quiesce point 213
to an image copy 213
to an RBA 213
to commonpoint 213
to current 213
to image copy 213
to quiesce 213
to RBA 213
XUNCHANGED option 210
recovery data collection report 615
recovery management solutions 46
Recovery Management
mirror revalidation 490
service level agreement options 98, 294
RECOVERY MANAGER (for DB2)
  Main Menu 85
  benefits 45
  concepts and functionality 46
  first time users 61
  option set 94
  repository 66, 288, 289
  software requirements 57
  task overview 62
RECOVERY MANAGER for DB2
  adding a subsystem 63
  and PACLOG 63
  authorizations 89
  help system 100
  keys 87
RECOVERY MANAGER log range file 356, 363, 432, 438
recovery point selection
  subsystem 677
  check unchanged 452
Recovery point, Progress Report 395
recovery resources report 136
recovery restart for concurrent jobs 223
recovery simulation 674, 675
  application spaces 206, 213
  for application objects 206, 217
  for system resources 294
recovery site copies, archive log 603
Recovery Site Del/Def configuration option 757
recovery time 70
recovery, backout 447
Redefine VCAT option 93
REDEFINE_VCAT_OBJ option 847
REDO option 599
referential integrity 132
referential integrity, including in group 534, 537, 540, 542, 543
REGION_SIZE option 847, 868
related publications 23
Remote Site has single LPAR option 331
RENAME GROUP command 511
RENAME GROUP option 561
REPAIR command, ARMBSET program 659
REPLACE 530
replacing objects 112
REPORT GROUP command 513
REPORT GROUP option 563
REPORT_STATISTICS option 870
REPORTARCHIVE option 451
REPORTHISTORY option 451
REPORTIX option 451
REPORTLOBS option 451
REPORTTRI option 451
reports
  batch revalidation 134
  data set recall 136
  data set recall list 136
  pick list report 493
  recall 136
  recall report 493
  recoverability 136
  recoverability report 493
  recovery resources 136
  resources 136
  resources report 493
reports, group revalidation 136
REPORTXML option 451
repository
  backups 350
  recovery 288
  recovery procedure 289
  RMGR 66
  tables 773
REPOSITORY option 510
repository tables 767
RESET option 512, 559
RESET_GRECP_LPL option 668
RESETMOD option 870
RESOLVE_INFLIGHTS option 450, 497
resources report, ARMBGPV 493
about 64
SCCAUTH program, authorization 90
Scope option 167
SCOPE option 868
sections in DB2 Product Configuration 734
security
mechanisms 98
SEPARATE_BY_PARTITION option 870
service level agreement options 98, 294
short IDs 112
SHOWCI command 122
SHRLEVEL 170, 820
SHRLEVEL option 870
simulate DR recovery option 217
SIMULATE option
ARMBGEN 446
ARMBSRR 709
Simulate Recovery option 357
simulation
about, application 206
application recovery 217
system resource recovery 294
simulation, recovery 674, 675
SINGLE LPAR option 700
SITETYPE RECOVERY 297
SMS options 186, 254
SMS VCAT configuration option 755
SNAP option 859, 870
software requirements 57
solution common code
about 64
maintenance 846
SORTDEVT 183
SORTKEYS option 249
sortworks, dynamic 238
source of highest logging rate 373
specifying objects
about 107
by exception status 107
by group name/pattern 107
by MVS volume name 107
by plan name/pattern 107
by storage group name/pattern 107
by table space name/pattern 107
using wildcard patterns 112
specifying objects using wildcard patterns 102
Split by partition 80
SQL
ARMSQL data set for batch 518
creating a group 113, 509
example JCL for group 577, 585
example output 577
example REPORT GROUP output 585
object specification 107
VIA SQL syntax for batch 535

