CA MIM™ Resource Sharing

Installation Guide

Release 12.0

Second Edition

CA Technologies Product References

This document references the following CA Technologies products:

- CA Chorus™ Software Manager (CA CSM)
- CA Chorus™
- CA Common Services for z/OS (CCS)
- CA Easytrieve® Report Generator (CA Easytrieve)
- CA MIA Tape Sharing (CA MIA)
- CA MIC Message Sharing (CA MIC)
- CA MII Data Sharing (CA MII)
- CA MIM™ Resource Sharing for z/OS (CA MIM)
- CA Remote Console™ (CA Remote Console)
- CA SYSVIEW® Performance Management (CA SYSVIEW)
- CA OPS/MVS® Event Management and Automation (CA OPS/MVS)

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Documentation Changes

The following documentation updates have been made since the first edition of this document.

- Overview > How the Installation Process Works (see page 17): updated the process for different versions of CA CSM
- Preparing for installation > Storage Requirements: removed subtopics describing target and distribution libraries
- Installing Your Product Using CA CSM (see page 27): updated the chapter

The following documentation updates have been made since the last release of this documentation:

Topics removed

- Installing Your Product from Tape—Removed from the guide.
- CA MIM Simple Configuration Example—Removed from the guide.
- CA MIM Largest Configuration Options Example—Removed from the guide.
- CA MIM Shared Runtime Library Example—Removed from the guide.

Topics Updated

- Updated the Software Requirements (see page 19) section.
- Updated the CA Common Services Requirements (see page 20) section.
- Updated the CAIRIM (see page 21) section.
- Updated the LXCFMAIN (see page 21) section.
- Easytrieve Services (see page 22)—Added to the guide.
- Updated the CA L-Serv (see page 22) section.
- Updated the Security Requirements (see page 23) section—Removed the bullet on tape install.
- Updated the Requisite IBM APARs (see page 25) section.
- Installing Your Product from Pax-Enhanced ESD—Replaced with the chapter Installing Your Product Using Pax ESD or DVD (see page 29).
- Installing Your Product from DVD—Replaced with the chapter Installing Your Product Using Pax ESD or DVD (see page 29).
- Updated the CA MIM Configuration Options using CA CSM (see page 53) section.
- Updated the Deploy the Startup JCL PROCs (see page 55) section.
- Updated the Running CA MIM in Mixed-Level Mode (see page 75) section.

- Updated the COMPATLEVEL Considerations (see page 76) section.
- CCS for z/OS Component Requirements—Added the CA zIIP Enablement Service (see page 81).
- Updated the CA Easytrieve Interface (see page 82) section.
- Updated the appendix CCS for z/OS Component Requirements (see page 81).

Contents

Chapter 1: Overview

This section contains the following topics:

CA MIM Overview

CA MIM for z/OS is the industry standard for sharing DASD, tape, and console resources safely and efficiently in z/OS and z/VM multiple-image environments. The product streamlines and automates many of the procedures involved in sharing resources and enables multiple-system sites to share data center resources across as many as 32 systems.

CA MIM performs the following functions:

- Provides a single point for interfacing with all systems and peripheral devices
- Enables multisystem sites to share DASD and tape devices across as many as 32 systems
- Enables multi-image sites to share unlimited consoles across a MICplex of as many as 128 systems. (This feature is available when using the ICMF communication method.)
- Provides resource and data integrity
- Improves user service by providing device sharing among systems, which increases resource availability, job throughput, and hardware usage efficiency rates

CA MIM is designed for sites with two or more physical CPUs running z/OS or z/VM that want to share and consolidate resources safely and efficiently. CA MIM also provides these benefits to sites with a single CPU, logically partitioned systems, or as a guest under a z/VM host.

CA MIM exploits your basic or parallel sysplex environment for maximum performance, allowing you to share resources among any mix of sysplex and non-sysplex systems, including resources shared across multiple sysplexes.

Components and Facilities

CA MIM comprises the following components that manage specific z/OS resources:

- CA MIA
- CA MIC
- CA MII
- A CA MIM Driver

CA MIA

CA MIA automates tape device sharing, allowing tape devices to be shared among z/OS sites, z/VM sites, and mixed z/OS and z/VM sites with CMS users and z/OS guests.

CA MIA performs the following tasks:

- Serializes access to tape devices across the complex, while guaranteeing data integrity.
- Automates the allocation recovery process by canceling or holding a job until a device is available, or by letting the operator respond with a device that is offline.
- Modifies the selection process for tape devices according to physical location, type of job, and other user criteria. This feature also lets operations reserve devices for important jobs.
- Integrates with Sun Storage Tek, Memorex Telex, and IBM Automated Tape Libraries, enabling z/OS or z/VM systems to share robotic devices transparently.
- Supports continuous operations by providing the capability to dynamically change the list of devices managed by CA MIA, without stopping and restarting the product's started task.
- Lets CMS users and z/OS jobs share tape devices in a mixed z/OS and z/VM site. To prevent data corruption, CA MIA coordinates global tape device allocation by keeping track of both z/OS and z/VM allocations.
- Provides SMF records with statistics about global device usage and mount times to assist in managing tape devices.

- Provides an Application Programming Interface (API), callable from an external program, to capture global information about the status of tape devices. This information includes the following:
  - Allocation status
  - The name of the volume mounted
  - The preferencing status
  - The online or offline status
  - The name of the allocating job

CA MIA Facilities

CA MIA consists of the following facilities:

Global Tape Allocation Facility (GTAF)

Prevents jobs on different systems from simultaneously allocating the same tape, and prevents jobs from allocating devices already in use on another system.

Tape Preferencing and Control Facility (TPCF)

Lets you influence device selection during the z/OS allocation process. TPCF also responds automatically to the messages z/OS issues when a job cannot allocate a suitable online device.

CA MIC

CA MIC provides cross-system command routing from any z/OS or z/VM command source. Use CA MIC to import messages from external systems and route them to local destinations.

CA MIC performs the following tasks:

- Consolidates all messages from multiple images into a single stream and delivers these messages to various user-defined destinations. The destinations include MCS and EMCS consoles, SYSLOG, TSO users, and vendor software products like CA OPS/MVS and CA Remote Console.
- Selects which messages are delivered to each destination based on various selection criteria. The selection criteria include message type, message ID, job name, route code, and the monitor type.
- Provides user-defined command routing to any combination of systems in the complex from various command sources. The sources include MCS and EMCS consoles, TSO users, and vendor software products such as CA Remote Console, CA OPS/MVS, and CA SYSVIEW.

- Allows your operators to identify the source system of a cross-system message. CA MIC can modify the color and highlighting of messages that are based on the originating system or sysplex. Operators can edit the job ID field of messages to contain the two-character alias of the originating system.
- Consolidates geographically dispersed systems for the enterprise console control using CA Common Services for z/OS.

CA MIC Facilities

CA MIC consists of the following facilities:

Global Command and Message Facility (GCMF)

Lets you route messages and commands to any or all systems in a complex

Intersystem Communication Facility (ICMF)

Lets you route cross-system commands and messages using an interface with the CA L-Serv component of CA Common Services for z/OS

CA MII

CA MII protects z/OS data integrity automatically, speeds resolution of resource conflicts in shared DASD environments, and adds additional integrity at a local system level.

CA MII performs the following tasks:

- Communicates selected ENQ requests to all images in a complex, so applications on different images cannot update a data set simultaneously.
- Converts specified RESERVE requests to ENQ requests to increase resource availability.
- Notifies operators, TSO users, or both when two or more jobs require the same data set, so that the conflict can be resolved more quickly.
- Enhances initiator utilization by requeuing jobs requiring data sets that are in use and then automatically rescheduling these jobs when their data sets become available.
- Provides additional local system data set integrity protection. This includes the capability to check the attributes of a data set specified by the user against the actual attributes of the data set. Sites can specify that only authorized applications or utility programs may read or update certain data sets or classes of data sets. The product serializes the data set for all update operations by automatically issuing an ENQ request when the data set is opened for output.

CA MII Facilities

CA MII consists of the following facilities:

Global Data Integrity Facility (GDIF)

Prevents simultaneous updates that occur when requests for resources are not communicated to all systems

Enqueue Conflict Management Facility (ECMF)

Helps Time Sharing Option (TSO) users and system operators identify and resolve conflicting requests for resources

Enhanced Data Set Integrity Facility (EDIF)

Prevents the most common sources of data set damage, such as damage to attributes or due to DISP=SHR updates

CA MIM Driver

The CA MIM Driver manages global activity of the product components by routing transactions across mainframe images through a control file residing on a shared DASD volume or a Virtual Control File (VCF) residing in CA MIM private storage on a selected master system.

The VCF architecture uses channel-to-channel (CTC) devices or the z/OS XCF component to pass a transaction data buffer between systems.

In a parallel sysplex environment with a CA MIM complex that is equal to or is a subset of the parallel sysplex complex, the control file can be placed in the coupling facility. This provides a significant performance enhancement through the reduction of I/O transfer times as compared to CTC, cached DASD, and non-cached DASD I/O operations.

CA MIM provides the capability to define backup communication methods. This furnishes data centers with the redundancy needed to guarantee uninterrupted resource integrity as the operating environment changes or during hardware outages. While the product is running, migrations can be initiated between DASD control files, between VCF, or between DASD and VCFs.

The CA MIM transaction processing architecture is based on a star configuration. With this architecture, CA MIM on each system needs only a single access to the control file to determine the global status of all managed resources. Frequency of access to the control file is based on the amount of resource activity on a particular system.

The CA MIM Driver can be defined as CA MIM address space control code, which supervises the activities of the CA MIM address space, regardless of which CA MIM facilities are activated. CA MIM Driver code is responsible for the following CA MIM address space activities:

Control:

- Command processing
- Message processing
- Synchronization
- Initialization
- Error recovery
- Diagnostics
- Performance
- Termination

Housekeeping:

- Storage management
- Subtask management
- Lock management

Global Communications:

- DASD I/O operations
- CTC I/O operations
- Coupling facility I/O operations

Audience

Readers of this book should have knowledge in the following areas:

- JCL
- TSO/ISPF
- z/OS environment and installing software in this environment
- Your organization's IT environment, enterprise structure, and region structure

You may need to work with the following personnel:

- Systems programmer for z/OS and VTAM definitions
- Storage administrator, for DASD allocations

How the Installation Process Works

CA Technologies has standardized product installations across all mainframe products. Installation uses the following process:

- Acquisition—Transports the software to your z/OS system.
- Installation using SMP/E—Creates an SMP/E environment and runs the RECEIVE, APPLY, and ACCEPT steps. The software is untailored.
- (For CA CSM Release 5.1 and earlier only) Deployment—Copies the target libraries to another system or LPAR.

  Note: This step is optional for CA CSM Version 6.0. For more information, see the scenario *Configuring Products Using CA CSM* that is available in the CA CSM Version 6.0 bookshelf at http://ca.com/support.
- Configuration—Creates customized load modules, bringing the software to an executable state.
- (For staging system configurations in CA CSM Version 6.0 only) Deployment—Makes configured run-time libraries available to a remote location where that software can be activated, bringing it to an executable state.

CA Chorus™ Software Manager (CA CSM) - formerly known as CA Mainframe Software Manager™ (CA MSM) - is an intuitive web-based tool that can automate and simplify many CA Technologies product installation activities on z/OS systems. This application also makes obtaining and applying corrective and recommended maintenance easier. A web-based interface enables you to install and maintain your products faster and with less chance of error. As a best practice, we recommend that you install mainframe products and maintenance using CA CSM. Using CA CSM, someone with limited knowledge of JCL and SMP/E can install a product.

Note: If you do not have CA CSM, you can download it from the Download Center at http://ca.com/support. Follow the installation instructions in the CA Chorus Software Manager documentation bookshelf on the CA Chorus Software Manager product page.

You can also complete the standardized installation process manually using pax files that are downloaded from http://ca.com/support or a product DVD.

To install your product, do the following tasks:

1. Prepare for the installation by confirming that your site meets all installation requirements.
2. Verify that you acquired the product using one of the following methods:
   - Download the software from http://ca.com/support using CA CSM.
   - Download the software from http://ca.com/support using Pax-Enhanced Electronic Software Delivery (Pax ESD).
   - Order a product DVD. To do so, contact your account manager or a CA Technologies Support representative.
3. Perform an SMP/E installation using one of the following methods:
   - If you used CA CSM to acquire the product, start the installation process from the SMP/E Environments tab in CA CSM.
   - If you used Pax ESD to acquire the product, you can install the product in the following ways:
     - Install the product manually.
     - Complete the SMP/E installation using the Add Product option in CA CSM.
   - If you used a DVD, install the product manually.

   Note: If a CA Recommended Service (CA RS) package is published for your product, install it before proceeding.
4. (For CA CSM Release 5.1 and earlier only) Deploy the target libraries.

