BMC Application Accelerator for IMS
User Guide

Supporting
Version 1.3 of BMC Application Accelerator for IMS

December 2015
Contacting BMC Software

Several methods are available for contacting BMC Software.

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

United States and Canada

<table>
<thead>
<tr>
<th>Address</th>
<th>BMC SOFTWARE INC</th>
<th>Telephone</th>
<th>Fax</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2103 CITYWEST BLVD</td>
<td>1 713 918 8800</td>
<td>1 713 918 8000</td>
</tr>
<tr>
<td></td>
<td>HOUSTON TX 77042-2827 USA</td>
<td>or</td>
<td>1 800 841 2031</td>
</tr>
</tbody>
</table>

Outside United States and Canada

<table>
<thead>
<tr>
<th>Telephone</th>
<th>+01 713 918 8800</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fax</td>
<td>+01 713 918 8000</td>
</tr>
</tbody>
</table>
Customer support

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

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

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

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

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

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

- Send an e-mail message to customer_support@bmc.com. (In the Subject line, enter SupID:yourSupportContractID, such as SupID:12345.)
- In the United States and Canada, call 1 800 537 1813. Outside the United States and Canada, contact your local support center for assistance.
## About this book
- Related publications ................................................................. 9
- Conventions .............................................................................. 10
- Syntax statements ....................................................................... 10

## Chapter 1  Overview of Application Accelerator
- What Application Accelerator does ............................................. 13
  - Optimization for IMS DLI and DBB job steps ......................... 13
  - Optimization for IMS BMP job steps .................................. 14
- Summary of Application Accelerator workflow ................................ 14
- Trial and production modes ..................................................... 15
  - Overview of trial mode ...................................................... 15
  - Overview of production mode ........................................... 17
- Installation and maintenance .................................................. 19
  - Installation System .......................................................... 19
  - Component maintenance ................................................... 19
- Requirements, considerations, and limitations .......................... 20
  - System and software requirements .................................... 20
  - Supported application and IMS elements ............................ 21
  - Memory requirements in the batch address space .................. 23
  - Additional considerations for product interactions ............... 23
- Information and support .......................................................... 24

## Chapter 2  Using Application Accelerator in trial mode
- Workflow summary for trial mode ............................................. 25
- Allocating and initializing the trial-mode repository .................... 27
- Modifying application JCL in trial mode .................................... 29
- APF authorization in trial mode ................................................ 32
- Troubleshooting problems in trial mode ................................... 33
  - Changing default values in trial mode ................................. 33
  - Reproducing the trial-mode repository for troubleshooting .... 34

## Chapter 3  Preparing to use production-mode components
- Transitioning from trial to production mode ............................. 35
- Overview of production-mode components ............................. 36
- Managing access authority for BMP jobs ................................ 38
Configuring and starting mainframe components ........................................................ 40
  Modifying startup procedures for existing subsystems and servers  .......... 40
  Enabling and disabling Application Accelerator in the CPC subsystem .... 41
  Starting subsystems and servers ................................................................. 42
Configuring the console ..................................................................................... 42
  Installing the console .................................................................................. 42
  Updating the console ................................................................................... 44
  Uninstalling the console ............................................................................ 44
Launching and exiting the console ..................................................................... 45
Managing BMC Database Management clients ................................................... 46
Overview of the console .................................................................................... 48
  Navigation window .................................................................................... 48
  Message windows ..................................................................................... 49
  Work area ............................................................................................... 50
  Selecting user options ............................................................................. 50
Setting up connections ..................................................................................... 51
  Overview of the Enterprise List and personal connections ....................... 51
  Managing the Enterprise List .................................................................. 51
  Using the Add Host Connection command ............................................... 52

Chapter 4    Using Application Accelerator in production mode  55
Workflow for production mode .......................................................................... 55
  Default and site values ........................................................................... 56
  Policies ................................................................................................. 57
Setting default and site values in the console .................................................. 58
  Setting default and site values in the Setup Wizard ................................. 58
  Setting default values in the Manage Defaults window ............................ 59
  Setting site values in the Maintain Parameters window ............................ 59
Setting default values with the IBOINIT utility .............................................. 60
Managing policies through the console ............................................................ 67
  Accessing the Manage Policy window ....................................................... 67
  Adding and editing policies .................................................................... 68
  Rearranging the Policies table ................................................................. 69
  Filtering displayed policies ..................................................................... 70
  Adding policies to the Exclude table ......................................................... 71
  Exporting policy information .................................................................. 71
Managing policies through the CPCBATCH utility ......................................... 72
  Adding or changing policies with the CPCBATCH utility ......................... 72
  CPCBATCH functions for the POLICIES command .................................. 74
  CPCBATCH JCL requirements .................................................................. 76
  CPCSYSIN control statements ................................................................. 77
About this book

This book contains detailed information about the BMC Application Accelerator for IMS product. This book is intended for:

- IMS system administrators
- IMS database administrators (DBAs)
- MS application developers
- IBM z/OS systems programmers
- z/OS operations teams

Like most BMC documentation, this book is available in printed and online formats. To request printed books or to view online books and notices (such as release notes and technical bulletins), see the support website at http://www.bmc.com/support.

Note

Online books are formatted as PDF or HTML files. To view, print, or copy PDF books, use the free Adobe Reader from Adobe Systems. If your product installation does not install the reader, you can obtain the reader at http://www.adobe.com.

The software also offers online Help. To access Help, press F1 within any product or click the Help button in graphical user interfaces (GUIs).

Related publications

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


- View Quick Course videos (short overviews of selected product concepts, tasks, or features), which are available from the following locations:
— Documentation Center (primary center and secured center)

— Support Central (at http://www.bmc.com/support/mainframe-demonstrations)

— BMC Mainframe YouTube channel (https://www.youtube.com/user/BMCSoftwareMainframe)


Products with online interfaces also offer online Help via the F1 key or, for graphical user interfaces (GUIs), via a Help button.

Tip
If you prefer hardcopy documentation, you can order it from your BMC sales representative or from Support Central. Also, from Support Central you can subscribe to receive proactive e-mail alerts when BMC issues notices.

Conventions

This document uses the following special conventions:

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

■ Variable text in path names, system messages, or syntax is displayed in italic text: testsys/instance/fileName

■ Menu sequences use a symbol to convey the sequence. For example, Actions => Create Test instructs you to choose the Create Test command from the Actions menu.

Syntax statements

This topic explains conventions for showing syntax statements.

A sample statement follows:

```command
COMMAND KEYWORD1 [KEYWORD2 | KEYWORD3] KEYWORD4={YES | NO} fileName...
```
The following table explains conventions for syntax statements and provides examples:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Example</th>
</tr>
</thead>
</table>
| Items in italic type represent variables that you must replace with a name or value. If a variable is represented by two or more words, initial capitals distinguish the second and subsequent words. | alias
databaseDirectory
serverHostName |
| Brackets indicate optional items. Do not type the brackets when you enter the option. A comma means that you can choose one or more of the listed options. You must use a comma to separate the options if you choose more than one option. | [tableName, columnName, field]
[-full, -incremental, -level] |
| Braces indicate that at least one of the enclosed items is required. Do not type the braces when you enter the item. | {DBDName | tableName}
UNLOAD device={disk | tape, fileName | deviceName}
{-a | -c} |
| A vertical bar means that you can choose only one of the listed items. In the example, you would choose either commit or cancel. | {commit | cancel} |
| An ellipsis indicates that you can repeat the previous item or items as many times as necessary. | columnName... |
Overview of Application Accelerator

This chapter introduces the BMC Application Accelerator for IMS product.

What Application Accelerator does

By improving the performance and efficiency of job steps in IMS batch applications, Application Accelerator can significantly reduce:

- CPU usage
- I/O activity
- Application processing time
- Time required for tuning job steps

Based on established policies or JCL specifications, Application Accelerator interacts with eligible job steps as follows:

- Monitors applications to analyze the characteristics and behavior of individual job steps for the number of executions that you designate

- Optimizes applications, based on observed past behavior

  After observing real-world characteristics of each application job step, Application Accelerator automatically chooses optimization techniques that are best suited for that step.

Optimization for IMS DLI and DBB job steps

For batch job steps that execute in IMS DLI or DBB regions, Application Accelerator can optimize performance by modifying runtime attributes as follows:

- Optimize DFSVSAMP buffer pools
- Enable OSAM buffer pools
Intercept DL/I calls and perform them by using a more efficient set of I/O routines

No JCL changes are required. For more information, see “Supported application types” on page 21.

Optimization for IMS BMP job steps

For IMS batch message processing (BMP) job steps, Application Accelerator can optimize performance by using a more efficient set of I/O routines to perform DL/I "get" calls under optimized buffering.

No JCL changes are required. For more information, see “Supported application types” on page 21.

Summary of Application Accelerator workflow

Application Accelerator operates while an application’s job steps are executing. For each eligible step, Application Accelerator completes the following basic workflow:

1. **Initialize operations in the job step:**
   a. Gain control.
   b. Determine the type of processing to perform.
   c. Set up and access required resources.

   **Note**
   You can specify that Application Accelerator ignore specific jobs and steps or exclude specific jobs.

2. **Monitor the job step for a designated (user-defined) number of executions.**

   Application Accelerator collects the following types of information about the job step and stores the data in a repository:
   
   - Types, patterns, and number of DL/I calls that the step issues
   - I/O statistics
Patterns of usage for various buffer subpools

3 **Optimize the monitored step.**

After the designated number of executions, Application Accelerator analyzes the repository data and modifies the runtime attributes for the job step as needed. For example, depending on the application type, the following modifications might be made:

- Intercept IMS get-type calls for BMP, DLI, and DBB applications, and perform these calls with a more efficient set of I/O routines.
- Optimize DFSVSAMP buffer pools for DLI and DBB applications.
- Enable OSAM sequential buffering for DLI and DBB applications that mainly access databases sequentially.

**Note**

While optimizing the job step, Application Accelerator also monitors the step to collect information that can help refine the attribute values for the next execution. In addition, Application Accelerator collects information about the effects of optimizing activities.

---

**Trial and production modes**

You can run Application Accelerator in **trial mode** or **production mode**:

- Trial mode is useful for initially evaluating Application Accelerator or quickly testing a new version of the product.
- Production mode is suitable for deploying Application Accelerator in your production environment, and for testing Application Accelerator against a large number of applications.

**Overview of trial mode**

Trial mode is a fast, simple way to implement Application Accelerator. Trial mode lets you start quickly, without requiring a system programmer to install and manage production-mode components.
**Trial-mode repository**

You must allocate a *trial-mode repository*. This VSAM data set contains product setup and control information and the data that the product collects during execution.

You specify setup and control information by using a batch utility to initialize or update option values in the repository. Options include, for example, the action that Application Accelerator should perform on behalf of the job step, and how to allocate temporary data sets for Application Accelerator.

Although the trial-mode repository is easy to allocate when evaluating and testing the product, the following disadvantages make the repository unsuitable for large-scale production use:

- You must identify the trial-mode repository in the JCL to execute a batch application job step.
- You must set up your own procedures for backing up the repository data.
- This repository supports simultaneous updates within a single z/OS image but does not support simultaneous updates across multiple z/OS systems.
- Repository data is limited to 4 gigabytes. Because Application Accelerator data requirements are limited, this concern is minor during testing but could be problematic during production use.

**Application JCL changes in trial mode**

In trial mode, you must modify the JCL for executing an IMS batch application job to identify the following items:

- Library that contains the executable load modules for the product
- Trial-mode repository, using the $IBO$VSM DD statement
- Other data sets as applicable

**APF authorization in trial mode**

In trial mode, you must provide a way for Application Accelerator to execute in an APF-authorized environment. You can use any valid authorization technique, including the following alternatives:

- You can authorize all data sets in the STEPLIB or JOBLIB concatenation.
- You can use a standard BMC started task, such as the Cross Product Connectivity (CPC) subsystem or the Database Utilities subsystem (DBUSS), to provide authorization.
Overview of production mode

Production mode is a robust way to implement Application Accelerator without making application JCL changes. This technique is suitable to use if you are deploying Application Accelerator in a production environment, or if you want to test the power and ease of using Application Accelerator for a large number of applications.

BMC subsystems and servers

In production mode, Application Accelerator works with the following BMC subsystems and servers:

- Cross Product Connectivity (CPC) subsystem, which is also known as the BMC Consolidated Subsystem (BCSS)
- BMC Primary Subsystem (BMCP)
- Advisor (ADV) server
- User Interface Middleware (UIM) server

Application Accelerator uses these subsystems and servers to:

- Screen IMS batch application jobs and steps
- Manage product setup and control information and the data that the product collects during execution
- Perform APF-authorized functions

The subsystems and servers are shared with other BMC products in the same environment, such as:

- MAXM Database Advisor for IMS
- MAXM Reorg solutions
- Backup and Recovery Solution for IMS

Note
If the JCL of an IMS batch application job step contains a DD statement identifying a trial-mode repository, the CPC subsystem does not participate in the job step.
Database Management Console

In production mode, you can use the BMC Database Management Console (a **graphical user interface**, or **GUI**) to set up and control Application Accelerator and view reports.

If you choose not to use the console, you can use a batch utility, instead.

The console runs on a client workstation under the Microsoft Windows operating system. The console communicates with the UIM server through TCP/IP technology.

CPC repositories

The first time that the CPC subsystem starts after Application Accelerator is installed, the subsystem allocates and initializes a set of CPC repositories. These repositories contain product setup and control information and the data that the product collects during execution.

Setup and control information includes default and site values for defining how Application Accelerator operates in your environment, and policies for defining how Application Accelerator interacts with your applications.

Using CPC repositories offers the following advantages:

- You do not need to identify the repositories in the JCL for executing the application job step. The CPC subsystem provides access to the repositories.
- You can use CPC procedures to automate backups of the repository data.
- CPC repositories support simultaneous updates across multiple z/OS systems.
- CPC repositories can accommodate very large amounts of data because the physical data is broken up into multiple data sets.

**Note**

Application Accelerator maintains only a small amount of data for each job step and uses the same set of data for multiple executions of the same job step. Therefore, this advantage might matter to you only if you have large numbers of IMS batch jobs executing in your environment.

SAF definitions

You can set up system authorization facility (SAF) definitions to secure and permit access to resources.
For more information, see “Managing access authority for BMP jobs” on page 38.

### Installation and maintenance

You install and maintain Application Accelerator libraries by using the BMC standard Installation System for mainframe products.

Other product components require little maintenance.

#### Installation System

You use the Installation System to install the Application Accelerator product modules. After prompting you for required values, the Installation System creates JCL that you can execute to prepare SMP/E zones.

To enable Application Accelerator execution on a specific CPU, you must install an authorization password from BMC. The three-character product code that you use to install Application Accelerator passwords is IBO.

Periodically, BMC distributes updates for Application Accelerator product modules in the form of PTFs. You use the Installation System to apply PTFs to your SMP/E zones.

For more information about these topics, see the Installation System documentation. Also, for specific information about installing Application Accelerator (including relevant FMIDs), see the Application Accelerator release notes.

#### Component maintenance

Application Accelerator trial-mode components typically need no maintenance during a brief trial period. If you use Application Accelerator in trial mode for an extended amount of time, the trial-mode repository might need the periodic maintenance and backups that you would perform for any VSAM data set.

In production mode, Application Accelerator product components are mostly self-maintaining:

- When the console on individual user systems should be updated, the console automatically detects that need and performs the update automatically. User intervention is seldom required.
The volume of Application Accelerator data in the CPC repositories is typically small. Data remains in the repositories for 30 days before being purged automatically.

