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
2101 CITYWEST BLVD
HOUSTON TX 77042-2827
USA

Telephone: 713 918 8800 or 800 841 2031

Fax: 713 918 8000

Outside United States and Canada

Telephone: (01) 713 918 8800

Fax: (01) 713 918 8000

© Copyright 2009 BMC Software, Inc.

BMC, BMC Software, and the BMC Software logo are the exclusive properties of BMC Software, Inc., are registered with the U.S. Patent and Trademark Office, and may be registered or pending registration in other countries. All other BMC trademarks, service marks, and logos may be registered or pending registration in the U.S. or in other countries. All other trademarks or registered trademarks are the property of their respective owners.

CICS, DB2, IMS, RACF, VTAM, and z/OS are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

BMC Software considers information included in this documentation to be proprietary and confidential. Your use of this information is subject to the terms and conditions of the applicable End User License Agreement for the product and the proprietary and restricted rights notices included in this 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 UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES.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, 2101 CITYWEST BLVD, HOUSTON TX 77042-2827, USA. Any contract notices should be sent to this address.
Customer support

You can obtain technical support by using the BMC Software Customer Support website or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, see “Before contacting BMC.”

Support website

You can obtain technical support from BMC 24 hours a day, 7 days a week at http://www.bmc.com/support_home. 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 issues similar to yours and possible solutions
- order or download product documentation
- download products and maintenance
- report an issue or ask a question
- subscribe to receive proactive e-mail alerts when new product notices are released
- 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 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 issue
- 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, such as file system full
  - messages from related software
License key and password information

If you have questions about your license key or password, contact BMC as follows:

- *(USA or Canada)* Contact the Order Services Password Team at 800 841 2031, or send an e-mail message to ContractsPasswordAdministration@bmc.com.

- *(Europe, the Middle East, and Africa)* Fax your questions to EMEA Contracts Administration at +31 20 354 8702, or send an e-mail message to password@bmc.com.

- *(Asia-Pacific)* Contact your BMC sales representative or your local BMC office.
## Contents

### About this book
- How this book is organized ................................................. 13
- Related documentation ................................................... 14
  - Online books ......................................................... 15
  - Online Help ......................................................... 15
- Conventions ............................................................... 16
  - General conventions .................................................. 16
  - Syntax statements .................................................... 18

### Chapter 1 Preparing to install Recovery Utility for VSAM
- Overview ........................................................................... 20
  - RUV distribution tape .................................................... 20
  - Installation requirements .............................................. 21
  - Space allocation for RUV distribution data sets .................. 22
  - Rules for running multiple product releases in a single subsystem .................................................. 23
- Installation process .......................................................... 24
  - Installation methods ...................................................... 25
  - Distribution method ...................................................... 25
  - International language support ....................................... 25
  - Version, release, and maintenance levels ......................... 25
  - Installation paths .......................................................... 26
- Customer support ............................................................. 28

### Chapter 2 Configuring Recovery Utility for VSAM
- Overview ........................................................................... 31
  - Getting started .................................................................. 32
    - Accessing RUV data sets .............................................. 32
    - Executing the ISPF interface ......................................... 32
  - Setting up security .......................................................... 34
    - Defining new resources .............................................. 34
    - Enabling access .......................................................... 35
  - Performing implementation tasks ..................................... 37
    - Implementation scenarios ............................................ 38
    - Starting BMCP and BCSS subsystems ......................... 39
    - Defining transactions and programs to CICS .................. 41
    - Customizing CICS startup and shutdown ...................... 42
    - Verifying CICS initialization parameters ....................... 43
    - Updating CICS startup JCL ......................................... 44

---

### Online Help
- Online books ......................................................... 15
- Online Help ......................................................... 15

### Conventions
- General conventions .................................................. 16
- Syntax statements .................................................... 18

---

### Chapter 1 Preparing to install Recovery Utility for VSAM
- Overview ........................................................................... 20
  - RUV distribution tape .................................................... 20
  - Installation requirements .............................................. 21
  - Space allocation for RUV distribution data sets .................. 22
  - Rules for running multiple product releases in a single subsystem .................................................. 23
- Installation process .......................................................... 24
  - Installation methods ...................................................... 25
  - Distribution method ...................................................... 25
  - International language support ....................................... 25
  - Version, release, and maintenance levels ......................... 25
  - Installation paths .......................................................... 26
- Customer support ............................................................. 28
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coding CICS file definitions</td>
<td>45</td>
</tr>
<tr>
<td>Initializing and enabling the BWO feature</td>
<td>51</td>
</tr>
<tr>
<td>Initializing and enabling the Logging feature</td>
<td>54</td>
</tr>
<tr>
<td>Using the RUVZFAC macro</td>
<td>55</td>
</tr>
<tr>
<td>Installing and implementing the SNAPSHOT UPGRADE FEATURE</td>
<td>58</td>
</tr>
<tr>
<td>SUF Requirements</td>
<td>58</td>
</tr>
<tr>
<td>SUF Authorization</td>
<td>59</td>
</tr>
<tr>
<td>Using the installation verification procedure</td>
<td>60</td>
</tr>
<tr>
<td>VSAM files that are used by the IVP</td>
<td>65</td>
</tr>
<tr>
<td>Allocated data sets</td>
<td>66</td>
</tr>
<tr>
<td>Migrating RUV from a test environment to a production environment</td>
<td>67</td>
</tr>
<tr>
<td>Migrating RECOVERY PLUS for CICS/VSAM to RUV</td>
<td>68</td>
</tr>
<tr>
<td>RUVZCJC utility</td>
<td>68</td>
</tr>
<tr>
<td>RUVZCDA utility</td>
<td>70</td>
</tr>
<tr>
<td>Migration considerations</td>
<td>74</td>
</tr>
<tr>
<td>Using rule sets</td>
<td>81</td>
</tr>
<tr>
<td>Rule set components</td>
<td>82</td>
</tr>
<tr>
<td>Component relationships</td>
<td>83</td>
</tr>
<tr>
<td>Implementing the RUV repository</td>
<td>84</td>
</tr>
<tr>
<td>Starting BMCP and BCSS subsystems at IPL</td>
<td>85</td>
</tr>
<tr>
<td>Appendix A Installation instructions</td>
<td>87</td>
</tr>
<tr>
<td>Tape installation</td>
<td>88</td>
</tr>
<tr>
<td>EPD installation</td>
<td>89</td>
</tr>
<tr>
<td>Post-release fixes for RUV 4.1.00</td>
<td>95</td>
</tr>
<tr>
<td>Appendix B BMC Software product authorization</td>
<td>97</td>
</tr>
<tr>
<td>Overview</td>
<td>98</td>
</tr>
<tr>
<td>Product Authorization utility</td>
<td>99</td>
</tr>
<tr>
<td>Product-authorization passwords</td>
<td>101</td>
</tr>
<tr>
<td>Permanent passwords</td>
<td>101</td>
</tr>
<tr>
<td>Temporary passwords</td>
<td>102</td>
</tr>
<tr>
<td>How to apply passwords</td>
<td>102</td>
</tr>
<tr>
<td>Product licensing</td>
<td>103</td>
</tr>
<tr>
<td>Product trials and permanent licenses</td>
<td>103</td>
</tr>
<tr>
<td>CPU upgrades</td>
<td>103</td>
</tr>
<tr>
<td>CPU failures</td>
<td>104</td>
</tr>
<tr>
<td>Product maintenance or version upgrades</td>
<td>105</td>
</tr>
<tr>
<td>How to obtain passwords</td>
<td>106</td>
</tr>
<tr>
<td>Online product authorization</td>
<td>108</td>
</tr>
<tr>
<td>Processing a permanent password</td>
<td>108</td>
</tr>
<tr>
<td>Adding authorization for a processor</td>
<td>110</td>
</tr>
<tr>
<td>Deleting authorization for a processor</td>
<td>112</td>
</tr>
<tr>
<td>Replacing authorization for a processor</td>
<td>114</td>
</tr>
<tr>
<td>Modifying authorization for an existing processor</td>
<td>116</td>
</tr>
<tr>
<td>Resetting authorization for all processors</td>
<td>118</td>
</tr>
<tr>
<td>Processing a temporary password</td>
<td>119</td>
</tr>
<tr>
<td>Displaying product authorization</td>
<td>120</td>
</tr>
<tr>
<td>Displaying current processor information</td>
<td>122</td>
</tr>
</tbody>
</table>
Figures

PLTPI entries for all CICS versions ............................................. 43
PLTSD entries for all CICS versions .......................................... 43
CICS file definition options for CICS TS environments ............... 50
User variable changes that are required for IVP$CHGS ............... 64
Sample JCL for RUVZCJC ......................................................... 69
Sample JCL for RUVZCDA .......................................................... 71
Sample JCL for the RUV data set name and group definition and report .... 73
Rule set component relationships ............................................ 83
JCL to extract files from the RUV product tape .................... 88
JCL to preallocate the TRS files ............................................. 92
JCL to unterse files downloaded from the Electronic Product Distribution site .. 93
JCL to copy modules .................................................................. 94
JCL to download the RUV 4.1.00 FIX files .............................. 95
JCL to copy modules .................................................................. 96
Product Authorization Primary Menu (SECEPPRI) .................. 108
ADD Authorization for a Processor panel (SECEPADD) ............. 110
Product authorization ADD message ........................................ 111
DELETE Authorization for a Processor panel (SECEPDEL) .......... 112
Product authorization DELETE message .................................. 113
REPLACE Authorization for a Processor panel (SECEPREP) ....... 114
Product authorization REPLACE message ................................. 115
MODIFY Authorization for an Existing Processor panel (SECEPUPD) ... 116
Product authorization MODIFY message .................................. 117
Product authorization RESET message .................................. 118
Product authorization temporary password message .................. 119
Product Authorization Display panel (SECEPTBL) ..................... 120
Current Processor Information panel (SECEPCPU) ................... 122
Sample JCL for running batch product authorization .................. 124
Tables

System resources .......................................................... 21
Space allocation for RUV distribution data sets .................. 22
Installation and configuration documentation .................... 24
Implementation tasks ...................................................... 37
Implementation scenarios ................................................ 38
CICS file definition parameters for CICS TS environments ..... 50
Installation verification procedure .................................... 61
VSAM files that are used by IVP members ......................... 65
Supported file structure requirements ................................. 74
Repository information for VSAM file recovery .................... 74
Subsystem requirements .................................................. 75
Rule set requirements ..................................................... 75
Security requirements ..................................................... 76
ISPF interface requirements .............................................. 76
BMC Software subsystem requirements ............................... 76
Batch journaling requirements and job step logging requirements . 77
Master data set backup requirements .................................. 78
Purge processing requirements .......................................... 78
Archiving requirements ................................................... 79
RUV recovery requirements .............................................. 79
RUV report requirements ................................................ 80
Miscellaneous requirements ............................................. 80
File sizes to preallocate ................................................. 91
Permanent password functions ........................................ 101
Password-required situations .......................................... 106
Product-authorization work sheet ...................................... 107
Product Authorization Primary Menu options ....................... 109
Product Authorization Display panel fields ......................... 120
Information for JCL script .............................................. 125
Control statement keywords .......................................... 126
Return codes .............................................................. 126
Resolutions for common installation and configuration issues .... 135
About this book

This book contains detailed information about configuring the Recovery Utility for VSAM (RUV) product.

This book is intended for use by system programmers and database administrators who install and configure RUV. The terms and concepts in this book require a general knowledge of your operating system, CICS, job control language (JCL), Interactive System Productivity Facility (ISPF), and VSAM.

How this book is organized

This book is organized as follows:

<table>
<thead>
<tr>
<th>Chapter/appendix</th>
<th>Description</th>
</tr>
</thead>
</table>
| Chapter 1, “Preparing to install Recovery Utility for VSAM” | Includes the following topics:  
  - installation overview  
  - understanding the installation process  
  - installation considerations |
| Chapter 2, “Configuring Recovery Utility for VSAM”      | Explains prerequisites and installation considerations that you should know before configuring this product. Includes procedures for using the installation verification process, performing implementation tasks, using rule sets, migrating from a test environment to a production environment, and migrating RPCV to RUV. |
| Appendix A, “Installation instructions”                 | Describes the process that you use for a tape installation, an ESD installation, and installation of post-release fixes. |
| Appendix B, “BMC Software product authorization”         | Describes the licensing of this product.                                    |
| Appendix C, “Work sheets for rule sets”                  | Provides information-gathering forms for the construction of RUV rule sets. |
| Appendix D, “Resolutions for common installation and configuration issues” | Provides solutions to common installation problems. |
Related documentation

This product is supported by several types of documentation:

- online books
- online Help
- release notes and other notices

**NOTE**
The messages that the RUV installation generates are available in a z/OS data set that is downloaded during installation. For each message, the data set includes an explanation and suggests a user response. The z/OS data set is called HLQ.MSGS (HLQ is the high-level qualifier that is specified during installation).

The following books provide information about RUV:

- *Recovery Utility for VSAM Installation and Customization Guide*
- *Recovery Utility for VSAM User Guide*
- *Recovery Utility for VSAM Messages Manual*

The following BMC Software books provide information about the SNAPSHOT UPGRADE FEATURE (SUF) component that is used for Instant Snapshot copy processing:

- *EXTENDED BUFFER MANAGER and SNAPSHOT UPGRADE FEATURE General Information*
- *EXTENDED BUFFER MANAGER and SNAPSHOT UPGRADE FEATURE Installation and Customization Guide*
- *EXTENDED BUFFER MANAGER and SNAPSHOT UPGRADE FEATURE User Guide*

These publications are available for download from the BMC Software EPD site and are included on the RUV documentation CD.
Online books

The books that accompany this product are available in online format. If you are a Windows or Unix user, you can view online books with Acrobat Reader from Adobe Systems. The reader is provided at no cost.

You can access online books from the World Wide Web or from the documentation CD. Online books are formatted as Portable Document Format (PDF) files. You can view them, print them, or copy them by using Acrobat Reader.

**NOTE**
For information about downloading the free reader from the Web, go to the Adobe Systems site at [www.adobe.com](http://www.adobe.com).

Online Help

RUV includes online Help. In the installation ISPF interface, you can access Help by pressing F1 from any ISPF panel.
Conventions

This section provides examples of the conventions that are used in this book and explains how to read syntax statements.

General conventions

This book uses the following general conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>information that you are instructed to type</td>
<td>Type <strong>SEARCH DB</strong> in the designated field.</td>
</tr>
<tr>
<td>specific (standard) keyboard key names</td>
<td>Press <strong>Enter</strong>.</td>
</tr>
<tr>
<td>field names, text on a panel</td>
<td>Type <strong>the appropriate entry</strong> in the <strong>Command</strong> field.</td>
</tr>
<tr>
<td>directories, file names, Web addresses</td>
<td>The home page of BMC Software is at <a href="http://www.bmc.com">www.bmc.com</a>.</td>
</tr>
<tr>
<td>nonspecific key names, option names</td>
<td>Use the HELP function key.</td>
</tr>
<tr>
<td></td>
<td><strong>KEEPDICTIONARY</strong> option</td>
</tr>
<tr>
<td>z/OS calls, commands, control statements,</td>
<td>Use the <strong>SEARCH</strong> command to find a particular object.</td>
</tr>
<tr>
<td>keywords, parameters, reserved words</td>
<td>The product generates the SQL TABLE statement next.</td>
</tr>
<tr>
<td>code examples, syntax statements, system</td>
<td><strong>//STEPLIB DD</strong></td>
</tr>
<tr>
<td>messages, screen text</td>
<td>The table <strong>table_name</strong> is not available.</td>
</tr>
<tr>
<td>emphasized words, new terms, variables</td>
<td>The instructions that you give to the software are called <strong>commands</strong>.</td>
</tr>
<tr>
<td></td>
<td>In this message, the variable <strong>file_name</strong> represents the file that</td>
</tr>
<tr>
<td></td>
<td>caused the error.</td>
</tr>
<tr>
<td>GUI menu sequence</td>
<td>Choose <strong>File =&gt; Open</strong>.</td>
</tr>
</tbody>
</table>
This book uses the following types of special text:

--- **NOTE**
Notes contain important information that you should consider.

--- **WARNING**
Warnings alert you to situations that could cause problems, such as loss of data, if you do not follow instructions carefully.

--- **TIP**
Tips contain useful information that may improve product performance or that may make procedures easier to follow.
Syntax statements

Here is a sample syntax statement:

```
COMMAND KEYWORD1 [KEYWORD2|KEYWORD3] KEYWORD4=(YES|NO)
  file_name...
```

This book uses the following conventions for syntax statements:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Items in italic type represent variables that you must replace with a name</td>
<td><code>dtsbackup control_directory</code></td>
</tr>
<tr>
<td>or a value. Use an underscore for variables with more than one word.</td>
<td></td>
</tr>
<tr>
<td>Braces { } indicate a group of options for RUV. You can choose at least</td>
<td><code>{ LOCATION(locname8) }</code></td>
</tr>
<tr>
<td>one of the items in the group, but none of them is required. Do not type</td>
<td><code>{ COMMENT(comment_fields) }</code></td>
</tr>
<tr>
<td>the braces when you enter the option. A comma means that you can choose</td>
<td></td>
</tr>
<tr>
<td>one or more of the listed options. You must use a comma to separate the</td>
<td></td>
</tr>
<tr>
<td>options if you choose more than one option.</td>
<td></td>
</tr>
<tr>
<td>An asterisk * indicates a wildcard. A question mark ? indicates a wildcard</td>
<td><code>dsn??,*,...</code></td>
</tr>
<tr>
<td>substitution for one character. An ellipsis indicates a repetition of the</td>
<td></td>
</tr>
<tr>
<td>listing.</td>
<td></td>
</tr>
<tr>
<td>Commands in bold specify required items. You must enter the items as part</td>
<td><code>BACKUP REPOSITORY</code></td>
</tr>
<tr>
<td>of the overall command statement. Do not type the bold when you enter the</td>
<td></td>
</tr>
<tr>
<td>item.</td>
<td></td>
</tr>
<tr>
<td>A vertical bar means that you can choose only one of the listed items. In</td>
<td>`{ commit</td>
</tr>
<tr>
<td>the example, you would choose commit or cancel.</td>
<td>`{ -commit</td>
</tr>
<tr>
<td>An ellipsis indicates that you can repeat the previous item or items as</td>
<td><code>column_name ...</code></td>
</tr>
<tr>
<td>many times as necessary.</td>
<td></td>
</tr>
<tr>
<td>Default values for RUV are listed first. An underline indicates a default.</td>
<td>`KEYWORD4({YES</td>
</tr>
<tr>
<td>In the example, YES is the default. Do not type the underscore.</td>
<td></td>
</tr>
</tbody>
</table>
Preparing to install Recovery Utility for VSAM

This chapter describes requirements for installation of the Recovery Utility for VSAM (RUV) product.

NOTE
For the most current installation notes and supplementary materials, contact Customer Support.

This chapter contains the following topics:

Overview ................................................................. 20
  RUV distribution tape .............................................. 20
  Installation requirements ......................................... 21
  Space allocation for RUV distribution data sets ............... 22
  Rules for running multiple product releases in a single subsystem ............. 23
Installation process .................................................. 24
  Installation methods .............................................. 25
  Distribution method .............................................. 25
  International language support .................................. 25
  Version, release, and maintenance levels ....................... 25
  Installation paths .................................................. 26
Customer support .................................................... 28
Overview

This chapter provides an overview of the installations that you can perform.

NOTE
RUV can work with the BMC Software SNAPSHOT UPGRADE FEATURE (SUF) component to produce Instant Snapshot copies of VSAM data sets. To use SUF with RUV, you must install, customize, and implement SUF. This component is packaged for distribution with the BMC Software EXTENDED BUFFER MANAGER (XBM) product. For information about the documents and media that are required for understanding, installing, customizing, and implementing SUF and managing the XBM subsystem, see the EXTENDED BUFFER MANAGER and SNAPSHOT UPGRADE FEATURE Release Notes.

RUV distribution tape

Recovery Utility for VSAM is distributed on a tape.

NOTE
Do not use an installation data set from a prior installation to install what is now called Recovery Utility for VSAM. Tapes that were labeled SIT or SIS still exist and contain DB2 products. Those products may be ordered and unloaded separately as before.
Installation requirements

Table 1 lists system resources that are required or recommended for operation of RUV.

RUV 4.1 supports all current IBM-supported releases of z/OS and CICS TS at the time of this publication. Refer to the Release Notes for further details.

Table 1 System resources

<table>
<thead>
<tr>
<th>Resource</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>hardware</td>
<td>RUV can operate on any processor that supports an IBM-supported z/OS operating system. This product is licensed to run on specific CPUs.</td>
</tr>
<tr>
<td>operating system</td>
<td>The following IBM software is required:</td>
</tr>
<tr>
<td></td>
<td>■ IBM-supported release of z/OS</td>
</tr>
<tr>
<td></td>
<td>■ IEBCOPY utility</td>
</tr>
<tr>
<td></td>
<td>■ ISPF/PDF version 4.2 or later</td>
</tr>
<tr>
<td></td>
<td>A region size of 3072 KB is required, and a region size of 4096 KB is recommended.</td>
</tr>
<tr>
<td></td>
<td>■ TSO/E version 2.1 or later</td>
</tr>
<tr>
<td></td>
<td>■ APF-authorized library</td>
</tr>
<tr>
<td>DASD storage</td>
<td>RUV libraries require approximately 250 cylinders of 3390 DASD storage. The amount of DASD that is required for the master VSAM repository varies according to the number of VSAM data sets that are controlled and the frequency and lifetime of backups and archives that have been taken.</td>
</tr>
</tbody>
</table>
Space allocation for RUV distribution data sets

This section describes the estimated DASD requirements for the RUV distribution data sets.

Table 2 lists the data sets that the installation process creates for RUV and the recommended space allocation for each data set.

NOTE
HLQ represents the high-level qualifier that you have chosen for RUV.

<table>
<thead>
<tr>
<th>Data set</th>
<th>RECFM</th>
<th>LRECL</th>
<th>BLKSIZE</th>
<th>Estimated tracks</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLQ.SOURCE</td>
<td>FB</td>
<td>80</td>
<td>23440</td>
<td>30</td>
</tr>
<tr>
<td>HLQ.CNTL</td>
<td>FB</td>
<td>80</td>
<td>3120</td>
<td>15</td>
</tr>
<tr>
<td>HLQ.LOAD</td>
<td>U</td>
<td>0</td>
<td>6144</td>
<td>1440</td>
</tr>
<tr>
<td>HLQ.DLIB</td>
<td>FB</td>
<td>80</td>
<td>3120</td>
<td>45</td>
</tr>
<tr>
<td>HLQ.BPSLOAD</td>
<td>U</td>
<td>0</td>
<td>6144</td>
<td>15</td>
</tr>
<tr>
<td>HLQ.BCSLOAD</td>
<td>U</td>
<td>0</td>
<td>6144</td>
<td>90</td>
</tr>
</tbody>
</table>
Rules for running multiple product releases in a single subsystem

You may have to perform acceptance testing on a new release before you can upgrade it to production. To facilitate testing, you can run multiple releases of some products within a single z/OS image. If you plan to use the Full installation path to install multiple releases of the products, follow these rules:

- Use a unique high-level qualifier (HLQ) that is different from the qualifier in use. The installation process creates a new set of product libraries to support the new release.