   Note: This step is optional for CA CSM Version 6.0. For more information, see the scenario *Configuring Products Using CA CSM* that is available in the CA CSM Version 6.0 bookshelf at http://ca.com/support.
5. Configure your product using CA CSM or manually.
6. (For staging system configurations in CA CSM Version 6.0 only) Deploy configured run-time libraries, and activate your product.

   Note: Configuration is considered part of starting your product.

Chapter 2: Preparing for Installation

This section describes what you need to know and do before you install the product.

This section contains the following topics:

Hardware Requirements

CA MIM 12.0 can be installed on hardware that supports the software described in the section Software Requirements.

CTC Communication

If you will be using a CTC communication method, then each system in the complex must be connected to a port on an IBM 3088-type device, to an extended mode ESCON CTC device, or to a FICON CTC device.

Software Requirements

The following software is required for CA MIM:

- z/OS 1.12, 1.13, or 2.1

  The operating system can be configured as stand-alone systems, as logically partitioned (LPARed) systems, or as guests under a z/VM host.

  For the most current list of supported operating systems, visit CA Support Online.

CA Common Services Requirements

The following CA Common Services are used with CA MIM:

- CAICCI
- CAIRIM
- CA LMP
- LXCFMAIN
- CAISDI Service
- Easytrieve Service
- CA L-Serv
- CA Health Checker Common Service

Note: If other CA products are installed at your site, some of these services may already be installed. For detailed CCS FMID requirements, see CCS for z/OS Component Requirements (see page 81).

CAICCI

Provides CA enterprise applications with a common communications software layer that insulates the applications from dealing with protocol specifics, error recovery, and system connection establishment.

CAIRIM

Prepares your operating system environment for all CA applications and starts them. The common driver for a collection of dynamic initialization routines eliminates the need for user SVCs, SMF exits, subsystems, and other installation requirements commonly encountered when installing systems applications.

Integral parts of CAIRIM are CAISSF, CA LMP, and CA zIIP Enablement Services.

Note: These modules must reside in LINKLIST or as part of the CA MIM STEPLIB concatenation.

CAISSF

Provides an external security mechanism for controlling and monitoring access to all system and application resource processes. CAISSF is integrated into many CA enterprise applications and is also used by other CCS for z/OS services. CAISSF provides security services for user logon, resource access control, process use control, and recording and monitoring of violation activity.

CA LMP

Provides a standardized and automated approach to the tracking of licensed software and is provided as an integral part of CAIRIM. After CAIRIM is installed, you have access to Technical Support for all CA LMP-supported products.

CA zIIP Enablement Services

Provides a common service for CA products to allow their code to run on zIIP processors, if available.

LXCFMAIN

CA MIM uses the CCS module LXCFMAIN when using the XCF communication option. The LXCFMAIN module acts as an interface to the IBM XCF services. The LXCFMAIN module is delivered on the CCS product tape.

Note: This module must reside in LINKLIST or as part of the CA MIM STEPLIB concatenation.

CAISDI

Provides a set of services that opens CA Service Desk requests from the z/OS environment. The requests can be opened directly by CA products or they can be opened on their behalf, depending on the requirements of each specific product using the interface.

CAISDI comprises the following components:

CAISDI/soap

The z/OS Simple Object Access Protocol (SOAP) client communicates with CA Service Desk. It manages the communication using TCP/IP to CA Service Desk. It also provides the basic mechanisms that allow CA products to open CA Service Desk tickets. This component is required for all CA Service Desk integration.

CAISDI/els

The Event Library Support (ELS) component provides a mechanism for CA products to open CA Service Desk tickets for events they detect directly. Supported events are defined in an event library that contains the customizable text and symbolic parameters to be used in the CA Service Desk request. The BrightStor z/OS product family requires installation of this component.

CAISDI/med

The Mainframe Event Director (MED) component monitors the z/OS environment and opens CA Service Desk tickets on behalf of CA products and/or other system components. This provides a way to open CA Service Desk tickets when the CA product is unable to do so, such as the case of an abend. It also opens solicited tickets for the CA products that use this interface component.

Easytrieve Service

Provides a powerful productivity language for both business and information processing professionals containing easy-to-use information retrieval, sophisticated report writing, and comprehensive application development capabilities. The Easytrieve Service is a limited version of the full-featured CA Easytrieve product, which lets you modify the contents of an Easytrieve application that is provided with another CA product. If you have CA Easytrieve product already installed at your site, you do not need to install the Easytrieve Service from CCS.

CA L-Serv

CA L-Serv provides standard services that various products of CA Technologies use. These products include:

- CA Endevor Software Change Manager
- CA Bundl
- CA TPX Session Management for z/OS
- CA Balancing
- CA MIC Message Sharing

These services include centralized logging and messaging facilities, VSAM file management, cross-system communications, and SQL table management.

Note: If you use the ICMF communication method for CA MIC, then CA L-Serv must be available on each system you want to communicate with. CA L-Serv communicates between systems using VTAM cross-domains.

CA Health Checker

Provides a simple and consistent method for CA products to create health checks to run under the IBM Health Checker for z/OS. The IBM Health Checker for z/OS helps you identify potential problems in your z/OS environment by checking system or product parameters and system status against recommended settings. CA has joined other vendors in creating checks for CA z/OS products. CA MIM health checks are automatically activated on the target system when the product is started on a system where the following components are installed and configured:

- CA Health Checker Common Service
- IBM Health Checker for z/OS

For more information on installing the CA Health Checker Common Service, see the *CA Common Service Installation Guide.*

For more information about the IBM Health Checker for z/OS, see the *IBM Health Checker for z/OS User Guide*.

Security Requirements

To complete the tasks in this guide, you need the following security privileges:

- Update authority for the DASD-resident product distribution data sets with high-level qualifier of your choice, if installing using Pax-Enhanced ESD
- Update authority for the SMP/E-installed data sets with high-level qualifier of your choice
- Update authority for the deployed runtime copies of the SMP/E-installed Target libraries with high-level qualifier of your choice
- Update authority to a JCL PROCLIB, such as SYS2.PROCLIB

Storage Requirements

This section describes storage needed to install and run CA MIM.

Virtual Storage

We strongly recommend that you use a region size of 4096 KB or larger.

Common Storage Area

When all CA MIM facilities are active, a minimum of 47 KB of system queue area (SQA) and 75 KB of extended system queue area (ESQA) are permanently allocated. Also, if you have a large peak of cross-system message traffic, CA MIC can temporarily use a variable amount of extended common storage area (ECSA).

Note: The total amount of storage used by all CA MIM components is difficult to estimate. It depends on system activity, the number of systems in a complex, and the total number of managed resources as defined using CA MIM parameters.

Note: For more information, see the appropriate *Programming Guide*.

DASD Data Sets

You need to allocate:

- A data set to contain CA MIM parameters
- An authorized load library to contain CA MIM load modules
- Non-shared checkpoint data set files when using CTCONLY or XCF communication methods or if you will be using the ECMF job requeue feature

You may optionally allocate:

- Data sets that provide trace data collection and collect diagnostic information about CA MIM
- A separate data set to contain CA MIM message facility parameters

DASD Control File Communication

If you will be using a DASD communication method, you need to allocate at least one DASD control file.

USS Space Requirements

Ensure that you have sufficient free space in the USS file system that you are using for Pax ESD to hold the directory that the pax command and its contents create. You need approximately 3.5 times the pax file size in free space.

If you do not have sufficient free space, you receive error message EDC5133I.

Other Requirements

This section describes miscellaneous other requirements.

Requisite CA MIM APARs

Important! Compatibility PTFs for previous CA MIM releases must be installed before installing CA MIM Release 12.0. For more information, see Migration Information (see page 75).

Requisite IBM APARs

The following IBM APARS are required to provide important basic support and general stability for CA MIM Version 12.0.00 and should be considered requisites to starting CA MIM 12.0.00 on z/OS 1.12, 1.13, and 2.1 systems.

For z/OS 1.12:

- None.

For z/OS 1.13:

- None.

For z/OS 2.1:

- None.

For a list of IBM APARS that impact CA MIM by component, see informational solution RI08812 located on the CA Support.

XCF Communication Method

If you will be using the XCF communication method, each system defined to CA MIM must belong to the same sysplex.

Note: For XCF requirements see the chapter Advanced Topics in the *CA MIM Programming Guide*.

Concurrent Releases

You can install this release of your product and continue to use an older release in another SMP/E environment. If you plan to continue to run a previous release, consider the following points:

- When you install the product into an existing SMP/E environment, this installation deletes previous releases in that environment.
- If you acquired your product with Pax ESD, select different target and distribution zones for your new release from where your current release is installed. The new zones use different libraries than your current release.

  Note: CA CSM installs a product into a new SMP/E environment by default. You can select an existing SMP/E environment from your working set. For more information, see the online help that is included in CA CSM.
- Define DDDEF entries in your new zones to point SMP/E to the proper libraries for installation. Ensure that they point to the new release libraries.

Chapter 3: Installing Your Products Using CA CSM

As a system programmer, your responsibilities include acquiring, installing, maintaining, deploying, and configuring CA Technologies mainframe products on your system.

CA CSM is an application that simplifies and unifies the management of your CA Technologies mainframe products on z/OS systems. As products adopt the CA CSM services, you can install your products in a common way according to industry best practices.

If you do not have CA CSM installed, download it from the Download Center at http://ca.com/support. This web page also contains links to the complete documentation for CA CSM.

You can use the following scenarios to guide you through the product installation process (see page 17) using CA CSM:

- Acquiring Products Using CA CSM
- Installing Products Using CA CSM
- Maintaining Products Using CA CSM
- Configuring Product Using CA CSM

These scenarios are available in the CA CSM Version 6.0 bookshelf at http://ca.com/support. For additional information about how to use CA CSM, use the online help.

Chapter 4: Installing Your Product Using Pax ESD or DVD

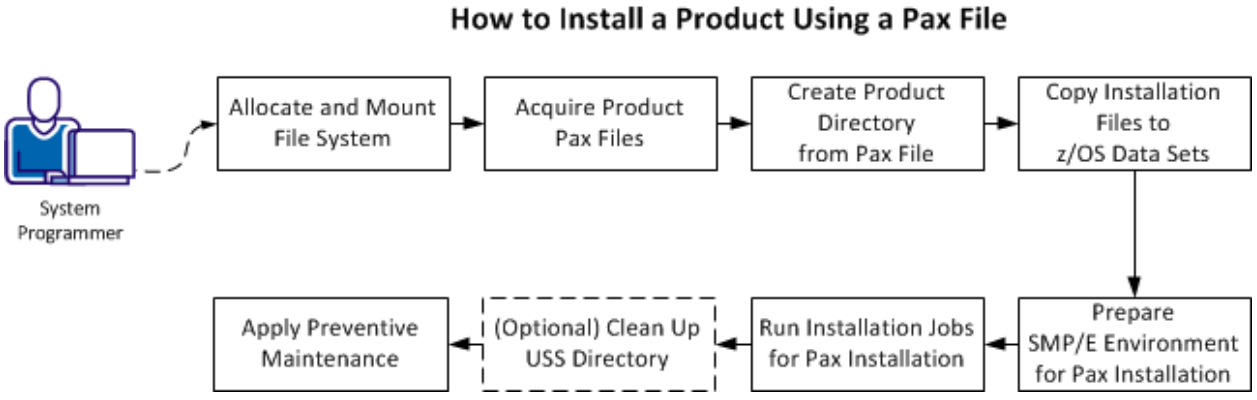
This section contains the following topics:

How to Install Your Product Using a Pax File

As a system programmer, your responsibilities include installing products on your mainframe system. With this option, you acquire a product pax file from http://ca.com/support or from a product DVD.

The DVD contains a folder that includes the pax file for the product. Product updates may have occurred after you acquired the product DVD. The files on the online site always have the most current product updates. To determine if you have the latest updates, go to http://ca.com/support and click Download Center.

You perform the following tasks to install a product with a pax file:



1. Allocate and mount the file system (see page 31).
2. Acquire the product pax files (see page 33).

3. Create a product directory from the pax file (see page 38).
4. Copy the installation files to z/OS data sets (see page 39).
5. Prepare the SMP/E environment for a pax installation (see page 41).
6. Run the installation jobs for a pax installation (see page 43).
7. (Optional) Clean up the USS directory (see page 44).
8. Apply preventive maintenance (see page 45).

USS Environment Setup

You need a UNIX System Services (USS) directory and a file system with adequate space to perform the following tasks:

- Receive product pax files from http://ca.com/support.
- Perform utility functions to unpack the pax file into MVS data sets that you can use to complete the product installation.