You can automate CPC repository backups by setting parameters and a schedule in the console.

Ongoing operation of the CPC subsystem and its affiliated started tasks is managed in the same way that you manage other started tasks in the z/OS environment.

When Application Accelerator runs in production mode (without modification of application JCL), a single load module library is the source of all Application Accelerator modules. The CPC subsystem selects the first Application Accelerator library from its STEPLIB or JOBLIB concatenation as this single-source library. If you need to use Application Accelerator maintenance or special-test fixes in production mode, you must install those fixes in this single-source library.

For more information about maintaining production-mode components, see the Database Products for IMS Customization Guide.

Requirements, considerations, and limitations

The following requirements, considerations, and limitations apply to Application Accelerator.

System and software requirements

The following system and software requirements apply to Application Accelerator.

For requirements related to the CPC subsystem and other related components, see the Installation System documentation.

Table 1: Requirements for Application Accelerator

<table>
<thead>
<tr>
<th>Category</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Any version of IBM z/OS that IBM supports</td>
</tr>
<tr>
<td>IMS</td>
<td>IBM IMS Version 11 or later</td>
</tr>
</tbody>
</table>
### Security

APF authorization, provided through standard techniques

For example, for Application Accelerator in trial mode, you can authorize all libraries in the STEPLIB concatenation or use the CPC subsystem or DBUSS. In production mode, the CPC subsystem is APF authorized and performs authorized functions.

### Database Management Console

Personal computer that meets the following requirements:

- Oracle Java SE 6
- Microsoft Windows XP or later
- 105 MB of disk space
- 256 MB of RAM
- SVGA or higher video resolution (800 by 600, 256 color)
- A web browser

---

### Supported application and IMS elements

Application Accelerator supports the following application and IMS elements.

#### Supported application types

Application Accelerator supports applications that issue IMS Data Language/I (DL/I) calls in the following types of IMS regions:

- BMP
- DLI
- DBB

#### BMP applications

BMP job steps are eligible for optimization if the following conditions exist:

- The job step uses a program specification block (PSB) that includes a program communication block (PCB) specifying PROCOPT=GO* ("read without integrity").
- The job step issues a minimum of 100,000 DL/I calls through the PROCOPT=GO* PCB.
- If the PSB also includes a PCB that specifies update intent, that PCB is not used for the same database as the PROCOPT=GO* PCB (or is not used for any calls at all).
**DLO and DBB applications**

DLO and DBB job steps are eligible for optimization as follows:

- Applications that issue more than 1,000 calls are eligible for optimization of DFSVSAMP buffer pools.
- Applications that issue more than 25,000 calls are eligible for optimization of OSAM buffer pools.
- Applications that issue more than the default number of 100,000 calls are eligible for optimization with enhanced I/O techniques. You can change this default number of calls.

**Supported database types**

Application Accelerator supports all types of IMS databases except the following types:

- Fast Path databases
- Databases that are defined with logical DBDs
- Databases that are defined with virtual logical segments
- Hierarchical sequential access method (HSAM) databases
- Simple hierarchical sequential access method (SHSAM) databases
- Generalized sequential access method (GSAM) databases

**Support for Boolean AND operators**

You can now use Boolean AND operators in segment search arguments (SSAs). Application Accelerator supports a maximum of 10 AND operators in each SSA.

*Note*

The product does not support other Boolean operators.

**Support for the PROCSEQ parameter**

Application Accelerator now supports using the PROCSEQ parameter. However, you cannot use:

- PROCSEQs that require logical database definitions
- PROCSEQs for databases with hierarchical pointers when the target of the index is not the root
- PROCSEQs for databases with symbolic pointers when the target of the index is not the root
Support for command code F in SSAs

Application Accelerator now supports database calls that include command code F in SSAs.

Note
Application Accelerator does not support database calls that include other SSA command codes (L, M, N, O, U, or V).

Support for PSBs that specify POS=M

Application Accelerator now supports program specification blocks (PSBs) that specify the multiple positioning option with POS=M.

Memory requirements in the batch address space

The region size allocated to the batch address space (the DLI, DBB, or BMP address space in which the application code executes, not the IMS DLS address space where BMP DL/I calls are normally processed) determines how many DBPCBs that Application Accelerator can optimize in parallel through enhanced I/O techniques.

Application Accelerator requires the following amounts of below-the-line memory in the batch address space:

- 1.5 MB for the first DBPCB that is eligible for enhanced I/O optimization
- 200 KB for each additional DBPCB that is eligible for enhanced I/O optimization

If the total available below-the-line memory (after forcing REGION=0M) is insufficient to satisfy this requirement, Application Accelerator limits optimization to the DBPCBs that are best suited and that fit within the available memory.

Additional considerations for product interactions

Application Accelerator works with other BMC products that operate during execution of IMS batch applications, including:

- APPLICATION RESTART CONTROL for IMS
- MainView Batch Optimizer

To ensure that the products work together correctly and that required maintenance is applied, see the release notes for Application Accelerator and the other products.
Note
Application Accelerator does not optimize batch jobs that execute under the control of the BMC Application Director for IMS product.

Information and support

When using Application Accelerator, you can access the following resources for additional information or assistance:

- Application Accelerator provides context-sensitive online Help in the Database Management Console. The topics in the Help system provide:
  - Procedural information for completing product tasks
  - Conceptual information to support you while you are using the console
  - Complete reference information for all panels and fields
    You can access Help topics by clicking the **Help** button in the interface.

- Application Accelerator can issue error, warning, and informational messages during execution of your IMS batch applications. These messages can appear in the application log. The CPC subsystem and affiliated subsystems and servers can issue messages to your system console. For help with these messages, see the message topics in the BMC Documentation Center. The Documentation Center is available from BMC Support Central at [www.bmc.com/support](http://www.bmc.com/support). A stand-alone version is also available and does not require Internet access.

- BMC backs its products with outstanding customer support. You can speak directly with an on-call support specialist who has years of experience with IMS. You can e-mail BMC Customer Support, or you can obtain technical support on the BMC Support Central website or by telephone 24 hours a day, 7 days a week.
Using Application Accelerator in trial mode

This chapter provides instructions for implementing the BMC Application Accelerator for IMS product in trial mode.

Workflow summary for trial mode

You must complete the following basic tasks to use Application Accelerator in trial mode:

1. **Allocate the trial-mode repository, and initialize it with default values.**

   The trial-mode repository is a VSAM key-sequenced data set (KSDS). Application Accelerator uses the repository to store setup and control information, and data about monitored job steps.

   You can use the IBOINIT batch program to specify the default values that control how Application Accelerator operates in your environment and interacts with your applications. For example, you can use default values to:

   - Set the minimum number of application executions that Application Accelerator must monitor before optimizing
   - Specify how to allocate temporary data sets for Application Accelerator

   During initialization, the product obtains the default values from the repository. While the application is executing, the product captures relevant information about the characteristics of the run-time environment and stores that data in the repository.

   For more information, see “Allocating and initializing the trial-mode repository” on page 27.

2. **Modify the JCL for executing the application.**
In this step, you identify where the Application Accelerator load modules, trial-mode repository, and other relevant data sets reside.

In trial mode, Application Accelerator automatically initializes in every job step in which the Application Accelerator load modules are present. Typically, you will add the required product libraries to the STEPLIB concatenation in the job step JCL.

To identify the trial-mode repository to the job step, you add the $IBO$VSM DD statement to the job step JCL. BMC recommends that you also add DD statements for trace data sets.

If the load library is present for a job step that Application Accelerator should not process, you must include the IBOIGNOR DD DUMMY statement in the JCL for that step.

For more information, see “Modifying application JCL in trial mode” on page 29.

3 Provide an APF-authorized environment for Application Accelerator operations.

During application execution, Application Accelerator performs certain tasks that require APF authorization.

For more information, see “APF authorization in trial mode” on page 32.

4 Execute your application jobs, and evaluate the results of using Application Accelerator.

To identify and analyze the effects of Application Accelerator activities on your application executions, you can manually compare statistics from the same job step, executed with and without the product.

Application Accelerator writes a report to the $IBO$RPT data set that can help with evaluation efforts. For more information, see “$IBO$RPT report” on page 93.

5 If needed, troubleshoot problems.

For more information, see “Troubleshooting problems in trial mode” on page 33.
Allocating and initializing the trial-mode repository

You must allocate and initialize the trial-mode repository by creating a job that:

- Executes the IDCAMS utility to define a VSAM cluster to contain the repository
- Executes the IBOINIT utility to initialize the repository with default values that tell Application Accelerator how to interact with eligible batch application jobs and steps

In the Application Accelerator sample library, member IBOVINIT contains a job to perform these tasks. The following example JCL includes the significant statements from this member, but omits the explanatory comments:

```jcl
//IBOVINIT JOB ...  
//*  
//ALLOCATE EXEC PGM=IDCAMS  
//SYSPRINT DD SYSOUT=*  
//SYSIN DD *  
DELETE BMCIBO.TRIAL.$IBO$VSM CLUSTER PURGE  
DEFINE CLUSTER(NAME(BMCIBO.TRIAL.$IBO$VSM) -  
  INDEXED KEYS(62 6) SHAREOPTIONS (3,3) VOLUMES( DEV080 ) -  
  CYL(1 1) RECORDSIZE(2000 8185) NOREUSE -  
  CISZ(8192) FREESPACE(70 20) UNORDERED ) -  
  DATA(NAME(BMCIBO.TRIAL.$IBO$VSM.DATA)) -  
  INDEX(NAME(BMCIBO.TRIAL.$IBO$VSM.INDEX))  
//*  
//INIT EXEC PGM=IBOINIT  
//STEPLIB DD DISP=SHR,DSN=&hlqLOAD$IBO$VSM DD DISP=SHR,DSN=BMCIBO.TRIAL.$IBO$VSM  
//SYSUDUMP DD SYSOUT=*  
//IBOTRACE DD SYSOUT=*  
//BMCSYSIN DD *  
* CODE AN ASTERISK IN COLUMN 1 TO TREAT THE LINE AS A COMMENT  
CAPTURES=1  
IBOTRIG=1000  
TEMPUNIT=unitName  
MGMTCLAS=mgmtClas  
STRGCLAS=strgClas  
DATACLAS=dataClas  
//*  
//DISPLAY EXEC PGM=IDCAMS,COND=EVEN  
//SYSPRINT DD SYSOUT=*  
//SYSIN DD *  
LISTC ENT(BMCIBO.TRIAL.$IBO$VSM) ALL  
PRINT IDS(BMCIBO.TRIAL.$IBO$VSM)
```
Note

In trial mode, allocate a separate trial-mode repository for each z/OS system in which you will use Application Accelerator. Separate repositories prevent different systems from attempting to update a repository simultaneously.

In contrast, production mode requires only one set of repositories for multiple z/OS systems because the Cross Product Connectivity (CPC) subsystem manages the updates.

To allocate and initialize the trial-mode repository

1. Create a new member by copying member IBOVINIT of the Application Accelerator sample library.

2. Change the JOB statement as needed to suit the standards of your organization.

3. Change the name of the repository data set as needed to suit your trial environment.

   The default data set name in the sample JCL is BMCIBO.TRIAL.$IBO$VSM.

4. In the STEPLIB statement of the INIT step, specify the data set name of the library that contains the Application Accelerator executable load modules.

5. In the BMCSYSIN statement, either accept the default keyword values for the IBOINIT utility, or change the values (see Table 2 on page 28) to suit your trial environment.

   The IBOINIT utility uses the values that you specify to update the default values in the repository.

Table 2: Trial-mode keywords for the IBOINIT utility

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAPTURES</td>
<td>Any numeric value</td>
<td>3</td>
<td>Specify the number of times that Application Accelerator should gather statistics during execution of an application job step before recommending the job step for optimizing. The internal default value for this keyword is 3, but the suggested value for a trial-mode implementation is 1.</td>
</tr>
</tbody>
</table>
### Modifying application JCL in trial mode

In trial mode, you must make required changes to the JCL for executing each IMS batch application program that Application Accelerator should monitor or optimize.

#### Table of Keywords

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBOTRIG</td>
<td>Any numeric value</td>
<td>1000</td>
<td>Specify the number of IMS DL/I calls that a batch job step must make in a single execution before Application Accelerator will save monitoring data or evaluate optimizing for the job step. Application Accelerator benefits applications that issue a sufficient number of DL/I calls to offset the overhead of Application Accelerator processing. This parameter prevents Application Accelerator from participating in job steps that issue too few DL/I calls to benefit from that participation.</td>
</tr>
<tr>
<td>TEMPUNIT</td>
<td>Any valid unit name in your environment</td>
<td>SYSDA</td>
<td>Specify the unit name of the device that Application Accelerator should use when allocating small temporary data sets during processing.</td>
</tr>
<tr>
<td>MGMTCLAS</td>
<td>Any valid management class name in your environment</td>
<td>None</td>
<td>Specify the SMS management class of the device that Application Accelerator should use when allocating small temporary data sets during processing.</td>
</tr>
<tr>
<td>STRGCLAS</td>
<td>Any valid storage class name in your environment</td>
<td>None</td>
<td>Specify the SMS storage class of the device that Application Accelerator should use when allocating small temporary data sets during processing.</td>
</tr>
<tr>
<td>DATACLAS</td>
<td>Any valid data class name in your environment</td>
<td>None</td>
<td>Specify the SMS data class of the device that Application Accelerator should use when allocating small temporary data sets during processing.</td>
</tr>
</tbody>
</table>

6. Submit the edited JCL.

7. Verify that the job completed without errors and produced the expected results.

You can view the SYSPRINT data set from the DISPLAY step to see the contents of the repository.
These changes allow Application Accelerator to receive control during execution and access the trial-mode repository.

You can also make other changes to provide output data sets for diagnostic information, which can be helpful during the product trial period.

**Note**

In production mode, no application JCL changes are required. The Cross Product Connectivity (CPC) subsystem enables Application Accelerator to receive control and provides access to required and optional data sets.

In the Application Accelerator sample library, member IBOBATCH contains a sample DLI batch application job that runs with Application Accelerator. The following example JCL includes the significant statements from this member, omits the explanatory comments, and highlights the changes for Application Accelerator:

```java
//IBOBATCH JOB ...
//*
//DLT0GN EXEC PGM=DFSRRC00,PARM='DLI,DFSDDLTO,HIDALO2,,,,,,,,,,N',
//      REGION=5120K
//STEPLIB DD DSN=hlq.LOAD,DISP=SHR
//         DD DSN=hlq.IMLIB,DISP=SHR
//         DD DSN=hlq.PASSWORD,DISP=SHR
//         DD DSN=IMS.RESLIB,DISP=SHR
//DFSRESLB DD DSN=IMS.RESLIB,DISP=SHR
//$IBO$VSM DD DISP=SHR,DSN=BMCIBO.TRIAL.$IBO$VSM
//IMS DD DSN=IBS.TEST.PSBLIB,DISP=SHR
//      DD DSN=IBS.TEST.DBDLIB,DISP=SH
//*
//HID00002 DD DSN=IBS.TEST.HID00002,HID00002,DISP=SHR
//HIDIX02 DD DSN=IBS.TEST.HIDIX02,HIDIX02,DISP=SHR
//HIDSIX02 DD DSN=IBS.TEST.HIDSIX02,HIDSIX02,DISP=SHR
//HIDSIX03 DD DSN=IBS.TEST.HIDSIX03,HIDSIX03,DISP=SHR
//IEFRDER DD DSN=&LOG,DISP=(NEW,PASS),UNIT=SYSDA,
//      SPACE=(TRK,(1,1))
//*
//PRINTDD DD SYSOUT=*  
//SYSDUMP DD SYSOUT=*  
//DFSSTAT DD SYSOUT=*  
//IBOTRACE DD SYSOUT=*  
//$IBODIAG DD SYSOUT=*  
//DLITRACE DD SYSOUT=*  
//*
//*IBOIGNOR DD DUMMY
//*IBO#xxxx DD DUMMY
//*
//DFSVSAMP DD *  
//12288,6
//SYSIN DD *
S 1 1 1 1 1 1 HID00002
L 9999 GN HIDRUT02
```

**To modify an IMS batch application job step**

1. In the STEPLIB statement, at or near the beginning, insert the library (or libraries) that contain the BMC executable load modules for:
■ Application Accelerator
■ MAXM Reorg solutions
■ CPC components

The data set names and library contents depend on the options selected when the products were installed in your environment.