- Do not copy the load modules to the APF library that is in use.

- Use the custom installation mode.

- Use naming conventions that are different from the conventions which are in use for plans and collection IDs.

The installation process generates a new ISPF interface to access the newly installed set of products. You must continue to use the old ISPF interface to access the older set of products.
**Installation process**

RUV is installed on z/OS by unloading the RUV product tape.

Before you install RUV, you must gather certain information and consider how and where you plan to run the product. This section provides the following information to help you plan your installation:

- prerequisites for installing, configuring, and running RUV
- more information that you should consider before you install or configure RUV

Table 3 lists the documentation that is required for installing and configuring RUV.

**Table 3  Installation and configuration documentation**

<table>
<thead>
<tr>
<th>Task</th>
<th>Documentation to use</th>
</tr>
</thead>
</table>
| prepare to install and customize RUV | ■ current release notes, flashes, and technical bulletins  
■ Product Authorization Letter  
■ Chapter 1, *Recovery Utility for VSAM Installation and Customization Guide* |
| customize RUV, and prepare it for use | Chapter 2, *Recovery Utility for VSAM Installation and Customization Guide*           |
Installation methods

The RUV install process supports the following methods of installation:

- Standard installation provides a fast IEBCOPY installation process that copies a partitioned data set.

NOTE
The SMP/E method is currently not available for installing RUV.

Distribution method

Recovery Utility for VSAM is distributed on a tape.

Every RUV shipment includes the following item:

- Recovery Utility for VSAM distribution tape containing the ISPF panels and programs that are required for customizing the installation of RUV

International language support

The installation process automatically invokes international language support if the terminal type is 32xxKN. For terminal type 32xxKN, installation panels are displayed in uppercase letters.

NOTE
The terminal type is set from the ISPF main menu.

Version, release, and maintenance levels

The complete product code for this product is distributed on a tape. You receive the same tape whether you request a product trial or have already purchased a permanent license, and whether you are installing the product for the first time or you are installing maintenance for a product that you already have.
Installation paths

The installation path that you select (full or maintenance) depends on whether you are installing a product for the first time or installing a new version of an existing product. Use this section to determine the best installation path for RUV.

Full path

Use the full installation path to install one or more products by BMC Software on a single subsystem. The full installation path has the following characteristics:

- creates new product libraries
- allows an interface with other BMC Software products
- allows data migration from a previous release when you specify the custom installation panel mode (not available for all products)
- allows you to retain security authorization from a previous release

**NOTE**

If this is your initial install of RUV, you must obtain new RUV passwords from your sales representative.

- allows use of your own VSAM data sets (when supported by the selected product and when you specify the custom installation panel mode)

Use the full installation path for the following reasons:

- You are installing a new version or new release of a product for the first time (initial installation).
- You completed the initial installation and now want to install the product on other z/OS images. However, you do not want to share the product libraries that were created during the initial installation.
You are installing maintenance updates under the following conditions:

— You want to overlay your current installation.

— You want to create a separate maintenance library to be used in concatenation with your current installation (you do not want to overlay the existing installation).

— You cannot share the product libraries that were created during the initial installation with the maintenance update.

**Maintenance path**

If you are installing maintenance updates and want to overlay your current installation, use the maintenance installation path. To use the maintenance installation path, you must have an installed version of RUV.

The maintenance installation path has the following characteristics:

- allows data migration from a previous release
- allows retention of security authorization from a previous release

 ***NOTE***

You will need new product licenses for RUV if you have not previously installed RUV.

- (optional) creates new product libraries

Maintenance installation comprises the following types:

- initial maintenance installation

  Use the initial maintenance installation path to apply a maintenance update for the first time. Select this option to unload the distribution tape, create a new set of product libraries or overlay existing libraries, and generate new default options.

- subsequent maintenance installation

  Use the subsequent maintenance installation path to apply maintenance to other installation sites that share the product libraries which were created during the initial maintenance application. If the subsequent installation sites do not share product libraries, you must use the initial maintenance installation path.
Customer support

Technical support analysts are on call 24 hours a day, 7 days a week, 365 days a year. The support number is listed on the back of the title page.

You can help the technical support analyst resolve your problem more rapidly by having the following information available when you call:

- JCL that you were using to run the job
- product messages that were issued by the job
- reports that are located in SYSPRINT
- version, release, and maintenance level of RUV
- version, release, and maintenance level of CICS (if applicable)
- version, release, and maintenance level of ISPF (if applicable)
- version, release, and maintenance level of z/OS (if applicable)

For detailed information, see the following books:

- *Recovery Utility for VSAM User Guide*
- *Recovery Utility for VSAM Messages Manual*
Chapter 2 Configuring Recovery Utility for VSAM

This chapter describes configuration requirements for the Recovery Utility for VSAM (RUV) product. This chapter contains the following topics:

Overview .................................................. 31
Getting started ............................................. 32
  Accessing RUV data sets .................................. 32
  Executing the ISPF interface .............................. 32
Setting up security ........................................ 34
  Defining new resources .................................. 34
  Enabling access .......................................... 35
Performing implementation tasks .......................... 37
  Implementation scenarios ................................. 38
  Starting BMCP and BCSS subsystems .................... 39
Defining transactions and programs to CICS ............... 41
Customizing CICS startup and shutdown .................... 42
Verifying CICS initialization parameters ................... 43
Updating CICS startup JCL ................................ 44
Coding CICS file definitions ............................... 45
Initializing and enabling the BWO feature ................. 51
Initializing and enabling the Logging feature ............... 54
Using the RUVZFAC macro ............................... 55
Installing and implementing the SNAPSHOT UPGRADE FEATURE ........................................ 58
  SUF Requirements ....................................... 58
  SUF Authorization ...................................... 59
Using the installation verification procedure ............... 60
  VSAM files that are used by the IVP .................... 65
  Allocated data sets ..................................... 66
Migrating RUV from a test environment to a production environment ........................................ 67
Migrating RECOVERY PLUS for CICS/VSAM to RUV .................................................. 68
  RUVZCJC utility ......................................... 68
  RUVZCDA utility ........................................ 70
  Migration considerations ................................. 74
Using rule sets ............................................ 81
Rule set components ................................................................. 82
Component relationships .......................................................... 83
Implementing the RUV repository ................................................ 84
Starting BMCP and BCSS subsystems at IPL ................................. 85
Overview

RUV provides quick and easy VSAM data recovery processing. With RUV you can focus on the data that you are recovering, not on the process that recovers the data. You can use RUV to recover VSAM data sets in the following situations:

- Physical VSAM data is lost or damaged, which may occur for the following reasons:
  - A hardware failure occurred.
  - A part of a DASD is not readable.
  - The data was deleted.
  - A natural disaster, such as a fire or a storm, occurred.

- Logical data integrity is lost, which may occur for the following reasons:
  - An application error occurred.
  - A failure occurred during CICS transaction backout.

**NOTE**

Loss of logical data integrity occurs when a CICS application incorrectly updates your VSAM spheres. CICS transaction backout maintains logical data integrity by backing out partially completed transactions from VSAM data sets. For example, if I/O errors occur as CICS accesses a VSAM data set, transaction backout will fail.

RUV recovers your damaged VSAM data sets if these situations occur. RUV recovers all updates that were made before the problem occurred. Recovery is a sequence of tasks that begins with the last usable backup and may include logs, journals, and archival data sets. Recovery is complete when data has been recovered and processing capability has been restored.

Before you start RUV, see the following installation topics:

- proper security setup (see “Setting up security” on page 34)
- customizing RUV for different implementation scenarios (see “Performing implementation tasks” on page 37)
- installing and implementing the SNAPSHOT UPGRADE FEATURE (SUF) (see “Installing and implementing the SNAPSHOT UPGRADE FEATURE” on page 58)
- verifying your RUV installation (see “Using the installation verification procedure” on page 60)
- using rule sets on “Using rule sets” on page 81)
Getting started

You can use the RUV ISPF interface to set repository options, to set global options, to create JCL for RUV recoveries, to access RUV message information, and to display or set operating environment options.

Accessing RUV data sets

You can access RUV data sets that are required for the ISPF interface by using either of the following methods. Use the method that is most appropriate for your site.

- Let RUV determine the high-level qualifier (HLQ) definition.
- Add the following libraries to the TSO LOGON procedure:
  - Add HLQ.DLIB to these DDs:
    - ISPMLIB (messages)
    - ISPPLIB (panels)
    - ISPSLIB (JCL skeletons)
    - ISPTLIB (tables)
  - Add HLQ.LOAD to these load DDs:
    - ISPLLIB
    - STEPLIB

Executing the ISPF interface

To initiate an RUV session at the ISPF interface, perform the following steps:

1. Ensure that the BMC subsystems are running (see “Starting BMCP and BCSS subsystems” on page 39).

2. On an ISPF panel that accepts TSO commands, type one of the following commands:
   - TSO EX ‘HLQ.DLIB(RUVISPF)’

   HLQ is the high-level qualifier from the initial installation.
EXECUTING THE ISPF INTERFACE

Chapter 2 Configuring Recovery Utility for VSAM

- **TSO RUVISPF**

  Use this command if you have placed the EXEC in SYSEXEC or SYSPROC.

  **NOTE**

  If your site preallocates ISPCTL* ddnames to the TSO logon procedure, execute TSO EX 'HLQ.DLIB(RUVISPFZ)'. This module will issue a TSO FREE DD for ISPCTL0 through ISPCTL5 at RUVISPF startup and TSO ALLOCATE these ddnames at RUVISPF termination.

  3 Press ENTER.

  4 (optional) Copy the RUVISPF CLIST from the HLQ.DLIB library to another location, such as the system procedure (SYSPROC) library, or rename the HLQ.DLIB library.

  As distributed by BMC Software, the CLIST expects the RUV load library to have the same high-level qualifier as the library from which the CLIST is executed.

  If you execute the unmodified CLIST from a library that has a different high-level qualifier, the following message is issued:

 IKJ56709I INVALID DATA SET NAME, '.DLIB'

  If you receive the preceding message, you must modify the RUVISPF CLIST to provide the data set name of the RUV load library, as shown in the following example:

  ```
  Parse Var revdsn llq "." revdsn /* Parse past the LLQ */
  ruvhlq = Reverse(revdsn) /* Flip the HLQ */

  If ruvhlq = " " Then /* MODIFICATION TO */
  ruvhlq = "HLQ.LOAD" /* DEFINE LOADLIB DSN */

  "CONTROL ERRORS RETURN"
  Address TSO /* Set TSO execute mode */
  ```

  ...
Setting up security

This section describes how to set up security for RUV.

In many processes that RUV performs, update access to secured files is required. If your site uses a security package, you may need to alter access availability for users who submit backups or other processing requests. You must ensure that all jobs which perform logging have access to the journal and log data sets.

RUV components, especially repository data sets, often require security beyond the security that is provided by the system security manager. The system security manager is typically compatible with the Resource Access Control Facility (RACF).

**TIP**

Use of RUV product security is not required, but BMC Software recommends that you use it.

RACF/ACF2 requirements should be documented for the IEESYS process that is used by the BLDINDEX function.

For information about the correct procedure for defining a user ID for a started task in your environment, see your security system (RACF/ACF2) manual. The IEESYSAS associated address space has its security profile propagated from the address space that spawns it, and it has the same access as the spawning address space.

The ISPF interface can run in INQUIRE mode or UPDATE mode. Mode selection is determined at execution by preestablished RACF authorization to the subsystem data sets (REGISET and REGISET2).

**NOTE**

To customize ACF2, you can use the RUVSAF macro. See examples in the RUVZFAC member of the HLQ.SOURCE library.

Defining new resources

To define new resources, enter the following commands. `ssid` is the actual SSID that you will use. You must repeat these commands for every subsystem that you use.

```
RDEF FACILITY BMCRUV.SUBSYS.ssidd
RDEF FACILITY BMCRUV.COMMAND
RDEF FACILITY BMCRUV.REPOSITORY
```
To permit a user (userid) to access the defined resources, type the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERMIT BMCRUV.SUBSYS.ssid</td>
<td>Class(FACILITY) ID(userid) ACCESS(ALTER)</td>
</tr>
<tr>
<td>PERMIT BMCRUV.COMMAND</td>
<td>Class(FACILITY) ID(userid) ACCESS(ALTER)</td>
</tr>
<tr>
<td>PERMIT BMCRUV.REPOSITORY</td>
<td>Class(FACILITY) ID(userid) ACCESS(ALTER)</td>
</tr>
</tbody>
</table>

**NOTE**

To refresh in-core facility authorizations, users must associate a user ID with the subsystem started task. Make this association in the started task table or with the RACF ADDUSER command. You then must permit this user ID to have alter access to the FACILITY resources.

To use RUV on a trial basis, you must define at least one RACF profile. If no SECURITY profile exists, access to all resources is denied.

**Enabling access**

To enable ALLOW_ALL access, enter the following command:

```
RDEF FACILITY BMC.RU(TM).FOR.VSAM.ALLOW.ALL.ACCESS + UACC(ALTER)
```

If you define the resource with this command, processing proceeds as follows:

- If the stated resource is the only (applicable) resource profile that is found, all access is allowed.
- If a profile exists for new resources, that profile is used as the basis for allowing or denying users access.
- If no profile exists, another check is made to determine whether the new resource exists in the facility class. If the resource exists in the facility class, the user is granted access to it at the UACC (specified) level. Otherwise, access is denied.

If you accidentally leave the ALLOW_ALL profile defined but decide that you want to protect entry with real resource profiles, control is given to the predefined profiles and is passed to the ALLOW_ALL profile only if the real resource profile does not exist. If the real resource profile exists but denies access to the user, the user is denied access even if the ALLOW_ALL profile exists.

If you do not define any profiles, the request is denied.
To implement ACF2 with RUV, use the sample control cards that are contained in HLQ.CNTL member ACF2INFO.

To implement Top Secret with RUV, use the sample control cards that are contained in HLQ.CNTL member TSSINFO.
Performing implementation tasks

To use RUV, you must start the RUV subsystem. If you are using the CICS Backup-While-Open (BWO) feature, you must define the relevant CICS resources as described below. You do not need to define CICS resources if you are not using CICS and use the RUV Batch Journalling Facility (BJF).

Table 4 lists requirements and tasks for customizing and completing RUV implementation.

Table 4  Implementation tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>RUV subsystems</td>
<td></td>
</tr>
<tr>
<td>starting BMCP and BCSS subsystems(^a)</td>
<td>“Starting BMCP and BCSS subsystems” on page 39</td>
</tr>
<tr>
<td>Note: This task allows RUV, BWO, and BJF to function properly.</td>
<td></td>
</tr>
<tr>
<td>BWO feature</td>
<td></td>
</tr>
<tr>
<td>defining transactions and programs to CICS(^b)</td>
<td>“Defining transactions and programs to CICS” on page 41</td>
</tr>
<tr>
<td>customizing CICS startup and shutdown(^b)</td>
<td>“Customizing CICS startup and shutdown” on page 42</td>
</tr>
<tr>
<td>verifying CICS initialization parameters(^b)</td>
<td>“Verifying CICS initialization parameters” on page 43</td>
</tr>
<tr>
<td>updating CICS startup JCL(^b)</td>
<td>“Updating CICS startup JCL” on page 44</td>
</tr>
<tr>
<td>defining CICS file resources(^c)</td>
<td>“Coding CICS file definitions” on page 45</td>
</tr>
<tr>
<td>initializing and enabling BWO(^b)</td>
<td>“Initializing and enabling the BWO feature” on page 51</td>
</tr>
<tr>
<td>BJF feature</td>
<td></td>
</tr>
<tr>
<td>initializing and enabling logging(^d)</td>
<td>“Initializing and enabling the Logging feature” on page 54</td>
</tr>
<tr>
<td>SNAPSHOT UPGRADE FEATURE (SUF)</td>
<td></td>
</tr>
<tr>
<td>installing and implementing SUF</td>
<td>“Installing and implementing the SNAPSHOT UPGRADE FEATURE” on page 58</td>
</tr>
</tbody>
</table>

\(^a\) required  
\(^b\) required only if you will be using BWO  
\(^c\) required if you must recover CICS VSAM files (If BWO is implemented, the CICS file definition must have browse and update attributes.)  
\(^d\) required only if you must initialize and enable batch journaling
Implementation scenarios

Table 5 lists several implementation scenarios and the tasks that you might have to perform.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>entire RUV product</td>
<td>defining transactions and programs to CICS</td>
</tr>
<tr>
<td></td>
<td>customizing CICS startup and shutdown</td>
</tr>
<tr>
<td></td>
<td>verifying CICS initialization parameter</td>
</tr>
<tr>
<td></td>
<td>updating CICS startup JCL</td>
</tr>
<tr>
<td></td>
<td>coding CICS file definitions</td>
</tr>
<tr>
<td></td>
<td>initializing and enabling BWO</td>
</tr>
<tr>
<td></td>
<td>initializing and enabling logging</td>
</tr>
<tr>
<td>BWO feature</td>
<td>defining transactions and programs to CICS</td>
</tr>
<tr>
<td></td>
<td>customizing CICS startup and shutdown</td>
</tr>
<tr>
<td></td>
<td>verifying CICS initialization parameter</td>
</tr>
<tr>
<td></td>
<td>updating CICS startup JCL</td>
</tr>
<tr>
<td></td>
<td>coding CICS file definitions</td>
</tr>
<tr>
<td></td>
<td>initializing and enabling BWO</td>
</tr>
<tr>
<td>BJF</td>
<td>initializing and enabling logging (non-CICS)</td>
</tr>
<tr>
<td>recovery of CICS</td>
<td>coding CICS file definitions</td>
</tr>
<tr>
<td>files</td>
<td>initializing and enabling BWO</td>
</tr>
<tr>
<td>SUF</td>
<td>installing, customizing, and implementing SUF</td>
</tr>
</tbody>
</table>

**NOTE**

You do not need to define any CICS resources or modify CICS tables if you are not using CICS or if you have no recoverable files defined to CICS.
Starting BMCP and BCSS subsystems

This task starts two BMC Software standard subsystems for resource management: the BMC Primary Subsystem (BMCP) and the BMC Consolidated Subsystem (BCSS). Refer to the relevant BMCP and BCSS manuals available on the BMC product support site.

Overview

To use RUV, you must start the BMCP and BCSS subsystems. RUV and other BMC Software products share the BMCP; you can have only one copy of the BMCP on a z/OS system. You must designate one (and only one) BCSS on a z/OS image as a public BCSS. All other BCSSs must be designated as private subsystems.

A major benefit of using a subsystem architecture is that virtual storage requirements in user address spaces are reduced significantly. The RUV subsystem architecture offers an additional benefit: increased ability to ensure data integrity.

BMCP and BCSS subsystems are maintained independently of RUV and other BMC Software products. BMCP and BCSS maintenance levels do not correspond to the maintenance levels of RUV. When you install a new or updated product that uses the BMCP and the BCSS, use the Installation Check program to compare the level of the existing subsystems to the level of the subsystems from the distribution tape.

Within the RUV ISPF interface, you can select option 8, “Subsystem detail,” to obtain relevant subsystem information or you can obtain a report through a batch command. Always use the highest available level of subsystem code.

For all parts of RUV to function, the subsystems must be started.

Subsystem initialization

This section describes how to initialize BMCP and BCSS subsystems. The BMCP interacts with the BCSS and the z/OS operating system. The BCSS supports RUV and its BWO and Logging features.

NOTE

The BMCP and the BCSS are required for implementation and use of RUV and for access to the REGISET repositories. Required RUV records must be present in the REGISET for correct access to records when you use the ISPF interface and for correct processing during application program execution.

It is recommended that you initialize two REGISET repositories (REGISET, REGISET2). For more information about the relationships between the BMCP and BCSS and the RUV repository, see the Recovery Utility for VSAM User Guide.
The BMCP is initiated first; then the BCSS is initiated. When the BCSS is running, the BMCP may be terminated.

**TIP**

It is recommended that you start the BCSS as a PRIVATE copy, which uses the default SSID RUVS. The procedure to create a PRIVATE subsystem is called BCSSRUVS$ and is located in the CNTL library.

To initialize BMCP and BCSS subsystems, perform the following steps:

1. If you changed the default PROC name during installation, specify the correct PROC name.

**NOTE**

BMCP is the default PROC name for the BMCP subsystem procedure. BCSS is the default PROC name for the BCSS subsystem procedure.

2. To start the BMCP subsystem, enter the following command at the z/OS console:

   S BMCP

   **NOTE**

   BMCP is the default subsystem ID for the BMCP subsystem. Only one BMCP subsystem can execute on a z/OS image.

When the BMCP subsystem is successfully started, the following message is displayed:

BMC10101I Subsystem ready. BMCP

**NOTE**

If the BCSS is already installed, step 3 may not be required.

3. To start the PRIVATE RUVS subsystem, enter the following command at the z/OS console:

   S BCSSRUVS$
When the RUV$ subsystem is started successfully, the following message is displayed:

```
BMC111001 Subsystem ready. RUV$
```

### Defining transactions and programs to CICS

This task is required only if you want to use the RUV BWO feature.

Depending on your CICS version, use the correct member to add CICS program and transaction resource definitions. If you change the default transaction IDs, modify the appropriate members of the HLQ.CNTL data set to define RUV transactions and programs to CICS.

**NOTE**

If you do not use an RDO definition that matches your CICS level, your CICS transactions and/or CICS region may abend.

If you use RDO to add transactions and programs, define RUV transactions and programs to CICS by modifying the following member:

- VRMRD31 (CICS Transaction Server version 3.1)
- VRMRD32 (CICS Transaction Server version 3.2)
Customizing CICS startup and shutdown

This task is required only if you want to use the RUV BWO feature. RUV provides the BWO feature with a high level of data integrity. RUV uses the CICS region to read the VSAM file, and RUV writes journals and updates before the backup is complete.

**NOTE**

RUVB8020, the BWO CICS Controller, communicates with the RUV subsystem and is required in the PLTPI and PLTSD.

You must start the BCSS subsystem before starting CICS, or you can start the RUV subsystem immediately after a z/OS system IPL. To initialize and enable the Backup subsystem, see “Initializing and enabling the BWO feature” on page 51.

An SVC number is dynamically selected to support the BWO feature. To use a predetermined SVC number, see “Initializing and enabling the BWO feature” on page 51.

Before you can use the BWO feature, you must authorize the RUV load libraries:

1. Based on the version of CICS that you are using, add the appropriate entries to your PLTPI and PLTSD tables. Sample PLTPI and PLTSD statements are supplied in the RUVPLTPI and RUVPLTSD members of the HLQ.CNTL library.