We recommend that you allocate and mount a file system that is dedicated to Pax ESD. The amount of space that you need for the file system depends on the following variables:

- The size of the pax files that you intend to download.
- Whether you plan to keep the pax files after unpacking them. We do not recommend this practice.

We recommend that you use one directory for downloading and unpacking pax files. Reusing the same directory minimizes USS setup. You need to complete the USS setup only one time. You reuse the same directory for subsequent downloads. Alternatively, you can create a directory for each pax download.

Important! Downloading pax files for the SMP/E installation as part of the Pax ESD process requires write authority to the UNIX System Services (USS) directories that are used for the Pax ESD process. In the file system that contains the Pax ESD directories, you also need free space approximately 3.5 times the pax file size to download the pax file and unpack its contents. For example, to download and unpack a 14 MB pax file, you need approximately 49 MB of free space in the file system hosting your Pax ESD directory.

Allocate and Mount a File System

The product installation process requires a USS directory to receive the pax file and to perform the unpack steps. We recommend that you allocate and mount a file system that is dedicated to the product acquisition and create the directory in this file system.

You can use the zSeries File System (zFS) or hierarchical file system (HFS) for product downloads.

This procedure describes how to perform the following tasks:

- Allocate a zFS or an HFS.
- Create a mount point in an existing maintenance USS directory of your choice.
- Mount the file system on the newly created mount point.

  Note: You must have either SUPERUSER authority, or the required SAF profile setting to allow you to issue the USS mount command for the file system.
- Optionally, permit write access to anyone in the same group as the person who created the directory.

  Important! USS commands are case-sensitive.

Follow these steps:

1. Allocate the file system by customizing one of the following samples to your site requirements:

   - On a zFS, use the following sample:

```
//DEFINE   EXEC PGM=IDCAMS
//SYSPRINT DD   SYSOUT=*
//SYSUDUMP DD   SYSOUT=*
//AMSDUMP  DD   SYSOUT=*
//SYSIN    DD *
  DEFINE CLUSTER ( +
  NAME(your_zFS_data_set_name) +
  STORAGECLASS(class) +
  LINEAR +
  CYL(primary secondary) +
  SHAREOPTIONS(3,3) +
  )
/*
//FORMAT   EXEC PGM=IOEAGFMT,REGION=0M,
// PARM=('-aggregate your_zFS_data_set_name -compat')
//SYSPRINT DD   SYSOUT=*
//SYSUDUMP DD   SYSOUT=*
//STDOUT   DD   SYSOUT=*
//STDERR   DD   SYSOUT=*
//CEEDUMP  DD   SYSOUT=*
//*
```

- On an HFS, use the following sample:

```
//ALCHFS EXEC PGM=IEFBR14
//CAPAX  DD   DSN=yourHFS_data_set_name,
//         DISP=(NEW,CATLG,DELETE),UNIT=3390,
//         DSNTYPE=HFS,SPACE=(CYL,(primary,secondary,1))
```

The file system is allocated.

Note: Ensure that the zFS or HFS data set name that you use conforms to your data set naming conventions for USS file systems. If the allocation of the file system data set fails, it is because of environmental settings not allowing for the allocation. On an HFS, try using the ISPF 3.2 Data Set Utility to allocate your HFS data set.

2. Create a mount point for the file system. This example shows how to create a /CA/CAPAX directory in an existing directory, /u/maint. From the TSO OMVS shell, enter the following commands:

```
cd /u/maint/
mkdir CA
cd CA
mkdir CAPAX
```

Note: This document refers to this structure as *yourUSSpaxdirectory*.

The mount point is created.

3. Mount the file system by customizing one of the following samples to your site requirements:

- On a zFS, use the following sample:

```
MOUNT FILESYSTEM('your_zFS_data_set_name')
   MOUNTPOINT('yourUSSpaxdirectory')
   TYPE(ZFS)  MODE(RDWR)
   PARM(AGGRGROW)
```

- On an HFS, use the following sample:

```
MOUNT FILESYSTEM('your_HFS_data_set_name')
   MOUNTPOINT('yourUSSpaxdirectory')
   TYPE(HFS)  MODE(RDWR)
```

The file system is mounted.

4. (Optional) Set security permissions for the directory. You can use the chmod command to let other users access the Pax ESD directory and its files. For example, to allow write access to the Pax ESD directory for other users in your USS group, from the TSO OMVS shell, enter the following command:

```
chmod -R 775 /yourUSSpaxdirectory/
```

Write access is granted.

Note: For more information about the chmod command, see the IBM *z/OS UNIX System Services User Guide* (SA22-7802).

Acquire the Product Pax Files

To begin the CA Technologies product installation procedure, copy the product pax file into the USS directory that you set up.

Important! Downloading pax files for the SMP/E installation as part of the Pax ESD process requires write authority to the UNIX System Services (USS) directories that are used for the Pax ESD process. Also, you must have available USS file space before you start the procedures in this guide.

Use one of the following methods:

- Download the product pax file from http://ca.com/support to your PC (see page 34), and then upload it to your USS file system.

  If you download a zip file, you must unzip it before uploading to your USS file system.
- Download the pax files from http://ca.com/support directly to your USS file system (see page 34).
- Download the pax file from the product DVD to your PC, and then upload the pax files to your USS file system. (see page 37)

This section includes the following information:

- A sample batch job to download a product pax file from the CA Support Online FTP server directly to a USS directory on your z/OS system
- Sample commands to upload a pax file from your PC to a USS directory on your z/OS system

Important! The FTP procedures vary due to local firewall and other security settings. Consult your local network administrators to determine the appropriate FTP procedure to use at your site.

Ensure that sufficient free space is available in the USS file system that you are using to hold the product pax file. If you do not have sufficient free space, error messages similar to the following appear:

```
EZA1490I Error writing to data set
EZA2606W File I/O error 133
```

When the download finishes, the pax file size in your USS directory matches the value in the Size column for the corresponding pax file on the CA Technologies Products Download window.

Download Files to a PC Using Pax ESD

You can download product installation files from http://ca.com/support to your PC.

Follow these steps:

1. Log in to http://ca.com/support, and click Download Center.

   The Download Center web page appears.

2. Under Download Center, select Products from the first drop-down list, and specify the product, release, and gen level (if applicable), and click Go.

   The CA Product Download window appears.

3. Download an entire CA Technologies product software package or individual pax files to your PC. If you download a zip file, you must unzip it before continuing.

   Note: For traditional installation downloads, see the *Traditional ESD User Guide*. For information about download methods, see the Download Methods and Locations article. Go to http://ca.com/support, log in, and click Download Center. Links to the guide and the article appear under the Download Help heading.

Download Using Batch JCL

You download a pax file from http://ca.com/support by running batch JCL on the mainframe. Use the sample JCL attached to the PDF file as CAtoMainframe.txt (see page 36) to perform the download.

Important! The PDF version of this guide includes sample JCL jobs that you can copy directly to the mainframe. To access these jobs, click the paper clip icon at the left of the PDF reader. A window displaying attachments opens. Double-click a file to view a sample JCL. We recommend that you use the latest version of Adobe Reader for viewing PDF files.

Note: We recommend that you follow the preferred download method as described on http://ca.com/support. This JCL procedure is our preferred download method for users who do not use CA CSM. We also include the procedure to download to the mainframe through a PC in the next section.

Follow these steps:

1. Replace *ACCOUNTNO* with a valid JOB statement.
2. Replace *yourTCPIP.PROFILE.dataset* with the name of the TCP/IP profile data set for your system. Consult your local network administrators, if necessary.

   The job points to your profile.

3. Replace *YourEmailAddress* with your email address.

   The job points to your email address.

4. Replace *yourUSSpaxdirectory* with the name of the USS directory that you use for Pax ESD downloads.

   The job points to your USS directory.

5. Locate the product component to download on the CA Support Product Download window.

   You have identified the product component to download.

6. Click Download for the applicable file.

   Note: For multiple downloads, add files to a cart.

   The Download Method window opens.

7. Click FTP Request.

   The Review Download Requests window displays any files that you have requested to download.

   Note: We send you an email when the file is ready to download or a link appears in this window when the file is available.

8. Select one of the following methods:

   Preferred FTP

   Uses CA Technologies worldwide content delivery network (CDN). If you cannot download using this method, review the security restrictions for servers that company employees can download from that are outside your corporate network.

   Host Name: ftp://ftpdownloads.ca.com

   Alternate FTP

   Uses the original download servers that are based on Long Island, New York.

   Host Name: ftp://scftpd.ca.com for product files and download cart files and ftp://ftp.ca.com for individual solution files.

   Both methods display the host, user name, password, and FTP location, which you then can copy into the sample JCL.

   Note: The following links provide details regarding FTP: the FTP Help document link in the Review Download Requests window and the Learn More link available in the Download Methods window.

9. Submit the job.

   Important! If your FTP commands are incorrect, it is possible for this job to fail and still return a zero condition code. Read the messages in the job DDNAME SYSPRINT to verify the FTP succeeded.

   After you run the JCL job, the pax file resides in the mainframe USS directory that you supplied.

Example: CAtoMainframe.txt, JCL

The following text appears in the attached CAtoMainframe.txt JCL file:

```
//GETPAX   JOB (ACCOUNTNO),'FTP GET PAX ESD PACKAGE',
//         MSGCLASS=X,CLASS=A,NOTIFY=&SYSUID
//*********************************************************************
//* This sample job can be used to download a pax file directly from  *
//* CA Support Online to a USS directory on your z/OS system.         *
//*                                                                   *
//* When editing the JCL ensure that you do not have sequence numbers *
//* turned on.                                                        *
//*                                                                   *
//* This job must be customized as follows:                           *
//* 1. Supply a valid JOB statement.                                  *
//* 2. The SYSTCPD and SYSFTPD JCL DD statements in this JCL may be   *
//*    optional at your site. Remove the statements that are not      *
//*    required. For the required statements, update the data set     *
//*    names with the correct site-specific data set names.           *
//* 3. Replace "Host" based on the type of download method.           *
//* 4. Replace "YourEmailAddress" with your email address.           *
//* 5. Replace "yourUSSpaxdirectory" with the name of the USS         *
//*    directory used on your system for Pax ESD downloads.           *
//* 6. Replace "FTP Location" with the complete path                  *
//*    and name of the pax file obtained from the FTP location        *
//*    of the product download page.                                  *
//*********************************************************************
//GETPAX   EXEC PGM=FTP,PARM='(EXIT TIMEOUT 120',REGION=0M
//SYSTCPD  DD   DSN=yourTCPIP.PROFILE.dataset,DISP=SHR
//SYSFTPD  DD   DSN=yourFTP.DATA.dataset,DISP=SHR
//SYSPRINT DD   SYSOUT=*
//OUTPUT   DD   SYSOUT=*
//INPUT    DD   *
Host
anonymous YourEmailAddress
lcd yourUSSpaxdirectory
binary
get FTP_location
quit
/*
```

Download Files to Mainframe through a PC

You download the product installation files to your PC and transfer them to your USS system.

Follow these steps:

1. Download the product file to your PC using one of the following methods:

   - Pax ESD (see page 34). If you downloaded a zip file, first unzip the file to use the product pax files.
   - DVD. Copy the entire product software package (or individual pax files) to your PC.

   The pax file resides on your PC.

   Note: Do *not* change the format of the pax.Z.

2. Open a Windows command prompt.

   The command prompt appears.

3. Customize and enter the following FTP commands:

   FTP *mainframe*
   userid
   password
   bin
   lcd C:*PC\folder\for\thePAXfile*
   cd */yourUSSpaxdirectory/*
   put *paxfile*.pax.Z
   quit
   exit

   mainframe

   Specifies the z/OS system IP address or DNS name.

   userid

   Specifies your z/OS user ID.

   password

   Specifies your z/OS password.

   C:*PC\folder\for\thePAXfile*

   Specifies the location of the pax file on your PC.

   Note: If you specify a location that has blanks or special characters in the path name, enclose that value in double quotation marks.

yourUSSpaxdirectory

Specifies the name of the USS directory that you use for Pax ESD downloads.

***paxfile*.pax.Z**

Specifies the name of the pax file to upload.

The pax file is transferred to the mainframe.

Create a Product Directory from the Pax File

The pax command performs the following actions:

- Extracts the files and directories that are packaged within the pax file.
- Creates a USS directory in the same directory structure where the pax file resides.
- Automatically generates a product and level-specific directory name.

Set the current working directory to the directory containing the pax file, and create a directory in your USS directory by entering the following command:

pax -rvf *pax-filename*

Use the sample JCL that is attached to the PDF file as Unpackage.txt (see page 39) to extract the product pax file into a product installation directory.

Important! The PDF version of this guide includes sample JCL jobs that you can copy directly to the mainframe. To access these jobs, click the paper clip icon at the left of the PDF reader. A window displaying attachments opens. Double-click a file to view a sample JCL. We recommend that you use the latest version of Adobe Reader for viewing PDF files.