2 If you store your CPU ID authorization passwords in a separate data set, include that data set in the STEPLIB concatenation.

3 Add a $IBO$VSM DD statement to identify the data set name of the previously allocated trial-mode repository.

4 (optional) Add DD statements for diagnostic output data sets.

For the following data sets, specify the standard SYSOUT parameter in your environment. BMC recommends that you add these DD statements to all modified job steps during trial mode testing.

<table>
<thead>
<tr>
<th>DD statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLITRACE</td>
<td>In job steps that use BMC proprietary access methods for optimization, MAXM modules use DLITRACE to trace code flow and provide details when errors occur.</td>
</tr>
<tr>
<td>IBOTRACE</td>
<td>Application Accelerator modules use IBOTRACE to trace code flow and provide details when errors occur.</td>
</tr>
<tr>
<td>$IBODIAG</td>
<td>Application Accelerator modules use $IBODIAG to provide details when errors occur.</td>
</tr>
</tbody>
</table>

5 Only if needed, add the IBOIGNOR DD DUMMY statement to job steps that you do not want Application Accelerator to optimize.

The IBOIGNOR statement prevents Application Accelerator from:

■ Optimizing the current execution of this job step
■ Recommending optimization for the next execution

If you remove this DD statement from a subsequent job step execution, Application Accelerator does not optimize that execution but considers recommending optimization for subsequent executions. Monitoring continues.
APF authorization in trial mode

Application Accelerator must execute in an APF-authorized environment. When Application Accelerator runs in trial mode, it attempts to establish an APF-authorized environment through the following sequence:

- Application Accelerator determines whether the product modules are being loaded from an APF-authorized STEPLIB, JOBLIB, or LNKLST concatenation of libraries.
  
  If all of the concatenated libraries are APF-authorized, the product obtains authorization through conventional methods.
  
  To use an APF-authorized library, typically you must request help from the individuals who manage the z/OS environment for your organization. They might use an existing APF-authorized library for the product load modules, or create a new one. Instructions for authorizing load module libraries with APF are located in the IBM z/OS documentation set.
  
- If the libraries are not APF authorized, Application Accelerator determines whether the default Database Utilities subsystem (DBUSS) is active.
  
  The subsystem ID of the default DBUSS is DBUZ. If DBUZ is active, the utility obtains authorization through DBUZ.
  
  For more information about setting up and using a DBUSS, see the *Database Products for IMS Customization Guide*.
  
- If DBUZ is not active, Application Accelerator searches for an active CPC subsystem.
  
  If an active CPC subsystem is found, the product obtains authorization through that subsystem.
  
  For more information about setting up and using a CPC subsystem, see the *Database Products for IMS Customization Guide*.
  
- If no APF authorization is available by any of these methods, Application Accelerator issues an error message, and the application job continues without Application Accelerator participation.

*Note*

In production mode, Application Accelerator automatically uses the Cross Product Connectivity (CPC) subsystem for authorization.
Troubleshooting problems in trial mode

If a problem occurs while you are using Application Accelerator in trial mode, you can refer to the following topics.

Changing default values in trial mode

The default values that you use in trial mode are not likely to require changes. If they do, you can use the IBOINIT utility as outlined in the following procedure to change the values without affecting any data that Application Accelerator collected during monitoring and optimizing activities.

In the Application Accelerator sample library, member IBOVMUPD contains a sample job that updates the default values in the trial-mode repository. The following example JCL includes the significant statements from this member, but omits the explanatory comments:

```jcl
//IBOVMPUD JOB ...
//*
//MUPDATE EXEC PGM=IBOINIT
//STEPLIB DD DISP=SHR,DSN=&hlq.LOAD
//$IBO$VSM DD DISP=SHR,DSN=BMCIBO.TRIAL.$IBO$VSM
//IBOTRACE DD SYSOUT=* 
//SYSUDUMP DD SYSOUT=* 
//BMCPRT DD SYSOUT=* 
//BMCSYSIN DD *
* CODE AN ASTERISK IN COLUMN 1 TO TREAT THE LINE AS A COMMENT
CAPTURES=1
IBOTRIG=1000
TEMPUNIT=unitName
MGMTCLAS=mgmtClas
STRGCLAS=strgClas
DATACLAS=dataClas
//*
//DISPLAY EXEC PGM=IDCAMS,COND=EVEN
//SYSPRT DD SYSOUT=* 
//SYSIN DD *
LISTC ENTR(BMCIBO.TRIAL.$IBO$VSM) ALL PRINT IDS(BMCIBO.TRIAL.$IBO$VSM)
```

To change default values in the trial-mode repository

1. Create a new member by copying member IBOVMUPD of the Application Accelerator sample library.

2. Change the JOB statement as needed to suit the standards of your organization.

3. Specify the name of your repository data set.

   The default data set name in the sample JCL is BMCIBO.TRIAL.$IBO$VSM. You can issue a global change command to change this name throughout the job.
4 In the STEPLIB statement of the MUPDATE step, specify the data set name of the library that contains the Application Accelerator executable load modules.

5 In the BMCSYSIN statement, change or accept the values of the keywords that control the IBOINIT utility, as needed.

For more information, see “Allocating and initializing the trial-mode repository” on page 27.

6 Submit the edited JCL.

7 Verify that the job completed without errors and produced the expected results.

You can view the SYSPRINT data set from the DISPLAY step to see the contents of the repository.

Reproducing the trial-mode repository for troubleshooting

For problem diagnosis, BMC Customer Support might ask you to send your trial-mode repository to BMC. You can use the IDCAMS utility to reproduce the contents of the trial-mode repository to a sequential data set for easy transmission.

To reproduce the trial-mode repository

1 If requested by Customer Support, use the REPRO command to reproduce the contents of the trial-mode repository.

The following example JCL allocates the sequential data set and executes the REPRO command.

```
//IBOVRPRO JOB ...
//*
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=* 
//INDD DD DSN=BMCIBO.TRIAL.$IBO$VSM,DISP=SHR 
//OUTDD DD DSN=BMCIBO.TRIAL.$IBO$VSM.REPRO,UNIT=SYSDA, 
// DISP=(NEW,CATLG,DELETE), 
// DCB=(RECFM=VB,LRECL=32004,BLKSIZE=32008), 
// SPACE=(CYL,(10,10),RLSE) 
//SYSIN DD * 
REPRO INFILE(INDD) OUTFILE(OUTDD)
```
Preparing to use production-mode components

This chapter provides instructions for preparing BMC Application Accelerator for IMS product components for use in production mode.

Note

Unless you are responsible for setting up and administering the Cross Product Connectivity (CPC) subsystem and its affiliated subsystems, servers, and repositories, you can skip all sections of this chapter except “Installing the console” on page 42 and “Launching and exiting the console” on page 45. Use of the console is intuitive, especially if you are already familiar with standard Windows-based applications.

Transitioning from trial to production mode

When ready to transition from trial mode to production mode, use the following workflow:

1 Implement production-mode components as described in “Preparing to use production-mode components” on page 35.

2 Define required production-mode values and policies as described in “Using Application Accelerator in production mode” on page 55.

3 To test or phase in the use of the CPC subsystem and repositories, use the IBO#cpcID DD DUMMY statement to identify a CPC subsystem for an application job step.

For more information, see “Using production mode without screening application jobs” on page 96 and “Using a different set of CPC repositories” on page 97.
Overview of production-mode components

In production mode, Application Accelerator uses the Cross Product Connectivity (CPC) subsystem and its associated subsystems and servers for the following purposes:

- Screening and intercepting IMS application jobs so that Application Accelerator can receive control during application execution
- Managing Application Accelerator data
- Establishing an APF-authorized environment for Application Accelerator processing
- Communicating with the BMC Database Management Console

Note
Numerous BMC products can share the production-mode components, including these products:

- MAXM Database Advisor *for IMS*
- Backup and Recovery Solution *for IMS*
- BMC Fast Path suites
- MAXM Reorg solutions
- BMC System Administration *for IMS*
- Energizer *for IMS Connect*

If you are already using these components with another BMC product, no action is required other than to update the components as needed to support Application Accelerator.
Figure 1 on page 37 shows the production-mode components and the relationships between them.

**Figure 1: Production-mode components**

- You can use the Database Management Console (a graphical user interface, or GUI) to specify default and site values, establish policies, and view Application Accelerator reports. The console runs on a client workstation under the Microsoft Windows operating system and communicates with the User Interface Middleware (UIM) server through TCP/IP technology.

- The UIM server translates communication between the console and theADV server. The UIM server runs on the z/OS platform.

- The ADV server communicates with the UIM server, the repositories, IMS subsystems, and Application Accelerator. The ADV server runs on the z/OS platform.

- ADV repositories contain Application Accelerator setup and control information and the data that the product collects during execution. ADV repositories are VSAM data sets that are shared by all ADV servers in a sysplex. One ADV server is assigned to handle I/O to the repositories.

- The CPC subsystem manages APF-authorized functions and processes intercepted operating-system requests. Application Accelerator communicates with the CPC subsystem, which works with the ADV server to manage product data. The CPC
subsystem communicates with CPC subsystems on other z/OS images through the z/OS coupling facility.

- Although not shown in Figure 1 on page 37, the BMC Primary Subsystem (BMCP) establishes supervisory services for many BMC products. The BMCP communicates in the background with the CPC subsystem. One BMCP is installed on each z/OS image and is shared by all BMC products that are running on that image.

Managing access authority for BMP jobs

You can set up system authorization facility (SAF) definitions to secure and permit access to resources.

For Application Accelerator to participate in the execution of IMS batch message processing (BMP) jobs, the submitting user ID of that job must have the authority to access various IMS data sets, including but not limited to these data sets:

- IMS RECON data sets
- All libraries in IMS subsystem’s STEPLIB concatenation
- IMS subsystem’s MODSTAT or OLCSTAT data sets
- IMS subsystem’s ACBLIBA, ACBLIBBB, and ACBLIB libraries
- IMS database data sets
  IMS database data sets (only if Application Accelerator is optimizing the job step)

BMP jobs that execute without Application Accelerator run under the IMS control region, which already has the required authority to access these data sets. When Application Accelerator participates in the execution, by default the security access facility (SAF) grants the access that is defined for the user ID that submitted the job. This user ID typically does not have access authority to the required data sets.

Instead of defining SAF rules that allow access to each data set for each user ID that will submit a BMP job, you can define a single SAF resource that allows Application Accelerator to access the required data sets. If you define this resource, Application Accelerator extracts the Installation Data value from the resource definition and uses that value only to access the required data sets. For all other access, the job uses the default authority of the user ID that submitted the job.

Application Accelerator handles access requests as follows:

---

**Note**

For the following patterns, **.*.*.*** can be used for MVSID.IMSID.PSBNAMEN.
1 During initialization, Application Accelerator attempts to retrieve the resource profile, based on the job step values for the operating system ID, the IMS ID, and the program specification block (PSB) name. Specifically, Application Accelerator searches for a facility that has the pattern BBM.SDBA.*.*.*.AAOR.

2 If the BBM.SDBA.*.*.*.AAOR facility is found, Application Accelerator uses the value in the installation data field or the application data field. One of these fields must contain the user ID that Application Accelerator should use to access the required IMS resources.

3 If the BBM.SDBA.*.*.*.AAOR facility is _not_ found, Application Accelerator searches for a facility that has the pattern BBM.AAI.*.*.*.

4 If the BBM.AAI.*.*.* facility is found, Application Accelerator uses the user ID that is specified for IMS to access IMS resources.

5 If the BBM.AAI.*.*.* facility is _not_ found, Application Accelerator is provided with the default access authority for the job step.

   If Application Accelerator cannot access a required data set because of insufficient authority, the product switches to IGNORE mode and issues a message.

For more information about using SAF definitions with Application Accelerator, see the _Database Products for IMS Customization Guide_.

### To define a SAF resource rule for Application Accelerator

1 Identify a user ID that has access to the required libraries and other data sets in the online IMS subsystem.

   You can use the information in message IEF695I to identify a user ID, as shown in the following example:

   IEF695I START MXOAIMS WITH JOBNAME MXOAIMS IS ASSIGNED TO USER STCUSER , GROUP STCGROUP

2 Define a resource rule as follows:

   ■ Specify CLASS = FACILITY.

   ■ Specify PROFILE = BBM.SDBA.mvsid.imsid.psbname.AAOR.
      You can specify the operating system ID, IMS ID, and PSB name as generic values by using wildcard characters (* and %).

   ■ In the installation data field, specify the previously identified user ID.
Define the SAF profile as shown in the following example:

```
SETROPTS CLASSACT(FACILITY)
SETROPTS RACLST(FACILITY) REFRESH
RLIST FACILITY BBM.SDBA.*.MXOA.*.AAOR
```

You can specify the operating system ID, IMS ID, and PSB name values as specific or generic values.

### Configuring and starting mainframe components

Before you can use Application Accelerator in production mode, a system programmer or other person with the required level of authority must install and configure the CPC subsystem, BMCP, UIM server, and ADV server.

Over time, mainframe systems personnel operate these components and maintain them as needed. In addition, your system security specialist can establish system authorization facility (SAF) security for Application Accelerator resources to prevent unauthorized access and use.

For specific instructions about installing, configuring, and operating the mainframe components, see the Installation System documentation and the Database Products for IMS Customization Guide.

### Modifying startup procedures for existing subsystems and servers

Certain BMC products, such as MAXM Database Advisor for IMS and Backup and Recovery Solution for IMS, share subsystems and servers with Application Accelerator. If these products are already installed in your environment, you might need to modify the startup procedures for your existing subsystems and servers to work with Application Accelerator.

**Note**

If you are using startup procedures that were created when the Installation System installed Application Accelerator, the Installation System has already modified the startup procedures as needed to support Application Accelerator.
To modify startup procedures

1. In the startup procedure for the CPC subsystem, ensure that the following command is included in the COMMANDS data set:

   ```
   CPC IBO ENABLE
   ```

   This command should follow the REINIT CPC command.

2. In the startup procedure for the ADV server, ensure that the library containing the rules-engine load modules (distributed in FMID SMREnnn) is included in the STEPLIB concatenation.

Enabling and disabling Application Accelerator in the CPC subsystem

You can use commands to enable and disable Application Accelerator in the CPC subsystem.

You can modify the procedure for starting the CPC subsystem to issue the CPC IBO ENABLE command automatically when the subsystem is started or restarted (as explained in “Modifying startup procedures for existing subsystems and servers” on page 40), or you can issue the command manually.

You can disable Application Accelerator through the Turn off Application Accelerator parameter in the Database Management Console, or you can issue the CPC IBO DISABLE command manually.

When the CPC subsystem starts or restarts, it issues the following message to indicate that Application Accelerator is enabled:

```
BMC251550I IBO, SUBSYSTEM INTERCEPTS ACTIVATED
```

To enable Application Accelerator manually

1. From the z/OS system console, enter the following command:

   ```
   cpcID CPC IBO ENABLE
   ```

   `cpcID` is the subsystem identifier.