**WARNING**

The order and placement of the PLTPI and PLTSD statements with respect to the DFHDELIM statement is important. When a SHUT IMMEDIATE must be performed, the PLTSD cannot terminate RUV subtasks. Before issuing the CEMT SHUTDOWN,I command, you must use the RUVM shutdown transaction to shut down the BWO connection. For more information, see “Terminating the CICS connection” on page 54.
Figure 1 shows sample PLTPI entries, and Figure 2 shows sample PLTSD entries. Use of RUVB8020 in either type of entry is optional.

**Figure 1** PLTPI entries for all CICS versions

```
* PLTAPI DFHPLT TYPE=INITIAL, X
  SUFFIX=4A
* DFHPLT TYPE=ENTRY,PROGRAM=DFHDELIM
  DFHPLT TYPE=ENTRY,PROGRAM=RUVB8020
* DFHPLT TYPE=FINAL
  END
```

**Figure 2** PLTSD entries for all CICS versions

```
* PLTASD DFHPLT TYPE=INITIAL, X
  SUFFIX=A4
* DFHPLT TYPE=ENTRY,PROGRAM=RUVB8020
  DFHPLT TYPE=ENTRY,PROGRAM=DFHDELIM
* DFHPLT TYPE=FINAL
  END
```

For more information, see “Removing RUV from execution” on page 52.

2 Add the RUVPLTxx members into your DFHPLT assemblies, or use the RUVPLTxx members as COPY members.

**Verifying CICS initialization parameters**

This task is only required if the PLT tables containing your RUV entries are not currently coded in your System Initialization Table (SIT) or the SIT overrides.

Verify that the following parameters are specified in the SIT for CICS TS environments (cc is a valid CICS table suffix):

- PLTPI=cc
- PLTSD=cc

**NOTE**

The cc table suffix refers to the suffixes that were assigned in “Customizing CICS startup and shutdown” on page 42.
Updating CICS startup JCL

This task is required for reference to the load library at startup. CICS regions that run RUV must reference the LOAD library in CICS startup JCL to enable RUV features.

Before including your load library in your CICS startup JCL, consider the following factors:

- If you use two LOAD libraries (one APF authorized, and one not authorized) you must place both libraries in the DFHRPL list.

- Because RUV libraries that are loaded through the DFHRPL DD statement do not require APF authorization, concatenating the RUV authorized LOAD library with unauthorized libraries will not affect operation.

**NOTE**

REGISET and REGISET2 repositories (created at installation) are used by the BCSS but are not required to be included in the CICS startup JCL.

Add the following statements to the concatenation of data sets in the DFHRPL and STEPLIB concatenations in your CICS startup JCL:

```
//DFHRPL DD DISP=SHR,DSN=yourname.LOAD
//STEPLIB DD DISP=SHR,DSN=yourname.LOAD
```
Coding CICS file definitions

This task defines the recovery attributes for files that are managed in a CICS-only transaction server environment or an ESA environment.

Transaction server environment

In a transaction server environment you specify recovery options, including forward recovery, in the Integrated Catalog Facility (ICF) catalog (if you are using DFSMS version 1.3 or later) or in the CICS file resource definition. You can use either of the following methods to code the CICS file definitions in a transaction server environment:

- If your VSAM data sets are accessed by CICS in a non-record-level-sharing (non-RLS) mode, you can define recovery attributes in the CICS file resource definition.

- If your VSAM data sets are accessed by CICS in record-level-sharing (RLS) mode, you must define the recovery attributes in the ICF catalog.

CICS uses the recovery attributes from the ICF catalog only when you specify RLS=YES as a system initialization parameter. If you specify RLS=NO, recovery attributes are always taken from the CICS file resource definition.

**TIP**

Before you archive the log stream, it is recommended that you enter the CEMT SET JOU(DFHJnn) FLUSH command. For more information, see the IBM CICS Supplied Transactions manual.

Non-RLS mode

RUV can archive recovery records when you use the CICS file definition options in the following example to capture forward recovery and backout recovery records.

To archive the log stream of user journals 01 through 99, use the RUV ARCHIVE command (data source formats).

Example:

```
FWDRECOVLOG(nn) == nn - forward recovery journal
RECOVERY(ALL)
```
Using the preceding CICS file definition options, you will get a forward recovery archive summary that is similar to the following example for file TEST01, which was updated with five add records, two update records, and one delete record:

<table>
<thead>
<tr>
<th>Fileid: TEST01</th>
<th>DSN: BMCRCM.DEVB.TEST01</th>
</tr>
</thead>
<tbody>
<tr>
<td>BACKOUT RECORDS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 Adds</td>
</tr>
<tr>
<td></td>
<td>0 Updates</td>
</tr>
<tr>
<td></td>
<td>0 Deletes</td>
</tr>
<tr>
<td>FORWARD RECORDS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5 Adds</td>
</tr>
<tr>
<td></td>
<td>2 Updates</td>
</tr>
<tr>
<td></td>
<td>1 Delete</td>
</tr>
</tbody>
</table>

In the preceding example and the following non–RLS mode examples, BACKOUT RECORDS and FORWARD RECORDS are the number of transactions that are added, updated, or deleted by CICS and that are recorded by the CICS Journaling Facility.

Example:

```yaml
FWDRECOVLOG(NO)
RECOVERY(NONE)
JNLREAD(UPDATEONLY)
JNLUPDATE(YES)
JNLADD(BEFORE)
JOURNAL(nn)
```

Using the preceding CICS file definition options, you will get a backout recovery archive summary that is similar to the following example for file TEST02, which was updated with five add records, two update records, and one delete record:

<table>
<thead>
<tr>
<th>Fileid: TEST02</th>
<th>DSN: BMCRCM.DEVB.TEST02</th>
</tr>
</thead>
<tbody>
<tr>
<td>BACKOUT RECORDS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5 Adds</td>
</tr>
<tr>
<td></td>
<td>2 Updates</td>
</tr>
<tr>
<td></td>
<td>1 Delete</td>
</tr>
<tr>
<td>FORWARD RECORDS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 Adds</td>
</tr>
<tr>
<td></td>
<td>0 Updates</td>
</tr>
<tr>
<td></td>
<td>0 Deletes</td>
</tr>
</tbody>
</table>
Using the preceding CICS file definition options, you will get a forward and backout recovery archive summary that is similar to the following example for file TEST03, which was updated with five add records, two update records, and one delete record:

**RLS mode**

If you specify `RLSACCESS(YES)` in the CICS file definitions, some attributes such as `RECOVER` and `FWDRECOVLOG` are ignored by transaction server. Recovery attributes are obtained from the ICF catalog for an RLS file. The `FWDRECOVLOG` log stream name is obtained from the ICF catalog for an RLS file.

To archive the log stream of user journals 01 through 99, use the RUV ARCHIVE command (data source formats).

**Example:**

```sql
DEFINE CLUSTER
  LOGSTREAMID(lsn)
  LOG(ALL)
```
Using the preceding IDCAMS options, you will get a forward recovery archive summary that is similar to the following example for file TEST04, which was updated with five add records, two update records, and one delete record:

<table>
<thead>
<tr>
<th>Fileid: TEST04</th>
<th>DSN: BMCRCM.DEVB.TEST04</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BACKOUT RECORDS</strong></td>
<td></td>
</tr>
<tr>
<td>0 Adds</td>
<td></td>
</tr>
<tr>
<td>0 Updates</td>
<td></td>
</tr>
<tr>
<td>0 Deletes</td>
<td></td>
</tr>
<tr>
<td><strong>FORWARD RECORDS</strong></td>
<td></td>
</tr>
<tr>
<td>5 Adds</td>
<td></td>
</tr>
<tr>
<td>2 Updates</td>
<td></td>
</tr>
<tr>
<td>1 Delete</td>
<td></td>
</tr>
</tbody>
</table>

In the preceding example and the following RLS mode examples, BACKOUT RECORDS and FORWARD RECORDS are the number of transactions that are added, updated, or deleted by CICS and that are recorded by the CICS Journaling Facility.

Example:

| JOURNAL($n) |
| JNLREAD(UPD) |
| JNLUPDATE(YES) |
| JNLDADD(BEFORE or AFTER) |
| DEFINE CLUSTER |
| LOG(ALL) |
| LOGSTREAMID($sn) |

Using the preceding CICS file definition and IDCAMS options, you will get a forward and backout recovery archive summary that is similar to the following example for file TEST05, which was updated with five add records, two update records, and one delete record:

<table>
<thead>
<tr>
<th>Fileid: TEST05</th>
<th>DSN: BMCRCM.DEVB.TEST05</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BACKOUT RECORDS</strong></td>
<td></td>
</tr>
<tr>
<td>5 Adds</td>
<td></td>
</tr>
<tr>
<td>2 Updates</td>
<td></td>
</tr>
<tr>
<td>1 Delete</td>
<td></td>
</tr>
<tr>
<td><strong>FORWARD RECORDS</strong></td>
<td></td>
</tr>
<tr>
<td>5 Adds</td>
<td></td>
</tr>
<tr>
<td>2 Updates</td>
<td></td>
</tr>
<tr>
<td>1 Delete</td>
<td></td>
</tr>
</tbody>
</table>
Example:

```
JOURNAL(nn)
JNLUPDATE(ALL)
JNLADD(ALL)

DEFINE CLUSTER
LOG(ALL)
LOGSTREAMID(\$sn)
```

Using the preceding CICS file definition and IDCAMS options, you will get a forward and backout recovery archive summary that is similar to the following example for file MXNRLS2, which was updated with five add records, one update record, and one delete record:

```
<table>
<thead>
<tr>
<th>Fileid: MXNRLS2</th>
<th>DSN: BMCRCM.DEVB.MXNRLS2</th>
</tr>
</thead>
<tbody>
<tr>
<td>BACKOUT RECORDS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10 Adds</td>
</tr>
<tr>
<td></td>
<td>3 Updates</td>
</tr>
<tr>
<td></td>
<td>1 Deletes</td>
</tr>
<tr>
<td>FORWARD RECORDS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5 Adds</td>
</tr>
<tr>
<td></td>
<td>1 Updates</td>
</tr>
<tr>
<td></td>
<td>1 Deletes</td>
</tr>
</tbody>
</table>
```

---

**NOTE**

Double backout records (before-image and after-image records) are created with the preceding CICS file definition options for the transaction server.
CICS TS environment

To code CICS file definitions for a CICS TS environment, define the parameters for the CICS TS file definitions as shown in Figure 3.

Figure 3  CICS file definition options for CICS TS environments

CEDA DEFINE FILE(fileid) GROUP(groupname)
.  
  FWDRECOVLOG(nn)  <= nn - forward recovery journal
  RECOVERY(ALL)
  .  

NOTE
To recover data sets by using the recovery utility, you must specify FWDRECOVLOG(nn) and RECOVERY(ALL).

Table 6 describes the parameters for the CICS TS file definitions. In the CICS TS environment or the transaction server environment, you can define files by using Resource Definition Online (RDO). For information about using RDO, see the IBM CICS Transaction Server Resource Definition (Online) manual.

NOTE
Underlined parameters are CICS defaults and will not ensure forward or backout recovery. These parameters apply to logging and journaling only and do not include all the parameters that you can define for the CICS file definition options. The rest of the entries in the CICS file definitions are table definitions for managing VSAM journal records that are generated by CICS.

Table 6  CICS file definition parameters for CICS TS environments (Part 1 of 2)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILE(fileid)</td>
<td>identifies the file that is defined in this CICS file definition entry</td>
</tr>
<tr>
<td></td>
<td>The name must be from one to eight characters. The recovery utility uses</td>
</tr>
<tr>
<td></td>
<td>this file name to correlate a data set and the corresponding journal</td>
</tr>
<tr>
<td></td>
<td>records during a recovery.</td>
</tr>
<tr>
<td>FWDRECOVLOG(NO</td>
<td>nn)</td>
</tr>
<tr>
<td></td>
<td>This identifier is used if ALL was specified on the RECOVERY parameter.</td>
</tr>
<tr>
<td></td>
<td>The parameter nn writes journal records to user journal nn, a range of</td>
</tr>
<tr>
<td></td>
<td>numbers from 01 through 99.</td>
</tr>
</tbody>
</table>
Chapter 2 Configuring Recovery Utility for VSAM

Initializing and enabling the BWO feature

This task is required if you want to use the RUV BWO feature.

The RUV product subsystem provides inter-address space communication services that allow a batch job to invoke the BWO feature for a VSAM file which is allocated and open to an online CICS region. RUV provides BWO capability with a high level of data integrity by using the CICS region to read the VSAM file and by writing journals and updates to it before backup is complete.

**NOTE**

To use a predetermined SVC number, you can use the RUVSAF macro. See examples in the RUVZFC member of the HLQ.SOURCE library.

Before you can activate RUV, you must APF-authorize its libraries. You can configure RUV so that you can initialize the BWO feature without an IPL.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>JNLADD(ALL)</td>
<td>defines the ADD operations that you want to be recorded on the journal which is specified by the JOURNAL parameter. ALL journals the file control WRITE operation before and after the VSAM I/O operation.</td>
</tr>
<tr>
<td>JNLREAD(UPDATEONLY)</td>
<td>defines the read operations that you want to be recorded on the journal which is specified by the JOURNAL parameter. UPDATEONLY journals only READ UPDATE operations.</td>
</tr>
<tr>
<td>JNLUPDATE(YES)</td>
<td>specifies that you want REWRITE and DELETE operations recorded on the journal which is specified by the JOURNAL parameter.</td>
</tr>
<tr>
<td>JOURNAL(nn)</td>
<td>indicates whether you want automatic journaling for the file, and identifies the journal to which journal records are written. The parameter <em>nn</em> writes journal records to user journal <em>nn</em>, a range of numbers from 01 through 99. Automatic journaling is independent of logging that is performed to the system and forward recovery logs, as specified by RECOVERY and FWDRECOVLOG parameters.</td>
</tr>
<tr>
<td>RECOVERY(ALL)</td>
<td>defines recovery requirements for this file. ALL writes before-images to the system log and after-images to the journal which is specified in the FWDRECOVLOG parameter.</td>
</tr>
</tbody>
</table>
The BWO feature must be enabled. To enable the feature, use the operator commands that are described in the *Recovery Utility for VSAM User Guide*.

The BWO feature backs up files while they are open to CICS. To make files eligible for this type of processing, define them to CICS in the CICS file definitions with the BROWSE and UPDATE service request.

CICS and RUV messages can help you determine whether the Backup subsystem was installed properly. During CICS initialization, the following messages are written to the CSMT log when a connection is established between CICS and the RUV product subsystem:

<table>
<thead>
<tr>
<th>Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC202884I</td>
<td>RUV applid Initialization of the connection to the Backup Subsystem (subsys_id) is in progress.</td>
</tr>
<tr>
<td>BMC202847I</td>
<td>RUV applid taskid backup subtask is active.</td>
</tr>
</tbody>
</table>

When an RUV backup job is run, the following message is issued to the backup job SYSPRINT to verify the communication between the job and the CICS:

<table>
<thead>
<tr>
<th>Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC202751I</td>
<td>RUV A Backup Subsystem connection to applid has been established to back up fileid.</td>
</tr>
</tbody>
</table>

CICS verifies the Backup subsystem communication link by writing the following messages to CSMT:

<table>
<thead>
<tr>
<th>Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BMC202850I</td>
<td>RUV APPLID &lt;applid&gt; Taskid &lt;taskid&gt; backup subtask has opened FCT=fileid DSN=dataset.</td>
</tr>
<tr>
<td>BMC202851I</td>
<td>RUV APPLID &lt;applid&gt; Taskid &lt;taskid&gt; backup subtask completed for DDName &lt;fileid&gt;. Records &lt;rcdcnt&gt;.</td>
</tr>
<tr>
<td>BMC202847I</td>
<td>RUV APPLID &lt;applid&gt; Taskid &lt;taskid&gt; backup subtask is active.</td>
</tr>
</tbody>
</table>

If you do not receive these messages, check your CICS table definitions. Ensure that you have completed the steps for initializing the subsystem.

**Removing RUV from execution**

If RUV is not used with a particular CICS, remove RUVB8020 from the PLTPI and PLTSD tables.
Reinitializing the BWO feature

To reinitialize and start the BWO feature without an IPL (for example, after installing a new release of RUV), enter the following commands from a TSO SDSF session or operator console:

/ssid REINIT RUV

/ssid RUV ENABLE BACKUP

For other RUV operator commands, see the *Recovery Utility for VSAM User Guide*.

Controlling the BWO feature

RUV provides a command interface that you can use to control the BWO feature from a CICS region. The command interface can be executed as a CICS transaction by entering an RUVM transaction from a CICS terminal, by entering it from a z/OS console that is defined to CICS, or by using the z/OS MODIFY command.

Starting the CICS connection

To start the BWO control subtask within CICS that communicates with the Backup subsystem, enter the following Backup subsystem CICS transaction (CICS transid) from your CICS terminal:

RUVM START

To start the CICS connection, you can also enter the following command from your z/OS system console:

F cicsjob,RUVM START

Displaying BWO status

To display the status of any active BACKUP control subtask within CICS that communicates with the Backup subsystem, enter the following Backup subsystem CICS transaction (CICS transid) from your CICS terminal:

RUVM STATUS
You can also enter the following commands from your z/OS system console or SDSF to display the status of the CICS connection:

```
F cicsjob,RUVM STATUS
/ssid RUV STATUS BACKUP
```

**Terminating the CICS connection**

When CICS is terminated normally, the RUVB8020 program entry in the PLTSD causes the CICS BWO subtask to terminate.

To stop the CICS connection, you can enter the following command from your z/OS system console:

```
/ssid RUVM SHUTDOWN
```

**Initializing and enabling the Logging feature**

This task is required if you want to use the RUV Logging feature. You can perform this task by using the RUV ISPF interface from the z/OS console or by running a batch job.

The Logging feature journals changes to VSAM files as the files are updated by batch application programs. The Logging feature gives you the same journaling capability for your batch VSAM processing that CICS gives to online users. You control the types, destinations, and methods of creating journal records through RUV rule sets and their components.

The Logging feature uses rule sets to define and control journaling for VSAM batch jobs. The rule sets control which VSAM data sets to journal and which processing parameters to use. For more information, see “Using rule sets” on page 81.

To initialize and enable the Logging feature, perform the following steps:

1. Ensure that the subsystem has been started and is available.

2. Enable the Logging feature by using the RUV ISPF interface or by entering the following command:

```
/ssid RUV ENABLE LOGGING
```

You may want to use a Job Level rule set, which is active for only the batch job in which it is activated. For more information, see the *Recovery Utility for VSAM User Guide*. 
3 Use a batch job to activate the Logging feature by using a rule set, or use the ISPF interface.

For instructions about building rule sets and components, see the Recovery Utility for VSAM User Guide.

4 To check logging status, enter the following command:

    /ssid RUV STATUS LOGGING

---

### NOTE

To route subsystem messages to a different destination, you can use the RUVSAF macro. See examples in the RUVZFAC member of the `HLQ.SOURCE` library.

---

### TIP

It is recommended that you run IVP processes to determine whether RUV is functioning properly. For more information, see “Using the installation verification procedure” on page 60.

---

## Using the RUVZFAC macro

You can use the RUVZFAC macro to change the following RUV elements:

- Backup-While-Open (BWO) feature SVC (to be dynamic or specific)
- RACROUTE RESPONSE return code
- RACROUTE CLASS name
- route and description codes for RUV subsystem messages

---

### NOTE

The RUVSAF macro applies only to the ACF2 facility. The RUVSAF macro should be included in your SYSLIB library concatenation during the assembly process.
Using the RUVZFAC macro

Code the macro as shown in the following example:

```plaintext
RUVZFAC RUvfsAF TYPE=INITIAL
  RUvfsAF TYPE=BWO, =>
    BWOSVC=249
  RUvfsAF TYPE=SAF, =>
    CLASS=FACILITY, =>
    RESPONSE=0
  RUvfsAF TYPE=WTO, =>
    ROUTCDE=(11), =>
    DESC=6
  RUvfsAF TYPE=FINAL
END
```

The TYPE=INITIAL statement must be the first statement, and the TYPE=FINAL statement must be the last statement. The macros (and other sample source members) are distributed in the HLQ.SOURCE library.

For example, to change the BWO feature to use SVC 249:

```plaintext
RUvfsAF TYPE=BWO, =>
  BWOSVC=249
```

For example, to change the BWO feature to find the first available Type 3 SVC:

```plaintext
RUvfsAF TYPE=BWO, =>
  BWOSVC=DYNAMIC
```

For example, to disable the BWO feature, specify “NONE” on the BWOSVC statement:

```plaintext
RUvfsAF TYPE=BWO, =>
  BWOSVC=None
```

RUV expects return code 0 from the RACROUTE authorization check. The following example changes the return code to 4:

```plaintext
RUvfsAF TYPE=SAF, =>
  CLASS=FACILITY, =>
  RESPONSE=4
```
The following example changes the RACROUTE Class name that RUV uses from the default name FACILITY to FAC:

| RUVSAF TYPE=SAF, | ==> |
| CLASS=FAC, | ==> |
| RESPONSE=0 |

To modify route and description codes for RUV subsystem messages, change the values on ROUTCDE and DESC under TYPE=WTO. For example:

| RUVSAF TYPE=WTO, | ==> |
| ROUTCDE=(11), | ==> |
| DESC=6 |
Installing and implementing the SNAPSHOT UPGRADE FEATURE

RUV includes the BMC Software SNAPSHOT UPGRADE FEATURE (SUF) at no additional cost. SUF enables RUV to produce Instant Snapshot copies of VSAM data sets. SUF provides a subset of the functions and features of the BMC Software EXTENDED BUFFER MANAGER (XBM) product.

SUF Requirements

To use SUF with RUV you must install, customize, and implement SUF. SUF is packaged for distribution with XBM and is installed by using the XBM installation process. A separate product authorization password for SUF is not required if you are using SUF with RUV. You can use the password for RUV to authorize the components that are required by SUF for snapshot processing:

- SUF component that enables Instant Snapshot functions
- Storage Systems Integration (SSI) component that enables the use of intelligent storage devices for snapshot processing

For information about installing, customizing, and implementing SUF, see the following books:

- EXTENDED BUFFER MANAGER and SNAPSHOT UPGRADE FEATURE Installation and Customization Guide
- EXTENDED BUFFER MANAGER and SNAPSHOT UPGRADE FEATURE User Guide
SUF Authorization

When you perform the product authorization step for SUF, you need a product authorization password. You install the password in a password module and place the password module in the XBM load library.

A permanent password module is named `pppTBL3P`, and a temporary password module is named `pppTBL3T`. `ppp` can be XBC, XBA, or VRU when you are authorizing SUF processing for RUV. These passwords enable total SUF functionality for snapshot processing, including the XBM SSI component.

You should copy the password module that has the latest expiration date into the XBM load library to ensure product authorization for the longest term. If the XBM password and the VRU password have the same expiration dates, use the VRU password. Then, when the VRU password is renewed, you must replace only one password instead of two.