Follow these steps:

1. Replace *ACCOUNTNO* with a valid JOB statement.
2. Replace *yourUSSpaxdirectory* with the name of the USS directory that you use for product downloads.

   The job points to your specific directory.
3. Replace *paxfile.pax.Z* with the name of the pax file.

   The job points to your specific pax file.
4. Submit the job.

   The job creates the product directory.

Note: If the PARM= statement exceeds 71 characters, uncomment and use the second form of UNPAXDIR instead. This sample job uses an X in column 72 to continue the PARM= parameters to a second line.

Example: JCL File, Unpackage.txt, to Customize

The following text appears in the attached Unpackage.txt JCL file:

```
//ESDUNPAX JOB (ACCOUNTNO),'UNPAX PAX ESD PACKAGE',
// MSGCLASS=X,CLASS=A,NOTIFY=&SYSUID
//*********************************************************************
//* This sample job can be used to invoke the pax command to create   *
//* the product-specific installation directory.                      *
//*                                                                   *
//* This job must be customized as follows:                           *
//* 1. Supply a valid JOB statement.                                  *
//* 2. Replace "yourUSSpaxdirectory" with the name of the USS         *
//*    directory used on your system for Pax ESD downloads.           *
//* 3. Replace "paxfile.pax.Z" with the name of the pax file.         *
//* NOTE: If you continue the PARM= statement on a second line,       *
//*       start entering characters in column 16 and make sure        *
//*       the 'X' continuation character is in column 72.             *
//*********************************************************************
//UNPAXDIR EXEC PGM=BPXBATCH,
// PARM='sh cd /yourUSSpaxdirectory/; pax -rvf paxfile.pax.Z'
//*UNPAXDIR EXEC PGM=BPXBATCH,
//* PARM='sh cd /yourUSSpaxdirectory/; pax                          X
//*           -rvf paxfile.pax.Z'
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
```

Copy Installation Files to z/OS Data Sets

Use this procedure to invoke the SMP/E GIMUNZIP utility to create MVS data sets from the files in the product-specific directory.

The file UNZIPJCL in the product directory contains a sample job to GIMUNZIP the installation package. You edit and submit the UNZIPJCL job to create z/OS data sets.

Follow these steps:

1. Locate and read the product readme file or installation notes, if applicable, which resides in the product-specific directory that the pax command created. This file contains the product-specific details that you require to complete the installation procedure.

   You have identified the product-specific installation details.

2. Use ISPF EDIT or TSO ISHELL to edit the UNZIPJCL sample job. You can perform this step in one of the following ways:

   - Use ISPF EDIT. Specify the full path name of the UNZIPJCL file.
   - Use TSO ISHELL. Navigate to the UNZIPJCL file and use the E line command to edit the file.

   The job is edited.

3. Change the SMPDIR DD PATH to the product-specific directory created by the pax command.

   Your view is of the product-specific directory.

4. If ICSF is not active, perform the following steps:

   a. Change the SMPJHOME DD PATH to your Java runtime directory. This directory varies from system to system.

   b. Perform one of the following steps:

      - Change the SMPCPATH DD PATH to your SMP/E Java application classes directory, typically /usr/lpp/smp/classes/.
      - Change HASH=YES to HASH=NO on the GIMUNZIP parameter.

   One of the following occurs: ICSF is active or you are using Java.

5. Change all occurrences of y*ourHLQ* to the high-level qualifier (HLQ) for z/OS data sets that the installation process uses. We suggest that you use a unique HLQ for each expanded pax file to identify uniquely the package. Do *not* remove CAI after *yourHLQ*. Do *not* use the same value for y*ourHLQ* as you use for the SMP/E RELFILEs.

   All occurrences of y*ourHLQ* are set to your high-level qualifier for z/OS data sets.

6. Submit the UNZIPJCL job.

   The UNZIPJCL job completes with a zero return code. Messages GIM69158I and GIM48101I in the output and IKJ56228I in the JES log are acceptable.

   GIMUNZIP creates z/OS data sets with the high-level qualifier that you specified in the UNZIPJCL job. You use these data sets to perform the product installation. The pax file and product-specific directory are no longer needed.

   Note: For more information, see the IBM *SMP/E for z/OS Reference (SA22-7772)*.

Prepare the SMP/E Environment for a Pax Installation

The following steps describe the process to install products using native SMP/E JCL:

1. Download external HOLDDATA.
2. Allocate product data sets and SMP/E data sets.
3. Create an SMP/E environment.
4. Receive base functions and HOLDDATA.
5. Download and RECEIVE PTFs from http://ca.com/support.
6. Run an SMP/E APPLY CHECK operation.
7. Apply base functions using SELECT GROUPEXTEND.
8. Run an SMP/E ACCEPT CHECK operation.
9. Accept base functions using SELECT GROUPEXTEND.
10. Configure the product according to your site requirements.

Note: Steps 1 through 3 of this process are documented in detail in this section. Steps 4 through 9 are documented in the section describing how to run installation jobs for a Pax installation. If applicable to your product, Step 10 is documented in the section describing starting your product.

The members that are used in this procedure prepare the data sets, initialize the zones, and create the DDDEFs for your product.

Establishing a hierarchical file system (HFS) may be required as part of the product installation or required as a feature of the product.

For information about the members, see the comments in the JCL.

Follow these steps:

1. Customize the macro BTDSEDIT with your site-specific information and then copy the macro to your SYSPROC location. Replace the rightmost parameters for each ISREDIT CHANGE command. Each time you edit an installation member, type BTDSEDIT on the command line, and press Enter to replace the defaults with your specifications.

   The macro is ready to customize the *yourHLQ*.SAMPJCL members.

   Note: Set the DASD HLQ to the same value specified for *yourHLQ* within the JCL that is used to unzip the pax file.

   Note: The following steps include instructions to execute the BTDSEDIT macro each time you open a new SAMPJCL member. To edit all SAMPJCL members simultaneously, read and follow the instructions in the BTDAREAD member, and submit the BTDEDALL member.

2. Open the SAMPJCL member BTD1HOLD in an edit session and execute the BTDSEDIT macro from the command line.

   BTD1HOLD is customized.

3. Submit BTD1HOLD.

   This job downloads the error and FIXCAT HOLDDATA from http://ca.com/support.

4. Open the SAMPJCL member BTD2ALL in an edit session and execute the BTDSEDIT macro from the command line.

   BTD2ALL is customized.

5. Submit BTD2ALL.

   This job produces the following results:

   - The target and distribution data sets for your product are created.
   - Unique SMPLTS, SMPMTS, SMPSCDS, and SMPSTS data sets for this target zone are created.

6. If your product requires a USS file system or if you want to install a feature of the product that requires a USS file system, allocate and mount the file system:

   Note: You can customize the supplied HFS JCL to zFS, if your site requires it.

   a. Open the SAMPJCL member *ccc2*ALLU in an edit session and execute the BTDSEDIT macro from the command line.

      Note: All instances of *ccc* in this section indicate a three-character component code based on the FMID.

      *ccc2*ALLU is customized.

   b. Submit *ccc2*ALLU.

      This job allocates your HFS or zFS data sets.

   c. Open the SAMPJCL member *ccc3*MKD in an edit session and execute the BTDSEDIT macro from the command line.

      *ccc3*MKD is customized.

   d. Submit *ccc3*MKD.

      This job creates all directories and mounts the file system.

7. Open the SAMPJCL member BTD3CSI in an edit session and execute the BTDSEDIT macro from the command line.

   BTD3CSI is customized.

8. Submit BTD3CSI.

   This job produces the following results:

   - The CSI data set is defined.
   - The SMPPTS and SMPLOG data sets are allocated.

- The global, target, and distribution zones are initialized.
- The DDDEF entries for your product are created.
- The DDDEFs for the required SMP/E data sets are created.

9. If your product requires HFS or if you want to install a feature of the product that requires HFS, add the DDDEFS that are required for the file system to your SMP/E environment:

   a. Open the SAMPJCL member *ccc*3CSIU in an edit session and execute the BTDSEDIT macro from the command line.

      *ccc*3CSIU is customized.

   b. Submit *ccc*3CSIU.

      This job customizes the CSI by adding the DDDEFs associated with the directory.

Run the Installation Jobs for a Pax Installation

Submit and run these SAMPJCL members in sequence. Do not proceed with any job until the previous job has completed successfully.

Note: The following steps include instructions to execute the BTDSEDIT macro each time you open a new SAMPJCL member. To edit all SAMPJCL members simultaneously, read and follow the instructions in the BTDAREAD member, and submit the BTDEDALL member.

Follow these steps:

1. Open the SAMPJCL member BTD4RECD in an edit session, and execute the BTDSEDIT macro from the command line.

   BTD4RECD is customized.

2. Submit BTD4RECD to receive SMP/E base functions and error HOLDDATA.

   Your product is received and now resides in the global zone.

3. If an FMID was placed in error, download and receive PTFs (see page 45) from http://ca.com/support.

4. Open the SAMPJCL member BTD5APP in an edit session, and execute the BTDSEDIT macro from the command line.

   BTD5APP is customized.

5. Submit BTD5APP to apply SMP/E base functions with the CHECK option. If you find unresolved hold errors, we recommend that you note these errors and verify that resolving PTFs are applied before implementing products in production. Update the JCL to BYPASS the unresolved hold error IDs. After successful completion, rerun APPLY with the CHECK option removed.

   Your product is applied and now resides in the target libraries.

6. Open the SAMPJCL member BTD6ACC in an edit session, and execute the BTDSEDIT macro from the command line.

   BTD6ACC is customized.

7. Submit BTD6ACC to accept SMP/E base functions with the CHECK option. After successful completion, rerun APPLY with the CHECK option removed.

   Your product is accepted and now resides in the distribution libraries.

Clean Up the USS Directory

Important! This procedure is optional. Do not use this procedure until you complete the entire installation process.

To free file system disk space for subsequent downloads after downloading and processing the pax files for your CA Technologies product, we recommend removing the files from your USS directory and deleting unnecessary MVS data sets. You can delete the following items:

- Pax file
- Product-specific directory that the pax command created and all of the files in it
- SMP/E RELFILEs, SMPMCS, and HOLDDATA MVS data sets

  These data sets have the HLQ that you assigned in the UNZIPJCL job.

Note: Retain non-SMP/E installation data sets such as *yourHLQ*.INSTALL.NOTES for future reference.

Follow these steps:

1. Navigate to your Pax ESD USS directory.

   Your view is of the applicable USS directory.

2. Delete the pax file by entering the following command:

   rm *paxfile*

   paxfile

   Specifies the name of the CA Technologies pax file that you downloaded.

   The pax file is deleted.

3. Delete the product-specific directory by entering the following command:

   rm -r *product-specific_directory*

   product-specific_directory

   Specifies the product-specific directory that the pax command created.

   The product-specific directory is deleted.

Note: You can also use TSO ISHELL to navigate to the pax file and product-specific directory, and delete them using the D line command.

Apply Preventive Maintenance

Important! We strongly recommend that you use CA CSM to maintain your CA Technologies z/OS-based products. The procedure that is discussed in this section is fully automated when you use CA CSM.

CA Support Online at http://ca.com/support has maintenance and HOLDDATA published since the installation data was created. After the maintenance process completes, the product is ready to deploy.

Use this procedure during product installation and for ongoing preventive maintenance in non-installation use cases according to your maintenance strategy.

Note: To review the CA Technologies mainframe maintenance philosophy, see your *Best Practices Guide* or visit the CA Next-Generation Mainframe Management page.

This procedure directs you to use the CAUNZIP utility. The CAUNZIP utility processes ZIP packages directly on z/OS without the need for an intermediate platform, such as a Microsoft Windows workstation. If you are not familiar with this utility, see the *CA Common Services for z/OS Administration Guide*. This guide includes an overview and sample batch jobs. To use this utility, you must be running CA Common Services for z/OS Version 14.0 with PTF RO54887 or CA Common Services for z/OS Release 14.1 with PTF RO54635 and RO58216. These PTFs are included in CA Common Services for z/OS Release 14.1 at the S1401 Service Update level.

Follow these steps:

1. Check the Download Center at http://ca.com/support for PTFs that have been published since this release was created. If the base release was created recently, no PTFs will have been published yet. If PTFs exist, add published solutions for your product to your Download Cart, and click Checkout.
2. Specify that you want a complete package.

   When processing completes, a link appears on the Review Download Requests page. You also receive an email notification.

3. Click the Alternate FTP link for your order to obtain FTP login information and the ZIP file location. Download the ZIP file into a USS directory on your z/OS system.
4. Run the CAUNZIP utility.