   If successful, this command enables Application Accelerator through subsequent restarts of the CPC subsystem, but the command does not persist across an IPL.

To disable Application Accelerator manually

1. From the z/OS system console, enter the following command:

   ```
   cpcID CPC IBO DISABLE
   ```
Starting subsystems and servers

Required subsystems and servers for Application Accelerator execute as started tasks.

In a production environment, you should automate submission of these started tasks.

You can use the manual Start (S) command to submit JCL for the BMCP, the CPC subsystem, and the UIM server for execution. These tasks typically remain running until stopped by a command (or an IPL). Initializing the CPC subsystem automatically starts the ADV server.

**Example**
The following commands start the BMCP, a CPC subsystem named IBOP, and a UIM server named IBOPUIM:

```
S BMCP
S IBOP
S IBOPUIM
```

Configuring the console

After the UIM server is installed and running, you can install, configure, and update the console. The UIM server downloads the required code to the console computer.

The default installation location is determined by the computer’s operating system. For example, the default installation location could be `C:\Users\user\Program Files\BMC Software\Database Management Console`.

Installing the console

Use the following procedure to install the console on a personal computer.

**Before you begin**

You will need:

- The name of the host computer on which the UIM server is running

```cpcID``` is the subsystem identifier.
To install the console

1. From your web browser, enter the URL for the UIM server on the mainframe.
   
   For example, enter `http://uimServerHostName:uimPortNumber/dna/index.html`, replacing the variables with your information.

2. On the resulting web page, click **Install Local Client**.

3. Based on the browser that you are using, either open the file from its current location, or save the file to your hard drive and then run the program:
   
   - If you are using Internet Explorer, perform the following steps:
     
     1. In the File Download dialog box, select **Run this program from its current location**, and click **OK**.
     
     2. If the Security Warning dialog box is displayed, click **Yes**.
   
   - If you are using Mozilla Firefox or a similar browser, clicking **Install Local Client** downloads the .exe file to the browser's default downloads folder. Browse to the saved file, and double-click it.

4. Enter the requested information in the BMC Database Management Wizard.

5. On the last page of the wizard, select **Launch the Database Management Console**.

6. Click **Finish**.

   The BMC Database Management Console starts.

7. On the last page of the wizard, select **Launch the Database Management Console**.

8. Click **Finish**.
The BMC Database Management Console starts.

If you implemented TLS prior to installing the console, a pop-up might be displayed indicating that the certificate is not trusted. This can occur if the certificate is self-signed, the intermediate certificate authority certificates are not in the RACF key ring, or the root certificate authority certificate is not in the trust store.

a (optional) Accept the certificate.

b Cut and paste the provided Uniform Resource Locator (URL) into your browser’s address bar and press Enter.

A pop-up is displayed that provides the URL.

The BMC Database Management Console starts.

Where to go from here

Tip
You can also implement TLS after you have installed the console. For more information, see the Database Products for IMS Advisors and Toolkit User Guide.

Updating the console

When you launch the console, the files on your personal computer (local client) are compared to the file residing on the UIM server. If a change in the installed products has occurred, the console is updated automatically.

Note
If the program that installs the console on the UIM server is updated, you must enter the URL for the UIM server on the mainframe to update the console. For more information, see “Installing the console” on page 42.

Uninstalling the console

Use the following procedure if you need to remove the console from your personal computer.

To uninstall the console

1 In Microsoft Windows, open the Control Panel.
For more information, see the documentation for your version of Windows.

2 In the Control Panel window, click Programs and Features.

3 Select and uninstall BMC Database Management Console (user).

The value of user is Admin if the console was installed by an administrator; otherwise, the value is the relevant user name.

Launching and exiting the console

The console uses z/OS authentication. When you launch the console, you must provide a valid SAF user ID and password. The security administrator for your site manages the SAF account information for users.

To launch the console

1 From the Start menu, select All Programs => BMC Software => Database Management Console.

The host selection dialog box (Figure 2 on page 45) or the logon dialog box (Figure 3 on page 46) is displayed.

Figure 2: BMC Database Management Console host selection dialog box
2 If the host selection dialog box (Figure 2 on page 45) is displayed, in the Host list, select your UIM host name.

**Tip**

You can manage the host list by clicking Manage Client Installs. For more information, see “Managing BMC Database Management clients” on page 46.

The logon dialog box (Figure 3 on page 46) is displayed.

3 In the logon dialog box, enter the user ID and password that is valid on the selected host (z/OS) system.

4 *(optional)* Enter your group and account information.

5 Click OK.

**To exit the console**

1 Click *userID => Logout*.

2 On the Quit? dialog box, click Yes.

---

**Managing BMC Database Management clients**

When you launch the console, you can manage your BMC Database Management clients from the host selection dialog box.
To manage the BMC Database Management clients

1. From the Start menu, select All Programs => BMC Software => Database Management Console.

The BMC Database Management Console host selection dialog box is displayed.

2. Click Manage Client Installs.

The Manage UIM Server Clients dialog box is displayed (Figure 4 on page 47).

Figure 4: Manage UIM Server Clients dialog box

3. Take the appropriate actions to manage your host list:

   - To add a host, click Add and complete the fields in the Details area.
   - To edit a host, select it in the host list, click Edit and change the fields in the Details area as needed.
   - To delete a host, select it in the host list, and click Delete.
   - To define a host to display as the default host when logging into the console, click the check box next to the host name.
Overview of the console

The BMC Database Management Console is a Windows application. You can use it to access console-enabled BMC products through a single interface.

Figure 5: BMC Database Management Console

The console interface features the following windows:

- Navigation window
- Messages window
- Work area

Navigation window

The Navigation window displays the data sources to which you have defined connections and other related objects.

To access commands from a pop-up menu, right-click an object in the tree.
Message windows

You can view messages in the Display Messages window, and in the Messages window at the bottom of the console. When you press CTRL-M, the Display Messages window appears as a separate window. These message windows list messages that the product generates during the current session.

The most recent message is displayed at the top of either window. To view detailed Help, click any message number.

Table 3 on page 49 describes the information that is displayed in the Display Messages and Messages windows.

Table 3: Message windows

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severity</td>
<td>Type of message: informational, warning, or error</td>
</tr>
<tr>
<td>Source</td>
<td>Where the message originated</td>
</tr>
<tr>
<td>When</td>
<td>Time that the message was generated</td>
</tr>
<tr>
<td>Message</td>
<td>Numerical identifier for the message and the message text</td>
</tr>
</tbody>
</table>

Message icons (Table 4 on page 49) indicate the severity of a message.

Table 4: Message icons

<table>
<thead>
<tr>
<th>Display Messages window icon</th>
<th>Messages window icon</th>
<th>Severity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Icon]</td>
<td>![Icon]</td>
<td>Informational</td>
<td>Status of your system or product</td>
</tr>
<tr>
<td>![Icon]</td>
<td>![Icon]</td>
<td>Warning</td>
<td>Noncritical problem that might interfere with system or product processes</td>
</tr>
<tr>
<td>![Icon]</td>
<td>![Icon]</td>
<td>Error</td>
<td>Critical problem that might interfere with system or product processes</td>
</tr>
</tbody>
</table>

From the Display Messages window, you can:

<table>
<thead>
<tr>
<th>Task</th>
<th>Click</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save the messages as an XML file</td>
<td>![Icon]</td>
</tr>
<tr>
<td>Clear all messages</td>
<td>![Icon]</td>
</tr>
<tr>
<td>Filter messages according to severity and source</td>
<td>![Icon]</td>
</tr>
</tbody>
</table>
To access Help or clear all messages from the Messages window, right-click a message.

**Work area**

Use the work area to work with the data sources that you select from the Navigation window, or with console tools.

The work area contains a window for each item that you select, as indicated by the windows tabs at the bottom of the work area (Figure 5 on page 48). Clicking a tab lets you see the information on that tab.

---

**Tip**

When you finish using a window, close it. Open windows consume computer memory.

---

**To sort columns in the work area**

1. To sort a column, click the column heading.

   An up or down arrow in the column heading shows whether the sort is ascending (up arrow) or descending (down arrow).

**To select rows in the work area**

1. To select rows, perform one of the following actions:

   - To select contiguous rows, click anywhere within the row and drag the mouse over the rows.

   - To select noncontiguous rows, press CTRL while selecting each row.

**Selecting user options**

The Set Options dialog box, which you open by choosing userID => Tools => Options, provides the ability to show the taskbar buttons in the work area.

You can display taskbar buttons for windows that are open in the work area. Taskbar buttons are displayed along the bottom edge of the work area. The default is to show the buttons.
Setting up connections

When you launch the console, you must define at least one host connection. After defining a host connection, you can work with resources on that host. The connection definition remains available whenever you start the console and log on.

In the IMS and DB2 perspectives of the console, the names of all defined connections are listed in the Connections folder in the Navigation window.

In the Jobs and Data Sets perspectives, you can define filters that determine which jobs or data sets are displayed. When you define a filter, you specify the connection on which to perform the filter. You can define more than one filter, and specify the same connection for all of them. For more information about adding filters to the Jobs or Data Sets perspectives, see the console Help.

In the UIMCommands perspective, you can specify which UIM servers appear as filters in the perspective by specifying the connection. The connection you specified when you logged on to the console appears by default as a filter (and not as a host connection) in the UIMCommands perspective.

Overview of the Enterprise List and personal connections

Host connections for individual users are managed separately from host connections for the entire enterprise. This separation makes it easier to isolate activities in different environments (such as testing systems versus production systems or application systems versus other application systems).

The console uses a shared Enterprise List of connections. This list resides on the UIM server. Users who have the appropriate security authority can add, delete, and edit connection information in the Enterprise List. When any user launches the console, the Enterprise List identifies the host connections that are defined.

Each user has a personal list of connections. You can define a connection in your personal list manually by entering connection information (such as the host name and port number). You can also add connections to your personal list by selecting predefined connections from the Enterprise List. After you define a host connection in your personal list, that connection definition remains available whenever you start the console and log on.

Managing the Enterprise List

If you have the appropriate authority, use the following procedure to add, edit, or delete connections in the shared Enterprise List on the UIM server.
To manage the Enterprise List

1 In the Navigation window, right-click the Connections folder and select **Manage Host Connections in the Enterprise List**.

The Manage Enterprise Connections dialog box is displayed.

2 Add, edit, or delete a connection as follows:

<table>
<thead>
<tr>
<th>To</th>
<th>Perform these steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a connection</td>
<td>1 Click <strong>Add</strong>.</td>
</tr>
<tr>
<td></td>
<td>2 Enter the connection information in the Details area.</td>
</tr>
<tr>
<td></td>
<td>3 Click <strong>Done</strong>.</td>
</tr>
<tr>
<td>Edit a connection</td>
<td>1 Select the connection, and click <strong>Edit</strong>.</td>
</tr>
<tr>
<td></td>
<td>2 Change the connection information in the Details area.</td>
</tr>
<tr>
<td></td>
<td>3 Click <strong>Done</strong>.</td>
</tr>
<tr>
<td>Delete a connection</td>
<td>1 Select the connection.</td>
</tr>
<tr>
<td></td>
<td>2 Click <strong>Delete</strong>.</td>
</tr>
</tbody>
</table>

3 Click **OK** to apply your changes to the shared Enterprise List and close the dialog box.

Using the Add Host Connection command

To define a single connection quickly, use the **Add Host Connection** command. Use this method to create a connection definition when you know the required connection information, such as host and port.

To add a connection

1 In the Navigation window, right-click the Connections folder and click **Add Host Connection**.

The Add Connection dialog box is displayed.
2 In the **Host** box, enter the name of the host.

3 In the **Port** box, enter the UIM port number.

   The value in the **Display Name** field is generated automatically from the host name and the port number.

4 *(optional)* In the Description box, enter a descriptive name for the connection.

5 In the Login Information area, select whether to connect by using console credentials or with credentials that you specify.

   __Note__

   Console credentials are the user name and password that you used to log on to the console. If you select **Connect using these credentials**, you must enter a user name and password. If necessary, you can also enter the account and group to which the user name belongs.

6 *(optional)* To connect automatically whenever the console starts, select **Autoconnect at console start**.

7 Click **OK**.

   The new host connection is displayed in the Connections folder in the Navigation window.

**To connect to a host**

1 In the Navigation window, right-click the host connection and click **Connect**.

   The Connect dialog box displays the connection information for the selected host.

2 In the Login Information area, select whether to connect by using console credentials or with credentials that you specify.

   __Note__

   Console credentials are the user name and password that you used to log on to the console. If you select **Connect using these credentials**, you must enter a user name and password. If necessary, you can also enter the account and group to which the user name belongs.

3 *(optional)* To connect automatically whenever the console starts, select **Autoconnect at console start**.

4 Click **OK**.
Using Application Accelerator in production mode

This chapter provides instructions for using the BMC Application Accelerator for IMS product in production mode.

Workflow for production mode

To use Application Accelerator in production mode, you perform the following basic tasks:

1. Set default values and site values to control basic Application Accelerator operations.

   For more information, see “Default and site values” on page 56.

2. Create and manage policies to tell Application Accelerator which application jobs are eligible for processing and how to handle them.

   For more information, see “Policies” on page 57.

3. After Application Accelerator optimizes one or more application job steps, view information that can help you evaluate the results of Application Accelerator activities.

   For more information, see “Evaluating Application Accelerator results” on page 82.

4. If needed, troubleshoot problems.

   For more information, see “Troubleshooting problems in production mode” on page 95.
Note
You can use the BMC Database Management Console to perform many production-mode tasks. For tasks in the console, this guide provides only basic instructions. For details, including field-level descriptions, see the online Help.

Default and site values

In production mode, you must define default values and site values that control basic Application Accelerator operations in your environment.

Note
Default and site values are sometimes called *parameters* or *options*.

Default values are specific to Application Accelerator and define how the product behaves in your environment. For example, you can use default values to:

- Enable or disable various Application Accelerator functions
- Enable or disable processing for various types of applications
- Set the minimum number of application executions that Application Accelerator must monitor before optimizing
- Specify how to allocate temporary data sets for Application Accelerator

Site values are shared with certain other BMC products in your environment and define how those products, including Application Accelerator, work with resources in your environment. For example, you can use site values to:

- Specify a valid JOB statement and control region sizes for various types of jobs
- Specify how to allocate work files and other data sets
- Define user variables
- Schedule backups and other maintenance tasks for product components

Default and site values reside in the Cross Product Connectivity (CPC) repository. The CPC subsystem provides access to this repository.

To set default and site values, you can use the Database Management Console. To set default values, you have the option of using a batch utility (IBOINIT). Use the method that you prefer:

- “Default and site values In production mode, you must define default values and site values that control basic Application Accelerator operations in your environment.”
Policies

In production mode, you use the CPC subsystem and policies to manage how Application Accelerator interacts with specific application jobs and steps in your environment.

A policy identifies an application by its identifying characteristics, such as the job name or program specification block (PSB) name. You create entries in the repository to define Include policies and Exclude policies.

The CPC subsystem screens all eligible IMS batch application jobs in the system. Application Accelerator uses a two-phase process to determine whether and how to interact with jobs and steps:

- **Phase 1**—Using the Exclude table and the job names or PSB names and z/OS System IDs from the Policies table, Application Accelerator builds the Include table. (You set a default value to specify whether to use job names or PSB names). You create entries in the Exclude table to specify the full names (with no wildcard characters) of jobs or PSBs to eliminate from further consideration. A job or PSB that matches an entry in the Exclude table is ineligible for further Application Accelerator processing.

  By using the job name or PSB name and the z/OS system ID from the Include table, you can limit the selection of batch jobs that Application Accelerator manages. If a job’s characteristics do not match an entry in the Include table, the job is also ineligible for further Application Accelerator processing.

- **Phase 2**—Application Accelerator uses the Policies table. You create entries in this table to specify the names or IDs of jobs, programs, PSBs, IMS systems, and z/OS systems. Your specifications can be generic (using a pattern of literal and wildcard characters) or specific (using all literal characters). The entries also specify the action that Application Accelerator should perform for job steps that match these qualifiers.

  Be cautious when using generic specifications. Because the Policies table lists the policies in the order that Application Accelerator will apply them, wildcard characters can cause policies to apply to unintended job steps. For example, if the first policy in the list contains an asterisk in all policy fields, that policy will apply to all job steps except those in the exclude list.

You can use the Database Management Console to set up and manage policies for Application Accelerator. The Manage Policy window on the console provides an interactive way to work with policies. Alternatively, you have the option of using a
batch utility (CPCBATCH) to set up and manage policies. Use the method that you prefer:

- “Managing policies through the console” on page 67.
- “Managing policies through the CPCBATCH utility” on page 72.

## Setting default and site values in the console

Use the following procedures to set up and maintain Application Accelerator default values and site values through the Database Management Console.

You can choose any of the following methods:

- “Setting default and site values in the Setup Wizard” on page 58 automatically leads you through the process of setting every default and site value. BMC recommends using this method for first-time setups.
- “Setting default values in the Manage Defaults window” on page 59 and “Setting site values in the Maintain Parameters window” on page 59 provide fast access to individual values. BMC recommends these methods when you want to change one or a few values.

### Setting default and site values in the Setup Wizard

Use this procedure to access the Setup Wizard and let it guide you through setup for the default values and site values.

**Note**

The Setup Wizard accesses and uses parameters and site values that are shared with other BMC products. If you change existing site values, you could affect the operation of those products.

#### To set or change default and site values through the Setup Wizard

1. In the Navigation window, right-click the IMS Applications object.

2. Select **Setup Wizard**.

3. View the displayed information and values on each page, and change the values as needed.
Tip
Click Next to move from one wizard page to the next.

4 On the last page of the wizard, click Finish to save your changes.

Setting default values in the Manage Defaults window

Use the following procedure if you prefer to set and change your Application Accelerator default values through the Manage Defaults window in the console.

Note
The parameters in the Manage Defaults window are identical to those on the Configure Defaults page of the Setup Wizard.

To set or change default values in the Manage Defaults window

1 In the Navigation window, right-click the IMS Applications object.

2 Select Manage Defaults.

3 Accept or change the displayed values for the parameters.

4 Click Save, or close the window and confirm that you want to save your changes.

Setting site values in the Maintain Parameters window

Use the following procedure if you prefer to use the Maintain Parameters window in the console to set and change site values. However, keep in mind that site values are shared with other BMC products. Changing existing site values could affect the operation of those products.

Note
Unless you are already familiar with the Maintain Parameters window, BMC recommends that you use the Setup Wizard to set site values.

To set or change site values in the Maintain Parameters window

1 In the Navigation window, right-click the IMS Databases object.

2 Choose Configuration => Maintain Parameters.
3 On the left side of the Maintain Parameters window, click the category of the parameters that you want to edit.

The following categories apply to Application Accelerator:

- Job Control
- Region
- Work Files
- DD Names
- User Variables
- Schedule Repository Backups

4 On the right side of the Maintain Parameters window, edit the displayed parameter values.

5 Click **Save**, or close the window and confirm that you want to save your changes.

### Setting default values with the IBOINIT utility

Use the following procedure if you prefer to use the IBOINIT batch utility (instead of the console) to set Application Accelerator default values.

**Note**

IBOINIT does not set site values. To set site values, see “Setting default and site values in the Setup Wizard” on page 58 or “Setting site values in the Maintain Parameters window” on page 59.

In the Application Accelerator sample library, member IBOCINIT contains a job to initialize the values. Member IBOCMUPD contains a job to update the values. The following example JCL includes the significant statements from this member, but omits the explanatory comments:

```
//INIT EXEC PGM=IBOINIT
//STEPLIB DD DISP=SHR,DSN=&hlqLOAD
//SYSUDUMP DD SYSOUT=* 
//BMCPRINT DD SYSOUT=* 
//IBOTRACE DD SYSOUT=* 
//BMCSYSIN DD */

CPCID=CPCX
CAPTURES=3
IBOTRIG=1000
TEMPUNIT=unitName
MGMTCLAS=mgmtClas
STRGCLAS=strgClas
DATACLAS=dataClas
IBOFF=N
SVCOFF=N
SVCEXP=N
PROCOPTA=Y
```
To initialize and update default values by using the IBOINIT utility

1. Create a new member by copying member IBOCINIT or IBOCMUPD of the Application Accelerator sample library.

2. Change the JOB statement as needed to suit the standards of your organization.

3. In the STEPLIB statement of the INIT or MUPDATE step, specify the data set name of the library that contains the Application Accelerator executable load modules.

4. In the BMCSYSIN statement, change or accept the values of the keywords that control the IBOINIT utility.

    Choose values to suit the requirements and conventions of your production environment. For descriptions of the keywords, see Table 5 on page 61.

    **Tip**

    To treat a line in the BMCSYSIN data set as a comment, insert an asterisk in column 1.

5. Submit the edited JCL.

6. Verify that the job completed without errors and produced the expected results.

### Table 5: Production-mode keywords for the IBOINIT utility

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPCID</td>
<td>Valid CPC ID</td>
<td>None</td>
<td>Specify the CPC identifier of the CPC subsystem that controls the repository to be updated.</td>
</tr>
<tr>
<td>CAPTURES</td>
<td>A numeric value</td>
<td>3</td>
<td>Specify the number of times that Application Accelerator should gather statistics during execution of an application job step before recommending the job step for optimizing.</td>
</tr>
<tr>
<td></td>
<td>The minimum value is 1.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------</td>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| IBMPRF   | N, Y            | N             | You can specify N if you are not running Application Accelerator under the IBM Program Restart Facility (PRF). Specify Y to run Application Accelerator under the IBM Program Restart Facility (PRF).  
**Note:** Attempting to run Application Accelerator under PRF when IBMPREF is set to N causes Application Accelerator to receive abend 806. |
| IBOTRIG  | A numeric value | 1000          | Specify the number of IMS (DL/I) calls that a batch job step must make in a single execution before Application Accelerator will save monitoring data or evaluate optimizing for the job step.  
Application Accelerator benefits applications that issue a sufficient number of (DL/I) calls to offset the overhead of Application Accelerator processing. This parameter prevents Application Accelerator from participating in job steps that issue too few (DL/I) calls to benefit from that participation. |
<p>| TEMPUNIT | Any valid unit name in your environment | SYSDA | Specify the unit name of the device that Application Accelerator should use when allocating small temporary data sets during processing. |
| MGMTCLAS | Any valid management class name in your environment | None | Specify the Storage Management Subsystem (SMS) management class of the device that Application Accelerator should use when allocating small temporary data sets during processing. |
| STRGCLAS | Any valid storage class name in your environment | None | Specify the SMS storage class of the device that Application Accelerator should use when allocating small temporary data sets during processing. |
| DATACLAS | Any valid data class name in your environment | None | Specify the SMS data class of the device that Application Accelerator should use when allocating small temporary data sets during processing. |</p>
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBOFF</td>
<td>N, Y</td>
<td>N</td>
<td>Specify N to enable Application Accelerator to process applications of all types. You can use other keywords to prevent processing of specific application types. Specify Y to disable Application Accelerator for all applications, regardless of their type. You might specify this value if a problem is occurring. <strong>Note:</strong> This parameter applies to applications for which Application Accelerator was implemented in production mode. It does not apply if Application Accelerator was implemented in trial mode.</td>
</tr>
</tbody>
</table>
| SVCOFF  | N, Y            | N             | Specify N to enable dynamic screening of IMS applications to determine whether Application Accelerator should participate in their executions. Specify Y to disable dynamic screening. You might specify this value if a problem is occurring with dynamic screening, or if you will never want to use dynamic screening. **Note:** This parameter does not disable Application Accelerator if either of the following conditions exists:  
  - The job step is using the product in trial mode.  
  - The job step JCL includes the IBO#cpcID DD statement, and the product load modules are included in the STEPLIB, JOBLIB, or LNKLST concatenation. |
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVCEXP</td>
<td>N, Y</td>
<td>N</td>
<td>Specify <strong>N</strong> to disable extended screening. This value limits Application Accelerator to reviewing attaches of DFSRRC00 (the IMS batch region controller) that occur from job step task control blocks (TCBs) only. Specify <strong>Y</strong> to enable extended screening. Application Accelerator will review all invocations of DFSRRC00, in addition to invoking it by specifying PGM=DFSRRC00 in the job step JCL. When you use this value, Application Accelerator reviews any attach of DFSRRC00, whether the attach is related to a job step execution or not.</td>
</tr>
<tr>
<td>PROCOPTA</td>
<td>Y, N</td>
<td>Y</td>
<td><strong>Note:</strong> This parameter does not apply to BMP applications. Specify <strong>Y</strong> to enable users to specify policies that use BMC custom I/O techniques for DLI and DBB applications that have update intent (through program control blocks, or PCBs, that have a PROCOPT other than G). If the application attempts to update a database, the custom I/O techniques dynamically disengage from the process. Specify <strong>N</strong> to disable use of custom I/O techniques for PCBs that have a PROCOPT other than G. For eligible applications, Application Accelerator can replace native (DL/I) call processes with BMC custom I/O techniques. An application is eligible if the application makes no update calls to any database. Application Accelerator automatically identifies PCBs for which a get-type PROCOPT is specified, and evaluates whether to replace the (DL/I) calls with custom I/O techniques. However, an application developer might code a PCB with a PROCOPT other than G, even if the application performs only read-type (DL/I) calls. To improve optimization results for these applications, you can use this keyword to attempt to use custom I/O techniques for PCBs that perform only read-type (DL/I) calls but are specified with a PROCOPT other than G.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default value</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------</td>
<td>---------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>SKIPRPT</td>
<td>N, Y</td>
<td>N</td>
<td>Specify N to dynamically allocate the $IBO$RPT data set (which contains information about job step characteristics and Application Accelerator activities) for each application job step. Specify Y to bypass (skip) dynamic allocation of the $IBO$RPT data set.</td>
</tr>
</tbody>
</table>
| SVCJOB and SVCPSB | Y, N          | SVCJOB= Y, SVCPSB= N | Use the SVCJOB and SVCPSB keywords to specify the characteristic that identifies applications to exclude from Application Accelerator participation:  
  - Specify SVCJOB= Y and SVCPSB= N to use the name of the application job, as specified with the JOB statement in the JCL for executing the application.  
  - Specify SVCJOB= N and SVCPSB= Y to use the name of the program specification block (PSB) that the application uses to access IMS resources.  
  **Warning:** If you change either of these values, Application Accelerator must delete existing entries in the Exclude table. |
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
</table>
| SVCBMP  | Y, N            | Y            | Specify Y to enable Application Accelerator to process applications that execute in an IMS batch message processing (BMP) region. Specify N to disable Application Accelerator from processing BMP applications. You might specify this value if one or more of the following conditions exist:  
  - You are phasing in Application Accelerator processing, based on application type.  
  - Your organization does not have any BMP applications, or never wants to use Application Accelerator for BMP applications.  
  - A problem is occurring when Application Accelerator processes a BMP application (but not other applications); you need to disable processing for BMP applications temporarily. |
| SVCDLI  | Y, N            | Y            | Specify Y to enable Application Accelerator to process applications that execute in an IMS batch DLI-type region. Specify N to disable Application Accelerator from processing DLI applications. You might specify this value if one or more of the following conditions exist:  
  - You are phasing in Application Accelerator processing, based on application type.  
  - Your organization does not have any DLI applications, or never wants to use Application Accelerator for DLI applications.  
  - A problem is occurring when Application Accelerator processes a DLI application (but not other applications); you need to disable processing for DLI applications temporarily. |
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVCDBB</td>
<td>Y, N</td>
<td>Y</td>
<td>Specify Y to enable Application Accelerator to process applications that execute in an IMS batch DBB-type region. Specify N to disable Application Accelerator from processing DBB applications. You might specify this value if one or more of the following conditions exist:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- You are phasing in Application Accelerator processing, based on application type.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Your organization does not have any DBB applications, or never wants to use Application Accelerator for DBB applications.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- A problem is occurring when Application Accelerator processes a DBB application (but not other applications); you need to disable processing for DBB applications temporarily.</td>
</tr>
</tbody>
</table>

### Managing policies through the console

In the console’s Manage Policy window, you can use the Policy tab to manage the Policies table, and the Exclude tab to manage the Exclude table.

Use the following procedures to work with policies through the console:

- “Accessing the Manage Policy window” on page 67
- “Adding and editing policies” on page 68
- “Rearranging the Policies table” on page 69
- “Filtering displayed policies” on page 70
- “Adding policies to the Exclude table” on page 71
- “Exporting policy information” on page 71

### Accessing the Manage Policy window

Use the following procedure to use the Manage Policy window in the console.
To access the Manage Policy window

1. In the Navigation window, right-click the IMS Applications object.
2. Select Manage Policy.
3. Click either the Policy tab or the Exclude tab.
4. View and change the tables as needed.

Tip

Clicking Refresh redisplays the tab with the latest values from the repository.

5. Click Save to store your changes in the repository.

This action saves any changes that you have made in both tabs.

Adding and editing policies

In the console’s Manage Policy window, the Policies table (in the lower portion of the Policy tab) lists the policies that define how Application Accelerator handles an eligible application jobs.

You can specify a maximum of 1000 policies.

Use the following procedures to add or edit policies. You can do so via the Add/Edit dialog, or by selecting application jobs from a list of job steps that Application Accelerator has been monitoring.

To add or edit a policy by using the Add/Edit dialog

1. In the Manage Policy window, click the Policy tab if it is not already displayed.
2. Take one of the following actions:
   - To edit an existing policy, click the corresponding row in the Policies table, and then click the Edit button.
   - To add a new policy that uses the values that you have entered in the Policy display filter, click the Add Like button.

The Add Like button can help you to insert a new policy in the correct position within the list of existing policies. The order of the policies is important when
determining which one will take effect; Application Accelerator uses the first policy that matches the job step.

The Add Like button is also convenient when you want to see if a policy with the required values already exists, and create one if not.

- To add a new policy without using filter values, click the Add button.

3 In the Add/Edit Policy dialog, specify values for the selection fields, and specify the action to take when the policy applies to a job step.

**Tip**

For most of the selection fields, you can specify wildcard (masking) characters to create generic policies. For more information and examples, see the online Help.

4 Click OK to store your changes in the Policies table, or click Cancel to abandon your changes.

**To add a policy by using the recommended jobs list**

1 In the Manage Policy window, click the Policy tab if it is not already displayed.

2 Click Recommend Jobs.

3 In the Recommended Jobs dialog, select a displayed job step.

**Note**

Application Accelerator must monitor and capture statistics for a job for a specified minimum number of runs before it can list the job in this dialog. The job steps that will provide the best resource savings when optimized are listed at the top.

4 Click Create Optimize Policy to create a policy that is based on the selected job step.

The Action field in the new policy is set to Optimize.

**Rearranging the Policies table**

The Policies table lists the policies in the order that Application Accelerator will apply them.

Application Accelerator attempts to match an eligible application job or step with these specifications, starting with the first policy in the list and continuing down the list until a policy matches. The first match determines the action that Application Accelerator will perform for the job or step. For this reason, specific policies will
precede generic policies in the Policies table. You can change the order of the policies in the table.

**To rearrange the Policies table**

1. In the Manage Policy window, click the Policy tab if it is not already displayed.
2. Select a policy by clicking the corresponding row in the Policies table.
3. Click **Move Up** or **Move Down** as needed to reposition the row in the table.

   **Note**

   You cannot move specific policies (those that contain all literal characters) below generic policies (those that contain one or more wildcard characters.)

4. Click **Save** to store your changes in the repository.

   **Tip**

   To abandon your changes, close the Manage Policies window without saving.

**Filtering displayed policies**

You can use the Policy Display Filter (in the upper portion of the Policy tab) to limit the displayed policies to those that match a specified filter.

Policies that do not match are still present in the Policies table, and still apply during Application Accelerator processing. The filter can make it easier for you to maintain the order of policies when the table contains a large number of policies.

**To filter displayed policies in the Policies table**

1. On the Policy tab, click **Add Like**.
2. Enter a values in a field in the Policies Display Filter table.

   As you enter a value in a field, the value immediately takes effect to limit the records displayed in the Policies table. Any policies that do not match this value are hidden until this field is cleared.

3. Click **OK** to apply the specified filters.

   **Tip**

   You can remove a filter and display all policies in the Policies table by clicking **Clear**.
Adding policies to the Exclude table

The Exclude table (in the Exclude tab of the Manage Policy window) defines the policies that Application Accelerator uses when determining whether to exclude an eligible job and bypass further processing of that job.

The Exclude table’s policies apply before Application Accelerator determines whether to monitor, optimize, or ignore eligible IMS batch application jobs and steps, based on the entries in the Policy tab.

Exclude policies are never generic; the job name or PSB name field cannot contain wildcard characters. However, you can specify an IBM MVS ID to exclude a job whenever it runs on the specified system. If you omit the MVS ID, the job is always excluded, regardless of the system on which the job is running.

Use the following procedure to add an exclusion policy. You can specify a maximum of 100 exclusion policies.

**To add an exclusion policy**

1. In the Manage Policy window, click the Exclude tab if it is not already displayed.
2. Click **Add**.
3. In the Add Exclude Policy dialog, specify values for the selection fields.
4. Click **OK** to add the policy to the Exclude List.

Exporting policy information

You can export the policy information in your Include and Exclude tables to an external file on your personal computer.

**To export policy information**

1. In the Manage Policies window, click the **Export** button to display the Export Data dialog.
2. In the **Select File Type** field, select **csv** to insert commas between fields in the exported information, or **txt** to insert tabs.
3. In the **File Name** field, specify the name of the file to contain the exported information.
4 Click Browse and navigate to the folder where you want to store the exported information.

5 Click Export to write the information to the specified file and folder.

Managing policies through the CPCBATCH utility

You can use the CPCBATCH utility to manage Application Accelerator policies through a batch job. This utility works with the Cross Product Connectivity (CPC) subsystem and repositories.

To use the CPCBATCH utility

1 Identify the tasks that you want to accomplish with the utility, and the functions to specify.

   For more information, see “Adding or changing policies with the CPCBATCH utility” on page 72.

2 Create JCL to execute the utility.

   For more information, see “CPCBATCH JCL requirements” on page 76.

3 Specify control statements in the CPCSYSIN data set.

   For more information, see “CPCSYSIN control statements” on page 77.

4 For the IMPORT or EXPORT function, work with the sequential data set that contains policy records.

   For more information, see “Sequential data set for the EXPORT and IMPORT functions” on page 79.

5 For the EVALUATE function, work with the sequential data set that contains policy records.

   For more information, see “Results of the EVALUATE function” on page 81.

Adding or changing policies with the CPCBATCH utility

You can use the CPCBATCH utility to perform the following tasks:
■ Implement policies in a new repository
■ Change existing policies in a repository
■ Test effects of policies

To implement policies in a new repository

1 Create a sequential data set (also known as a *flat file*), and specify policy records in this data set.

   For more information, see “Sequential data set for the EXPORT and IMPORT functions” on page 79.

2 Use the IMPORT function to import the records from the sequential data set to the repository.

   For more information, see “Importing policies” on page 74.

3 Use the REFRESH function to activate the new policies for Application Accelerator use.

   For more information, see “Refreshing policies” on page 75.

To change existing policies in a repository

1 Use the EXPORT function to export policies from the repository to a sequential data set.

   For more information, see “Exporting policies” on page 74.

2 Edit the policies as needed.

   For more information, see “Sequential data set for the EXPORT and IMPORT functions” on page 79.

3 Use the IMPORT function to import the edited policies.

   For more information, see “Importing policies” on page 74.

4 Use the REFRESH function to activate the new policies for Application Accelerator use.

   For more information, see “Refreshing policies” on page 75.

To test effects of policies

1 Use the EVALUATE function to evaluate existing policies and determine the action that Application Accelerator will take for a specific application job.
CPCBATCH functions for the POLICIES command

To work with Application Accelerator policies, you specify the following CPCBATCH utility functions by using the POLICIES command:

- EXPORT
- IMPORT
- REFRESH
- EVALUATE

Exporting policies

The EXPORT function exports existing policy records from the repository to a sequential data set. The data set can be an existing data set, or the utility can allocate a new data set.

Example

The following CPCSYSIN statement exports all policies from the repository to a data set that is named MY.POLICIES.FILE:

POLICIES RUNTYPE(EXPORT) DSN(MY.POLICIES.FILE)

Importing policies

The IMPORT function imports policy records from an existing sequential data set to the repository.

WARNING

The IMPORT function deletes all existing policies in the repository before importing the set of policies from the sequential data set. If the repository contains existing policies that you want to keep, export the policies to the sequential data set, and make changes to that set of exported policies.

Example

The following CPCSYSIN statement deletes all existing policies from the repository and imports the policies from a data set that is named MY.NEW.POLICIES.FILE to the repository:

POLICIES RUNTYPE(IMPORT) DSN(MY.NEW.POLICIES.FILE)
Refreshing policies

The REFRESH function activates a set of newly imported policies in the repository so that Application Accelerator can use them.

Example
The following CPCSYSIN statement refreshes the policies:

POLICIES RUNTYPE(REFRESH)

Note
Newly imported policies do not take effect until you run the REFRESH function or until the CPC subsystem is restarted.

Evaluating policies

Use the EVALUATE function to test a set of policies to determine the action (optimize, monitor, or ignore) that Application Accelerator will perform for a specified application job.

The CPCBATCH utility writes the results to the CPCLOG data set.

Note
The EVALUATE function does not consider the effect of Exclude policies when determining the action.

EVALUATE can be useful when you are testing policies to ensure that they perform as expected, or to determine whether a problem exists.

Example
You need to determine the action that Application Accelerator will take for an application job that:
- Executes with job name TESTJOB1
- Uses a program specification block (PSB) named PSB01
- Executes program TESTPGM1
- Executes in an IMS system that has IMS identifier IMSA
- Executes in a z/OS system that has z/OS identifier SYSX

Specify the following CPCSYSIN statement:

POLICIES RUNTYPE(EVALUATE) JOBNAME(TESTJOB1) PSB(PSB01) - PROGNAME(TESTPGM1) IMS(IMSA) MVSID(SYSX)
CPCBATCH JCL requirements

The following JCL statements are valid for the CPCBATCH utility:

<table>
<thead>
<tr>
<th>Statement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXEC</td>
<td>*(required) Specify PGM=CPCBATCH.</td>
</tr>
<tr>
<td>STEPLIB DD</td>
<td>*(required) Identify the library that contains the CPC load modules.</td>
</tr>
<tr>
<td>CPCLOG DD</td>
<td><em>(required for EVALUATE functions; recommended for other functions) Identify an output data set to contain the CPC message log and the policy report that the utility produces for the EVALUATE function. Typically, this statement specifies a standard SYSOUT=</em> data set.</td>
</tr>
<tr>
<td>ddname DD</td>
<td>*(POLICIES: optional for EXPORT and IMPORT functions) Identify a sequential data set to contain the policies to export, or one that contains the policies to import. *(JOBSTEPS: optional) Identify a sequential data set to contain the resource savings report. For more information about using the JOBSTEPS command to generate a batch report, see Generating a batch report on page 85</td>
</tr>
<tr>
<td>CPCSYSIN DD</td>
<td>*(required) Identify the data set that contains control statements for the CPCBATCH utility. Typically, this statement specifies an in-line data set (CPCSYSIN DD *).</td>
</tr>
</tbody>
</table>

Example job for POLICIES

The following example job shows a CPCBATCH job that specifies policies for Application Accelerator:

```cpcbatch
//CPCBATCH JOB ...
//POLICIES EXEC  PGM=CPCBATCH,REGION=region
//STEPLIB DD DISP=SHR,DSN=cpcOptionsLibrary
// DD DISP=SHR,DSN=bmcLoadLibrary
//CPCLOG DD DISP=SHR,DSN=yourPolicyExportDsn
//CPCSYSIN DD *
POLICIES -
RUNTYPE(EXPORT) -
DDN(PLCYEXPT)
/*
```

Example job for JOBSTEPS

The following example job shows a CPCBATCH job that creates a resource savings report:

```cpcbatch
//CPCBATCH JOB ...
//POLICIES EXEC  PGM=CPCBATCH,REGION=region
//STEPLIB DD DISP=SHR,DSN=cpcOptionsLibrary
// DD DISP=SHR,DSN=bmcLoadLibrary
//CPCLOG DD DISP=SHR,DSN=yourResourceSavingsDsn
//CPCSYSIN DD *
```
CPCSYSIN control statements

The data set for the CPCSYSIN control statements contains 80-character, fixed-length records that control the actions of the CPCBATCH utility.

CPCSYSIN syntax

In the CPCSYSIN data set, you can specify control statements by using the following syntax elements:

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commands</td>
<td>The control statement must begin with a valid command. A command can have keywords and comments, separated where necessary with separators and continuation characters (if statements continue to the next line). A separator must follow a command. You can enter commands anywhere in positions 1 through 72 of the input statement; positions 73 through 80 are ignored.</td>
</tr>
<tr>
<td>Keywords</td>
<td>Keywords follow a command and invoke options. All keywords are nonpositional. You specify a keyword with a value following it in parentheses. The keyword value can be any character string up to 255 characters. All alphanumeric and special characters are allowed.</td>
</tr>
<tr>
<td>Comments</td>
<td>Comments consist of an alphanumeric character string beginning with a slash-asterisk (/<em>) and ending with an asterisk-slash (</em>/). Comments cannot start in position 1 of an input statement.</td>
</tr>
<tr>
<td>Separators</td>
<td>When you require a separator, use a blank, a comma, or a comment. You can use more than one separator between keywords. Do not use a separator between a keyword and its value.</td>
</tr>
<tr>
<td>Continuation</td>
<td>Valid continuation characters are the plus sign (+) and the minus sign (-). Use them to continue control statements and comments that do not fit on a single line of input. The continuation character must be the last nonblank character. Use the continuation characters as follows:</td>
</tr>
<tr>
<td>characters</td>
<td>- Use the plus sign (with no spaces before it) to continue values for a single keyword to the next line. The plus sign deletes leading separators from the continued line.</td>
</tr>
<tr>
<td></td>
<td>- Use the minus sign (with a space before it) to continue a list of keywords for a single command. The minus sign does not delete leading separators from the continued line.</td>
</tr>
</tbody>
</table>
To specify policies for Application Accelerator, you specify the POLICIES command. The following keywords are valid with the POLICIES command:

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted Values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RUNTYPE</td>
<td>EXPORT</td>
<td>None</td>
<td>Specify the function for the CPCBATCH utility to perform. This keyword is required for all functions.</td>
</tr>
<tr>
<td></td>
<td>IMPORT</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>REFRESH</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EVALUATE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DDNAME (or DDN)</td>
<td>ddname</td>
<td>None</td>
<td>If a DD statement in the utility JCL identifies a file that will contain exported data or that contains the data to import, specify this keyword to identify the ddname of that statement. If you specify this keyword, do not specify the DSNAME keyword. This keyword is valid for EXPORT and IMPORT functions only.</td>
</tr>
<tr>
<td>DSNAME (or DSN)</td>
<td>Data set name</td>
<td>None</td>
<td>If no DD statement in the utility JCL identifies a file that will contain exported data or that contains the data to import, specify this keyword to identify the data set name of the file. If you specify this keyword, do not specify the DDNAME keyword. For the EXPORT function, the CPCBATCH utility will create the data set if it does not already exist. For the IMPORT function, the data set must exist before you execute the CPCBATCH utility. This keyword is valid for EXPORT and IMPORT functions only.</td>
</tr>
<tr>
<td>JOBNAME</td>
<td>Job name</td>
<td>None</td>
<td>Specify the job name of the application job to evaluate. This keyword is valid for EVALUATE functions only.</td>
</tr>
<tr>
<td>PSB</td>
<td>PSB name</td>
<td>None</td>
<td>Specify the program specification block (PSB) name that is associated with the application job to evaluate. This keyword is valid for EVALUATE functions only.</td>
</tr>
<tr>
<td>PROGNAME</td>
<td>Program name</td>
<td>None</td>
<td>Specify the program name of the application job to evaluate. This keyword is valid for EVALUATE functions only.</td>
</tr>
<tr>
<td>IMS</td>
<td>IMSID</td>
<td>None</td>
<td>Specify the IMS identifier of the IMS system that is associated with the application job to evaluate. This keyword is valid for EVALUATE functions only.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted Values</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>MVSID</td>
<td>System ID</td>
<td>None</td>
<td>Specify the system identifier of the z/OS system that is associated with the application job to evaluate. This keyword is valid for EVALUATE functions only.</td>
</tr>
</tbody>
</table>

**Sequential data set for the EXPORT and IMPORT functions**

You can edit the sequential data set that contains data for the EXPORT and IMPORT functions to change, add, or remove policies.

In the following example, a sequential data set is open for editing in the ISPF Edit panel. The COLS ON command displays a column counter that can help with data entry.