Whichever type of password you use, do not use a combination of temporary (`pppTBL3T`) and permanent (`pppTBL3P`) password modules. Expiring temporary passwords cause a product to issue warning messages that can cause doubt about the true authorization status of the product when a permanent password is also present. You should delete (or rename) the temporary password module from the load library when you are adding a permanent password.
Using the installation verification procedure

When you have installed RUV and have performed post-installation tasks, and RUV is ready to use, consider performing the installation verification procedure (IVP) in Table 7 on page 61. The IVP is a set of jobs with manual steps to guide you through several production scenarios. The IVP ensures a successful installation and introduces RUV features.

**NOTE**
Perform the IVP only when all required installation and implementation steps have been completed.

**TIP**
If your site coded ISP control data sets (ISPCTLxx) in the TSO logon procedure, use RUVISPFZ to launch the RUV online interface.
Complete the steps in Table 7 in order. All members are in the CNTL library that was installed from the RUV product distribution tape.

### Table 7  
**Installation verification procedure (Part 1 of 3)**

<table>
<thead>
<tr>
<th>Step</th>
<th>Member</th>
<th>Task</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IVP$CHGS</td>
<td>Edit and modify the part of IVP$CHGS that is indicated in the listing, and save it.</td>
<td>IVP$CHGS is ready for step 2.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> To view user variable changes that are required for IVP$CHGS, see Figure 4 on page 64.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>IVP$CHGS is an EDIT macro that is invoked by the IVP$EDIT REXX EXEC which modifies IVP members in the CNTL library for the IVP jobs. The top half contains several ISPF CHANGE variables that you must modify to reflect your environment and site requirements.</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>IVP$EDIT</td>
<td>Enter the following TSO command:</td>
<td>The remaining IVP members are modified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>TSO EX 'yourname.CNTL(IVP$EDIT)</strong>**</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> After performing steps 1 and 2, you do not have to modify the remaining members; just submit them. Members IVP$JB04 (step 6), IVP$JB13 (step 15), and IVP$JB15 (step 17) are text members that contain instructions for more steps. Member IVP$EDIT invokes all required JCL changes, except for those noted. The remaining executable members must be submitted in order.</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>IVP$JB01</td>
<td>Submit the member for processing.</td>
<td>BMCP and BCSS procedures are copied to your PROCLIB.</td>
</tr>
<tr>
<td>4</td>
<td>IVP$JB02</td>
<td>Submit the member for processing.</td>
<td>IVP REGISET and REGISET2 files for the subsystem repository are defined. (ssid=RUV$)</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> Before performing steps 5 and 6, see “Setting up security” on page 34.</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>IVP$JB03</td>
<td>Submit the member for processing.</td>
<td>VSAM files and sequential files for copies of VSAM files are defined.</td>
</tr>
<tr>
<td>6</td>
<td>IVP$JB04</td>
<td>Read the IVP$JB04 text, and follow the instructions.</td>
<td>The IVP$JB04 text instructs users how to start BMCP and BCSS subsystems for the IVP. The REINIT command is executed from the COMMANDS data set.</td>
</tr>
<tr>
<td>7</td>
<td>IVP$JB05</td>
<td>Submit the member for processing.</td>
<td>A rule set is created and is added to the RUV repository.</td>
</tr>
</tbody>
</table>
I Dod the installation verification procedure

**Table 7  Installation verification procedure (Part 2 of 3)**

<table>
<thead>
<tr>
<th>Step</th>
<th>Member</th>
<th>Task</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>IVP$JB06</td>
<td>Submit the member for processing.</td>
<td>RUV performs a backup of VSAM files (backup #1).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Record the “Registration Time” of the VSAM file KSDS01, which may be</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>obtained from the JES listing for this job or from the ISPF</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>interface by using selection 5.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> You will use this time in Step 17.</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>IVP$JB07</td>
<td>Submit the member for processing.</td>
<td>RUV performs batch journaling #1.</td>
</tr>
<tr>
<td>10</td>
<td>IVP$JB08</td>
<td>Submit the member for processing.</td>
<td>RUV performs VSAM file backup #2.</td>
</tr>
<tr>
<td>11</td>
<td>IVP$JB09</td>
<td>Submit the member for processing.</td>
<td>RUV performs batch journaling #2.</td>
</tr>
<tr>
<td>12</td>
<td>IVP$JB10</td>
<td>Submit the member for processing.</td>
<td>RUV performs batch journaling #3.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Step JNLS0E4 forces an S0C4 abend, so a recovery job will be</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>created and saved as member IVP$JB11. This recovery job will</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>back out only the changes that are made in JNLS0E4, but not the</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>previous steps.</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>IVP$JB11</td>
<td>Review this jobstream.</td>
<td>IVP$JB11 is created when IVP$JB10 runs and abends.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Submit the member to back out the changes that have been made in</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>the abended step of the previous job.</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>IVP$JB12</td>
<td>Submit the member for processing.</td>
<td>RUV provides an output report for the following commands:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ REPORT ARCHIVE_FILE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ REPORT BACKUP_FILE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ REPORT VSAM_FILE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>■ REPORT RULE_SET</td>
</tr>
<tr>
<td>15</td>
<td>IVP$JB13</td>
<td>Invoke the RUV ISPF interface.</td>
<td>The RUV ISPF interface is invoked.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Exercise the following options:</td>
<td>These options result in job JCL that is generated for forward recovery.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Browse options 2, 4, and 5 to view the items that were registered</td>
<td><strong>Note:</strong> For more information about these options, see member</td>
</tr>
<tr>
<td></td>
<td></td>
<td>during the previous IVP jobs.</td>
<td>IVP$JB13 in the sample library</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Select option 1 to perform a VSAM forward recovery.</td>
<td><strong>HLQ.CNTL.</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>After recovery, compare the resulting reports with the printout of</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>the file.</td>
<td></td>
</tr>
</tbody>
</table>
### Table 7  Installation verification procedure (Part 3 of 3)

<table>
<thead>
<tr>
<th>Step</th>
<th>Member</th>
<th>Task</th>
<th>Result</th>
</tr>
</thead>
</table>
| 16   | IVP$JB14| *(optional)* Change the SET parameter to INACTIVE status in the repository records for backup #2 (the three KSDS VSAM files).  
  **Note:** This change is required if you ship a set of backup tapes offsite.  
  Submit the member for processing.                                      | RUV recoveries do not select these backups as being available for use. |
| 17   | IVP$JB15| **Invoke the RUV ISPF interface.**  
  Exercise the following options:  
  ▪ Select option 1 to generate backout recovery JCL.  
  ▪ Select option 2 to perform a VSAM backout recovery.  
  ▪ Type the registration time that was recorded in step 8 as the start time.  
  After recovery, compare the resulting reports with the printout of the file. | The RUV ISPF interface is invoked.  
These options result in a backout of all updates that have been made to the VSAM files in this set of IVP jobs. The files should have all the original data records.  
**Note:** For more information about these options, see member IVP$JB15 in the sample library HLQ.CNTL. |
| 18   | IVP$JB16| Submit the member for processing.                                                        | RUV executes miscellaneous repository commands.                        |
| 19   | IVP$JB99| *(optional)* Submit the member for processing.                                           | IVP DASD data sets are deleted.                                        |

**NOTE**

For resolutions to common installation and configuration issues, see Appendix D, “Resolutions for common installation and configuration issues.”
Using the installation verification procedure

Figure 4 shows user variable changes that are required for IVP$CHGS.

**Figure 4  User variable changes that are required for IVP$CHGS**

```plaintext
/*--------------------------------*/
/*  JES job card related.        */
/*--------------------------------*/
'ISREDIT change "<acct>"       "nnnn"                 ALL'
'ISREDIT change "<ssid>"       "RUV$"                 ALL'
'ISREDIT change "<jobnm6>"     "RUVivp"               ALL'
'ISREDIT change "<msgclass>"   "X"                    ALL'
'ISREDIT change "<jobclass>"   "A"                    ALL'
/*--------------------------------*/
/*  JES DD statement related.    */
/* Max of 22 chars on IVPHLQ. */
/*--------------------------------*/
'ISREDIT change "<sysout>"     "*"                    ALL'
'ISREDIT change "<RUVhlq>"     "yourname"             ALL'
'ISREDIT change "<ivphlq>"     "yourivp.name"         ALL'
'ISREDIT change "<ivpvol>"     "vol001"               ALL'
/*--------------------------------*/
/* Regiset/REGISET2 DASD volumes, */
/* ** Can NOT be the same !! ** */
/*--------------------------------*/
'ISREDIT change "<regisetvol>" "ivp001"                ALL'
'ISREDIT change "<regiset2vol>" "ivp002"               ALL'
/*--------------------------------*/
/* Miscellaneous items.         */
/*--------------------------------*/
'ISREDIT change "<proclib>"    "your.proclib.dsn"     ALL'
/*--------------------------------*/
/* * * * * * Change variables after this point */
/* * * * * * Change variables after this point */
/* * * * * * Change variables after this point */
/*---------------------------------------------------------------------*/
/* * * * * * Change variables before this point */
/* * * * * * Change variables before this point */
/* * * * * * Change variables before this point */
/*---------------------------------------------------------------------*/
```
VSAM files that are used by the IVP

Table 8 lists the VSAM files (and attributes) that are used by IVP members.

Table 8  VSAM files that are used by IVP members

<table>
<thead>
<tr>
<th>File</th>
<th>Record size</th>
<th>Keys</th>
</tr>
</thead>
<tbody>
<tr>
<td>KSDS01</td>
<td>(100 100)</td>
<td>(6 0)</td>
</tr>
<tr>
<td>KSDS02</td>
<td>(100 100)</td>
<td>(6 0)</td>
</tr>
<tr>
<td>KSDS03</td>
<td>(100 100)</td>
<td>(6 0)</td>
</tr>
</tbody>
</table>

Data is loaded with IDCAMS from CNTL library member IVP$DATA.
Allocated data sets

The following data sets are allocated to support this set of IVP tests:

- IVPHLQ.KSDS01
- IVPHLQ.KSDS02
- IVPHLQ.KSDS03
- IVPHLQ.KSDS01.BKUP1
- IVPHLQ.KSDS02.BKUP1
- IVPHLQ.KSDS03.BKUP1
- IVPHLQ.KSDS03.COPY
- IVPHLQ.KSDS03.BEFRECOV
- IVPHLQ.KSDS03.AFTRECOV
- IVPHLQ.KSDS01.BKUP2
- IVPHLQ.KSDS02.BKUP2
- IVPHLQ.KSDS03.BKUP2
- IVPHLQ.SSID.REGISET
- IVPHLQ.SSID.REGISET2
- IVPHLQ.JOBNM607.JNLSSEP1.JNL
- IVPHLQ.JOBNM607.JNLSSEP2.JNL
- IVPHLQ.JOBNM607.JNLSSEP3.JNL
- IVPHLQ.JOBNM609.JNLSSEP1.JNL
- IVPHLQ.JOBNM609.JNLSSEP2.JNL
- IVPHLQ.JOBNM609.JNLSSEP3.JNL
- IVPHLQ.JOBNM610.JNLSSEP1.JNL
- IVPHLQ.JOBNM610.JNLSSEP2.JNL
- IVPHLQ.JOBNM610.JNLSSEP3.JNL
- IVPHLQ.JOBNM610.JNLSSEP4.JNL
- IVPHLQ.JOBNM607.JNLSSEP1.LOG
- IVPHLQ.JOBNM607.JNLSSEP2.LOG
- IVPHLQ.JOBNM607.JNLSSEP3.LOG
- IVPHLQ.JOBNM609.JNLSSEP1.LOG
- IVPHLQ.JOBNM609.JNLSSEP2.LOG
- IVPHLQ.JOBNM609.JNLSSEP3.LOG
- IVPHLQ.JOBNM610.JNLSSEP1.LOG
- IVPHLQ.JOBNM610.JNLSSEP2.LOG
- IVPHLQ.JOBNM610.JNLSSEP3.LOG
- IVPHLQ.JOBNM610.JNLSSEP4.LOG
Migrating RUV from a test environment to a production environment

Before you migrate RUV to a production environment (or a product-acceptance testing environment), perform the following tasks to ensure that RUV functions in the new environment:

- Ensure that all references to product data sets indicate the name of the new data sets in the production environment where the product modules reside.

- Ensure that the products being migrated are authorized for use on the CPU where the production system resides. For questions about CPU authorization, contact your sales representative. For more information about the authorization process, see Appendix B, “BMC Software product authorization.”

- Ensure that the implementation tasks beginning with “Performing implementation tasks” on page 37 have been completed.

- To use the security facility that is provided with RUV, determine whether you want to use the internal security component. For more information about product security, see “Setting up security” on page 34.

- *(optional)* Place the load library in a load list.

- Use your site’s standard methods to copy the test libraries into the production environment.

- Initiate the RUV ISPF interface:

  A  On an ISPF panel that accepts TSO commands, type the following TSO command:

  
  TSO EX 'yourname.DLIB(RUVISPF)'

  This step is especially important if the systems at your site do not share DASD.

  B  Press Enter.
Migrating RECOVERY PLUS for CICS/VSAM to RUV

To help you with conversion from RECOVERY PLUS for CICS/VSAM (RPCV) to RUV, RUV provides the following utilities:

- RUVZCJC migrates JCT/OPT tables to RUV rule sets (see “RUVZCJC utility” on page 68).
- RUVZCDA migrates VSAM data set name, group definitions, and user variables from the ISICDS data sets (see “RUVZCDA utility” on page 70).

For similarities and differences between RPCV and RUV, see “Migration considerations” on page 74.

RUVZCJC utility

RPCV requires assembly of an ISIOPT and a VSAM DSN selection with ISIJCT. ISIOPT is loaded into the common service area (CSA) and is used to provide a Fast Path rejection of RPCV interest in a VSAM file. When a VSAM file is opened, ISIJCT is loaded into the address space. ISIJCT provides job, step, and program filters by DSN, journaling needs (before and/or after images), and journal allocation specifications. This manual process is cumbersome and error prone.

Creating RUV rule sets with RUVZCJC

RUVZCJC is an ISIJCT/ISIOPT migration utility that provides templates and models for converting your existing ISIJCT load module to RUV rule sets. RUVZCJC provides approximately 80 percent of your rule sets from the information in your existing ISIJCT load module.

NOTE

Because of fundamental differences between the ways that RPCV and RUV view journaling (one journal per VSAM cluster and one journal per JOB step), RUVZCJC cannot provide 100 percent of the information that RUV requires from the information in ISIJCT alone. When migration is complete, the resulting rule set will require modification for accurate results.
Figure 5 shows sample JCL for executing RUVZCJC. The source for this sample is located in HLQ.CNTL(CONVJCT). The ISIJCT load module is located in the SYSLIB DD statement. The new RUV rule set is placed in the CARDS DD statement.

Figure 5  Sample JCL for RUVZCJC

```plaintext
// . . . jobcard . . .
/*
//CONVJCT EXEC PGM=RUVZCJC
//STEPLIB DD DISP=SHR,DSN=HLQ.LOAD <= RECOVERY UTILITY loadlib
//SYSLIB DD DISP=SHR,DSN=HLQ.RPCV.load <= Loadlib with ISIJCT
//CARDS DD DSN=HLQ.output.ruleset.name.<== New VSAM Rule Set
//    SPACE=(TRK,(5,5),RLSE).
//    DISP=(NEW,CATLG),
//    UNIT=SYSDA
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(5,5))
//SYSUT2 DD UNIT=SYSDA,SPACE=(CYL,(5,5))
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(5,5))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(5,5))
//SYSUT5 DD UNIT=SYSDA,SPACE=(CYL,(5,5))
//SYSPRINT DD SYSOUT=*        /*
//SYSSIN DD *
ISIJCT
/*

Executing RUVZCJC

To execute RUVZCJC, perform the following steps:

1 Review the JCL to verify that the UNIT parameters match your system requirements.

2 Modify the JOB statement to meet your system requirements.

3 Point the STEPLIB statement to the load library where you installed RUV.

4 Point the SYSLIB statement to the load library where your current ISIJCT load module is located.

5 Modify the PARM statement to include PARM=’!LMOD=ISIJCT’, or specify the SYSIN statement as follows:

```plaintext
//SYSIN DD *
ISIJCT
/*
```

6 Point the CARDS statement to a data set that is defined with RECFM=FB and LRECL=80.
The data set is a PDS member or a sequential data set:

```
//CARDS DD DISP=SHR,DSN=HLQ(RUVCARDS)
```

```
//CARDS DD DISP=SHR,DSN=HLQ.sequential.dataset
```

7 Submit the job to convert the existing CICS journal definitions to the new RUV rule set format.

8 Review the CARDS output, and modify the rule set to meet your site standards.

9 Finish building the entire rule set.

10 Activate the rule set as described in the *Recovery Utility for VSAM User Guide*.

You can use the journals that are created by RPVC in RUV if you use the ARCHIVE RPCV_IN command to convert them to a format that RUV can use. For more information, see the *Recovery Utility for VSAM User Guide*.

**RUVZCDA utility**

RUVZCDA converts RPCV data set name, group definitions, and user variables to produce RUV control statements. A set of system variables is also included. All unique RPCV data set names that are contained in the ISICDS are converted to RUV ADD VSAM_FILE statements. All RPCV groups are converted to RUV ADD VSAM_GROUP_DEFINITION statements. All RPCV user variables are converted to RUV ADD USER_VARIABLES statements.

The following parameters that are associated with ADD VSAM_FILE are *not* migrated from RPCV:

- COMMENT
- SELECTION_EXIT
- DAYS

The following VSAM_GROUP_DEFINITION parameters *are* migrated from RPCV:

- VSAM_FILE
- COMMENT

---

**NOTE**

COMMENT is included only when it exists in RPCV.
User variables are given a name indicating the application ID, the file ID, and/or the group that they are associated with.

---

**NOTE**

Because of fundamental differences between RPCV and RUV, RUVZCDA cannot provide 100 percent of the information that RUV requires from the information in RPCV alone. When RUVZCDA is executed, the resulting RUV statements will require modification for accurate results.

---

**Creating data set, group definition, and user variable statements**

Figure 6 shows sample JCL for executing RUVZCDA. The source for this sample is located in `HLQ.CNTL(CONVDSN1)`. The utility writes the new RUV data set, group definition, and user variable statements to the SYSPUNCH data set.

---

**Figure 6  Sample JCL for RUVZCDA**

```plaintext
//   . . . jobcard . . .
/*
 ** NOTE: See RUV User Guide Appendix B for instructions.
 ** */
/**
** CONVDSNS EXEC PGM=RUVZCDA
** STEPLIB DD DISP=SHR,DSN=HLQ.LOAD <= RUV Loadlib
** DD DISP=SHR,DSN=HLQ.LOAD <= RPCV Loadlib
** ISICDS DD DISP=SHR,DSN=HLQ.RPCV.ISICDS <= RPCV CDS pair
** ISICDS1 DD DISP=SHR,DSN=HLQ.RPCV.ISICDS1 <= RPCV CDS pair
** SYSPUNCH DD DSN=HLQ.output.dsns.groups.name, <= RUV DSNs/GRPs stmts
**   SPACE=(TRK,(5,5),RLSE).
**   DCB=(RECFM=FB,LRECL=80,BLKSIZ=0).
**   DISP=(NEW,CATLG),UNIT=SYSDA
** SYSPRINT DD SYSOUT=* <= Report of DSNs and Groups in ISICDS
** CRDPRINT DD SYSOUT=* <= Listing of SYSPUNCH stmts
*/
```

You can specify the ISICDS in either of the following ways:

- You can code the ISICDS DD statement and, if required, the ISICDS1 DD statement in the utility JCL. (Older versions of RPCV use one ISICDS, and ISICDS1 is not required.)
- You can allow the utility to use the ISICDSs that are coded in the ISICOPT by omitting ISICDS and ISICDS1 DD statements from the utility JCL. If you use this method, ISICOPT must be present in the STEPLIB.
Executing RUVZCDA

To execute RUVZCDA, perform the following steps:

1. Review the JCL to verify that UNIT parameters match your system requirements.

2. Modify the JOB statement to meet your system requirements.

3. Point the first DD in the STEPLIB concatenation to the load library where you installed RUV.

4. Point the second DD in the STEPLIB concatenation to your RPCV library that contains the RPCV load lib.

5. Point the ISICDS and ISICDS1 DD statements to your primary and alternate CDS libraries.

   **NOTE**

   You can omit ISICDS and ISICDS1 DD statements if they are coded in the ISICOPT.

6. Point the SYSPUNCH DD statement to a data set that is defined with RECFM=FB and LRECl=80.

   The data set is a PDS member or a sequential data set:

   //SYSPUNCH DD DISP=SHR,DSN='HLQ.PDS(RUVDSNG)'

   //SYSPUNCH DD DISP=SHR,DSN='HLQ.PDS.sequential.dataset'

7. Submit the job.

   The utility writes RUV data set name and group definition statements to SYSPUNCH.
Defining data sets, groups, and user variables to RUV

When you have executed RUVZCDA, you are ready to use the control statements that are produced by the utility to define the data sets and groups to RUV. Use the sample JCL that is shown in Figure 7 as a model. The source for this sample is located in HLQ.CNTL(CONVDSN2).

Figure 7 Sample JCL for the RUV data set name and group definition and report