   CAUNZIP unzips the package of published solutions and creates a SMPNTS file structure that the SMP/E RECEIVE FROMNTS command can process. For sample JCL to run the utility that is located in *yourHLQ*.CAW0JCL(CAUNZIP), see the *CA Common Services for z/OS CAUNZIP Administration Guide*. After execution completes, the ZIPRPT data set contains the summary report. The summary report does the following:

   - Summarizes the content of the product order ZIP file.
   - Details the content of each data set and the z/OS UNIX files produced.
   - Provides a sample job to receive the PTFs in your order.
5. Review the sample job that is provided in the CAUNZIP output ZIPRPT file. Cut and paste the JCL into a data set, specify your SMP/E CSI on the SMPCSI DD statement and submit the job to receive the PTFs in your order.
6. Verify that you have the values from the base installation in the BTDSEDIT macro that was customized in the installation steps.
7. Open the SAMPJCL member BTD1HOLD in an edit session and execute the BTDSEDIT macro from the command line.

   Note: Update BTD1HOLD SAMPJCL to download the HOLDDATA file.

   BTD1HOLD is customized.
8. Submit BTD1HOLD.

   The job downloads the external HOLDDATA file.
9. Open the SAMPJCL member BTD7RECH in an edit session and execute the BTDSEDIT macro from the command line.

   BTD7RECH is customized.
10. Submit BTD7RECH.

    The job receives the external HOLDDATA file.

11. (CA Recommended Service (CA RS)) installation only) Do the following:
    a. Determine which ASSIGN statements to download.
        - The yearly CA RS ASSIGN statements are stored in the following file:

          ftp.ca.com/pub/ASSIGN/YEARLY/CAR*yyyy*.TXT
        - The quarterly CA RS ASSIGN statements are stored in the following file:

          ftp.ca.com/pub/ASSIGN/CAR*yymm*.TXT

    b. Open the SAMPJCL member BTD7CARS in an edit session, update BTD7CARS SAMPJCL to download ASSIGN statements from http://ca.com/support, and execute the BTDSEDIT macro from the command line.

    BTD7CARS is customized.
12. (CA RS installation only) Submit BTD7CARS.

    The job downloads the CA RS ASSIGN statements.
13. (CA RS installation only) Open the SAMPJCL member BTD7RECP in an edit session, manually add the data set that contains the ASSIGN statements to the SMPPTFIN DD, and execute the BTDSEDIT macro from the command line.

    BTD7RECP is customized.
14. (CA RS installation only) Submit BTD7RECP.

    The job receives the external HOLDDATA file and CA RS ASSIGN statements.
15. Open the SAMPJCL member BTD8APYP in an edit session and execute the BTDSEDIT macro from the command line.

    BTD8APYP is customized.
16. Submit BTD8APYP.

    The PTFs are applied.
17. (Optional) Open the SAMPJCL member BTD9ACCP in an edit session and execute the BTDSEDIT macro from the command line.

    BTD9ACCP is customized.
18. (Optional) Submit BTD9ACCP.

    The PTFs are accepted.

    Note: You do not have to submit the job at this time. You can accept the PTFs according to your site policy.

HOLDDATA

When you apply maintenance, you typically encounter SMP/E HOLDDATA. We use HOLDDATA to notify your SMP/E system of SYSMODs that have errors or special conditions. We support system and external HOLDDATA.

System HOLDDATA

System HOLDDATA indicates data that is an in-stream part of the SYSMOD, informing you of special conditions. The following reasons are used with SYSTEM HOLDDATA for your product:

ACTION

Indicates that you must perform special processing before or after you apply this SYSMOD.

AO

Affects automated operations. It changes either the message identifier or the displacement of a field inside the message.

DDDEF

Indicates that data sets and DDDEFs are being added or modified.

DELETE

Deletes the SYSMOD load module. You cannot reverse this type of SYSMOD with the SMP/E RESTORE command.

DEP

Indicates a dependency for this SYSMOD that you must externally verify.

DOC

Indicates a documentation change with this SYSMOD.

EXRF

Indicates that the SYSMOD must be installed in both the Active and Alternate Extended Recovery Facility Systems.

IPL

Indicates that an IPL is required for this SYSMOD to take effect. This is used only when there is no alternative for dynamic activation.

Code a bypass operand on your APPLY command to install SYSMODs that have internal holds. Code the bypass operand only after you have performed the required action, or if you are performing the action after the APPLY, if that is appropriate.

External HOLDDATA

External HOLDDATA is not part of the PTF. The HOLDDATA resides in a separate file and contains both error and FIXCAT HOLDDATA. The error HOLDDATA is commonly used for SYSMODs that have been distributed and later are discovered to cause problems. The FIXCAT HOLDDATA helps identify maintenance that is required to support a particular hardware device, software, or function.

Download the external HOLDDATA from http://ca.com/support to a DASD file, and allocate the file to the SMPHOLD DD statement. To take care of the external HOLDDATA, receive it into your SMP/E environment. SMP/E receives the HOLDDATA from CA-supplied jobs.

You can find JCL to download the external HOLDDATA in your SAMPJCL member. Open BTD1HOLD in an edit session and execute the BTDSEDIT macro on the command line. Then, submit the JCL.

Error HOLDDATA

If a SYSMOD has an unresolved hold error, SMP/E does not install it unless you add a bypass to your APPLY command. You can bypass an error hold in situations that are not applicable to you. Error holds that are not applicable to you can include a problem that happens only with a hardware device that you do not have or in a product feature that you do not use.

When CA Technologies publishes a SYSMOD that resolves the hold, the resolving SYSMOD supersedes the hold error. This action lets you apply the original SYSMOD in conjunction with the fixing SYSMOD.

The only manual task is running a REPORT ERRSYSMODS. This report identifies the following:

- Any held SYSMODs already applied to your system
- Any resolving SYSMODs that are in RECEIVE status

SMP/E identifies the SYSMOD to apply to correct the situation.

FIXCAT HOLDDATA

CA Technologies provides FIXCAT HOLDDATA to help identify maintenance that is required to support a particular hardware device, software, or function. Fix categories are supplied as SMP/E FIXCAT HOLDDATA statements. Each FIXCAT HOLDDATA statement associates an APAR and its related fixing PTF to one or more fix categories.

Chapter 5: Starting Your Product

This section contains the following topics:

Deploy Your Product

How to Complete Deployment With CA CSM

The topics in this section describe the manual tasks that you perform when deploying your product using CA CSM.

You can use CA CSM to deploy a runtime copy of all of the CA MIM SMP/E-installed Target libraries to one or all of the systems at your site.

You can choose to deploy CA MIM checkpoint files as custom data sets using CA CSM.

For more information, see the How to Deploy a Product.

How to Deploy Without CA CSM

The topics in this section describe the manual tasks that you perform if you are not deploying your product using CA CSM.

Deploy Runtime Libraries

You will need to deploy a runtime copy of the following CA MIM SMP/E-installed Target libraries:

- CAI.CBTDJCL
- CAI.CBTDPROC

CA MIM running on each system in your MIMplex must have access to a deployed runtime copy of each of the following CA MIM SMP/E-installed Target libraries:

- CAI.CBTDPARM
- CAI.CBTDMSEN
- CAI.CBTDLOAD

You can provide access to these libraries using one of the following methods:

- Allocate a single copy of these libraries on shared DASD that is accessible to all systems in the MIMplex
- Allocate a single copy of these libraries on one system, and deploy (or duplicate) them on each system in the MIMplex.
- Allocate a unique copy of these libraries on each system in the MIMplex

Choose the method that is appropriate for your environment.

You can customize sample JCL in CAI.CBTDJCL(DEPLOY) member to manually deploy a runtime copy of some or all of the CA MIM SMP/E-installed Target libraries to one or all of the systems in the MIMplex.

Configure Your Product

The topics in this section describe the manual task you perform whether you are configuring using CA CSM or manually.

How to Complete Configuration With CA CSM

The topics in this section describe the manual tasks that you perform when configuring your product using CA CSM.

You can use CA CSM to configure different combinations of CA MIM with CA MII and CA MIA. You can configure CA MIA alone, CA MII alone or both CA MIA and CA MII at the same time.

You can configure and implement a single MIAplex supporting up to 32 MIA address spaces using either DASDONLY or XCF communication, a single MIIplex supporting up to 32 MII address spaces using either DASDONLY or XCF communication, or any combination in between.

For this release of CA MIM, CA MIA and CA MII will be configured to run in separate address spaces. After you have selected what you want to configure, you will be prompted by the CA CSM Configuration Wizard to confirm and/or modify configurable CA MIA and/or CA MII target settings.

Data sets are created by CA CSM after successfully completing each step; that is the SMP/E installation, deployment, and configuration procedures. For more information on data sets created by CA CSM see Data Sets Created by CA CSM (see page 85).

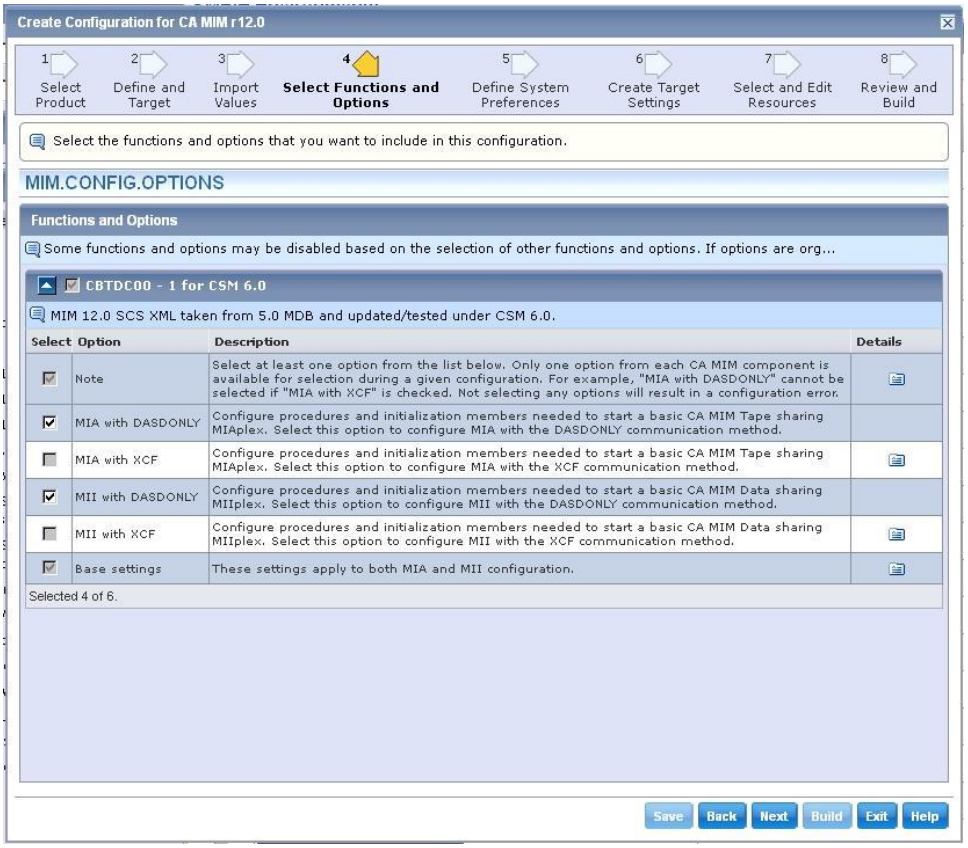
Note: CA Technologies strongly advises that you perform the initial installation and configuration of CA MIM and its components in a test environment as a precaution. This testing will let you detect any possible conflicts with other vendor products.

CA MIM Configuration Options Using CA CSM

Using CA CSM, you may configure the following types of CA MIM systems. During a given configuration run, you can configure one or two of the following:

- CA MIA with DASDONLY communications
- CA MII with DASDONLY communications
- CA MIA with XCF communications
- CA MII with XCF communications

The following Function and Option panel is used to select the type of MIM system to be configured.



To configure a CA MIA DASDONLY system select:

- CA MIA with DASDONLY
- Base settings

To configure a CA MIA XCF system select:

- CA MIA with XCF
- Base settings

To configure a CA MII DASDONLY system select:

- CA MII with DASDONLY
- Base settings

To configure a CA MII XCF system select:

- CA MII with XCF
- Base settings

For more information on how to configure using CA CSM see the section How to Configure a Product.

Once you have configured CA MIM using CA CSM configure from deployment, you must complete the following manual procedures.

APF-Authorize the Runtime Load Libraries

Each deployed runtime copy of the *rthlq*.CBTDLOAD load library must be APF-authorized on each system where it is going to be executed. *rthlq* is the run time high level qualifier that you used in the CA CSM configuration.

You authorize each library on each system using the IBM Authorized Program Facility.