SQL ID 86
SQL statements
  deleting rows from the BMCDICT table 816
  deleting rows from the BMCHIST table 820
  deleting rows from the BMCSYNC table 824, 832
  deleting rows from the BMCSYNC table for RECOVER UNLOADKEYS 825
  deleting rows from the BMCXCOPY table 837
  displaying BMC utilities 814
  querying BMCHIST table 814
  querying BMCXCOPY table 814
  terminating BMC utilities 814
SQUEEZE option 870
stack option 186
STACK option 418
stacked tape analysis 306, 337, 679, 681, 845
stacked tape options 186
Start time, Progress Report 395
START_MESSAGE option 870
starting the product 84
STATISTICS option 249
status, BMC utilities 814
status, of DB2 objects 842, 843
advisory rebuild pending 107
advisory reorg pending 107
ALTER IX 839
auxiliary check pending 107
auxiliary warning 107
BAD PART 839
BAD TYPE 220, 840
BADDSNUM 839
BADSHRL 840
copy pending (CHKP) 107
copy pending (COPY) 107
DEFER 840
DS LEVEL 840
error range detected 107
group recover pending 107
INFLIGHT 840
informational copy pending 107
INVRECP 841
logical page list (LPL) 107
NOCOPIES 841
NOTAVAIL 220, 841
NOTCLONED 841
NOTDEFND 131, 841
OK 839, 845
PEND DDL 841
persistent read only 107
read or replication only 107
rebuild pending 107
recover pending (RECP) 107
reorg pending 107
stopped error range (STOPE) 107
SYS OBJ 131, 842
TBLPART 842
TEMPDB 842
TS STAT 842
TSREORP 842
unacceptable status 839, 845
step library information 748
Steplib Addition configuration option 757
steplib data sets, additional 748
Steplib Override configuration option 757
steplib override data sets 748
STEPLIB_ADDITION configuration option 757
STEPLIB_OVERRIDE configuration option 757
STOGROUP, delete 233
stopped error range 107
storage group specification 107
STORCLAS option 417
subsystem
adding 63
backup options 98
options, authorization 93
options, editing 98
recovery options 98
recovery point 677
recovery point, default 677
service level agreement options 98
status 266
subsystem configuration options 755, 757
subsystem ID 57, 86
subsystem options 757
subsystem recovery
about 347
ARMBGPS program 347, 351, 363
backup strategy 349
basic information 347
batch programs 363
building groups 351
local recovery JCL 357
log range analysis 357
RECOVERY MANAGER repository 350
SUF, CONCURRENT option 870
summary of changes 25
SUPPORT_FULL_COPY_DDS option 870
SUPPORT_OUTSIZE option 870
symbolic variables
copy data set names 191, 259
in JOB card specification 51
JCARD 52
JCLOUT 52
output data sets 52
SYNC option 451
synchronization option 271
synchronization, logs 382
SYNCNAME option 707
syntax
syntax options 49
syntax rules 102
SYS OBJ status 131, 842
SYSAFF cards 331, 700
SYSCOPY search
before ARMBGEN 302
limiting 233
SYSIBM.SYSCOPY 70
SYSPLANDEP 320, 681
system data set password 279
system object status 131
system resource recovery 320, 673
system resources
about 265
authorization to access 92
recovery 68
recovery and maintenance 50, 157, 265
system validation report 691, 712
SYSUT2 DD statement 211
SYSUT4 DD statement 211

T

table names, determining 814
table space, specification 107
tables
BMCARM.BMCRMD_RECOVERY_EXCEPTIONS 807
coordinated disaster recovery 773
recovery history
IX_PART table 799
IXPSORT table 805, 806
JOB table 776
KEYSORT table 796
PHASE table 785
TS table 788
TS_PART table 790
TSPSORT table 798
UTILITY_RUN table 774
tables, BMC
backing up 813
BMCHIST 817
BMCSYNC 820
BMCTRANS 828
BMCUITL 829
BMXCOPY 832
considerations 813
determining names 814
querying 814
warnings 813
tables, BMC Common DB2 repository
GROUPAUTH 772
GRPOPTS 771
OBJSET_DEF 768
OBJSET_SQL 771
OBJSETS 767
PRODREG 772
tables, BMCDICT 816
tables, RECOVERY MANAGER repository
Tables, RMGR repository
- IX_PART 799
- IXPSORT 806
- JOB 776
- KEYSORT 796
- PHASE 785

Tables, shared
- BMCLGRNX 64
- BMCSYNC 64
- BMCUTIL 64
- BMCXCOPY 64

Tablespace, Progress Report 395
- Tape
disaster recovery options 317
for copy after recovery 186, 254

TAPE option 416
- TAPE UNIT option 418
- TBLPART status 842
- TEMPDB status 842
Temporals tables, grouping 477
temporary tables 64
temporary work file database 285
terminating BMC utilities 814
timestamp insertion, ARMBTSI 426
timestamp recovery 205
timestamp recovery option 500
timestamp, ARMBTSI program 719
timestamp, recovery option 452
TOCOMMONRECPT option 452
TOCOPY option 452, 500
TOCURRENT option 452, 500
TOFULLCOPY option 452, 500
TOLOGMARK option 452, 500
TOLOGPOINT 183
TOLOGPOINT option 452, 500, 728, 883
TOQUIESCE option 452, 500
TORESTARTRBA option 452, 500
Total bytes, Progress Report 395
TOTIMESTAMP option 452
trace
- batch 845
- online 845
- TRACE file 609
- TRTCCH compression 317