```zobjc
// . . . jobcard . . .
//JOBLIB    DD DISP=SHR,DSN=HLQ.LOAD   <= RUV Loadlib
//*
//DEFINE   EXEC PGM=RUVZSMO,REGION=64M
//SYSPRINT DD SYSOUT=*                <= From CONVDSN1
//SYIN     DD DSN=HLQ.output.dsns.groups.name, DISP=SHR
//*
//REPORT   EXEC PGM=RUVZSMO,REGION=64M
//SYSPRINT DD SYSOUT=*                <= From CONVDSN1
//SYIN     DD *
SET SUBSYSTEM(ssid);
SET REPORT(FULL);
REPORT VSAM_FILE(*);
REPORT VSAM_GROUP_DEFINITION(*);
```

To define data sets and groups to RUV, perform the following steps:

1. Review the SYSPRINT report and the SYSPUNCH output from RUVZCDA.
2. Modify the statements to meet your site requirements.
   Use the sample JCL in Figure 7 as a model for the job to generate the RUV data set name and group definitions.
3. Modify the JOBLIB to meet your system requirements.
4. Point the first SYSIN in the job to the data set that contains your modified RUV statements.
5. Substitute your site RUV subsystem ID for ssid in Figure 7, line SET SUBSYSTEM(ssid).
6. Submit the job.
7. Review the SYSPRINT of each step, and modify the RUV statements as required. Repeat this step until all definitions have been generated correctly.
Migration considerations

This section describes RPCV-to-RUV migration considerations.

Supported file structure requirements

RUV supports and recovers the following file types:

- KSDS
- ESDS
- RRDS
- VRRDS

Table 9 lists the requirements for supported file structures.

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>none</td>
</tr>
<tr>
<td>changed</td>
<td>none</td>
</tr>
<tr>
<td>obsolete</td>
<td>The BDAM file type is no longer supported.</td>
</tr>
</tbody>
</table>

Repository information for VSAM file recovery

The repository records information on data sets, associated journal records, and backups. The information is used by the RUV automatic recovery facility. Table 10 lists repository information for VSAM file recovery.

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>A new repository (and a REGISETt2 repository) are now required. The following repository administrative commands are available:</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>BACKUP REPOSITORY</td>
</tr>
<tr>
<td></td>
<td>RESTORE REPOSITORY</td>
</tr>
<tr>
<td></td>
<td>PURGE REPOSITORY</td>
</tr>
<tr>
<td>changed</td>
<td>none</td>
</tr>
<tr>
<td>obsolete</td>
<td>Migration program RUVZCDA converts data set names and group definitions from the CDS to the new RUV repository.</td>
</tr>
</tbody>
</table>
**RUV subsystem requirements**

RUV requires proper memory and storage for subsystems. Table 11 lists subsystem requirements.

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>The RUV subsystem requires a minimum common storage area of 568 KB for product modules.</td>
</tr>
<tr>
<td>changed</td>
<td>The RUV ISPF interface requires ISPF version 4.2 or later.</td>
</tr>
<tr>
<td>obsolete</td>
<td>none</td>
</tr>
</tbody>
</table>

**Rule set requirements**

RUV uses rule sets to determine which VSAM files should be protected. Table 12 lists rule set requirements.

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>The ADD USER_VARIABLES command creates user variables.</td>
</tr>
<tr>
<td></td>
<td>The ADD JOURNAL_MODEL command defines log and journal models that are named in the job set rule.</td>
</tr>
<tr>
<td></td>
<td>The ADD JOB_SET command identifies user jobs that will trigger batch journaling activity.</td>
</tr>
<tr>
<td></td>
<td>The ADD RULE_SET command identifies the set of rules that combines all options which a user has defined to control the batch journaling environment.</td>
</tr>
<tr>
<td></td>
<td>The ADD VSAM_SET command identifies recoverable VSAM files by DSN and the type of recovery to support at the DSN level.</td>
</tr>
<tr>
<td></td>
<td>The ADD JOB_JCL command creates a recovery JCL record to be used when a user abend occurs.</td>
</tr>
<tr>
<td></td>
<td>The ADD INTERNAL_READER command creates a rule set element that controls the processing of recovery JCL which is generated when a job abends.</td>
</tr>
<tr>
<td></td>
<td>The ISPF online interface and the ACTIVATE RULE_SET command activate a rule set.</td>
</tr>
<tr>
<td>changed</td>
<td>none</td>
</tr>
<tr>
<td>obsolete</td>
<td>none</td>
</tr>
</tbody>
</table>
Security requirements

RUV uses APF interfaces to allow flexibility for you to implement security. Table 13 lists security requirements.

Table 13  Security requirements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>RUV has specific security parameters (in addition to standard site security).</td>
</tr>
<tr>
<td>changed</td>
<td>The product load library must be APF authorized.</td>
</tr>
<tr>
<td>obsolete</td>
<td>none</td>
</tr>
</tbody>
</table>

ISPF interface requirements

With the ISPF interface you can perform a VSAM recovery, display backup and archive data sets, display a log of logs list, display subsystem and rule set information and create rule sets, and look up message information. Table 14 lists ISPF interface requirements.

Table 14  ISPF interface requirements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>The ISPF interface can be used as an automated recovery tool and provides online interaction for repository data.</td>
</tr>
<tr>
<td>changed</td>
<td>none</td>
</tr>
<tr>
<td>obsolete</td>
<td>none</td>
</tr>
</tbody>
</table>

BMC Software subsystem requirements

BMC Software subsystems activate or facilitate RUV and other BMC Software products. Table 15 lists requirements for using BMC Software subsystems.

Table 15  BMC Software subsystem requirements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>The BMC Software Primary Subsystem (BMCP) and the BMC Software Consolidated Subsystem (BCSS) are required to implement and use RUV and to access the repository.</td>
</tr>
<tr>
<td>changed</td>
<td>In RPCV, BMCP and BCSS subsystems were required only when the user implemented JOB STEP logging and backout features. Use the SET SUBSYSTEM command to connect a batch job to a specific subsystem.</td>
</tr>
<tr>
<td>obsolete</td>
<td>The RPCV subsystem (RPCS) and the Backup Subsystem are not valid for RUV.</td>
</tr>
</tbody>
</table>
Batch journaling requirements and job step logging requirements

Batch journaling and job step logging provide recovery data for batch processing. Table 16 lists requirements for batch journaling and job step logging.

Table 16  Batch journaling requirements and job step logging requirements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>Batch journaling requires that a rule set be defined in the RUV repository. The rule set contains all information that is required for indicating what VSAM files will be protected (journaled or logged). The default rule set is #DEFAULT. This member is located in the CNTL library. The BMCP and the BCSS are now required for this feature. Program RUVZCJC (see “RUVZCJC utility” on page 68) converts an ISIJCT rule set to an RUV rule set.</td>
</tr>
<tr>
<td>changed</td>
<td>One output journal or log per job step is created. Dynamic backout is provided if the proper recovery JCL is defined in the rule set and logging is requested for the step.</td>
</tr>
<tr>
<td>obsolete</td>
<td>ISIREP and ISI0192A are no longer required. ISIOPT, ISIJCT, and ISICOPT tables are no longer required. RPCV subsystem option tables (global option, write-ahead log, backout exclude data sets, and initialization list) are no longer required. An equivalent program to perform a backout to the beginning of a job step has been dropped.</td>
</tr>
</tbody>
</table>
Master data set backup requirements

The BACKUP command writes one or more copies of a VSAM base cluster to sequential data sets in RUV-prefixed format. Table 17 lists requirements for backing up master data sets.

Table 17 Master data set backup requirements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>The BACKUP command has a new syntax. Registration of a backup is performed automatically. The BMCP and the BCSS are now required for this feature and for the BWO feature. The CICS file definition for VSAM files that are to be processed through the BWO feature must include the update attribute.</td>
</tr>
<tr>
<td>changed</td>
<td>Program RUVZSM0 is used to process the BACKUP command. CICS RDO entries for PLTPI and PLTSD have changed. See members CNTL (RUVPLTPI and RUVPLTSD). If you plan on using BWO, RUVB8020 is required in the PLTPI and PLTSD. RUVB8020 is the BWO CICS Controller, which communicates with the RUV subsystem. The CICS transient data destination ID that is used for Subsystem Backup information messages has the default setting “CSMT.” The setting of the first three characters of RUV CICS transaction IDs has changed. The default transaction ID prefix is “RUV.” You must use the SET SUBSYSTEM command to connect a batch job to a specific subsystem when performing backup (or BWO) so that registration is performed in the correct repository.</td>
</tr>
<tr>
<td>obsolete</td>
<td>The CIMODE feature has been dropped.</td>
</tr>
</tbody>
</table>

Purge processing requirements

The PURGE command purges obsolete repository records. Table 18 lists requirements for purge processing.

Table 18 Purge processing requirements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>none</td>
</tr>
<tr>
<td>changed</td>
<td>The number of days or cycles per file can be set by using UPDATE VSAM_FILE and ADD VSAM_FILE commands or can be set globally by using the SET DEFAULT command.</td>
</tr>
<tr>
<td>obsolete</td>
<td>none</td>
</tr>
</tbody>
</table>
Archiving requirements

The ARCHIVE command registers recovery information in the repository. Table 19 lists requirements for archiving with RUV.

Table 19 Archiving requirements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>The ARCHIVE command causes a CICS journal or log stream to be read and to be reformatted into RUV archive records.</td>
</tr>
<tr>
<td>changed</td>
<td>Program RUVZSM0 is used to process the ARCHIVE command.</td>
</tr>
<tr>
<td>obsolete</td>
<td>The JOURNAL command is obsolete.</td>
</tr>
</tbody>
</table>

RUV recovery requirements

RUV recovery recovers VSAM files in parallel. Table 20 lists requirements for RUV recovery.

Table 20 RUV recovery requirements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>The RESTORE command has a new syntax.</td>
</tr>
<tr>
<td></td>
<td>The RECOVER command performs forward or backward recovery.</td>
</tr>
<tr>
<td></td>
<td>RUV recovery processing performs parallel restore, recovery, and BLDINDEX tasks. Use the Set READER_TASKS(nn) command to control the maximum number of parallel tasks per command.</td>
</tr>
<tr>
<td></td>
<td>RUV recovery processing restores the latest RUV or SMS backup master file.</td>
</tr>
<tr>
<td></td>
<td>The RESTORE VSAM_FILE command restores a VSAM master file from the latest RUV or SMS backup master file in preparation for a forward recovery.</td>
</tr>
<tr>
<td></td>
<td>The RECOVER FORWARD command performs a forward recovery.</td>
</tr>
<tr>
<td></td>
<td>The RECOVER BACKOUT command performs a backward recovery.</td>
</tr>
<tr>
<td>changed</td>
<td>none</td>
</tr>
<tr>
<td>obsolete</td>
<td>none</td>
</tr>
</tbody>
</table>
RUV report requirements

The REPORT command creates a report on data in the repository. Table 21 lists requirements for creating RUV reports.

Table 21  RUV report requirements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>RPCV-created journals and logs can be processed by the RUV utilities, but they must be registered by the ARCHIVE command or the REGISTER command. RUV reports have a new format. You can use the SET command with the REPORT option to set the reporting level (SUMMARY</td>
</tr>
<tr>
<td>changed</td>
<td>none</td>
</tr>
<tr>
<td>obsolete</td>
<td>none</td>
</tr>
</tbody>
</table>

Miscellaneous requirements

Table 22 lists miscellaneous requirements.

Table 22  Miscellaneous requirements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>Most commands allow wildcards.</td>
</tr>
<tr>
<td>changed</td>
<td>Group processing is supported. This option lets you process files within groups. Many features have been enhanced with more functions or other changes.</td>
</tr>
<tr>
<td>obsolete</td>
<td>Because of changes in the CICS transaction server, in-flight processing is not supported.</td>
</tr>
</tbody>
</table>

To use all functions of RUV, users with RPCV product data should convert their JCT/OPT tables to RUV rule sets. For more information, see “Rule set components” on page 82.
Using rule sets

You must use rule sets to customize your batch journaling environment. You tell RUV about the recovery requirements of your VSAM files, and you specify the jobs and programs to which those requirements apply. RUV requires more information, such as where to store recovery information, how to build backup JCL, and when to run recovery jobs. You use rule sets to provide this information.

RUV provides rule set samples during installation and provides forms to help you organize rules for VSAM sets, job sets, and journal models.

RUV supports the following levels of rule sets:

- subsystem level
- job level

RUV uses the subsystem level rule set to examine all jobs that are running in the z/OS system. A job level rule set applies to a single batch job only; it does not affect other jobs. RUV searches a job level rule set ahead of a subsystem level rule set. This convention makes testing of rule sets easier and simplifies customization of rule sets for specific tasks.

**TIP**

RPCV users can think of the rule set as a replacement for the JCT/OPT process that is used in RPCV.
Rule set components

RU V provides a TSO/ISPF interface and a batch interface to replace the RPCV OPT/JCT. A JCT-to-rule-set migration assistance utility is provided.

A rule set may consist of multiple job sets and multiple VSAM sets that are designated by the user. The product rule sets that you create are stored in the repository. The rule sets contain data, log/journal allocated templates, and JCL, and can include the following defined components:

- VSAM rules, determined by recoverability requirements
- job and program rules, which determine when and how logging and journaling occurs by job and by program
- user variables, which provide environmental information
- job JCL, which provides a JCL template for job step backout
- internal readers, which specify what is to be done with a recovery job
- journal models, which provide allocation information for logs and journals
- output models, which provide allocation information for backups
- system variables, which are available for use without any user definition

**NOTE**

The Step Name Selection feature has been dropped.
Component relationships

Figure 8 shows the relationships between rule set components.

Figure 8  Rule set component relationships

Appendix C, “Work sheets for rule sets,” provides information-gathering forms for RUV. These work sheets can help you avoid problems that might occur as you construct rule sets. You can maintain a written record of the VSAM_SET, JOB_SET, and journal model definition data that you have developed. For more information about rule sets, see the Recovery Utility for VSAM User Guide.
Implementing the RUV repository

RUV uses its own repository (the REGISET), which is created during installation. Records are created in the REGISET by using the RUV ISPF interface, by the RUVZSM0 utility, by backup, or by the Batch Journaling Facility. When the RUV repository is installed, it requires no further implementation. The online backup of the RUV repository is called REGISET2.

**TIP**
To better ensure data integrity, it is recommended that you run RUV with two REGISETs.

For more information about the RUV repository, see the *Recovery Utility for VSAM User Guide*. 

Starting BMCP and BCSS subsystems at IPL

When you move RUV to a production environment, you may want to initialize subsystems to load and start at IPL. To have the subsystems load and initialize at IPL, ensure that the authorized RUV load library is in the system link list (SYS1.PARMLIB(LNKLSTxx)).

For more information about starting and stopping BMCP and BCSS subsystems, see “Starting BMCP and BCSS subsystems” on page 39.
Starting BMCP and BCSS subsystems at IPL
Installation instructions

This appendix describes the process that you use for a tape installation, an EPD installation, and installation of post-release fixes.

This appendix contains the following topics:

Tape installation .......................................................... 88
EPD installation .......................................................... 89
Post-release fixes for RUV 4.1.00 ................................... 95
Tape installation

NOTE

Refer to the RUV$README.txt file on the ftp.bmc.com/pub/RUV server for any updated procedures. Otherwise, follow the instructions listed below.

Use the JCL shown in Figure 9 to extract the files from the RUV product tape.

Figure 9  JCL to extract files from the RUV product tape

```plaintext
//jobcard
// * THIS IS THE JCL TO INSTALL THE RUV 4.1.00 TAPE    ----- 
//IEBCOPY EXEC PGM=IEBCOPY
//INDD1 DD DISP=(OLD,KEEP),UNIT=(3490),LABEL=(1,SL...,EXPDT=2008/365),
// VOL=(,RETAIN...,SER=RVU410),DSN=RVU.V4100.SOURCE
//INDD2 DD DISP=(OLD,KEEP),UNIT=AFF=INDD1,
// LABEL=(2,SL...,EXPDT=2008/365),
// VOL=(,RETAIN...,SER=RVU410),DSN=RVU.V4100.CNTL
//INDD3 DD DISP=(OLD,KEEP),UNIT=AFF=INDD1,
// LABEL=(3,SL...,EXPDT=2008/365),
// VOL=(,RETAIN...,SER=RVU410),DSN=RVU.V4100.LOAD
//INDD4 DD DISP=(OLD,KEEP),UNIT=AFF=INDD1,
// LABEL=(4,SL...,EXPDT=2008/365),
// VOL=(,RETAIN...,SER=RVU410),DSN=RVU.V4100.DLIB
//INDD5 DD DISP=(OLD,KEEP),UNIT=AFF=INDD1,
// LABEL=(5,SL...,EXPDT=2008/365),
// VOL=(,RETAIN...,SER=RVU410),DSN=BMC.BPS.LOAD
//INDD6 DD DISP=(OLD,KEEP),UNIT=AFF=INDD1,
// LABEL=(6,SL...,EXPDT=2008/365),
// VOL=(,RETAIN...,SER=RVU410),DSN=BMC.BCS.LOAD
//OUTDD1 DD DISP=(NEW,CATLG,DELETE),DSN=hlq.RUV4100.SOURCE,
// UNIT=SYSDA,SPACE=(CYL,(2,1,25))
//OUTDD2 DD DISP=(NEW,CATLG,DELETE),DSN=hlq.RUV4100.CNTL,
// UNIT=SYSDA,SPACE=(CYL,(6,3,125))
//OUTDD3 DD DISP=(NEW,CATLG,DELETE),DSN=hlq.RUV4100.LOAD,
// UNIT=SYSDA,SPACE=(CYL,(60,10,125))
//OUTDD4 DD DISP=(NEW,CATLG,DELETE),DSN=hlq.RUV4100.DLIB,
// UNIT=SYSDA,SPACE=(CYL,(60,10,125))
//OUTDD5 DD DISP=(NEW,CATLG,DELETE),DSN=hlq.RUV4100.BPSLOAD,
// UNIT=SYSDA,SPACE=(CYL,(20,10,25))
//OUTDD6 DD DISP=(NEW,CATLG,DELETE),DSN=hlq.RUV4100.BCSLOAD,
// UNIT=SYSDA,SPACE=(CYL,(20,10,25))
//SYSPRINT DD SYSOUT=* 
//SYSIN DD *
COPY INDD=INDD1,OUTDD=OUTDD1
COPY INDD=INDD2,OUTDD=OUTDD2
COPY INDD=INDD3,OUTDD=OUTDD3
COPY INDD=INDD4,OUTDD=OUTDD4
COPY INDD=INDD5,OUTDD=OUTDD5
COPY INDD=INDD6,OUTDD=OUTDD6
/*
```
Follow these steps to obtain the RUV files electronically:


2. Select the SUPPORT tab.


4. Select PRODUCT DOWNLOADS (EPD).

5. At this page, either enter your previously registered logon ID or select REGISTER NOW on the left side of the page.

   This step generates the ID that allows you to navigate to download the files.

6. Once you have registered and entered your ID and password into the LOGIN panel, you are taken to a page titled Export Validation & License Terms.

   A. Follow all of the instructions on this page.

   B. Be sure to read and select the two disclosures at the bottom of this page.

7. Export Validation & License Terms

   Complete the required fields and click Continue.
8 Select Product Page

My Entitled Products lists products that you have purchased or are authorized to trial.

A When coming to the site for the first time, or to see authorized trial products, click on the Manage Support IDs button.

B In the Manage My Support IDs screen, enter your support ID and password and click the Add button.

C Once your support ID appears in the list, click on the Back to My Entitled Products button to see your products.

D Click on the check box in front of the product that you want to download.

E Click Continue to proceed to the Select Product Version screen.

F Request Trial routes requests for new products trial requests to BMC representatives.

9 Select Product Version

A Scroll through the list or use search aids to find the product version.

B Click on the check box in front of the product version that you want to download.

C Click Continue to proceed to Download Files.

10 Download Files

Click on desired files to download.
Once you have downloaded the product files to your PC, transmit them to your appropriate mainframe system by using FTP.

Table 23 lists the file sizes that you need to preallocate before transmitting the files from your PC to the mainframe. Use the JCL shown in Figure 10 to preallocate these files.

<table>
<thead>
<tr>
<th>File name</th>
<th>Location</th>
<th>Size / Allocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCS.LOAD.TRS</td>
<td>EPD to PC</td>
<td>1429 KB</td>
</tr>
<tr>
<td>BCS.LOAD.TRS</td>
<td>PC to mainframe</td>
<td>2 cyls, lrecl=1024, blksze=6144, recfm=FB</td>
</tr>
<tr>
<td>BCS.LOAD (uncompressed)</td>
<td>mainframe</td>
<td>6 cyls, 10 dir blocks, 61 members*</td>
</tr>
<tr>
<td>BPS.LOAD.TRS</td>
<td>EPD to PC</td>
<td>95 KB</td>
</tr>
<tr>
<td>BPS.LOAD.TRS</td>
<td>PC to mainframe</td>
<td>1 cyls, lrecl=1024, blksze=6144, recfm=FB</td>
</tr>
<tr>
<td>BPS.LOAD (uncompressed)</td>
<td>mainframe</td>
<td>1 cyls, 5 dir blocks, 25 members*</td>
</tr>
<tr>
<td>RUV.CNTL.TRS</td>
<td>EPD to PC</td>
<td>88 KB</td>
</tr>
<tr>
<td>RUV.CNTL.TRS</td>
<td>PC to mainframe</td>
<td>2 cyls, lrecl=1024, blksze=6144, recfm=FB</td>
</tr>
<tr>
<td>RUV.CNTL (uncompressed)</td>
<td>mainframe</td>
<td>1 cyls, 11 dir blocks, 63 members*</td>
</tr>
<tr>
<td>RUV.DLIB.TRS</td>
<td>EPD to PC</td>
<td>217 KB</td>
</tr>
<tr>
<td>RUV.DLIB.TRS</td>
<td>PC to mainframe</td>
<td>2 cyls, lrecl=1024, blksze=6144, recfm=FB</td>
</tr>
<tr>
<td>RUV.DLIB (uncompressed)</td>
<td>mainframe</td>
<td>3 cyls, 38 dir blocks, 227 members*</td>
</tr>
<tr>
<td>RUV.LOAD.TRS</td>
<td>EPD to PC</td>
<td>24585 KB</td>
</tr>
<tr>
<td>RUV.LOAD.TRS</td>
<td>PC to mainframe</td>
<td>35 cyls, lrecl=1024, blksze=6144, recfm=FB</td>
</tr>
<tr>
<td>RUV.LOAD (uncompressed)</td>
<td>mainframe</td>
<td>72 cyls, 39 dir blocks, 230 members*</td>
</tr>
<tr>
<td>RUV.SOURCE.TRS</td>
<td>EPD to PC</td>
<td>191 KB</td>
</tr>
<tr>
<td>RUV.SOURCE.TRS</td>
<td>PC to mainframe</td>
<td>1 cyls, lrecl=1024, blksze=6144, recfm=FB</td>
</tr>
<tr>
<td>RUV.SOURCE (uncompressed)</td>
<td>mainframe</td>
<td>2 cyls, 3 dir blocks, 15 members*</td>
</tr>
</tbody>
</table>

* Note: Check with the RUV product support team to verify the current number of members in each library, as these numbers might change between releases.
Use the JCL shown in Figure 10 to preallocate the TRS files to help in the FTP transfer from your PC to the mainframe.

Figure 10  JCL to preallocate the TRS files