```
File  Edit  Edit_Settings  Menu  Utilities  Compilers  Test  Help
EDIT       RIHRXC.POLS228.CNTL                             Columns 00001 00072
Command ===>                                                  Scroll ===> CSR
=COLS> ----+----1----+----2----+----3----+----4----+----5----+----6----+----7--
****** ********************** Top of Data **********************
000001 E AAITESTX                            SYSP
000002 E AAITESTX                            IMSA
000003 I AAITEST3 PSB01    DFSDDLT0 BMC1     SYSP     I
000004 I AAITEST2 PSB01    DFSDDLT0 BMC1     SYSP     M
000005 I AAITEST1 PSB01    DFSDDLT0 BMC1     SYSP     O
****** ********************** Bottom of Data **********************
```

**Order of policies in the sequential data set**

Application Accelerator applies the policies in the order that they are specified in the data set. If the order is incorrect, the wrong policy might be applied to an application job. Specific policies should precede generic policies (those that contain wildcard characters).

**Format of the sequential data set**

The sequential data set has a logical record length (LRECL) of 80 bytes. When the CPCBATCH utility creates the data set, the data set has a fixed-blocked (FB) format with a half-track block size.

**Fields in policy records**

Each policy record in the sequential data set contains the following fields:
### Policy Type

Specify the type of policy as follows:

- **E** specifies an exclude policy, which causes Application Accelerator to bypass processing for a matching job. The following fields apply in an exclude policy and should be the only fields that are specified when the **Policy Type** field contains E:
  - **Job Name** or **PSB Name** (depending on an Application Accelerator default value)
  - **MVSID**

- **I** specifies an include policy, which causes Application Accelerator to continue processing for a matching job. All fields apply to an include policy.

### Columns | Field | Contents
---|---|---
1 | Policy Type | Specify the type of policy as follows:
3 through 10 | Job Name | Specify the name of the application job, as specified with the JOB statement in the JCL for executing the application.
12 through 19 | PSB Name | Specify the name of the program specification block (PSB) that the application uses to access IMS resources.
21 through 28 | Program Name | Specify the name of the application program, as specified with the PGM parameter of the EXEC statement in the JCL for executing the application.
30 through 37 | IMSID | Specify the IMS identifier to identify the IMS system that is specified in the EXEC parameter of the application JCL or in the IMS control blocks that are used in the eligible job step.
39 through 46 | MVSID | Specify the system identifier of the z/OS system in which the application job executes. Specify a complete, specific identifier, or leave the value blank to indicate that the policy applies to all z/OS systems.
48 | Action | Specify the action that Application Accelerator will take for an application job that matches this policy:
  - **O** specifies that Application Accelerator will optimize the application execution.
  - **M** specifies that Application Accelerator will monitor the application execution.
  - **I** specifies that Application Accelerator will ignore the application execution.
Specific and generic values

To identify a specific value for a job characteristic, such as the job name or IMS system, specify the complete name or identifier.

**Example**
Specify ACCTUPDT if you want the policy to apply only when the ACCTUPDT job is running.

To identify a generic value for a job characteristic, so that the value applies to any name or identifier that matches a pattern, specify wildcard characters as follows:

- Specify a percent sign (%) in any position to represent a single-character wildcard.

  **Example**
  Specify ACCT%%%1 if you want the policy to apply when the ACCTABC1 or ACCTDEF1 job is running, but not when the REPTABC1 or ACCTABC2 job is running.

- Specify an asterisk (*) as the last non-blank character to represent a multiple-character wildcard for the remainder of the ID.

  **Example**
  Specify ACCT* if you want the policy to apply when a job whose name starts with the string ACCT is running.

You can specify both % and * in the same value.

For the MVSID value in an include policy or exclude policy, you can specify a complete value, or leave the value blank to match all z/OS systems. No wildcard characters are permitted.

For the job name or PSB name value in an exclude policy, you must specify a complete name; no wildcard characters are permitted.

Results of the EVALUATE function

The EVALUATE function determines the effects of policies on specified application jobs, and writes the resulting information to the CPCLOG data set. The following example shows how Application Accelerator will handle selected application jobs:

- Optimize job AAITEST1
- Monitor job AAITEST2
- Ignore job AAITEST3
■ Do not participate in job AAITEST4, because no policy applies to the job

Evaluating Application Accelerator results

In production mode, you can evaluate the results of Application Accelerator activities as follows:

■ You can use the console’s Resource Savings window to view overall and individual savings.

For more information, see “Viewing information about resource savings” on page 83.
In the Job Step History window, you can view statistical information that Application Accelerator has collected about a selected job step over a selected period of time.

For more information, see “Viewing job step history information” on page 84.

For job step status information specific to applications, you can view the Job Status window on the console.

For more information, see “Viewing job step history information” on page 84.

You can generate batch reports that contain the same information as the Resource Savings window and the Job Status window.

For more information, see “Generating a batch report” on page 85.

You can evaluate results manually by comparing the job step’s statistics when executed with Application Accelerator and without Application Accelerator. Application Accelerator writes a report to the $IBO$RPT data set that can help with evaluation efforts. For more information, see “$IBO$RPT report” on page 93.

Viewing information about resource savings

After Application Accelerator optimizes one or more job steps in production mode, you can view information about the resources that were saved through optimization, such as:

- Total CPU service units and elapsed seconds saved
- CPU service units and elapsed seconds saved during individual executions

You can view this information in the Resource Savings window on the console:

- The upper portion of the window reports overall results for all optimized job steps.
- The lower portion displays results for each optimized job step.

To view resource-savings information on the console

1. In the Navigation window, right-click the IMS Applications object.
2. Select Resource Savings.
3. If you want to limit the report to a specified job (or jobs), enter a job name in the pop-up dialog.
You can use wildcard characters to specify a generic (masked) name. Use % to match any character in the specified position. Use * to match any characters from the specified position through the end of the name.

4 View the information in the Resource Savings window.

---

**Tip**

You can export the statistics to an external file by clicking the Export button. You can also view detailed status information about job steps by clicking the Job Status button.

---

**Viewing job step history information**

You can view statistical information that Application Accelerator has collected while monitoring or optimizing IMS batch application job steps over time. This information can help you evaluate actual and potential optimization efforts for the job step.

**To view job step history information**

1 In the Resource Savings window, select the job step of interest.

   For more information, see “Viewing information about resource savings” on page 83.

2 Click Show History.

3 In the Show Job Step History pop-up, select one of the following options to display job step information:

   - **Detail** displays detailed statistics.
   - **Monthly** displays monthly average statistics.
   - **Yearly** displays yearly average statistics.

   The information is displayed in the Job Step History window.

---

**Tip**

You can view the information in table format or in various chart formats by selecting the corresponding tab at the top of the window. You can also export the statistics to an external file by clicking the Export button.
Viewing job step status information

You can view information about the actions that Application Accelerator took for eligible applications. This information can help you understand why Application Accelerator handled job steps as it did.

To view job step status information

1. In the Navigation window, right-click the IMS Applications object.
2. Select Job Status.
3. If you want to limit the report to a specified job (or jobs), enter a job name in the Job Status pop-up.

You can use wildcard characters to specify a generic (masked) name. Use % to match any character in the specified position. Use * to match any characters from the specified position through the end of the name.

The information is displayed in the Job Status window.

Tip
You can export the statistics to an external file by clicking the Export button.

4. For more information, select a job step, and click the Details button.

The Detail Job Status window is displayed. You can also access this window from the Resource Savings window.

Generating a batch report

You can generate a batch report to view the overall and individual results of using Application Accelerator to optimize eligible job steps. You can request a batch report through the console or through the CPCBATCH utility. Application Accelerator generates the report and writes it to the specified mainframe data set.

To generate batch reports through the console

1. In the Navigation window, right-click the IMS Applications object.

Application Accelerator submits a batch job to generate the report. When the job completes successfully, Application Accelerator issues message
BMCMXO340683I, which indicates the name of the mainframe data set that contains the report. The information is displayed in the BMC Application Accelerator for IMS Savings History window.

---

**Tip**

If you want to access a previously generated report without regenerating it, select **Batch Reports => View Batch Report**, and enter the name of the data set that contains the report.

You can export the report to an external file by clicking the **Export** button.

---

## To generate batch reports through the CPCBATCH utility

1. Identify the report and content that you want to generate:
   - The Resource Savings Report, which shows the accumulated savings for the current profile of each job step
   - The JOBSTEP History Report, which shows the savings for each execution of each job step for which data is available

2. Create JCL to execute the utility.
   
   For more information, see “CPCBATCH JCL requirements” on page 76.

3. Specify control statements in the CPCSYSIN data set.
   
   For more information, see the following topics:
   - “CPCSYSIN control statements” on page 77
   - “CPCSYSIN keywords for the JOBSTEPS command” on page 86
   - “Example JOBSTEPS command statements” on page 89

4. Run the JCL and view the reports.
   
   For more information, see “Description of data columns on the resource savings reports” on page 90.

## CPCSYSIN keywords for the JOBSTEPS command

To create the resource savings reports for Application Accelerator, you specify the JOBSTEPS command. The following keywords are valid with the JOBSTEPS command:
### Keyword Table

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Accepted values</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>REPORT_TYPE</strong></td>
<td>SAVINGS</td>
<td>None</td>
<td>Type of report to create:</td>
</tr>
<tr>
<td></td>
<td>HISTORY</td>
<td></td>
<td>- REPORT_TYPE(SAVINGS) reports the most recent savings information and is a batch version of the request in the console’s Resource Savings window.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- REPORT_TYPE(HISTORY) reports all of the historical data available for the requested jobs. This keyword is required.</td>
</tr>
<tr>
<td><strong>NOTITLE</strong></td>
<td>Y</td>
<td>N</td>
<td>Whether to omit report title lines (Y for yes, or N for no) Indicates whether to include or omit report title lines. Use this keyword if you want to export the report to a spreadsheet. This keyword is optional.</td>
</tr>
<tr>
<td></td>
<td>N</td>
<td></td>
<td>Use this keyword if you want to export the report to a spreadsheet. This keyword is optional.</td>
</tr>
<tr>
<td><strong>METHODS</strong></td>
<td>Yes</td>
<td>No</td>
<td>Whether the report should indicate the methods used for optimization This keyword is optional.</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td></td>
<td>This keyword is optional.</td>
</tr>
<tr>
<td><strong>DETAIL_TYPE</strong></td>
<td>D (Detail)</td>
<td>All types</td>
<td>The type of repository records to process:</td>
</tr>
<tr>
<td></td>
<td>M (Monthly)</td>
<td></td>
<td>- D is detail</td>
</tr>
<tr>
<td></td>
<td>Y (Yearly)</td>
<td></td>
<td>- M is monthly</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Y is yearly</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>This keyword is used with REPORT_TYPE(HISTORY).</td>
</tr>
<tr>
<td><strong>DDName</strong></td>
<td>ddname</td>
<td>None</td>
<td>The ddname of the report data set This keyword is mutually exclusive with DSNAME. The JCL should include a DD statement with this ddname.</td>
</tr>
<tr>
<td><strong>DSName</strong></td>
<td>Data set name</td>
<td>None</td>
<td>The data set name of the report data set This keyword is mutually exclusive with DDNAME. If the specified data set does not exist, Application Accelerator creates a data set with this name.</td>
</tr>
<tr>
<td>Keyword</td>
<td>Accepted values</td>
<td>Default</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>JOBname</td>
<td>Job name</td>
<td>None</td>
<td>The job for which you want to generate the report. If you omit this keyword, the report will include all job steps for which data is available in the repositories. Additional considerations follow:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- If you specified REPORT_TYPE(SAVINGS), you can use wildcards in the JOBNAME value:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- A percent sign (%) represents any character.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- An asterisk (*) at the end of the value includes any names that start with the preceding characters.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- If you specified REPORT_TYPE(HISTORY), you must specify additional job step criteria (using one or more of the keywords specified in the subsequent rows of this table). Also, you cannot use wildcards.</td>
</tr>
<tr>
<td>STEPname</td>
<td>Job step name</td>
<td>None</td>
<td>The execution job step (to further qualify the requested job name)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>This keyword is used with REPORT_TYPE(HISTORY).</td>
</tr>
<tr>
<td>PROCNAME</td>
<td>Procedure step name</td>
<td>None</td>
<td>The executing procedure step if a procedure is used (to further qualify the requested job name)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>This keyword is used with REPORT_TYPE(HISTORY).</td>
</tr>
<tr>
<td>PSBname</td>
<td>PSB name</td>
<td>None</td>
<td>The name of the program specification block (PSB) that the application used to access the IMS databases (to further qualify the requested job name)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>This keyword is used with REPORT_TYPE(HISTORY).</td>
</tr>
<tr>
<td>PROGNAME</td>
<td>Program name</td>
<td>None</td>
<td>The name of the batch application program executed in the job step (to further qualify the requested job name)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>This keyword is used with REPORT_TYPE(HISTORY).</td>
</tr>
<tr>
<td>IMSid</td>
<td>IMSID</td>
<td>None</td>
<td>The ID of the IMS system specified on the job step EXEC parameter (to further qualify the requested job name)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>This keyword is used with REPORT_TYPE(HISTORY).</td>
</tr>
</tbody>
</table>
### Example JOBSTEPS command statements

The following examples show how to use the different keywords for the JOBSTEPS command:

**Example**

The following command reports the total Application Accelerator resource savings, and shows the savings for each job step. The report is placed in data set `yourReportDsn`.

```
JOBSTEPS REPORT_TYPE(SAVINGS) DSN(yourReportDsn)
```

**Example**

The following command reports the total Application Accelerator resource savings and shows the savings for each step whose job name is `MYJOB`. The report is placed in the data set provided with the AAISAVGS ddname.

```
JOBSTEPS REPORT_TYPE(SAVINGS) DDN(AAISAVGS) JOBNAME(MYJOB)
```

**Example**

The following command reports the total Application Accelerator resource savings and shows the savings for each job step whose job name begins with `AAI`. The report is placed in data set `yourReportDsn`.

```
JOBSTEPS REPORT_TYPE(SAVINGS) DSN(yourReportDsn) JOBNAME(AAI*)
```

**Example**

The following command shows the Application Accelerator resource savings of an execution of a job step and places it in data set `yourReportDsn`. The report contains all job steps for which data is available in the repositories.

```
JOBSTEPS REPORT_TYPE(HISTORY) DSN(yourReportDsn)
```

**Example**

The following command shows the Application Accelerator resource savings of an execution of a job step and places it in the data set provided with the AAIIHIST ddname. The report contains all job steps that match the specified criteria for which data is available in the repositories.