For example, this MVS operator command temporarily APF-authorizes the library *rthlq*.CBTDLOAD on the system where it is issued:

SETPROG APF,ADD,DSN=*rthlq*.CBTDLOAD,VOLUME=*volser*

If the load library is on an SMS managed volume the following command can be issued:

SETPROG APF,ADD,DSN=*rthlq*.CBTDLOAD,SMS

Deploy the Startup JCL PROCs

Deploy the JCL PROCs configured by CA CSM.

Follow these steps:

1. Copy MIIPROC (if it exists).

   Copy *rthlq*.MIMSCNTL(MIIPROC) to any JCL PROCLIB that is automatically searched as part of z/OS START command processing, such as SYS2.PROCLIB. You can rename the PROC.

   Note: If you use the CA MII Early Start Mechanism, you MUST start CA MII with a PROC named CAMIMGR and must be in a proclib allocated by MSTJCL.

2. Copy MIAPROC (if it exists).

   Copy *rthlq*.MIMSCNTL(MIAPROC) to any JCL PROCLIB that is automatically searched as part of z/OS START command processing, such as SYS2.PROCLIB. You can rename the PROC.

Syntax Check the Parameter Library

Scan the CA MIM parmlib members that CA CSM configured using the CA MIM SyntaxSCAN utility.

Follow these steps:

1. Copy MIISYN (if it exists).

   Copy *rthlq*.MIMSCNTL(MIISYN) to any JCL PROCLIB that is automatically searched as part of z/OS START command processing, such as SYS2.PROCLIB. You can rename the PROC.

2. Start your copy of the JCL PROC (if it created).

3. Copy MIASYN (if it exists).

   Copy *rthlq*.MIMSCNTL(MIASYN) to any JCL PROCLIB that is automatically searched as part of z/OS START command processing, such as SYS2.PROCLIB. You can rename the PROC.

4. Start your copy of the JCL PROC (if created).

5. Review the SYSLOGs and JOBLOGs for error messages. If no errors are found, then your CA MIM parmlib members are ready. If you find errors, correct the errors and rerun the SyntaxSCAN utility.

Complete the Configuration

You *must* follow the procedures under Required Manual Configuration to complete the configuration process.

How to Configure Without CA CSM

The topics in this section describe the manual tasks you perform if you are not configuring your product using CA CSM.

APF-Authorize the Runtime Load Libraries

Each deployed runtime copy of the *rthlq*.CBTDLOAD load library must be APF-authorized on each system where it is going to be executed. *rthlq* is the run time high level qualifier that you used in the CA CSM configuration.

You authorize each library on each system using the IBM Authorized Program Facility.

For example, this MVS operator command temporarily APF-authorizes the library *rthlq*.CBTDLOAD on the system where it is issued:

SETPROG APF,ADD,DSN=*rthlq*.CBTDLOAD,VOLUME=*volser*

If the load library is on an SMS managed volume the following command can be issued:

SETPROG APF,ADD,DSN=*rthlq*.CBTDLOAD,SMS

Customize Startup JCL PROCs

This step customizes JCL PROCs used to start CA MIM components on each system.

Some sites prefer to run all three CA MIM components (CA MIA, CA MIC, and CA MII) in a single CA MIM address space; however, we recommend that you run the three CA MIM components in two or three separate CA MIM address spaces.

Note: For more information about running CA MIM components in separate address spaces, see the *CA MIM Programming Guide*.

Data set CAI.CBTDPROC contains the following sample startup JCL PROCs:

- PROCMIA
- PROCMIC
- PROCMII
- PROCMIM

To customize the JCL PROCs

1. Copy one or more of the sample JCL PROCs in CAI.CBTDPROC to any JCL PROCLIB that is automatically searched as part of z/OS START command processing, such as SYS2.PROCLIB. You may rename the PROCs if you wish.

   Note: If you use the CA MII Early Start Mechanism, you MUST start CA MII with a PROC named SYS1.PROCLIB(CAMIMGR).

2. The sample JCL PROCs use the default data set names CAI.CBTDPARM, CAI.CBTDMSEN, and CAI.CBTDLOAD on the MIMPARMS, MIMMSGS, and STEPLIB DDs, respectively. Customize your copies of the JCL PROCs to use the actual data set names of the deployed runtime copies of those data sets.

Customize Parameter Library Members

The //MIMPARMS DD statement in the CA MIM startup JCL PROC points to the CA MIM parameter library. Several members of the CA MIM parmlib are read during CA MIM address space initialization. The members that will be read depend on your specifications in the CA MIM startup JCL PROC and in the CA MIM parmlib members themselves.

Some parmlib members contain initialization statements and commands that are read regardless of which CA MIM facilities are activated, while other CA MIM parmlib members are read only if specific CA MIM facilities are activated:

MIMINIT

Contains CA MIM initialization statements that are processed, regardless of which CA MIM facilities are activated

MIMCMNDS

Contains operating parameters that are processed, regardless of which CA MIM facilities are activated

MIMSYNCH

Contains any commands that need to be executed at CA MIM synchronization time, regardless of which CA MIM facilities are activated

EDIPARMS

Contains data set protection definitions used by EDIF of the CA MII component

GDIEXMPT

Contains ENQ and RESERVE exemption definitions used by GDIF of the CA MII component

MIMQNAME

Contains ENQ and RESERVE resource definitions used by GDIF of the CA MII component

MIMUNITS

Contains tape device definitions optionally used by GTAF of the CA MIA component

Notes:

- For more information about customizing the CA MIM parmlib members, see the respective component *CA MIM Programming Guide*.
- If you are upgrading from a previous release, then you may need to add or remove statements and commands. For more information, see the *CA MIM Release Notes*.

Check Parameter Library Members for Syntax Errors

Use the CA MIM SyntaxSCAN Utility to scan the MIMPARMS and MIMMSGS files for syntax errors.

Follow these steps:

1. Copy the SyntaxSCAN Utility sample JCL PROC located in CAI.CBTDPROC(PROCSYN) and any JCL PROCLIB that is automatically searched as part of the z/OS START command processing, such as SYS2.PROCLIB. You can optionally rename the PROC. Most sites name their CA MIM SynaxSCAN Utility JCL PROC MIMSCAN.

   The sample JCL PROC uses the default data set names CAI.CBTDPARM, CAI.CBTDLOAD, and CAI.CBTDMSEN on the MIMPARMS, STEPLIB, and MIMMSGS DDs, respectively. Customize your copy of the JCL PROC to use the actual data set names of the deployed runtime copies of those data sets.

2. Start your copy of the JCL PROC.
3. Review the SYSLOG and JOBLOG for error messages. If no errors are found, then your CA MIM parmlib members are ready. If you find errors, correct the errors and rerun the SyntaxSCAN Utility.

Note: For more information about using the CA MIM SyntaxSCAN Utility, see the *CA MIM Programming Guide*.

Allocate Control Files

CA MIM address spaces can communicate with one another through any one of several cross-system communication methods. A shared DASD file or a shared coupling facility structure can be used as a physical control file. Alternatively, CA MIM private storage can be used as a virtual control file (VCF) buffer and be passed between systems using natively allocated CA MIM CTC devices or XCF services.

Most sites use the simplest CA MIM cross-system communication method—a shared DASD control file—when installing CA MIM for the first time. This step describes that approach.

1. To allocate two CA MIM control files, customize and submit the job found in the CAI.CBTDJCL(ALLOCCF) member.
2. To ensure that the new control files are allocated to the CA MIM address spaces during startup, add two CA MIM ALLOCATE commands to the CAI.CBTDPARM(MIMINIT) parmlib member.

Important: CA MIM does not support control files on the high portion of the Extended Address Volume (EAV). If you use EAV, make sure you allocate data sets with EATTR=NO.

Note: For more information, see the comments in the CAI.CBTDJCL(ALLOCCF) member. For more information about other CA MIM control files and other cross-system communication methods, see the *CA MIM Programming Guide*.

Required Manual Configuration

This section contains steps you must perform manually, whether or not you used CA CSM to configure CA MIM.

Allocate Checkpoint Files

CA MIM checkpoint files are required if any one of the following is true:

- You intend to use the CTCONLY cross-system communication method. In this environment, the local checkpoint file on each system is used to store MIMplex system status information.
- You intend to use the XCF cross-system communication method. In this environment, the local checkpoint file on each system is used to store MIMplex system status information.
- You intend to use the Job Requeue feature of the ECMF. In this case, the local checkpoint file is used to track information associated with batch jobs that have been placed on hold (requeued) by ECMF.

Notes:

- If you are not sure whether to allocate checkpoint files at this point, then we suggest that you allocate them. The allocated checkpoint files can be deleted later if you decide not to use any of the CA MIM features that require CA MIM checkpoint files.
- CA MIM checkpoint files are not shared between CA MIM address spaces on different systems. You need at least one checkpoint file for each system in the MIMplex.

- CA MIM checkpoint files do not need to be allocated on shared DASD volumes.
- CA MIM serializes access to its checkpoint files by issuing ENQ SCOPE=SYSTEM requests; no hardware RESERVE requests are issued. CA MIM checkpoint files can be placed on DASD devices that have other data sets on them, provided that other applications do not have Hardware Reserve requests issued for the selected DASD unit.

Important: CA MIM does not support checkpoint files on the high portion of the Extended Address Volume (EAV). If you use EAV, make sure you allocate data sets with EATTR=NO.

To allocate checkpoint files

1. Customize and submit the job in the CAI.CBTDJCL(ALLOCKPT) member. This member also contains other helpful details.
2. Add CA MIM ALLOCATE commands to the CAI.CBTDPARM(MIMINIT) parmlib member.

   This ensures the new checkpoint files are allocated to the CA MIM address spaces during startup.

Notes:

- The usual naming convention for the CA MIM checkpoint files uses the system name as one of the data set name nodes.
- For more information about CA MIM checkpoint files and other cross-system communication methods, see the *CA MIM Programming Guide*.
- You may choose to allocate one or more checkpoint files for only one system in this step, and use CA MIM to deploy a copy to each system in the MIMPlex. For more information, see the *CA MIM Programming Guide*.

Set Up the CA MIM Early Start Mechanism

When running CA MII Global Data Integrity Facility (GDIF), we recommend using the CA MIM Early Start Mechanism.

If CA MII synchronization does not occur early in the IPL process, unconverted reserves, deadly embraces, or integrity exposures can occur. Use the CA MIM Early Start mechanism to prevent these issues.

The Early Start Mechanism uses SYS1.PARMLIB to install and execute the module MIMESNQX early in the IPL process. MIMESNQX internally starts the CAMIMGR address space to start monitoring global resources. All RESERVE and SYSTEMS ENQ requests are suspended until the CAMIMGR address space is fully initialized.

Important: If the CAMIMGR address space fails to initialize for any reason, such as a parameter error in the MIMINIT member, the IPL will not complete, because one or more system address spaces will be suspended by MIMESNQX. Attempts to start CAMIMGR manually after correcting the error on another system will also fail, because MIMESNQX has special handling for the ASID created through ASCRE. The only option will be to restart the IPL process from the beginning.

To set up the Early Start mechanism, edit the following members of the SYS1.PARMLIB data set:

IEALPAxx

To add the MIMESNQX module to LPA, add the following statement to the member:

```
INCLUDE LIBRARY(mimloadlibrary) VOLUME(xxxxxx) MODULES(MIMESNQX)
```

PROGxx

To execute MIMESNQX, add the following statement:

```
EXIT ADD EXITNAME(ISGNQXIT) MODNAME(MIMESNQX) STATE(ACTIVE) FIRST
```

IEASYSxx

When new members are created for IEALPAxx and PROGxx, add the member specifications to the PROG and MLPA parameters.

Review the following requirements:

- The PROC that starts CA MII GDIF must be named CAMIMGR, and must reside in a proclib allocated by MSTJCL.
- All data set references in the CAMIMGR PROC must be cataloged in the master catalog, or the CAMIMGR PROC must contain UNIT and VOLSER information. All data sets must be PDSs and not PDSEs.
- The CAMIMGR PROC must specify PGM=MIMDRBGN, (MIMDRRM when using Restart Manager), not MIMASCRE.
- The CA MIM address space is started with SUB=MSTR. The startup messages are present in the MIMLOG before JES starts.

- The startup JCL procedure must not reference any SYSOUT data sets.
- The address space is started as a permanent system address space.

When using the CA MIM Early Start Mechanism, consider the following restriction:

- The Early Start Mechanism can be implemented one system at a time.

Restart Manager Usage

The Restart Manager provides the capability to restart CA MIM components automatically after a failure or to perform a planned restart on a single system. When used with CA MII, the restart manager maintains the integrity of global resources throughout the restart process. Local ENQ/DEQ activity is suspended for managed QNAMEs from the instant a failure is detected or a SHUTDOWN RESTART command is issued. Activity remains suspended until the restart is complete and normal global operation resumes.