```jcl
//jobcard here.
//*-------------------------------------------------------------------+
//* Pre- allocate data sets for uploading data sets downloaded to your
//* PC from the EPD software download site.
//* Change hlq to your hilevel qualifier
//*-------------------------------------------------------------------+
//IEFBR14  EXEC PGM=IEFBR14
//* allocate BCS.TRS
//SYSUT1   DD DSN=hlq.BMC.BCS.TRS,
//*          DISP=(NEW,CATLG,DELETE),DCB=(BLKSIZE=6144,LRECL=1024,RECFM=FB),
//*          SPACE=(CYL,(10,10)),UNIT=SYSDA.VOL=SER=DEVXXX
//* allocate BPS.TRS
//SYSUT2   DD DSN=hlq.BMC.BPS.TRS,
//*          DISP=(NEW,CATLG,DELETE),DCB=(BLKSIZE=6144,LRECL=1024,RECFM=FB),
//*          SPACE=(CYL,(10,10)),UNIT=SYSDA.VOL=SER=DEVXXX
//* allocate RUV.CNTL.TRS
//SYSUT3   DD DSN=hlq.BMC.RUV.CNTL.TRS,
//*          DISP=(NEW,CATLG,DELETE),DCB=(BLKSIZE=6144,LRECL=1024,RECFM=FB),
//*          SPACE=(CYL,(10,10)),UNIT=SYSDA.VOL=SER=DEVXXX
//* allocate RUV.DLIB.TRS
//SYSUT4   DD DSN=hlq.BMC.RUV.DLIB.TRS,
//*          DISP=(NEW,CATLG,DELETE),DCB=(BLKSIZE=6144,LRECL=1024,RECFM=FB),
//*          SPACE=(CYL,(10,10)),UNIT=SYSDA.VOL=SER=DEVXXX
//* allocate RUV.LOAD.TRS
//SYSUT5   DD DSN=hlq.BMC.RUV.LOAD.TRS,
//*          DISP=(NEW,CATLG,DELETE),DCB=(BLKSIZE=6144,LRECL=1024,RECFM=FB),
//*          SPACE=(CYL,(50,10)),UNIT=SYSDA.VOL=SER=DEVXXX
//* allocate RUV.SOURCE.TRS
//SYSUT6   DD DSN=hlq.BMC.RUV.SOURCE.TRS,
//*          DISP=(NEW,CATLG,DELETE),DCB=(BLKSIZE=6144,LRECL=1024,RECFM=FB),
//*          SPACE=(CYL,(10,10)),UNIT=SYSDA.VOL=SER=DEVXXX
```
Once you have the product files on your mainframe system, use the JCL shown in Figure 11 to unterse them and copy them to your RUV hlq name.

**Figure 11** JCL to unterse files downloaded from the Electronic Product Distribution site

```plaintext
//STEP1 EXEC PGM=TRSMAIN,PARM=UNPACK,COND=(0,LT)  
//SYSPRINT DD SYSOUT=H  
//INFILE DD DISP=SHR,DSN=hlvl.BMC.RUVSOURCE.TRS  
//OUTFILE DD DISP=(NEW,CATLG),UNIT=SYSDA,  
// DSN=hlvl.RUV4100.SOURCE.UNP,  
// SPACE=(CYL,(120,50,50),RLSE)  
//STEP2 EXEC PGM=TRSMAIN,PARM=UNPACK,COND=(0,LT)  
//SYSPRINT DD SYSOUT=H  
//INFILE DD DISP=SHR,DSN=hlvl.BMC.RUVCNTL.TRS  
//OUTFILE DD DISP=(NEW,CATLG),UNIT=SYSDA,  
// DSN=hlvl.RUV4100.CNTL.UNP,  
// SPACE=(CYL,(120,50,150),RLSE)  
//STEP3 EXEC PGM=TRSMAIN,PARM=UNPACK,COND=(0,LT)  
//SYSPRINT DD SYSOUT=H  
//INFILE DD DISP=SHR,DSN=hlvl.BMC.RUVLOAD.TRS  
//OUTFILE DD DISP=(NEW,CATLG),UNIT=SYSDA,  
// DSN=hlvl.RUV4100.LOAD.UNP,  
// SPACE=(CYL,(120,50,150),RLSE)  
//STEP4 EXEC PGM=TRSMAIN,PARM=UNPACK,COND=(0,LT)  
//SYSPRINT DD SYSOUT=H  
//INFILE DD DISP=SHR,DSN=hlvl.BMC.RUVDLIB.TRS  
//OUTFILE DD DISP=(NEW,CATLG),UNIT=SYSDA,  
// DSN=hlvl.RUV4100.DLIB.UNP,  
// SPACE=(CYL,(100,20,50),RLSE)  
//STEP5 EXEC PGM=TRSMAIN,PARM=UNPACK,COND=(0,LT)  
//SYSPRINT DD SYSOUT=H  
//INFILE DD DISP=SHR,DSN=hlvl.BMC.BPSLOAD.TRS  
//OUTFILE DD DISP=(NEW,CATLG),UNIT=SYSDA,  
// DSN=hlvl.BMC.BPS.LOAD.UNP,  
// SPACE=(CYL,(120,50,150),RLSE)  
//STEP6 EXEC PGM=TRSMAIN,PARM=UNPACK,COND=(0,LT)  
//SYSPRINT DD SYSOUT=H  
//INFILE DD DISP=SHR,DSN=hlvl.BMC.BCLOAD.TRS  
//OUTFILE DD DISP=(NEW,CATLG),UNIT=SYSDA,  
// DSN=hlvl.BMC.BCS.LOAD.UNP,  
// SPACE=(CYL,(120,50,150),RLSE)
```
13 Use the JCL shown in Figure 12 to copy modules.

**Figure 12  JCL to copy modules**

```jcl
//STEP1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*  
//SYSUT3 DD UNIT=SYSDA,SPACE=(400,(25,10))
//SYSUT4 DD UNIT=SYSDA,SPACE=(400,(25,10))
//INDS1 DD DISP=SHR,DSN=hilvl.RUV4100.SOURCE.UNP
//OUTDS1 DD DISP=SHR,DSN=hilvl.RUV4100.SOURCE
//SYSIN DD *
COPY I=((INDS1,R)),O=OUTDS1
//STEP2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*  
//SYSUT3 DD UNIT=SYSDA,SPACE=(400,(25,10))
//SYSUT4 DD UNIT=SYSDA,SPACE=(400,(25,10))
//INDS1 DD DISP=SHR,DSN=hilvl.RUV4100.CNTL.UNP
//OUTDS1 DD DISP=SHR,DSN=hilvl.RUV4100.CNTL
//SYSIN DD *
COPY I=((INDS1,R)),O=OUTDS1
//STEP3 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*  
//SYSUT3 DD UNIT=SYSDA,SPACE=(400,(25,10))
//SYSUT4 DD UNIT=SYSDA,SPACE=(400,(25,10))
//INDS1 DD DISP=SHR,DSN=hilvl.RUV4100.LOAD.UNP
//OUTDS1 DD DISP=SHR,DSN=hilvl.RUV4100.LOAD
//SYSIN DD *
COPY I=((INDS1,R)),O=OUTDS1
//STEP4 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*  
//SYSUT3 DD UNIT=SYSDA,SPACE=(400,(25,10))
//SYSUT4 DD UNIT=SYSDA,SPACE=(400,(25,10))
//INDS1 DD DISP=SHR,DSN=hilvl.RUV4100.DLIB.UNP
//OUTDS1 DD DISP=SHR,DSN=hilvl.RUV4100.DLIB
//SYSIN DD *
COPY I=((INDS1,R)),O=OUTDS1
//STEP5 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*  
//SYSUT3 DD UNIT=SYSDA,SPACE=(400,(25,10))
//SYSUT4 DD UNIT=SYSDA,SPACE=(400,(25,10))
//INDS1 DD DISP=SHR,DSN=hilvl.BMC.BPS.LOAD.UNP
//OUTDS1 DD DISP=SHR,DSN=hilvl.BMC.BPS.LOAD.UNP
//SYSIN DD *
COPY I=((INDS1,R)),O=OUTDS1
//STEP6 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*  
//SYSUT3 DD UNIT=SYSDA,SPACE=(400,(25,10))
//SYSUT4 DD UNIT=SYSDA,SPACE=(400,(25,10))
//INDS1 DD DISP=SHR,DSN=hilvl.BMC.BCS.LOAD.UNP
//OUTDS1 DD DISP=SHR,DSN=hilvl.BMC.BCS.LOAD.UNP
//SYSIN DD *
COPY I=((INDS1,R)),O=OUTDS1
```
Post-release fixes for RUV 4.1.00

There might be fixes available to be pulled down and applied to the newly-installed RUV code.

1. Use the JCL shown in Figure 13 to download the RUV 4.1.00 FIX files.

--- WARNING ---

Contact Customer Support for the most recent RUV Service Pack file to download.

This file includes the most recent recommended fixes to apply to RUV.

Use the file name that is given to you by the Customer Support representative to fill in the D*** variable in the JCL shown in Figure 13.

--- Figure 13 ---

JCL to download the RUV 4.1.00 FIX files

```plaintext
//FTPIN EXEC PGM=FTP,REGION=3M
//OUTPUT DD SYSOUT=*  
//SYSOUT DD SYSOUT=*  
//SYSPRINT DD SYSOUT=*  
//INPUT DD *
ftp.bmc.com
anonymous
@bmc.com
cd /pub/RUV
locsite rec=fb lr=1024 blk=6144
locsite unit=storage cy pri=50 sec=100
binary
get RUV4100.SERVPACK.D***.TRS 'hlq.RUV4100.SERVPACK.D***.TRS'
quit
/*
//UNPCK1 EXEC PGM=TRSMAIN,PARM=UNPACK
//SYSPRINT DD SYSOUT=*  
//INFILE DD DISP=SHR,DSN=hlq.RUV4100.SERVPACK.D***.TRS
//OUTFILE DD DISP=(NEW,CATLG),UNIT=SYSALLDA,VOL=SER=xxxxxx,
//DSN=hlq.RUV4100.SERVPACK.D***.
//SPACE=(CYL,(50,10,1),RLSE)
```
2 Use the JCL shown in Figure 14 to copy the modules to your newly-installed RUV LOAD library.

Or, you can create a FIX library for the LOAD library and run it concatenated in front of your RUV LOAD library.

---

**WARNING**

Contact Customer Support for the most recent RUV Service Pack file to download.

This file includes the most recent recommended fixes to apply to RUV.

Use the file name that is given to you by the Customer Support representative to fill in the D*** variable in the JCL shown in Figure 14.

---

**NOTE**

If you bring in your FIX library as a concatenated library in front of your RUV LOAD library, you will need to copy password module RUVLINFO from your RUV LOAD library into your FIX library.

---

**Figure 14  JCL to copy modules**

```
//STEP1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*  
//SYSUT3 DD UNIT=SYSDA,SPACE=(400,(25,10))
//SYSUT4 DD UNIT=SYSDA,SPACE=(400,(25,10))
//INDS1 DD DISP=SHR,DSN=hlq.RUV4100.SERVPACK.D***
//OUTDS1 DD DISP=SHR,DSN=hlq.RUV.LOAD <==== customer's RUV LOAD library
//SYSPIN DD *
COPY I=((INDS1,R)),O=OUTDS1  
/*
```
BMC Software product authorization

This appendix describes the process that you use to apply passwords and to reconfigure your CPU for the RECOVERY UTILITY for VSAM (RUV) product. This appendix contains the following topics:

Overview ................................................................. 98
Product Authorization utility ............................................. 99
Product-authorization passwords ........................................ 101
  Permanent passwords .................................................. 101
  Temporary passwords ................................................. 102
  How to apply passwords ............................................. 102
Product licensing ......................................................... 103
  Product trials and permanent licenses .............................. 103
  CPU upgrades ......................................................... 103
  CPU failures .......................................................... 104
  Product maintenance or version upgrades .......................... 105
How to obtain passwords ................................................. 106
Online product authorization ............................................ 108
  Processing a permanent password ................................. 108
  Adding authorization for a processor .............................. 110
  Deleting authorization for a processor ............................ 112
  Replacing authorization for a processor ........................... 114
  Modifying authorization for an existing processor ............... 116
  Resetting authorization for all processors ....................... 118
  Processing a temporary password ................................. 119
  Displaying product authorization .................................. 120
  Displaying current processor information ....................... 122
Batch product authorization ........................................... 124
  Running batch product authorization .............................. 124
  Control statements and keywords ................................. 125
  Return codes ......................................................... 126
Overview

When BMC Software Contracts Administration processes a license agreement for a product, it issues CPU authorization passwords. These passwords authorize specific CPUs (processors) to run the licensed product. Because BMC Software licenses its products for use on individual CPUs, the passwords are product-specific and CPU-specific (one license per product per CPU). To delete or replace an authorized CPU, you must also have a password.

**NOTE**

You do not need to apply passwords or update CPU authorization when you install product maintenance or version upgrades.

Passwords can be processed in either of the following ways:

- as part of an online procedure
- in a batch interface that uses a job which is supplied on the product distribution tape
Product Authorization utility

You use the BMC Software Product Authorization utility to apply passwords and to change your CPU configuration. When you apply passwords, the BMC Software Product Authorization utility builds or updates product-authorization tables. Product-authorization tables are data files that are normally stored in the product LOADLIB. The Product Authorization utility uses passwords to create entries in the tables that define the authorization for the product and to validate software licenses. If you have authorization questions or concerns about the Product Authorization utility, contact your BMC Software sales representative.

Product-authorization tables comprise the following types:

- permanent product-authorization table

  The Product Authorization utility builds or updates a permanent product-authorization table when you install or apply a permanent password. The permanent table controls which CPUs are licensed to run the product, based on the serial number, the model number, and the submodel number of the unit.

- temporary product-authorization table

  The Product Authorization utility builds or updates a temporary product-authorization table when you apply a temporary password.

For more information about permanent and temporary passwords, see “Product-authorization passwords” on page 101.

Product-authorization tables are product specific and are identified by the three-character product code (VRU), as follows:

- VRUTBL3P (permanent)
- VRUTBL3T (temporary)

**NOTE**

Do not install product-authorization tables into partitioned data set extended (PDSE) load modules. PDSE files cannot contain load modules and data type objects together in the same file.

Installing product-authorization tables into load module PDSEs will cause a failure (as shown in the following detailed error message) when trying to apply the password to the product library:
This problem can be prevented by specifying a different library to contain the product-authorization tables. Allocate a separate PDS or PDSE for the authorization table, and specify this PDS or PDSE when installing the password.
Product-authorization passwords

BMC Software products require the following types of passwords:

- **Temporary passwords** are issued for product trials or in other special circumstances (for example, when a hardware failure prevents you from using the authorized CPU).

- **Permanent passwords** are issued when you convert to a permanent license, delete or replace a CPU, or modify the properties of a CPU or the product-authorization tables.

Valid passwords can include the following characters:

- alphanumeric character set, excluding the letters I and O to avoid confusion with the numbers one (1) and zero (0)

- equal sign (=), “at” sign (@), and plus sign (+)

  **TIP**
  If your keyboard does not have the “at” sign (@), you can use an asterisk (*) instead of @. You can use these two characters (@ and *) interchangeably when typing passwords.

Permanent passwords

Permanent passwords update a product’s permanent authorization table. Each permanent password has one of the functions that are listed in Table 24. When you apply a permanent password, the Product Authorization utility automatically recognizes the password function and prompts you accordingly.

**Table 24  Permanent password functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>authorizes one new CPU to run the product</td>
</tr>
<tr>
<td>Delete</td>
<td>removes one CPU from the table, preventing that CPU from running the product</td>
</tr>
<tr>
<td>Replace</td>
<td>replaces one CPU in the table with another CPU, allowing the new CPU to run the product in place of the old CPU</td>
</tr>
<tr>
<td>Modify</td>
<td>modifies one or more properties of one CPU that exists in the product-authorization table</td>
</tr>
<tr>
<td>Reset</td>
<td>modifies the global properties of the product-authorization tables</td>
</tr>
</tbody>
</table>
**Temporary passwords**

BMC Software issues temporary passwords to customers who are evaluating products on a trial basis or to customers who need to bypass product authorization to run a product temporarily on an unlicensed CPU. Temporary passwords have a specific expiration date, which is part of the password.

**How to apply passwords**

A password is an activation key for a software license. The password is not the software license. Because permanent passwords have a limited lifespan (typically 30 days), apply your new passwords as soon as possible after you receive them.

You can apply the new passwords before you completely install the product, if you have installed the Product Authorization utility and have downloaded the product load library.

You can apply the new passwords even if the product is not yet running on a specific CPU. For example, if your installation process requires that you install and run the product on a test system before migrating it to the production system, you can apply the password for the production system CPU, even though the product is not yet running there.
Product licensing

You must use the Product Authorization utility in the following situations:

- for product trials and permanent licensing
- when you are upgrading to a new CPU
- when an authorized CPU fails

**NOTE**

Although the Product Authorization utility is not required for product maintenance and version upgrades, you must consider certain issues. For more information, see “Product maintenance or version upgrades” on page 105.

Product trials and permanent licenses

During a trial period for a BMC Software product, you can install and use the product on any CPU by using a temporary password that you have obtained from your BMC Software sales representative. When you finish the trial and you want to obtain a product license, the following rules apply:

- You must purchase a product license for each CPU on which you will run the product.
- BMC Software Contracts Administration issues a permanent password for each combination of CPU and licensed product.
- To enable a product on a CPU, you must add the permanent password issued for that CPU. You do not need to reinstall and retest the product.
- You can install multiple passwords in the same product library. This capability lets you use the same load library to run a product on multiple CPUs or to install a product at a central site and run it at remote sites.

CPU upgrades

When you upgrade to a new CPU, you must obtain a new permanent password for each product that you want to use on that CPU. When you install the new password, the old entry in the authorization table for the product is replaced. The new table entry defines the authorization for the product.
CPU failures

If a hardware failure or a disaster-recovery situation prevents the use of a licensed CPU, BMC Software can provide a temporary license that lets the product run on a backup CPU for a limited time. Before the temporary license expires, you must obtain a permanent license for the new CPU or you must resume using the original CPU.

At the end of the grace period you will no longer be able to run the affected product on the temporary CPU. If this situation occurs, you must obtain a new password to reset the grace period.

Updating product-authorization tables

To trigger the grace period, the license validation process must update the authorization tables. If the product load library must be write-protected, problems could occur with updates. To avoid problems, you can place the authorization tables in another data set and concatenate that data set to the product load library.

The concatenated authorization-table library should have the same DCB attributes as the product’s load library. (The RECFM for the authorization-table library must be U.) If you have several BMC Software products, you can dedicate one library that includes all authorization tables for all products.

Before updating the library that contains the authorization tables, the license validation process determines whether the data set is in LNKLST. If the data set is in LNKLST, the license validation process will not attempt an update.

Running a product on an unlicensed processor

When you run a product on an unlicensed processor, a 15-calendar-day grace period can be triggered. When the grace period is triggered, the Product Authorization utility (online or in batch mode) and the affected product issue a message advising you of the expiration date. When this grace period expires, the product will not run or will run with diminished functions.

**NOTE**
The product will continue to function normally when it is run on a licensed CPU, even if the grace period has been triggered or has expired.

To avoid problems, obtain a RESET password from BMC Software Contracts Administration and apply the password before the grace period ends. The RESET password updates the product-authorization table and makes another 15-calendar-day grace period available.
Product maintenance or version upgrades

Installing a new maintenance level or upgrading the version or release level of a product has no effect on product authorization. No new passwords are required, but your authorization tables must reside in the new production libraries.

If you install products in a test environment before moving them to production, the product-authorization tables must also reside in the test libraries. If you try to run the product on a different CPU, that CPU must also be licensed. Copy the product-authorization tables from the “old” library to the “new” library that contains the product’s new maintenance or upgrade.

Although product-authorization tables typically reside in the product’s load library, these tables are not load modules. If you are running ISPF version 4.2 or later, you might not be able to copy these tables by using the ISPF Move/Copy utility (option 3.3). You might receive a STOW error or some of the following error messages:

<table>
<thead>
<tr>
<th>Error Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEW2515W</td>
<td>4731 DIRECTORY ENTRY FOR VRUTBL3n IDENTIFIED BY DDNAME ISPddname IS NOT MARKED AS LOAD MODULE.</td>
</tr>
<tr>
<td>IEW2522E</td>
<td>470E MEMBER VRUTBL3n IDENTIFIED BY DDNAME ISPddname... IS NOT A LOAD MODULE- (INVALID RECORD TYPE).</td>
</tr>
<tr>
<td>IEW2307S</td>
<td>1032 CURRENT INPUT MODULE NOT INCLUDED BECAUSE OF INVALID DATA.</td>
</tr>
<tr>
<td></td>
<td>COPY FAILED FOR MEMBER VRUTBL3n. FAILURE IN IEWBIND INCLUDE, RETURN CODE 8 REASON CODE 83000507</td>
</tr>
</tbody>
</table>

In these messages, the variable prd is the three-character product code and n is P (permanent) or T (temporary). For more information, see “Product Authorization utility” on page 99.

If you receive any of these messages, use the IEBCOPY utility to copy the tables. Do not use the IEBCOPY COPYMOD parameter when copying the tables.
How to obtain passwords

Table 25 describes situations in which you must obtain passwords. For each situation, the table indicates the type of password that you require (temporary or permanent), what the password does, and how to obtain it.

### Table 25 Password-required situations

<table>
<thead>
<tr>
<th>Situation</th>
<th>Password type</th>
<th>Password function</th>
<th>Contact</th>
</tr>
</thead>
</table>
| You want to begin a free trial period.                                    | temporary     | ■ temporarily bypasses authorization checking  
■ allows you to run the product on any CPU for a limited time                  | BMC Software sales representative                                         |
| You purchase a license for a new product.                                 | permanent     | adds a designated CPU to the list of CPUs that are authorized to run a licensed product | BMC Software Contracts Administration (which issues a password after processing the license agreement) |
| You stop using an authorized CPU.                                        | permanent     | removes a designated CPU from the list of CPUs that are authorized to run a licensed product | ■ BMC Software sales representative  
■ BMC Software Contracts Administration                                    |
| You upgrade to a new CPU.                                                | permanent     | authorizes the transfer of a license from one CPU to another                      | ■ BMC Software sales representative  
■ BMC Software Contracts Administration                                     |
| You want to run the product on an additional CPU.                         | permanent     | adds a designated CPU to the list of CPUs that are authorized to run a licensed product | BMC Software Contracts Administration (which issues a password after processing the license agreement) |
| The authorized CPU is not available because of an emergency (such as hardware failure). | temporary     | ■ temporarily bypasses authorization checking  
■ allows you to run the product on any CPU for a limited time                  | ■ BMC Software sales representative  
■ BMC Software Contracts Administration  
■ BMC Software Customer Support                                               |
| The grace period has been triggered and must be reset.                    | permanent     | resets the grace period                                                           | ■ BMC Software sales representative  
■ BMC Software Contracts Administration                                     |
When you request a permanent product license from BMC Software, you must furnish information about the affected CPUs. For each product that you want to license, use the work sheet in Table 26 to record the CPU information and the passwords that you receive from BMC Software. CPU information is not required for temporary passwords.

Table 26  Product-authorization work sheet

<table>
<thead>
<tr>
<th>CPU serial</th>
<th>CPU type</th>
<th>Version code</th>
<th>CPU model</th>
<th>Number of CPUs</th>
<th>Permanent password</th>
</tr>
</thead>
<tbody>
<tr>
<td>__________</td>
<td>________</td>
<td>_______</td>
<td>_______</td>
<td>__________</td>
<td>_______</td>
</tr>
<tr>
<td>__________</td>
<td>________</td>
<td>_______</td>
<td>_______</td>
<td>__________</td>
<td>_______</td>
</tr>
<tr>
<td>__________</td>
<td>________</td>
<td>_______</td>
<td>_______</td>
<td>__________</td>
<td>_______</td>
</tr>
<tr>
<td>__________</td>
<td>________</td>
<td>_______</td>
<td>_______</td>
<td>__________</td>
<td>_______</td>
</tr>
<tr>
<td>__________</td>
<td>________</td>
<td>_______</td>
<td>_______</td>
<td>__________</td>
<td>_______</td>
</tr>
<tr>
<td>__________</td>
<td>________</td>
<td>_______</td>
<td>_______</td>
<td>__________</td>
<td>_______</td>
</tr>
<tr>
<td>__________</td>
<td>________</td>
<td>_______</td>
<td>_______</td>
<td>__________</td>
<td>_______</td>
</tr>
</tbody>
</table>

For example, you would list a 9X2 with three processors and a CPU ID of 10309-9021-DA as follows:

<table>
<thead>
<tr>
<th>CPU Serial</th>
<th>CPU Type</th>
<th>Version Code</th>
<th>CPU Model</th>
<th>Number of CPUs</th>
<th>Permanent Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>10309</td>
<td>9021</td>
<td>DA</td>
<td>9X2</td>
<td>3</td>
<td>123,456,789,ABC</td>
</tr>
</tbody>
</table>

For information about determining your CPU ID, see “Displaying current processor information” on page 122 or use the LIST option of Batch Product Authorization (see Table 30 on page 126).
Online product authorization

This section describes the online interface that is used for product authorization. To apply passwords by using the batch interface, see “Batch product authorization” on page 124.

**NOTE**

You use the BMC Software Product Authorization utility to apply passwords and to change your CPU configuration. To launch this utility through the online interface, execute `HLQ.INSTALL(BMCINSTL)`. For more information, see the OS/390 and z/OS Installer Guide.

Processing a permanent password

Use the Product Authorization Primary Menu (Figure 15) to process passwords. You can also obtain pertinent information about the current processor and the authorization for that processor.

**Figure 15  Product Authorization Primary Menu (SECEPPRI)**

```
SECEPPRI <product-name> Product Authorization Primary Menu
COMMAND ===> ________________________________________________________________

Select an option. Type additional information if applicable. Then press Enter.