```
JOBSTEPS REPORT_TYPE(HISTORY) DDN(AAIIHIST) - JOBNAME(MYJOB) STEPNAM(STEP1) PROCSTEP(PROC1) - PROGNAME(MYPGM) PSBNAME(MYPSB) IMSID(XXX) MVSID(IMSX) TYPE(BMP)
```
Description of data columns on the resource savings reports

The IMS resource savings reports contain different information depending on which value you specified for the REPORT_TYPE keyword in the JOBSTEPS command:

- Specifying REPORT_TYPE(SAVINGS) generates the Resource Savings Report.
- Specifying REPORT_TYPE(HISTORY) generates the JOBSTEP History Report.

Data columns on the Resource Savings Report

The Resource Savings Report shows the accumulated savings for the current profile of each job step. The top portion of this report contains the overall results for all optimized job steps, as follows:

- TOTAL OPTIMIZED JOB STEPS
  The total number of steps for which Application Accelerator has found optimization data in the repository

- TOTAL CPU SERVICE UNITS
  The total number of CPU service units that optimization saved for all optimized job steps

- TOTAL ELAPSED TIME IN SECONDS
  The total number of elapsed seconds that optimization saved for all optimized job steps

- TOTAL EXCP’S
  The total number of execute channel program (EXCP) operations that optimization saved for all optimized job steps

The remainder of the report shows results for individual job steps:

- Columns that identify the job step
  JOBNAME, STEPNAME, PROCSTEP, PGMNAME, PSBNAME, IMSID, MVSID, and TYPE

- SAVED CPU SERVICE UNITS
  The calculated total number of CPU service units that optimization prevented this job step from consuming

- SAVED ELAPSED SECONDS
  The total number of elapsed seconds that optimization prevented this job step from consuming
- **SAVED EXCP’S**
  The total number of execute channel program (EXCP) operations that optimization prevented this job step from consuming.

- **TOTAL RUNS**
  The number of executions of this job step for which Application Accelerator found optimization data in the repository.

- **STATUS**
  Whether the job step was optimized or monitored.

**Data columns on the JOBSTEP History Report**

The JOBSTEP History Report shows the savings for each execution of each job step for which data is available. This report includes a row for each job step execution that is available in the repositories and that matches the specified criteria. If no criteria is specified, the report contains all job steps. The rows are formatted with comma-delimited columns to facilitate export to a spreadsheet. The report contains:

- **DETAIL_TYPE**
  The type of repository record. D indicates detail, M monthly, and Y yearly records.

- **JOBNAME, STEPNAME, PROCSTEP, PGMNAME, PSBNAME, IMSID, MVSID, and TYPE**
  Columns that identify the job step.

- **RUN_DATE**
  The date on which the job step executed.

- **RUN_TIME**
  The time at which the job step executed. Zeros are used for monthly and yearly records.

- **NO_EXECUTIONS**
  The number of executions (always displays 1 for detail records, and the number of executions accumulated for monthly and yearly records).

- **BASE_DLI**
  The number of DL/I calls that the job step made before Application Accelerator optimized it.
- **TOTAL_DLI**
  The total number of DL/I calls that were made during the execution of this job step. For monthly and yearly records, the number is an average for the number of executions.

- **BASE_CPU**
  The CPU service units that the job step used before Application Accelerator optimized it. For monthly and yearly records, the number is an average for the number of executions.

- **TOTAL_CPU**
  The total number of CPU service units that were consumed during execution of this job step. For monthly and yearly records, the number is an average for the number of executions.

- **SAVED_CPU**
  The calculated total number of CPU service units that optimization prevented this job step from consuming. For monthly and yearly records, the number is an average for the number of executions.

- **BASE_WALL**
  Elapsed seconds that the job step used before Application Accelerator optimized it. For monthly and yearly records, the number is an average for the number of executions.

- **TOTAL_WALL**
  The total number of elapsed seconds that were consumed during execution of this job step. For monthly and yearly records, the number is an average for the number of executions.

- **SAVED_WALL**
  The total number of elapsed seconds that optimization prevented this job step from consuming. For monthly and yearly records, the number is an average for the number of executions.

- **BASE_EXCP**
  The number of EXCPs that the job step used before Application Accelerator optimized it. For monthly and yearly records, the number is an average for the number of executions.

- **TOTAL_EXCP**
  The total number of EXCPs that were consumed during execution of this job step. For monthly and yearly records, the number is an average for the number of executions.
- **SAVED_EXCP**
  The total number of EXCPs that optimization prevented this job step from consuming. For monthly and yearly records, the number is an average for the number of executions.

- **TOTAL_OPTIMIZED_DLI**
  The total number of DL/I calls that optimization prevented this job step from making. For monthly and yearly records, the number is an average for the number of executions.

If you specified METHODS=Yes, the report also includes the following columns:

- **UAPI**
  Y means optimization was done using Custom I/O.

- **VSAMP**
  Y means optimization was done using DFSVSAMP tuning.

- **OSAMS**
  Y means optimization was done using OSAM Sequential Buffering.

### $IBO$RPT report

The $IBO$RPT report provides essential information about job step characteristics and Application Accelerator activities.

By default, Application Accelerator dynamically allocates the $IBO$RPT data set to SYSOUT=* for every job step that is eligible for Application Accelerator participation. The job step JCL can allocate the data set explicitly to a different location. You can prevent dynamic allocation of $IBO$RPT by specifying the SKIPRPT=Y option through the IBOINIT utility.

*Figure 6 on page 93* shows an example of the $IBO$RPT report.

#### Figure 6: $IBO$RPT report

```
*******************************************************************************
**                             AAI JOBSTEP REPORT                             **
** JOB=IBOV0190   STEP=DBREAD1    PROCSTEP=DLIUPROG    PROGRAM=DBREAD       **
** PSB=PIACGO     IMSID=MXOG       MVSID   =IMSA        RGNTYPE=DLI           **
*******************************************************************************
*******************************************************************************
TOTAL RUNS OF ALL TYPES      3
RUNS BEFORE OPTIMIZATION     1
THE CURRENT EXECUTION DID OPTIMIZE THIS STEP
```
The **$IBO$RPT** report contains the following sections:

- The job step section identifies the job step and shows overall resource usage and savings.
- The DBPCB section shows calls, optimizations, and reasons for optimization choices by DBPCB.
- The summary section shows calls and optimizations for all DBPCBs.
Troubleshooting problems in production mode

If a problem occurs while you are using Application Accelerator in production mode, check the following topics for troubleshooting tips.

Preventing Application Accelerator participation

When Application Accelerator is implemented in production mode, the CPC subsystem screens IMS batch applications and passes control to Application Accelerator to participate when an application is eligible.

You can use the following methods if you need to prevent Application Accelerator participation:

- Specify an exclude policy.
  This technique prevents Application Accelerator from participating in an entire job. For more information, see “Managing policies through the console” on page 67 or “Managing policies through the CPCBATCH utility” on page 72.

- Specify the IBOIGNOR DD DUMMY statement in the JCL for executing a job step.
  This technique is useful when you want to prevent Application Accelerator from optimizing a specific job step, while allowing optimizing in other steps in that job.
  The IBOIGNOR statement prevents Application Accelerator from:
    - Optimizing the current execution of this job step
    - Recommending optimization for the next execution
  If you remove this DD statement from a subsequent job step execution, Application Accelerator does not optimize that execution but considers recommending optimization for subsequent executions. Monitoring continues to occur.

Obtaining diagnostic trace information

If a problem is occurring, BMC Customer Support might ask you to add the following DD statements to your application job steps to produce diagnostic trace information.

Specify the standard SYSOUT parameter in your environment.
Using production mode without screening application jobs

In some cases, you might want to use Application Accelerator in production mode without screening service requests (SVCs) to determine whether they are eligible for processing.

(In this approach, the job step will technically use a hybrid of trial mode and production mode.) For example, you might want to suspend screening in the following cases:

- A problem is occurring with SVC screening.
- IMS batch applications are not executing with PGM=DFSRRC00 on the EXEC card in their job steps JCL.
- You do not want to use SVC screening for other reasons pertaining to systems operations.

You can disable SVC screening and still use Application Accelerator in production mode. You do so by using the IBO#cpcID DD statement to identify a CPC subsystem to use for an application job step. The last four characters of the DD name provide the subsystem identifier (cpcID) of the CPC subsystem that you want to use.

When you use the IBO#cpcID DD statement, the job step technically is using Application Accelerator in a hybrid of trial and production modes.

**Note**

When the IBO#cpcID statement is included, Application Accelerator automatically considers the job step as Included in Phase 1 of the process that determines the action to perform for the job step. Phase 2 proceeds, using the policies from the cpcID repositories.

**To use production mode without SVC screening**

1. In the JCL (or JCL procedure) for executing the application job step, add the IBO#cpcID DUMMY statement.
2 If needed, remove the $IBO$VSM DD statement.

---

*Note*

The $IBO$VSM DD statement identifies a trial-mode repository for Application Accelerator. If the JCL includes the $IBO$VSM DD statement, the IBO#cpcID statement is ignored.

---

3 In the cpcID CPC subsystem, disable SVC screening in the default values.

4 Ensure that the required load modules for Application Accelerator are included in the STEPLIB, JOBLIB, or LNKLST concatenation.

### Using a different set of CPC repositories

In some cases, you might want to use repositories other than those that are associated with the CPC subsystem that is currently screening executing jobs for Application Accelerator. (In this approach, the job step will technically use a hybrid of trial mode and production mode.)

For example, you might want to use different repositories in the following cases:

- You are testing a new level of Application Accelerator components.
- You want to use a set of policies that are different from the production set.

You can designate a different CPC subsystem to manage Application Accelerator data. You do so by using the IBO#cpcID DD statement to identify that CPC subsystem. The last four characters of the DD name provide the subsystem identifier (cpcID) of the CPC subsystem that you want to use.

---

*Note*

When the IBO#cpcID statement is included, Application Accelerator automatically considers the job step as *Included* in Phase 1 of the process that determines the action to perform for the job step. Phase 2 proceeds, using the policies from the cpcID repositories.

---

When you use the IBO#cpcID DD statement, the job step technically is using Application Accelerator in a hybrid of trial and production modes.

### To use a different set of CPC repositories

1 In the JCL (or JCL procedure) for executing the application job step, add the IBO#cpcID DUMMY statement.

2 If needed, remove the $IBO$VSM DD statement.
**Note**

The $IBO$VSM DD statement identifies a trial-mode repository for Application Accelerator. If the JCL includes the $IBO$VSM DD statement, the IBO#cpcID statement is ignored.

---

## Correcting problems with dimmed console items

Console items (such as menu selections, buttons, and commands) are dimmed when they are present in the interface but not accessible, given current conditions.

An item can appear dimmed (gray or lighter than other items) if any of the following conditions exist:

- You do not have the Security Access Facility (SAF) authority to use the item. For example, to manage policies and defaults for Application Accelerator, your user ID must have SAF update authority for resource FACILITY BBM.SDBA.IMSAA.*.AAS.
  
  To access the item, ask your security administrator to provide the correct authority. For more information about defining SAF resources for Application Accelerator, see the *Database Products for IMS Customization Guide*.

- The item does not apply to the selected object, or the focus is on a window in which the item does not apply. To access the item, try selecting a different object or window.
Index

* wildcard 81
/* and */ comment indicators 77
% wildcard 81
$IBOSRPT data set and report 93
$IBOSVSM data set 30
$IBODIAG DD statement 31, 95

A
access authority 38
Add/Edit dialog 68
ADV repositories 36
ADV server 36
  modifying startup procedure 40
AND Boolean operator 22
APF authorization 32
Application Accelerator
  initialization 14
  monitoring applications 14
  optimization 14
  purpose and uses 13
  workflow summary 14
APPLICATION RESTART Control 23
applications
  JCL changes for trial mode 30
  supported types 21
authentication, z/OS 45

B
BCSS 36
BMC Application Accelerator for IMS Savings
  History window 85
BMC Consolidated Subsystem 36
BMC Primary Subsystem 36
BMCP 36
BMP regions 21
BMPs
  access authority 38
Boolean operators 22
browser, installing console 42

C
CAPTURES keyword 28, 61
client, console 42
connection
  defining 52
  Enterprise List 51
  personal list 51
considerations 20
console
  dimmed items 98
  exiting 45
  Help system 24
  installation 42
  installing 42
  introduction 18
  launching 45
  production-mode overview 36
  requirements 20
  uninstalling 44
conventions, documentation 10
CPC repositories 97
  introduction 18
CPC subsystem
  APF authorization 32
  configuring 40
  introduction 17
  modifying startup procedure 40
  production-mode overview 36, 40
  starting 42
CPCBATCH utility
control statement syntax 77
evaluation results 81
POLICIES command 74
CPCSYSIN keywords
  policies 78
  reports 86
JCL requirements 76
overview 72
sequential data set 79
CPCID keyword 61
CPCLOG data set 81
CPCLOG DD statement 76
CPCSYSIN control statement syntax 77
CPCSYSIN DD statement 76
CPCSYSIN keywords
  policies 78
  reports 86
CPU ID passwords 30

D

database types, supported 22
DATACLAS keyword 29, 62
DBB regions 21
DBBPC limitations 23
DBUSS 32
DBUZ subsystem 32
DDNAME keyword 78, 87
default values
  console techniques for setting 58
  overview 56
Detail Job Status window 85
DETAIL_TYPE keyword 87
Display Messages window 49
DLI regions 21
DLITRACE DD statement 31, 95
documentation information 9
DSNAME keyword 78, 87

E

Enterprise List, managing 51
error messages 24
EVALUATE function of CPCBATCH utility 75
EXPORT function of CPCBATCH utility 74

F

F command code in SSA 23

H

Help
  online 9
hosts
  connecting 52
  defining 52

I

IBMPRF keyword 62
IBO#cpcid DD statement 96
IBOBATCH sample member 30
IBOCIMUPD sample member 60
IBOCINIT sample member 60
IBOIGNOR DD statement
  production-mode usage 95
  trial-mode usage 31
IBOINIT utility
  production mode 60
  production-mode keywords 60
  trial mode 27, 33
  trial-mode keywords 27
IBOOFF keyword 63
IBOTRACE DD statement 31, 95
IBOTRIG keyword 29, 62
IBOVINIT sample member 27
IVBOMUPD sample member 33
IMPORT function of CPCBATCH utility 74
IMS
  supported region types 21
  supported versions 20
IMS keyword 78
IMSID keyword 88
installation 19
  console 42
interfaces, user
  console 48

J

JCL changes for trial mode 30
K

keywords
  CAPTURES 28, 61
  CPCID 61
  DATACLAS 29, 62
  DDNAME 78, 87
  DETAIL_TYPE 87
  DSNNAME 78, 87
  IBMPRF 62
  IBOFF 63
  IBOTRIG 29, 62
  IMS 78
  IMSID 88
  JOBNAME 78, 88
  METHODS 87
  MGMTCLAS 29, 62
  MVSID 79, 89
  NOTITLE 87
  PROCNAME 88
  PROCOPTA 64
  PROGNAME 78, 88
  PSB 78, 88
  REPORT_TYPE 87
  RUNTYPE 78
  SKIPRPT 65
  STEPNAMES 88
  STRGCLAS 29, 62
  SVC BMP 66
  SVC DBB 67
  SVC DLI 66
  SVC EXP 64
  SVC JOB 65
  SVC OFF 63
  SVC PSB 65
  TEMPUNIT 29, 62

L

load libraries 30

M

Maintain Parameters window 59
maintenance 19
MainView Batch Optimizer 23
Manage Defaults window 59
Manage Policies window 67
Messages window 49
messages, Application Accelerator 24
METHODS keyword 87
MGMTCLAS keyword 29, 62
MVSID keyword 79, 89

N

Navigation window 48
NOTITLE keyword 87

O

online Help 9
operating system requirements 20
optimization
  DBB 13
  DLI 13
optimizing applications 14

P

policies
  console management of 67
  evaluating 75
  exclusion 71
  export 71
  exporting 74
  filtering 70
  importing 74
  overview 57
  rearranging 69
  wildcard values in 81
  adding 68, 72
CPCBATCH utility management of 72
  editing 68, 72
  refreshing 75
POLICIES command 78
POLICIES command of CPCBATCH utility 74
POS=M 23
PROCNAME keyword 88
PROCOPTA keyword 64
PROCSEQ parameter 22
production mode
  evaluation 82
  introduction 17
  workflow 55
  component overview 36
  policies 57
  transition from trial mode 35
PROGNAME keyword 78, 88
PSB keyword 78, 88
publications, related 9

R
Recommended Jobs dialog 68
REFRESH function of CPCBATCH utility 75
related publications 9
REPORT_TYPE keyword 87
reports
  $IBOSRPT 93
requirements 20
Resource Savings window 83
RUNTYPE keyword 78

S
SAF resource for Application Accelerator 98
SAF resources 38
samples
  IBOBATCH member 30
  IBOCINIT member 60
  IBOCMUPD member 60
  IBOVINIT member 27
  IBOVMUPD member 33
screening applications 96
sequential data set for policies 79
Setup Wizard 58
site values
  overview 56
SKIPRPT keyword 65
SSA command codes 23
startup procedures, modifying 41
STEPLIB changes for trial mode 30
STEPNAME keyword 88
STRGCLAS keyword 29, 62
SVC screening 96

T
TEMPUNIT keyword 29, 62
trace data sets 31
trial mode
  introduction 15
  workflow summary 25
  JCL changes 30
  transition to production mode 35
trial-mode repository
  introduction 16
  REPRO to a sequential data set 34
  $IBOSVSM DD statement 30
  allocating 27
  allocating one per z/OS system 27
  changing values 33
  initializing 27
troubleshooting
  production mode 95
  trial mode 33

U
UIM Server 36
  overview 42
uninstallation, console 44
update, console 44
User Interface Middleware (UIM) server 42
user options, console 50

V
values, default 56
values, site 56

W
windows
Display Messages 49
Messages 49
Navigation 48
work area 50

Z

z/OS authentication 45