TRTCCH option 418
truncation
- long names 88
- RECOVERY MANAGER option 88
Truncation Characters configuration option 757
Truncation Position configuration option 757
truncation, long names 757
TRUNCCHAR configuration option 757
TRUNCPOS configuration option 757
- TS STAT status 842
- TS table 788
- TS_PART table 790
- TSO command restrictions 89, 747
- TSPSORT table 798
- TSREORP status 842
- TSTAMP configuration option 757
type A authorization 91
type O authorization 91

U
- UCATIX option 728
- UID option 728
- UNCHANGE 842
- UNCHANGE status 842
unchanged analysis 452
Unicode support 111, 508
unit count 251, 254
Unit count option 317
unit option 186, 254
UNIT option 416
UNIT_COUNT option 870
UNITCNT option 417
UNLOADKEYS entries, cleaning up 825
UNLOADKEYS_BUILINDEX option 860
UNLOADKEYS, performance considerations 71
Unloadkeys/Buildindex option 230, 238
UNLOADKEYS/BUILINDEX option 238
UNRECOVER_RC option 449, 497
unsatisfactory object status 839
UPDATE GROUP command 431, 452, 512, 546, 560
UPDATE_BMCSTATS option 870
UPDATE_DB2_CATALOG option 870
Use CATALOG Parm configuration option 757
user ID 86
User Joblib configuration option 757
user-defined SQL
ARMSQL data set 518
batch 509
example JCL for group 577
example output 577
object specification 107
VIA SQL syntax 535
Using Multiple Job Optimization in Off-Site Recovery 73
USRLIB1 configuration option 757
utilities
backup 67
BSDS maintenance and recovery 68
image copy, post-recovery 67
integrity checking 67
option set 748
recovery 66
recovery point 67
repair 67
specifying 49
supported by RECOVERY MANAGER 66
utility options, override rules 125
UTILITY_RUN table 774

V

validation
object availability 131
object eligibility 131
object recoverability 50, 132
of application group 133
of DB2 objects 130
VARBIN 843
VARBIN status 843
variables, symbolic 191
VCAT configuration option 757
VCAT objects, redefine 233
verifying
object availability 131
object eligibility 131
object recoverability 132
versioning (history) objects, inclusion in groups 132
VIA EXCEPTION 531
VIA EXCEPTION option 573
VIA GROUP option 543
VIA INDEX 538
VIA PACKAGE 540
VIA PLAN option 542
VIA REPOSITORY 540
VIA REPOSITORY option 510
VIA SQL 535
VIA STOGROUP option 542
VIA TABLE option 531
VIA TABLESPACE option 531, 573
VIA VOLUMES option 581
viewing
  group definitions 126
  physical data set attributes 286
viewing backup time estimates 390
viewing recover time estimates 392, 394
viewing recovered, changed, and copied objects 386
volume count option 186, 254
volume group creation 107, 118
  in background mode 107, 118
  in foreground mode 107, 118
  procedure 119
  VVDS method 118
volume groups 473
VSAM catalog maintenance 279
VVDS
  method for ARMBGRP 118
  working with volume groups 118
  VVDS method, for ARMBGRP 509

W

wildcard characters
  in batch group renaming 511
  in coordinated recoveries 112
  supported 102
  usage in RECOVERY MANAGER 102, 112
work file
database data set 285
options, in disaster recovery 320
reallocating temporary space 285
space specification 97, 320
temporary data base 285
Work File Database configuration option 757
work file option descriptions 251
Work File Prefix configuration option 757
Work Unit configuration option 755
WORKFILE_ALLOCATION_TYPE option 862
WORKFILE_MAX_PRIMARY option 862
WORKFILE_PRIMARY_ALLOC option 862
WORKFILE_SECONDARY_ALLOC option 862
WORKFILE_WORK_UNIT option 862
WORKFILE_WORKDDN option 862
WORKFILE_WORKPREFIX option 862
WORKFLDB configuration option 757
WORKPREFIX configuration option 757
WRITE PENDING status 438, 691

**X**

XBMID option 238, 860, 870
XML data
  BMCSYNC table considerations 823
XML objects, inclusion in groups 132
XML recovery
  about 202
XML spaces, grouping 477
XUNCHANGED 452
XUNCHANGED option 452
  about 210
  in subsystem recovery 357

**Z**

z/OS requirement 57
zaps, determine applied 846
ZIIP configuration option 757
ZIIP option
  and ARMBARC 418
  and ARMBSRR 700
  and system resource recovery fields 332
  and XBM requirements 57
ZIIP Redirection configuration option 757
zoom action 87, 88