Options

_  1. Process password (Requires product load library and password)
_  2. Display product authorization (Requires product load library only)
_  3. Display current processor information
_  4. Help about...
_  5. Exit

Additional information

Product load library . . . ‘BMC.DIS.INSTALL’

Authorization password . . ___ ___ ___ ___

F1=Help  F2=Split  F3=Exit  F7=Bkwd  F8=Fwd  F9=Swap  F12=Cancel
```
Table 27 lists the options on the primary menu. Other sections of this appendix provide instructions for completing specific tasks, such as adding authorization to run the product on a new CPU.

Table 27  Product Authorization Primary Menu options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1      | processes a password that is provided by BMC Software Contracts Administration  
Use this option to add, delete, replace, modify, or reset authorization for a password. The Product Authorization utility automatically identifies the type of password and displays the appropriate panel. |
| 2      | displays a list of processors that are authorized to use the product  
The list also identifies when the authorization was last modified (and by whom) and the trial or temporary expiration date. |
| 3      | displays information about the current processor, including serial number, model number, version code (submodel), and number of available processors |
| 4      | displays version, copyright, and licensing information about the Product Authorization utility |
| 5      | exits the Product Authorization utility, and returns to the previous menu or panel |

**NOTE**

If you select option 1, you must also type the name of the product’s load library and the authorization password. The utility saves the library name in your ISPF profile and uses that name as the default library. The data set name must be fully qualified.
Adding authorization for a processor

Use the ADD Authorization for a Processor panel (Figure 16 is an example) to authorize a new CPU to run the product:

1. Access the ADD Authorization for a Processor panel:
   
   A. On the Product Authorization Primary Menu (Figure 15 on page 108), select option 1.
   
   B. At Product load library, type a fully qualified data set name.
   
   C. Press Tab.
   
   D. At Authorization password, type your permanent password.
   
   E. Press Enter.

   The ADD Authorization for a Processor panel is displayed.

   2. At New serial number, type the serial number of the processor for which you are adding authorization.

---

Figure 16   ADD Authorization for a Processor panel (SECEPADD)

SECEPADD  ADD Authorization for a Processor

Command ===> _________________________________________________________________

Supply information for all input fields. Then press Enter.

Authorization password . . : X04  UH9  KNG  JKE

New serial number . . . 10293
New model number . . . 9672   (for example, 9021, 9121, 3090)
3 At New model number, type the model number of the processor for which you are adding authorization.

4 Press Enter.

A pop-up message (Figure 17 is an example) explains that the product-authorization table was modified successfully.

Figure 17 Product authorization ADD message

5 To exit the Product Authorization utility and return to the previous menu or panel, press F3.
Deleting authorization for a processor

Use the DELETE Authorization for a Processor panel (Figure 18 is an example) to remove a CPU from the product-authorization table:

1 Access the DELETE Authorization for a Processor panel:

   A On the Product Authorization Primary Menu (Figure 15 on page 108), select option 1.

   B At Product load library, type a fully qualified data set name.

   C Press Tab.

   D At Authorization password, type your permanent password.

   E Press Enter.

The DELETE Authorization for a Processor panel is displayed.

Figure 18  DELETE Authorization for a Processor panel (SECEPDEL)

2 At Old serial number, type the serial number of the processor for which you are deleting authorization.
3 At **Old model number**, type the model number of the processor for which you are deleting authorization.

4 Press **Enter**.

   A pop-up message (**Figure 19** is an example) explains that the product-authorization table was updated successfully.

**Figure 19  Product authorization DELETE message**

5 To exit the Product Authorization utility and return to the previous menu or panel, press **F3**.
Replacing authorization for a processor

Use the REPLACE Authorization for a Processor panel (Figure 20 is an example) to replace one CPU in the product-authorization table with another CPU. This process allows the “new” CPU to run the product in place of the “old” CPU.

1 Access the REPLACE Authorization for a Processor panel:

   A On the Product Authorization Primary Menu (Figure 15 on page 108), select option 1.
   B At Product load library, type a fully qualified data set name.
   C Press Tab.
   D At Authorization password, type your permanent password.
   E Press Enter.

   The REPLACE Authorization for a Processor panel is displayed.

   Figure 20   REPLACE Authorization for a Processor panel (SECEPREP)

   Supply information for all input fields. Then press Enter.

   Authorization password . . : 4XY YAL AMB 48S

   Old serial number . . . 10293
   Old model number . . . 9672   (for example, 9021, 9121, 3090)

   New serial number . . . 10293
   New model number . . . 9652   (for example, 9021, 9121, 3090)

   F1=Help   F2=Split   F3=Exit   F7=Bkwd   F8=Fwd   F9=Swap   F12=Cancel

2 At Old serial number, type the serial number of the processor to be replaced.

3 At Old model number, type the model number of the processor to be replaced.
4 At New serial number, type the serial number of the processor that will replace the old processor.

5 At New model number, type the model number of the processor that will replace the old processor.

6 Press Enter.

A pop-up message (Figure 21 is an example) explains that the product-authorization table was updated successfully, replacing the old processor with the new processor.

Figure 21 Product authorization REPLACE message

7 To exit the Product Authorization utility and return to the previous menu or panel, press F3.
Modifying authorization for an existing processor

Use the MODIFY Authorization for an Existing Processor panel (Figure 22 is an example) to change one or more properties of an existing CPU in the product-authorization table. These properties include the version code, the number of significant digits for the serial number, the tier, the maximum number of processors, and the expiration date for the product license.

1 Access the MODIFY Authorization for an Existing Processor panel:

   A On the Product Authorization Primary Menu (Figure 15 on page 108), select option 1.

   B At Product load library, type a fully qualified data set name.

   C Press Tab.

   D At Authorization password, type your permanent password.

   E Press Enter.

   The MODIFY Authorization for an Existing Processor panel is displayed.

Figure 22    MODIFY Authorization for an Existing Processor panel (SECEPUPD)
2 At **Serial number**, type the serial number of the processor for which you want to modify the authorization.

3 At **Model number**, type the model number of the processor for which you want to modify the authorization and press **Enter**.

The properties are modified automatically. A pop-up message (is an example) explains that the product-authorization table was updated successfully.

**Figure 23**  **Product authorization MODIFY message**

<table>
<thead>
<tr>
<th>SECEPPRI &lt;product-name&gt; Product Authorization Primary Menu</th>
</tr>
</thead>
<tbody>
<tr>
<td>SELECT AN OPTION. TYPE ADDITIONAL INFORMATION IF APPLICABLE. THEN PRESS ENTER.</td>
</tr>
<tr>
<td>OPTIONS</td>
</tr>
<tr>
<td>1. Process password (Requires product load library and password)</td>
</tr>
<tr>
<td>2. Display product authorization (Requires product load library only)</td>
</tr>
<tr>
<td>3. Display current processor information</td>
</tr>
<tr>
<td>4. Help about...</td>
</tr>
<tr>
<td>5. Exit</td>
</tr>
<tr>
<td>ADDITIONAL INFORMATION</td>
</tr>
<tr>
<td>Product load library . . . 'BMC.DIS.LOAD'</td>
</tr>
<tr>
<td>Authorization password . . .</td>
</tr>
<tr>
<td>F1=Help F2=Split F3=Exit F7=Bkwd F8=Fwd F9=Swap F12=Cancel</td>
</tr>
</tbody>
</table>

4 To exit the Product Authorization utility and return to the previous menu or panel, press **F3**.
Resetting authorization for all processors

Use the Product Authorization Primary Menu (Figure 15 on page 108) to reset a global property (applying to all CPU IDs) of the authorization table:

1. On the Product Authorization Primary Menu, select option 1.
2. At Product load library, type a fully qualified data set name.
3. Press Tab.
4. At Authorization password, type your permanent password.
5. Press Enter.

A pop-up message (Figure 24 is an example) explains that the product-authorization table was updated successfully.

Figure 24  Product authorization RESET message

6. To exit the Product Authorization utility and return to the previous menu or panel, press F3.
Processing a temporary password

Use the Product Authorization Primary Menu (Figure 15 on page 108) to process a temporary password:

1. At Product load library, type a fully qualified data set name.
2. Press Tab.
3. At Authorization password, type your permanent password.
4. Press Enter.

A pop-up message (Figure 25 is an example) explains that the product-authorization table was built or updated successfully.

Figure 25  Product authorization temporary password message

5. To exit the Product Authorization utility and return to the previous menu or panel, press F3.
Displaying product authorization

Use the Product Authorization Display panel (Figure 26 is an example) to display the current authorization for a product:

1. On the Product Authorization Primary Menu (Figure 15 on page 108), select option 2.

2. At Product load library, type a fully qualified data set name.

3. Press Enter.

The Product Authorization Display panel is displayed.

Figure 26  Product Authorization Display panel (SECEPTBL)

4. To exit the Product Authorization utility and return to the previous menu or panel, press F3.

Table 28 lists the fields on the Product Authorization Display panel.

Table 28  Product Authorization Display panel fields  (Part 1 of 2)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load library</td>
<td>name of the product’s load library</td>
</tr>
<tr>
<td>Product code</td>
<td>code that BMC Software assigns to the product</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Last changed             | date and time that the product-authorization tables were last modified  

The variables are as follows:  
- \( mm \) represents the month (in the range 01 through 12)  
- \( dd \) represents the day (in the range 01 through 31)  
- \( yy \) represents the year (in the range 00 through 99)  
- \( hh:mm \) represents the hour (in the range 00 through 23)  
- \( mm \) represents minutes (in the range 00 through 59)  

| Last changed by          | user ID or job that requested the modification  

| Grace period ends        | date when the grace period (if triggered) will end  

The variables are as follows:  
- \( mm \) represents the month (in the range 01 through 12)  
- \( dd \) represents the day (in the range 01 through 31)  
- \( yyyy \) represents the year (in the range 0001 through 9999)  

**Note:** This line is displayed only if the failure mode is phased and the grace period has been triggered. The grace period can be triggered when you run a permanently licensed product on an unlicensed processor. To reset the grace period, apply a RESET password. Contact your BMC Software sales representative for assistance.  

| Temporary expiration     | date on which you will no longer be allowed to bypass the CPU ID check or trial the product  

The variables are as follows:  
- \( mm \) represents the month (in the range 01 through 12)  
- \( dd \) represents the day (in the range 01 through 31)  
- \( yyyy \) represents the year (in the range 0001 through 9999)  

**Note:** If this expiration date has not been reached, you can run this product on any processor. On the date shown, your trial period will end or (if you have licensed the product) you will be able to run the product only on authorized processors.  

| Licensed Processors      | list of properties for each licensed CPU  

The Version Code column reflects the hardware representation of the submodel. The Significant Digits column refers to the number of significant digits for the serial number. The Expiration Date column indicates the month and year through which you are licensed for the specific processor. In most cases, this value is NONE.  

**Note:** Most of the remaining processor information is provided for reference if you must contact BMC Software Customer Support.
Displaying current processor information

Use the Current Processor Information panel (Figure 27 is an example) to display information about the processor that you are using:

1 On the Product Authorization Primary Menu (Figure 15 on page 108), select option 3.

2 Press Enter.

The Current Processor Information panel is displayed.

Figure 27  Current Processor Information panel (SECEPCPU)

3 To exit the Product Authorization utility and return to the previous menu or panel, press F3.

The Current Processor Information panel displays the CPU serial and model numbers for the processor on which TSO is running. The panel also displays the version code of the processor.

The version code is the hardware representation of the submodel (for example, 942 in ES/9000-942 or 600 for a 3090-600 processor). If you experience problems, BMC Software Customer Support might require this information.
NOTE
Version code X'FF' indicates that MVS is running as a VM guest. X'FF' is not the real processor version code. To determine the real version code, run the LIST option of the Batch Product Authorization utility from an APF-authorized library. For more information, see Table 30 on page 126.

The Current Processor Information panel also displays the number of processors that are online to the current operating system. This information may apply to your BMC Software License Agreement.

NOTE
Information on the Current Processor Information panel might not refer to a computer on which you are licensed to run a BMC Software product. For example, if you log on to TSO on SYSA but run your BMC Software product on SYSB, your product-authorization entries might refer to SYSB.
Batch product authorization

This section describes the batch interface that is used for product authorization. To apply passwords by using the online interface, see “Online product authorization” on page 108.

Using the batch interface, you can perform the following tasks:

- process a password
- obtain current product authorization and processor information

Running batch product authorization

Figure 28 is a sample JCL script for running batch product authorization.

**Figure 28 Sample JCL for running batch product authorization**