To activate the Restart Manager, specify PGM=MIMDRRM on the EXEC statement of the CA MIM proc.

Additional requirements:

- Restart Manager requires the use of a STEPLIB; adding the CA MIM load library to the LINKLST is not sufficient. When no STEPLIB DD statement is present, CA MIM terminates with ABEND U40, reason code 164 (X'A4').
- Restart Manager requires that only partitioned data sets (PDS) are specified on the STEPLIB, MIMCNTL, and MIMMSGS DD statements. When any of those DD statements refer to a PDSE, CA MIM terminates with ABEND U40, reason code (X'A8').
- Restart Manager requires that CA MIM be started SUB=MSTR. This requirement is already met when MIM is started through the Early Start mechanism. When CA MIM is started through a START command, code the SUB=MSTR keyword on the command. When CA MIM is started without the SUB=MSTR keyword, MIM terminates with ABEND U40, reason code 169 (X'A9').

We recommend that GDIF be started under the control of the Restart Manager, with the Early Startup mechanism. This procedure provides the maximum protection for global resources by commencing global management as soon as possible during the IPL, and by maintaining global integrity during restart processing.

Note: GDIF is the only CA MIM facility that exploits the Restart Manager. All other CA MIM facilities function normally with the Restart Manager but do not necessarily take advantage of it.

Restart Manager behavior and usage:

- To activate changes to the CA MIM JCL PROC, do not use a SHUTDOWN RESTART command. Instead, issue a SHUTDOWN LOCAL command followed by an MVS START.
- Use the SHUTDOWN RESTART command to activate changes to INIT values that do not need to be consistent across systems. For example, you can change MIMINIT PAGEFIX=COND and do a SHUTDOWN RESTART and the change is in effect.
- During a restart, CA MIM can ignore ALLOCATE initialization statements for checkpoint files and control files. CA MIM continues to use control files and checkpoint files that are allocated before the restart.

- During a restart, CA MIM ignores MIMINIT CHKPTDSN parameters. MIM continues to use checkpoint files that are allocated before the restart.
- Restart Manager does not attempt a restart when CA MIM abends within 60 seconds of initialization. Abends that occur early after initialization are most likely the result of incorrect parameters that require user intervention.
- When used in conjunction with WAITSTATE, CA MIM places the system into a disabled wait state when the restart attempt fails.
- During a restart, Restart Manager forces REUSE=YES. To force REUSE=NO, issue a SHUTDOWN LOCAL, followed by an MVS START.

Start CA MII with the Address Space Creation Utility

Note: The Early Start Mechanism ensures that CA MII runs as a SYSTEM address space. However, if the Early Start Mechanism is in use and CA MII terminates, then restart CA MII using the Address Space Creation Utility.

In general, service providers should execute with a higher dispatching priority than service requestors. CA MIM provides vital system services for z/OS address spaces and users. Therefore, CA MIM must obtain the resources necessary to provide a time critical service. Execute the CA MIM address space as a system created address space to provide optimal throughput. System address spaces execute with a fixed dispatching priority of 255.

Utility Components

The CA MIM Address Space Creation Utility consists of two components:

MIMASCRE Program

MIMASCRE defines CA MIM as a system address space. MIMASCRE is included with the base CA MIM installation.

PROCASC JCL Procedure

The PROCASC sample JCL procedure executes the MIMASCRE program. PROCASC installs to the CAI.CBTDPROC data set and is included with the base CA MIM installation.

Using the ASCRE utility

- Use the z/OS START command to invoke the PROCASCR JCL procedure. As parameters, supply a CA MIM JCL procedure name and any JCL procedure overrides for use with the CA MIM JCL procedure. General syntax for the utility is as follows:

  START PROCASC,,,(*parameters*)

parameters

Specifies the CA MIM procedure name followed by any optional CA MIM JCL procedure overrides.

Examples:

1. Assume that CA MII is started using a JCL procedure named CAMIMGR.

   To use the CA MIM Address Space Creation Utility to start CAMIMGR as a system address space follow these steps:

   1. Customize the PROCASC JCL procedure to fit installation standards, rename it to MIMASC, and copy it to an appropriate procedure library.
   2. Start the utility. Assume that CAMIMGR is usually started with START CAMIMGR. To start CAMIMGR as a system address space issue:

      START MIMASC,,,CAMIMGR

The MIMASC task starts and ends quickly, but creates a system address space named CAMIMGR. The CAMIMGR system address space continues through normal CA MIM initialization and remains active with a dispatching priority of 255 (hex 'FF') providing optimal throughput.

2. Assume the desired effect is:

   - Start CAMIMGR as a system address space.
   - Format the checkpoint file
   - Reuse the existing CA MIM CSA intercepts.
   - Issue the following z/OS START command:

     ```
     START MIMASC,,,(CAMIMGR,REUSE=YES,FORMAT=CHKPT)
     ```

     If CA MII is expected to manage heavy enqueue activity, execute CA MII as a system address space.

     The CA MIM Address Space Creation Utility does not provide as much benefit for the CA MIA or CA MIC components. Therefore, use the CA MIM Address Creation Utility for the CA MII component. Do not use the utility for the CA MIA or CA MIC components.

     Code a WLM classification rule to explicitly assign CA MIM to the SYSTEM service class.

     Note: An installation can assign CA MIM to one of these service classes in a classification rule. For example, assign CA MIM to a report class.

     When CA MII executes as a system address space, the CA MII address space name does not appear in the z/OS DISPLAY A,L command response. If automated procedures depend upon the z/OS D A,L command response, use the z/OS D A,taskname command where taskname is the CA MII JCL procedure name (not the MIMASCRE JCL procedure name). The MIMASCRE utility address space is active for only a few seconds while it launches the CA MIM task specified on the START command. MIMASC terminates immediately after it creates the desired CA MII address space.

     For important information about WLM classification rules specification, see the *z/OS MVS Planning: Workload Management Guide*.

Place License Keys in the CA Common Services PPOPTION Data Set

During startup, CA MIM license validation is performed by calling CA LMP service of the CAIRIM component of CCS. For information about installing CAIRIM, activating CA LMP, and coding CA LMP keys, see the CA Common Services for z/OS documentation.

Place CA LMP keys for each of the CA MIM components (CA MIA, CA MIC, CA MII) that you intend to activate in the KEYS member of the PPOPTION data set, found in the CAS9 JCL procedure.

Follow these steps:

1. Examine the CA LMP Key Certificate you received with your CA MIM installation or maintenance tape.

   Your certificate contains the following information:

   Product Name

   Specifies the trademarked or registered name of CA MIM as licensed for the designated site and CPUs.

   Product Code

   Specifies a two-character code that corresponds to the CA MIM facility.

   Supplement

   Specifies the reference number of your license for the particular CA MIM facility in the format:

   nnnnnn - nnn

   This format differs slightly inside and outside North America, and in some cases may not be provided at all.

   CPU ID

   Identifies the specific CPU for which installation of CA MIM is valid.

   Execution Key

   Specifies an encrypted code required by CA LMP for CA MIM installation. During installation, it is referred to as the LMP code.

   Expiration Date

   Specifies the date your license for CA MIM expires. It is of the format: *ddmmmyy*, as in 27JAN13.

   Technical Contact

   Specifies the name of the designated technical contact at your site who is responsible for installation and maintenance of CA MIM. This is the person to whom CA addresses all CA LMP correspondence.

MIS Director

Specifies the name of the Director of MIS or the person who performs such a function at your site. If the title, but not the name of the individual is indicated on the certificate, then you should supply the actual name when correcting and verifying the certificate.

CPU Location

Specifies the address of the building in which the CPU is installed.

2. Define a CA LMP Execution Key to the CAIRIM parameters by modifying member KEYS in CAI.PPOPTION.

   The parameter structure for member KEYS is:

   PROD(*pp*) DATE(*ddmmmyy*) CPU(*tttt-mmmm/ssssss*)
   LMPCODE(*kkkkkkkkkkkkkkkk*)

   pp

   Specifies the two-character product code. This code is the same as the product code already in use by the CAIRIM initialization parameters for any earlier releases of CA MIM (if applicable). This is required.

   Valid values for *pp* are as follows:

   1A - CA MIA (GTAF and TPCF)

   1C - CA MIC (GCMF)

   1D - CA MII (GDIF, ECMF, and EDIF)

   1F - CA MIC (ICMF)

   ddmmmyy

   Specifies the CA LMP licensing agreement expiration date.

   tttt-mmmm

   Specifies the CPU type and model on which CA LMP is to run (for example, 2094-712). If the CPU type, model, or both, require less than four characters, then blank spaces are inserted for the unused characters. This is required.

   ssssss

   Specifies the serial number of the CPU on which CA LMP is to run. This is required.

   kkkkkkkkkkkkkkkk

   Specifies the execution key needed to run CA LMP. This CA LMP execution key is provided on the Key Certificate shipped with each CA LMP software solution.

Example: Code CA LMP Key

The following example shows you how to code a control statement for the CA LMP execution software parameter.

Note: The product code and execution key value will be different when you install CA MIM at your site.

```
PROD(1B) DATE (27MAR08) CPU(2094-712 /999999)
LMPCODE(52H2K06130Z7RZD6)
```

Review MIMGLOBL DASD Control File Serialization

If you are running the CA MIA or CA MIC address spaces without GDIF active, and those address spaces are using DASD control files (for example, COMMUNICATION=DASDONLY or CTCDASD), tasks in those address spaces issue hardware reserves with QNAME MIMGLOBL that are eligible for conversion by CA MII (running in a separate address space) or by IBM Global Resource Serialization (GRS).

You can optionally let CA MII convert the MIMGLOBL hardware reserves for the CA MIA or CA MIC address spaces if the following conditions are true:

- You run CA MIA or CA MIC in address spaces separate from the CA MII address space
- The CA MIA or CA MIC address space is using DASD control files

If you are using GDPS/Hyperswap, you need to convert all hardware reserves, so this may be one reason why you would choose to convert the MIMGLOBL reserve for the CA MIA and CA MIC address spaces.

Important! You should not convert the MIMGLOBL hardware reserves if any of the following conditions are true:

- The MIAplex or MICplex contains systems that are outside of the MIIplex
- The MIAplex or MICplex contains systems that are outside of the GRSplex. (This condition assumes that you are not running CA MII and that GRSRNL<> EXCLUDE.)
- The MIAplex or MICplex contains any z/VM systems.

If you are running CA MII, consider the following:

- If GDIINIT PROCESS=SELECT, and you choose to convert the MIMGLOBL reserve, then see the CAI.CBTDPARM(MIMQNAMS) parmlib member for instructions on converting the hardware reserve for QNAME MIMGLOBL.
- If GDIINIT PROCESS=ALLSYSTEMS, and you choose to convert the MIMGLOBL reserve, then no entry is needed in the MIMQNAME member.
- If GDIINIT PROCESS=SELECT, and you choose to not convert the MIMGLOBL reserve then, no entry is needed in the MIMQNAMS member.
- If GDIINIT PROCESS=ALLSYSTEMS, and you choose to not convert the MIMGLOBL reserve, then place the following in the MIMQNAME member:

  ```
  MIMGLOBL GDIF=NO
  ```

If you are running GRS, do *one* of the following, but not both:

- To convert the hardware reserve for QNAME MIMGLOBL, place the following entry in the GRSRNLxx member of SYS1.PARMLIB:

  ```
  RNLDEF RNL (CON) TYPE (GENERIC) QNAME (MIMGLOBL)
  ```

- To prevent the global enqueue for QNAME MIMGLOBL from being propagated globally and raising a reserve, place the following entry in the GRSRNLxx member of SYS1.PARMLIB

  ```
  RNLDEF RNL(EXCL) TYPE(GENERIC) QNAME(MIMGLOBL)
  ```

Grant Data Set Access

Before you start the CA MIM startup JCL PROC, make sure that it has the required security access to CA MIM product libraries created in previous installation steps.

The CA MIM startup JCL PROC requires UPDATE access to the following data sets:

- DASD control files created in Allocate Control Files in this chapter
- Checkpoint files created in Allocate Checkpoint Files in this chapter

The CA MIM startup JCL PROC requires READ access to the deployed runtime copies of the following data sets:

- CAI.CBTDPARM
- CAI.CBTDMSEN
- CAI.CBTDLOAD

Note: The CA MIM startup JCL PROC will fail if it is not granted appropriate mainframe security access to CA MIM product libraries.

Start Your CA MIM Address Spaces

NOTE: If CA MIM GDIF is active, we recommend starting CA MII using the Early Start Mechanism (see page 61) in this guide.

Use the z/OS START command to start the CA MIM address space on each system.