```
//JJJJJJJJ JOB ....................
//*
//BLFSEC3B EXEC PGM=BLFSEC3B,PARM=VRU <<== PRODUCT CODE
//STEPLIB DD DSN=BMC.PRODUCT.LOAD,DISP=SHR
//*
//SYSLIB DD DSN=BMC.PRODUCT.LOAD,DISP=SHR <<== REQUIRED
//SYSPRINT DD SYSOUT=*                     <<== REQUIRED
//*
//SYSIN DD *
**** PROCESS AN ADD PASSWORD AND LIST RESULTS ****
PSWD=AE@,82G,91#,C7$  NEWCPUID=11111-9021
**** PROCESS A DELETE PASSWORD AND LIST RESULTS ****
PSWD=BE@,AD0,32$,7C#  OLDCPUID=31091-9121
**** PROCESS A REPLACE PASSWORD AND LIST RESULTS ****
PSWD=ARF,56C,##1,C7$   OLDCPUID=31001-3390  NEWCPUID=31091-3381
**** PROCESS A RESET PASSWORD
PSWD=123,456,789,ABC
**** PROCESS A TEMPORARY PASSWORD AND LIST RESULTS ****
PSWD=AE@.B32,1C,07#
**** REPORT THE PROCESSOR INFORMATION AND AUTHORIZATION ****
LIST
```
Some tasks require different input parameters, depending on the type of password that you are installing. The sample JCL in Figure 28 on page 124 shows tasks that you can perform by using the batch version of product authorization. To include only the tasks that you want to perform, you must modify the JCL.

The following syntax rules apply to control statements:

- Control statements can begin in any column.

- Uppercase letters are required.

- You must insert at least one blank space between individual keywords and data fields. Multiple blanks are acceptable.

- To insert comments, type an asterisk (*) in column 1 of each line that contains the comment. Comments following keywords are not allowed.

- You cannot specify the LIST keyword on the same line as PSWD, NEWCPUID, and OLDCPUID.
**Return codes**

Table 30 lists control statement keywords.

**Table 30  Control statement keywords**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Data</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSWD</td>
<td>12-character password formatted as four fields of three characters each, separated by a comma or a blank (see sample JCL) Twelve continuous characters are also acceptable.</td>
<td>Valid characters are alphanumeric (excluding letters I and O. Valid special characters are =, +, and @. You can substitute the asterisk (*) for the “at” sign (@) when @ is not available on the keyboard.</td>
</tr>
<tr>
<td>NEWCPUID</td>
<td>five-digit serial number, followed by a hyphen and a four-digit model number</td>
<td>The serial number and the model number must be hexadecimal characters and must be separated by a single hyphen.</td>
</tr>
<tr>
<td>OLDCPUID</td>
<td>five-digit serial number, followed by a hyphen and a four-digit model number</td>
<td>The serial number and the model number must be hexadecimal characters and must be separated by a single hyphen.</td>
</tr>
<tr>
<td>LIST</td>
<td>not applicable</td>
<td>A printed report lists the contents of the product-authorization tables and information about the processor on which the job ran.</td>
</tr>
</tbody>
</table>

**Return codes**

Table 31 lists the return codes that you can receive.

**Table 31  Return codes**

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>All requests completed successfully. For messages about each operation, see the SYSPRINT output.</td>
</tr>
<tr>
<td>4</td>
<td>A LIST was requested, but no tables were residing in the load library.</td>
</tr>
<tr>
<td>8</td>
<td>An error prevented completion of all of your requests. For messages about the error and any completed operations, see the SYSPRINT output.</td>
</tr>
</tbody>
</table>
Work sheets for rule sets

This appendix provides work sheets that you can use to gather information for configuring the Recovery Utility for VSAM (RUV) product.

This appendix contains the following topics:

Overview ................................................................. 128
Work sheets ............................................................ 129
Overview

The following work sheets can help you gather information for RUV and avoid problems that might occur as you construct rule sets:

- Use the VSAM_SET work sheet (page 130) to identify VSAM DSNs and related information. To make VSAM files recoverable, you use VSAM rules. VSAM rules identify recoverable VSAM data sets and specify the type of recovery that can be performed against those data sets. You generally group VSAM rules together under one or more VSAM sets.

- Use the JOB_SET work sheet (page 131) to record each job name and related information. To identify the jobs and programs that use recoverable VSAM files and to identify the recovery actions that you want RUV to take if an abend occurs, you use job rules. Job rules control when RUV logging and journaling occur (by job and by program), and where RUV writes log and journal information. You generally group job rules together under one or more job sets.

  **NOTE**

  For detailed information about rule sets, see the *Recovery Utility for VSAM User Guide*.

- Use the Journal Model Definition work sheet (page 132) to record each journal model name and related information. To create templates that will construct a DD allocation statement for QSAM logs and journals, you use journal models. RUV uses logs to record information that is required for step backout, and RUV uses journals for forward recovery (long-term retention of data).

If you use these work sheets, you can maintain a written record of the VSAM_SET, JOB_SET, and journal model definition data that you have developed.
Work sheets

Make copies of the VSAM_SET work sheet (page 130), the JOB_SET work sheet (page 131), and the Journal Model Definition work sheet (page 132). Date each work sheet, and fill in the specific names of the VSAM_SET, the JOB_SET, and the journal model definition. Fill in the appropriate blocks.
<table>
<thead>
<tr>
<th>VSAM DSN</th>
<th>Protect/Exclude</th>
<th>Backout Recovery</th>
<th>Forward Recovery</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Appendix C Work sheets for rule sets

### JOB_SET ___________________________ Date: __________

<table>
<thead>
<tr>
<th>Job Name</th>
<th>Program Name</th>
<th>Forward Recovery?</th>
<th>Backout Recovery?</th>
<th>Journal Abend</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>No/Yes</td>
<td>No/Yes</td>
<td>Abend Continue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model:</td>
<td>Model:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL:</td>
<td>JES/PDS:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No/Yes</td>
<td>No/Yes</td>
<td>Abend Continue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model:</td>
<td>Model:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL:</td>
<td>JES/PDS:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No/Yes</td>
<td>No/Yes</td>
<td>Abend Continue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model:</td>
<td>Model:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL:</td>
<td>JES/PDS:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No/Yes</td>
<td>No/Yes</td>
<td>Abend Continue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model:</td>
<td>Model:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL:</td>
<td>JES/PDS:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No/Yes</td>
<td>No/Yes</td>
<td>Abend Continue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model:</td>
<td>Model:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL:</td>
<td>JES/PDS:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No/Yes</td>
<td>No/Yes</td>
<td>Abend Continue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model:</td>
<td>Model:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL:</td>
<td>JES/PDS:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No/Yes</td>
<td>No/Yes</td>
<td>Abend Continue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model:</td>
<td>Model:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL:</td>
<td>JES/PDS:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No/Yes</td>
<td>No/Yes</td>
<td>Abend Continue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model:</td>
<td>Model:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL:</td>
<td>JES/PDS:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No/Yes</td>
<td>No/Yes</td>
<td>Abend Continue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model:</td>
<td>Model:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL:</td>
<td>JES/PDS:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No/Yes</td>
<td>No/Yes</td>
<td>Abend Continue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model:</td>
<td>Model:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL:</td>
<td>JES/PDS:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No/Yes</td>
<td>No/Yes</td>
<td>Abend Continue</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Model:</td>
<td>Model:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>JCL:</td>
<td>JES/PDS:</td>
<td></td>
</tr>
</tbody>
</table>
### Journal Model Definition

<table>
<thead>
<tr>
<th>Journal Model Name</th>
<th>Activity Estimate</th>
<th>DSN</th>
<th>DASD Space/Tape</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Resolutions for common installation and configuration issues

This appendix provides resolutions for common issues that may raise concerns when you are installing and configuring the Recovery Utility for VSAM (RUV) product. This appendix contains the following topics:

Overview ................................................................. 134
Resolutions ............................................................... 135
Overview

Occasionally during installation and configuration, you may be concerned about the following common issues:

- rule set errors
- product security
- subsystem availability
- subsystem restart

For suggested resolutions to these and other common issues, see “Resolutions” on page 135.
# Resolutions

Table 32 lists resolutions for common installation and configuration issues.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>When you are trying to restart a subsystem that came down since the z/OS system performed an IPL, you receive a JCL error with code IEF196I.</td>
<td>Restart the subsystem by entering the following command:</td>
</tr>
<tr>
<td></td>
<td>/S ssid, SUB=JES2</td>
</tr>
<tr>
<td>When you are trying to start the TSO interface by executing TSO RUVISPF, you receive error message BMC202329E RUV subsystem &lt;ssid&gt; not available; RS (00000002).</td>
<td>Identify the condition of the subsystem, and perform the appropriate action:</td>
</tr>
<tr>
<td></td>
<td>- The subsystem is not active and must be started.</td>
</tr>
<tr>
<td></td>
<td>- The subsystem is active, but you may have insufficient RACF, ACF2, or TOPSECRET access.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> For security implementation instructions, see HLQ.CNTL, members ACF2INFO, RACFINFO, or TSSINFO.</td>
</tr>
<tr>
<td>During RUV batch journaling or logging, IVP job IVP$JB10 does not create job IVP$JB11 (automatic backout).</td>
<td>Check jobs IVP$JB07, IVP$JB09, and IVP$JB10 for RUV journal message BMC202426I RUV &lt;ssid&gt; Allocated &lt;text&gt; DDName: &lt;ddname&gt; DSName: &lt;dsname&gt;, and perform the appropriate action:</td>
</tr>
<tr>
<td></td>
<td>- If no journal messages are present in the jobs, check the ACF2 rules for the correct access to the ssid from the IVP batch job name and account code.</td>
</tr>
<tr>
<td></td>
<td>- If no journal messages are present in the jobs, determine that the IVP batch job is running on the same CPU as the ssid subsystem.</td>
</tr>
<tr>
<td></td>
<td>- If no journal messages are present in the jobs, determine that the RUV JOB_RULE in the VSAM_SET is the correct rule for the VSAM name which is defined in IVP$JB05. This VSAM name should match the KSDSIN in IVP$JB10.</td>
</tr>
<tr>
<td>When you are performing the installation process, the created job ($B05UNLD) may cause an S806-04 abend.</td>
<td>Edit $B05UNLD for HLQ.PLIB to specify RECFM=FB.</td>
</tr>
</tbody>
</table>
Glossary

This glossary contains terms relating to technology in this book.

A

ACB
See access method control block.

ACF2
Access Control Facility.

access method control block
ACB. A control block that links an application program to VSAM or ACF/VTAM.

action
One of the defined tasks that an application performs.

action bar
The area at the top of the primary window containing keywords that provide access to actions which are available in that window.

accumulation
A set of journal records in file ID, key, and date-time sequence. Also referred to as *ACCUM processing*.

AIX
See alternate index.

after-image
The journal record of a file transaction (add, delete, or update) after the transaction occurs. *Contrast with* before-image.

alternate index
A collection of index entries that are related to a base cluster and are organized by an alternate key (a key other than the primary key of the associated base cluster data records). The alternate index provides an alternate directory for finding records in the data component of a base cluster. See also path.
APPLICATION RESTART CONTROL for VSAM
A BMC Software product that provides checkpoint/restart support and repositioning of VSAM data sets at application restart. When APPLICATION RESTART CONTROL for VSAM is used with RUV, APPLICATION RESTART CONTROL for VSAM provides automatic logging and backout of changes to VSAM data sets.

Applid
An application ID that is defined to CICS for communication purposes.

archiving
The storage of backup files and any associated journal records, usually for a given time.

AR/CTL
The acronym for APPLICATION RESTART CONTROL.

ASID
A unique system-assigned identifier for an address space.

backout
The process of restoring a previous state of all or part of a system. All updates against protected resources are removed. Backout occurs during a restart in a CICS or z/OS image failure or a batch job step failure.

back up
To copy information, usually to a tape device, for safekeeping.

backup
A copy of information or data that is kept in case the original is changed or destroyed.

Backup-While-Open
A Recovery Utility for VSAM feature that backs up online CICS data sets while leaving the online data available (or open) to users.

backward file recovery
The reconstruction of an earlier version of a file by using a newer version and data that has been recorded in a journal (before-image journal records). Contrast with forward file recovery.

backward recovery
See backward file recovery.

base cluster
In systems with VSAM, a key-sequenced or entry-sequenced file over which one or more alternate indexes are built. See also cluster.
batch
A group of data processing jobs or records that are stacked together for processing in a non-interactive environment.

BCSS
BMC Consolidated Subsystem. The BCSS manages I/O to the registration data sets (collectively called the REGISET), manages APF-authorized functions, and performs processing for intercepted open, close, attach, and link requests.

before-image
A journal record of a file transaction (add, delete, or update) before the transaction occurs. Contrast with after-image.

BJF
Batch Journaling Facility.

BMCP
BMC Primary Subsystem. The BMCP establishes supervisory services for the BCSS and many BMC Software products. The BMCP allows interception of open, close, attach, and link requests in the system.

C
CA
Control area. The unit of space allocation for a VSAM data set.

CI
Control interval. A unit of data that is transferred between auxiliary storage and virtual storage, independent of physical record size.

CDS

CICS
Customer Information Control System.

command area
The area on the portion of the window which is composed of two elements: a command field prompt and a command entry field.

command entry field
The entry field on the portion of the window where you type commands. The entry field is preceded by a command field prompt.
Common User Access
A set of basic online interface principles, techniques, and components for software applications that have been developed to run on nonprogrammable terminals in IBM SAA operating environments.

cluster
In systems with VSAM, a named structure consisting of a group of related components. For example, a data component with its index component. See also alternate index, base cluster.

CSA
Common service area. The area that contains data which is addressable by all address spaces.

CUA
See Common User Access.

D
data integrity
Safety of data from unintentional damage or an out-of-synchronization condition.

DBMS
Database management system.

DTB
See dynamic transaction backout.

DUPLEX
The Recovery Utility for VSAM duplicate repository.

duplicate key
The presence of the same value in a key field or a composite key field in more than one record in a data set.

dynamic transaction backout
The cancellation of uncommitted changes that are made to a protected resource when a task abends.

E
ECSA
Extended common service area.

ESDS
Entry-sequenced data set. In systems with VSAM, a data set whose records are loaded without respect to their contents, and whose relative byte addresses cannot change. Records are retrieved and stored by addressed access, and new records are added at the end of the data set.
forward file recovery
The reconstruction of a file by updating an earlier version with data that has been recorded in a journal (after-image journal records). *Contrast with* backward file recovery.

forward recovery
*See* forward file recovery.

GDG
Generation data group. A collection of data sets with the same base name, such as PAYROLL, that are kept in chronological order. Each data set is called a *generation data set*.

generation data set
One generation of a generation data group.

ICF
Integrated catalog facility.

IDCAMS
Integrated Catalog Access Method Services facility.

in-flight
Refers to a task that was interrupted before it was completed when a CICS failure or an immediate shutdown occurred. An in-flight task causes records to be written to the system log, but no sync point records are written for the current logical unit of work.

ISPF
Interactive System Productivity Facility.

IVP
Installation verification procedure.

JES
Job entry subsystem. A *z/OS* subsystem that receives jobs into the system, converts them to internal format, selects them for execution, processes their output, and purges them from the system.

JCL
Job Control Language.
job name
The name that is assigned to a JOB statement. The job name identifies a job to the system.

journal
A chronological record of changes that have been made in a data set; the record can be used to reconstruct a previous version of the data set. In CICS, a set of one or more data sets to which records are written sequentially during a CICS run.

journal work data set
A data set that is created by the Recovery utility for the storage of journal records which are sorted by file ID, key, and date-time sequence.

journal record
An individual record of transaction changes to a file. Journal records can be used for forward or backward file recovery.

journaling
The process of recording information sequentially in a database.

K
key sequence
In VSAM, the collating sequence of data records as determined by the value of the key field in each data record. The key entry may be the same as, or different from, the entry sequence of the records.

key field
In VSAM, a field in each record of a data set that is used as the key for the record. The key identifies the record and establishes the order for the record in respect to other records.

KSDS
Key-sequenced data set. A VSAM file or data set whose records are loaded in key sequence and are controlled by an index.

L
logging
The replacement function for batch journaling.

LPA
Link pack area. An area of main storage containing re-enterable routines from system libraries.

LUW
Logical unit of work. The processing that a program performs between synchronization points. See also sync point.
M
master data set
A data set that is used as an authority in a given job and that is relatively permanent, even though its contents may change. For example, a data set, such as PAYROLL, would be the target of a Recovery utility Backup function.

P
path
In VSAM, a named, logical entity providing access to the records of a base cluster directly or through an alternate index.

PDSE
Partitioned data set extended. A data set organization storing many different but related members.

PLT
Program list table.

pop-up
A bordered area of the screen that supplements the dialog which is occurring in the primary window.

program name
The name of the program that is executing, as identified to the system. The program name is specified in the execute command that is used to call the program.

pull-down
An extension of the action bar that displays a list of available choices for a selected choice in the action bar.

Q
quiesce
To end a process by allowing operations to complete in a normal way.

R
RACF
Resource Access Control Facility.

RBA
Relative byte address.

RCM
BMC Software product code for RECOVERY PLUS for CICS/VSAM.
RDO
   Resource Definition Online.

read integrity
   Data access which ensures that the most current copy of the record is being read.

recovery
   The reconstruction of a database; for example by means of backup files and after-images.

Recovery Utility for VSAM
   The replacement product for the VSAM Recovery Services product.

REGISET
   The Recovery Utility for VSAM repository.

registration
   The recording of information regarding data sets, associated journal records, and backup and recovery information in the Recovery Automation feature’s control data set (CDS). The Recovery Automation feature uses the data to coordinate recoveries.

repository
   The vehicle for collecting primary registration data sets.

rollback
   A programmed return to a prior checkpoint. A rollback cancels the changes to all recoverable resources during the current logical unit of work.

RPCV
   The common acronym for RECOVERY PLUS for CICS/VSAM.

RUV
   The common acronym for Recovery Utility for VSAM.

S

SAA
   See Systems Application Architecture.

sequential data set
   A data set that has records organized by successive physical positions. It is typically used for sequential input or output, such as for journal, ledger, and reporting functions.

SHAREOPTION
   The attribute of a VSAM data set which controls concurrent sharing of its data by different programs running in the same CPU or different CPUs. The types of sharing can be at the read and/or update levels.
SIT
System Initialization Table.

SMS
Storage Management Subsystem. A component of z/OS that automates and centralizes the management of storage.

Systems Application Architecture
Systems Application Architecture (SAA) is a collection of selected software interfaces, conventions, and protocols that are published by IBM. It is IBM’s design for creating applications that work in multiple environments. There are four elements to this architecture: Common User Access, Common Programming Interface, Common Communication Support, and Common Applications.

sync point
An intermediate or end point during processing of a transaction at which an update or modification to one or more of the transaction’s protected resources is logically complete and error free. Synonym for synchronization point.

synchronization point
See sync point.

SYSLOG
System log. A data set or file for job-related information, operational data, descriptions of unusual occurrences, commands, and messages to or from the operator.

T
task
A basic unit of work. See also logical unit of work.

TSO

U
UACC

V
VIO
Virtual I/O.

VSAM
Virtual Storage Access Method. An access method for direct or sequential processing of fixed-length and variable-length records on direct access devices.
X

XBM
   EXTENDED BUFFER MANAGER.
Index

A
ACF2/RACF requirements 34
ACTIVATE RULE_SET command 75
ADD Authorization
  for a Processor panel 110
ADD Authorization for a Processor panel 110
ADD authorization message 111
ADD INTERNAL_READER command 75
ADD JOB_JCL command 75
ADD JOB_SET command 75
ADD JOURNAL_MODEL command 75
ADD RULE_SET command 75
ADD USER_VARIABLES command 75
ADD VSAM_FILE command 78
ADD VSAM_SET command 75
allocated data sets 66
ALLOW_ALL command 35
ARCHIVE command 79, 80
archiving requirements 79
authorized load library in DFHRPL 44
CICS (continued)
  shutdown 42
  startup 42, 53
  tables
    file definitions 45
    transient data destination ID 78
CICS file definition parameters
  FILE(fileid) 50
  FWDRECOVLOG(NO | nn) 50
  JNLADD(ALL) 51
  JNLREAD(UPDATEONLY) 51
  JNLUPDATE(YES) 51
  JOURNAL(nn) 51
  RECOVERY(ALL) 51
CICS file definitions, coding 45
CICS journal definitions, migration 68
CICS transaction backout 31
CICS transactions and programs, defining 41
CIMODE feature 78
coding CICS file definitions
  CICS TS environment 50
  non-RLS mode 45
  RLS mode 47
  Transaction Server environment 45
commands
  ACTIVATE RULE_SET 75
  ADD INTERNAL_READER 75
  ADD JOB_JCL 75
  ADD JOB_SET 75
  ADD JOURNAL_MODEL 75
  ADD RULE_SET 75
  ADD USER_VARIABLES 75
  ADD VSAM_FILE 78
  ADD VSAM_SET 75
  ALLOW_ALL 35
  ARCHIVE 79, 80
  BACKUP 78
  BACKUP REPOSITORY 74
  CEMT SHUTDOWN,I 42
  JOURNAL 79
  PURGE 78
  PURGE REPOSITORY 74
  RECOVER 79
  RECOVER BACKOUT 79
  RECOVER FORWARD 79
  REGISTER 80
B
BACKUP command 78
BACKUP REPOSITORY command 74
Backup-While-Open (BWO) feature 37
Batch Journaling Facility 81
Batch Journaling Facility (BJF) 37
batch journaling requirements 77
batch product authorization
  about 124
  JCL script 125
  return codes 126
  sample JCL 124
BCSS subsystem 39
BMC Software, contacting 2
BMCP subsystem 39
C
CEMT SHUTDOWN,I command 42
characters in passwords 101
checklist for installation verification procedure 60
CICS
  DFHRPL concatenation 44
  CICS (continued)
  shutdown 42
  startup 42, 53
  tables
    file definitions 45
    transient data destination ID 78
  CICS file definition parameters
    FILE(fileid) 50
    FWDRECOVLOG(NO | nn) 50
    JNLADD(ALL) 51
    JNLREAD(UPDATEONLY) 51
    JNLUPDATE(YES) 51
    JOURNAL(nn) 51
    RECOVERY(ALL) 51
  CICS file definitions, coding 45
  CICS journal definitions, migration 68
  CICS transaction backout 31
  CICS transactions and programs, defining 41
  CIMODE feature 78
coding CICS file definitions
  CICS TS environment 50
  non-RLS mode 45
  RLS mode 47
  Transaction Server environment 45
  commands
    ACTIVATE RULE_SET 75
    ADD INTERNAL_READER 75
    ADD JOB_JCL 75
    ADD JOB_SET 75
    ADD JOURNAL_MODEL 75
    ADD RULE_SET 75
    ADD USER_VARIABLES 75
    ADD VSAM_FILE 78
    ADD VSAM_SET 75
    ALLOW_ALL 35
    ARCHIVE 79, 80
    BACKUP 78
    BACKUP REPOSITORY 74
    CEMT SHUTDOWN,I 42
    JOURNAL 79
    PURGE 78
    PURGE REPOSITORY 74
    RECOVER 79
    RECOVER BACKOUT 79
    RECOVER FORWARD 79
    REGISTER 80
commands (continued)

REINIT 61
REPORT 80
REPORT ARCHIVE_FILE 62
REPORT BACKUP_FILE 62
REPORT RULE_SET 62
REPORT VSAM_FILE 62
RESTORE 79
RESTORE REPOSITORY 74
RESTORE VSAM_FILE 79
SET 80
SET DEFAULT 78
SET SUBSYSTEM 76, 78
SHUT IMMEDIATE 42
UPDATE VSAM_FILE 78

controlling Backup-While-Open 53
conventions
   general 16
   syntax statements 18

CPU
   failures (product license) 104
   properties (licensed processors) 121
   serial and model numbers 122
   upgrades 103
   version 122

Current Processor Information panel 122
customer support 3, 28
customizing CICS startup and shutdown 42

D

data set allocation 66
DCB attributes, authorization table library 104
defining transactions and programs to CICS 41
DELETE Authorization for a Processor panel 112
DELETE authorization, processor panel 112
determining installation paths 26
determining version, release, and maintenance levels 25
DFHPLT entries in PLT 43
DFHRPL DD statements 44
displaying Backup-While-Open status 53
displaying current processor 122
documentation, online Help 15

E

electronic documentation 15
enabling Logging 54
ESD installation 89
expiration dates, temporary 121
EXTENDED BUFFER MANAGER (XBM) 58

F

file definitions, CICS 45
file structure requirements 74

G

general conventions 16
grace period
   at temporary password expiration 104
   triggering 104

I

implementation of RUV repository 84
initial maintenance installation 27
initializing and enabling Backup-While-Open 51
initializing Logging 54
installation instructions 87
installation paths 26
installation process, understanding 24
installation tape 20
installation verification procedure 60
installation verification procedure checklist 60
installing maintenance (product authorization) 105
Instant Snapshot copies 58
interface requirements 76
international language support 25
IVP files used by VSAM 66
IVP$CHGS user variable changes 64

J

job step logging requirements 77
JOB_SET work sheet 128, 131
job-level rule sets 81
JOURNAL command 79
Journal Model Definition work sheet 128, 132

K

keywords (batch product authorization) 126

L

LIST keyword (batch authorization) 126
LNKLST library (product authorization) 104
LOAD library in CICS startup JCL 44
logging
   enabling 54
   initialization 54
M
maintenance
 initial installation 27
 subsequent installation 27
 version upgrades 105
master data set backup requirements 78
messages
 ADD authorization 111
 DELETE authorization 113
 errors copying authorization tables 105
 MODIFY 117
 MODIFY authorization 117
 online z/OS data set 14
 REPLACE 115
 RESET 118
 RESET authorization 118
 successful update, product-authorization table 119
 temporary password 119
migration
 CICS journal definitions 68
 from a test environment to a production environment 67
 OPT tables 68
 RPCV to RUV 68, 74
miscellaneous requirements 80
MODIFY authorization 116
MODIFY Authorization for an Existing Processor panel 116
multiple releases, rules for running 23

N
naming conventions, product authorization tables 99
NEWCPUID keyword (batch authorization) 126
non-RLS mode, coding CICS file definitions 45

O
OLDCPUID keyword (batch authorization) 126
online documentation 15
online Help 15
online product authorization 108
OPT table migration 68

P
panels
 ADD Authorization for a Processor 110
 Current Processor Information 122
 DELETE Authorization for a Processor 112
 MODIFY Authorization for an Existing Processor 116
 Product Authorization Display 120
parameters
 CICS file definitions 50
 FILE 50
 JNLADD(ALL) 51
 JNLREAD(UPDATEONLY) 51
 JNLUPDATE(YES) 51
 JOURNAL(nn) 51
 PLTPI 43
 PLTSD 43
 RECOVERY(ALL) 51
 verifying CICS initialization 43
password-required situations 106
passwords
 affected CPUs 107
 functions 101
 how to obtain 106
 permanent 101
 RESET 104
 temporary 102
 worksheet for obtaining 107
permanent passwords 101
PLTPI and PLTSD statement placement 42
PLTPI parameter 43
PLTSD and PLTPI statement placement 42
PLTSD parameter 43
post-release fixes 95
product authorization
 ADD Authorization for a Processor panel 110
 ADD message 111
 applying maintenance 105
 batch control statements 125
 batch method 124
 batch return codes 126
 CPU failures 104
 CPU information 107
 CPU properties 121
 CPU upgrades 103
 DELETE panel 112
 displaying current authorization 120
 displaying current processor 122
 LNKLST library 104
 MODIFY message 117
 MODIFY panel 116
 new CPUs 110
 online method 108
 options 109
 overview 98
 permanent licenses 103
 primary menu 108
 processing permanent passwords 108
 processing temporary passwords 119
 Product Authorization Display panel 120
 REPLACE panel 114
 RESET message 118
 RESET password 104
 tables, naming convention 99
 trials 103
 unlicensed CPU 102
product authorization (continued)
  updating authorization tables 104
utility
    batch authorization 124
    online authorization 108
Product Authorization Display panel 120
product authorization passwords
  permanent 101
  temporary 101
Product Authorization Primary Menu 108
Product Authorization utility
  overview 99
  product-authorization tables 99
product support 3
product-acceptance testing environment 67
product-authorization tables
  permanent 99
  successful update 119
  temporary 99
production environment 67
PSWD keyword (batch authorization) 126
PURGE command 78
purge processing requirements 78
PURGE REPOSITORY command 74

R
RACF/ACF2 requirements 34
reasons for VSAM data set recovery 31
RECOVER BACKOUT command 79
RECOVER command 79
RECOVER FORWARD command 79
recovery requirements 79
Recovery Utility for VSAM distribution tape 20
REGISET
  implementation 84
  information 74
REGISTER command 80
REINIT command 61
reinitializing Backup-While-Open 53
related documentation 15
removing RUV from execution 52
REPLACE
  Authorization for a Processor panel 114
  authorization, processor panel 114
REPORT ARCHIVE_FILE command 62
REPORT BACKUP_FILE command 62
REPORT command 80
REPORT option 80
report requirements 80
REPORT RULE_SET command 62
REPORT VSAM_FILE command 62
repository
  implementation 84
  information 74
requirements
  archiving 79
  batch journaling 77
  job step logging 77
  master data set backup 78
  miscellaneous 80
  purge processing 78
  recovery 79
  RUV reports 80
RESET
  authorization message 118
  product authorization 118
RESET password 104
resetting global properties 118
Resource Access Control Facility (RACF) 34
RESTORE command 79
RESTORE REPOSITORY command 74
RESTORE VSAM_FILE command 79
return codes (batch authorization) 126
RLS mode, coding CICS file definitions 47
RPCV migration to RUV 68, 74
rule sets
  component relationships 83
  components 82
  job level 81
  overview 81
  requirements 75
  Step Name Selection feature 82
  system level 81
  work sheets 83
rules for running multiple releases 23
RUV reports 80
RUV subsystem requirements 75
RUV to a production environment 67
RUVZCDA utility 70, 72
RUVZCJC utility 68, 69
RUVZFAC macro 55

S
SECEPADD 110
SECEPCPU 122
SECEPDEL 112
SECEPPRI 108
SECEPREP 114
SECEPTBL 120
SECEPUPD 116
security
  requirements 76
  setting up 34
SET command 80
SET DEFAULT command 78
SET SUBSYSTEM command 76, 78
setting up security 34
SHUT IMMEDIATE command 42
SNAPSHOT UPGRADE FEATURE (SUF) 58
starting BMCP and BCSS subsystems 39
starting CICS 53
starting subsystems at IPL 85
starting the BMCP subsystem 39
Step Name Selection feature 82
subsystems
  overview 39
  requirements 76
  RUV memory and storage requirements 75
  starting 39
SUF authorization 58
SUF components 58
SUF requirements 58
support 28
support, customer 3
symbols in passwords 101
syntax statement conventions 18
System Initialization Table (SIT) 43
system-level rule sets 81

T

tape contents 20
tape installation 88
technical support 3, 28
temporary expiration date 121
temporary passwords
  JCL for batch 124
  message 119
  processing online 119
testing environment 67
transient data destination 78

U

understanding the installation process 24
UPDATE VSAM_FILE command 78
updating CICS start-up JCL 44
updating product authorization tables 104
using the RUVZFAC macro 55
utilities
  RUVZCDA 70, 72
  RUVZCJC 68, 69

V

verifying CICS initialization parameters 43
version upgrades (product authorization) 105
version, release, and maintenance levels 25
VSAM data sets 54, 66
VSAM_SET work sheet 128, 130

W

work sheets
  JOB_SET 128
  Journal Model Definition 128
  product authorization 107
  VSAM_SET 128
worksheet for passwords 107