This step assumes that CA MIM starts on three systems as defined on the DEFSYS statement found in the CAI.CBTDPARM(MIMINIT) parmlib member. The DEFSYS statement example:

```
DEFSYS (PROD1,P1,SMF1),(PROD2,P2,SMF2),(TEST1,TT,SMF3)
```

The CA MIM startup JCL PROC is named MIMGR. The DASD control and checkpoint files are formatted when CA MIM first starts.

These steps start CA MIM on each system and format new control and checkpoint files.

Follow these steps:

1. On the first system, issue the following z/OS command:

   ```
   START MIMGR,FORMAT=BOTH
   ```

2. On the second system, issue the following z/OS command:

   ```
   START MIMGR,FORMAT=CHKPT
   ```

3. Repeat Step 2 for the third system.

Message MIM0023I displays on all three systems. An example message follows:

MIM0023I SYSTEM *sysid* IN FILE 00 SYNCHRONIZATION COMPLETE

The CA MIM tasks on each system are now active and communicating with one another.

Note: For more information, see the *CA MIM Programming Guide.*

Verify Your MIMplex

You need to verify that the CA MIM started tasks on each system in your MIMplex are active and synchronized.

Note: The following commands use the “at” sign (@) as the CA MIM command prefix character.

Follow these steps:

1. Issue the following command:

   @DISPLAY SYSTEMS

   A display of the status of all systems defined on your DEFSYS statement appears.

2. Issue the following command:

   @DISPLAY FILES

   A display of the status of all primary and secondary CA MIM control and checkpoint files appears.

3. Issue the following command:

   @DISPLAY FACILITIES

   A display of the CA MIM release number and service levels and a list of all active CA MIM facilities in the CA MIM address space appears.

Post-Installation Considerations

Now that CA MIM is successfully installed and started, consider the following:

- For more information about shutting down CA MIM, see the Global Shutdown Procedure section in the *CA MIM for z/OS Programming Guide*.
- Tune *CA MIM to optimize the handling of y*our workloads and processing requirements.

  Note: For more information, see the *CA MIM Programming Guide*.
- In addition to using shared DASD control files as a cross-system communication medium, CA MIM can use other types of hardware for this purpose. For most sites, using a shared DASD control file is adequate. However, when the environment requires faster throughput, consider implementing one of the other CA MIM cross-system communication methods.

  Note: For more information, see the *CA MIM Programming Guide*.
- CA MIM has a robust suite of commands to help systems personnel manage multisystem data-sharing environment.

  Note: For more information, see the *CA MIM Statement and Command Reference Guide*.

- The following CA MIM utilities help with implementation issues, boost address space performance, monitor performance, diagnose problems, or extend general processing capabilities.
  - Help Facility
  - CA Easytrieve Reports
  - SyntaxSCAN Utility
  - CTC Path Validation Utility
  - Message Localization Facility
  - DASD Reserve Validation Utility
  - Tape Device Status Reporting Interface
  - CA MIM Command Processing Interface for TSO Users
  - GDPS/PPRC GDIF HyperSwap Utility

  Note: For more information, see the *CA MIM Programming Guides*.

Chapter 6: Migration Information

This section contains the following topics:

Running CA MIM in Mixed-level Mode

CA MIM Release 12.0 can run concurrently with CA MIM Release 11.8, CA MIM Release 11.9, or CA MIM VM Release 11.5 on different systems in the same MIMplex.

When running the current release of CA MIM in mixed-level mode, upgrade all systems to the current Release 12.0 as quickly as possible to avoid any error conditions. We recommend, carrying out this upgrade over a period of days or weeks, not months.

CA MIM is a multisystem product. While CA MIM can run at different levels on each system, we recommend not to do so long term.

The cross-system communication architectures employed and the global functionality of each CA MIM component makes it impractical to run CA MIM tasks at different control logic levels in the same MIMplex.

For example, CTC IO recovery logic applied to one of five systems provides no benefit to the global MIMplex CTC recovery process. The same is true for global component functions provided by CA MIA, CA MIC, and CA MII.

We recommend running CA MIM at the current release on all systems.

COMPATLEVEL Considerations

CA MIM 12.0 is only compatible with external systems that are running with COMPATLEVEL=11.81, 11.9, or 12.0. Bringing the compatibility maintenance to the current CA MIM release can be required, before installing CA MIM 12.0.

The following CA MIM PTFs provide support for COMPATLEVEL=11.81:

- CA MIM MVS 11.8 - RO38458
- CA MIM VM 11.5 – RO26864

To determine the current COMPATLEVEL value of your active CA MIM tasks, use the F MIM,DISPLAY MIM INIT command.

The simplest method of upgrading to 12.0 is by changing COMPATLEVEL=12.0 in the INIT member. Then start CA MIM 12.0 on all systems with a control file and checkpoint file format. However, this change requires a global CA MIM outage. The following examples show alternate upgrade methods that avoid the global outage.

Example for upgrading from CA MIM 11.9 to CA MIM Release 12.0:

No special CA MIM 11.9 PTFs are required for CA MIM 11.9 before upgrading to 12.0. Follow the standard COMPATELVEL upgrade process. For more information, see the *CA MIM Statement and Command Reference Guide* and the *CA MIM Programming Guide.*

Example: Upgrading to CA MIM for z/OS Release 12.0 with VM systems in the MIMplex requires the following steps:

1. Issue F MIM,DISPLAY INIT to confirm all systems in the MIMplex are running at COMPATLEVEL 11.81.
2. Install RO51703 on all CA MIM VM 11.5 systems.
3. Install CA MIM 12.0 on all MVS systems.
4. Issue F MIM ACTIVATE COMPATLEVEL=11.82.

Note: CA MIM VM does not support COMPATLEVEL=11.9 or 12.0.

Note: For more information about the CA MIM ACTIVATE COMPATLEVEL process, see the *CA MIM Programming Guide* and the *CA MIM Statement and Command Reference*.

Example for upgrading from CA MIM 11.8 to CA MIM 12.0:

1. Confirm CA MIM 11.8 is running on all systems with COMPATLEVEL=11.8.
2. Install RO38458 on all systems, it can be rolled out one system at a time.
3. F MIM,ACTIVATE COMPATLEVEL=11.81
4. Shut down 11.8 and start 12.0 with COMPATLEVEL=11.81. This change can be done on one system at a time, a global shutdown is not required.

5. After MIM 12.0 is up and running on all systems, enter:
6. F MIM,ACTIVATE COMPATLEVEL=11.9
7. F MIM,ACTIVATE COMPATLEVEL=12.0

Chapter 7: CA MIM Product Maintenance

This section contains the following topics:

Product Maintenance Activation

CA MIM sustaining engineering delivers corrective product code and element changes using standard IBM SMP/E ++APAR and ++PTF protocols. Complete the SMP/E RECEIVE and APPLY processing, and refresh the run-time MIM loadlib. Stop and then restart the CA MIM address space to activate the corrective code. CA MIM does not currently support the F LLA, REFRESH command to refresh dynamically CA MIM run-time load modules. Therefore, it requires planning to verify that CA MIM service is activated correctly.

The following items are assumed default actions with regard to getting a CA MIM ++PTF activated:

- An IPL is not required.
- The ++PTF change can be activated by stopping CA MIM on a single system. A global MIMplex-wide shutdown is not required.
- Restarting the CA MIM address space with the REUSE=NO option is not required.
- Restarting the CA MIM address space with a Control File or Checkpoint Files FORMAT is not required.
- The ++PTF can be activated using the SHUTDOWN RESTART command. For details on this command, see the *CA MIM Statements and Commands Reference*, the *CA MIM Programming Guide*, and the *CA MIM Installation Guide*.

If there are exceptions to these actions, a ++HOLD SYSTEM statement is embedded in the ++PTF to alert clients regarding special installation/activation requirements.

For more information, see the *CA MIM Programming Guide* for detailed considerations regarding CA MIM address space termination and activation.

Appendix A: CCS for z/OS Component Requirements

This appendix provides the FMIDs of CA Common Services for z/OS (CCS) components upon which various CA MIM features rely.

This section contains the following topics:

CA LMP

The following FMID is required to validate product licensing for CA MIM:

CAS9E00 or CAS9E10

Specifies the CAIRIM component

CA zIIP Enablement Service

The following CCS FMIDs and PTFs are required for the interface between CA MIM and the CCS zIIP Enablement Service:

- CAS9E00 + PTF RO32488 + PTF RO57356
- CAS9E10 + PTF RO57370

Note: For information about additional setup and configuration steps that must be completed, see the *CA Common Services for z/OS Administration Guide*.

CA Easytrieve Interface

The following FMID is required to run the CA MIM SMF reports.

CEZ6400 or CDX8640 or CDX8E00 or CDX8E10

Note: For information about additional setup and configuration steps that must be completed, see the *CA Common Services for z/OS Administration Guid*e.

CA MIC Intersystem Communication Facility (ICMF)

The following CCS for z/OS component is required for the CA MIM interface to the ICMF.

CBUJE00

Specifies the CA L-Serv component

Note: For information about additional setup and configuration steps that must be completed, see the *CA Common Services for z/OS Administration Guid*e.

CA Service Desk Interface

The following FMIDs are required for the interface between CA MIM and CA Service Desk.

CAS9E00 or CAS9E10

Specifies the CAIRIM component

CAW1E00 or CAW1E10

Specifies the CAIENF component

CDYFE00 or CDYFE10

Specifies the CAISDI/med and CAI/soap components

CAW4E00 or CAW4E10

Specifies the CAICCI with SSL component

For information on additional setup and configuration steps that must be completed, see the CCS for z/OS documentation.

Interface to IBM Health Checker

The following CCS for z/OS component is required for the CA MIM interface to the IBM Health Checker.

CEF5E00 or CEF5E10

Specifies the CA Health Checker Common Service component

Note: For information on additional setup and configuration steps that must be completed, see the CCS for z/OS documentation.

Appendix B: Data Sets Created by CA CSM

This section contains the following topics:

Post SMP/E, Deployment, and Configuration Data Sets

Data sets are created by CA CSM after successfully completing each installation step, that is the SMP/E, deployment, and configuration procedures.

Each step in the installation creates the following data sets.

- SMP/E see the data sets in the POST SIS column in the following table. These data sets are collectively known as the CA MIM SMP/E environment.
- Deployment, see the data sets in the POST SDS column in the following table. These data sets are collectively known as the CA MIM deployment environment.
- Configuration, see the POST SCS column in the following table. These data sets are collectively known as the CA MIM runtime environment data sets. The major data set created by configuration step for CA MIM is RTHLQ.MIMSCNTL. RTHLQ by default is MIM.R119.RT. CA CSM configuration procedure for CA MIM will populate this data set with all the members needed to start either CA MIA, or CA MII, or both, depending on configuration options selected. This will include a startup procedure, a syntax check procedure, and all associated parameter members for only CA MIA, only CA MII, or both.

  Note: The control and checkpoint files creation is based on the configuration options selected on panel two of the configuration wizard.

Data Sets Table

POST SIS	POST SDS	POST SCS
SMPEHLQ.ABTDCLS0		
SMPEHLQ.ABTDEZTM		
SMPEHLQ.ABTDEZTR		
SMPEHLQ.ABTDHENU		

POST SIS	POST SDS	POST SCS
SMPEHLQ.ABTDJCL		
SMPEHLQ.ABTDMAC		
SMPEHLQ.ABTDMOD		
SMPEHLQ.ABTDMSEN		
SMPEHLQ.ABTDPARM		
SMPEHLQ.ABTDPENU		
SMPEHLQ.ABTDPROC		
SMPEHLQ.ABTDSAMP		
SMPEHLQ.ABTDXML		
SMPEHLQ.CBTDCLS0	SDSHLQ.CBTDCLS0	RTHLQ.CBTDCLS0
SMPEHLQ.CBTDEZTM	SDSHLQ.CBTDEZTM	RTHLQ.CBTDEZTM
SMPEHLQ.CBTDEZTR	SDSHLQ.CBTDEZTR	RTHLQ.CBTDEZTR
SMPEHLQ.CBTDHENU	SDSHLQ.CBTDHENU	RTHLQ.CBTDHENU
SMPEHLQ.CBTDJCL	SDSHLQ.CBTDJCL	RTHLQ.CBTDJCL
SMPEHLQ.CBTDLOAD	SDSHLQ.CBTDLOAD	RTHLQ.CBTDLOAD
SMPEHLQ.CBTDMAC	SDSHLQ.CBTDMAC	RTHLQ.CBTDMAC
SMPEHLQ.CBTDMSEN	SDSHLQ.CBTDMSEN	RTHLQ.CBTDMSEN
SMPEHLQ.CBTDPARM	SDSHLQ.CBTDPARM	RTHLQ.CBTDPARM
SMPEHLQ.CBTDPENU	SDSHLQ.CBTDPENU	RTHLQ.CBTDPENU
SMPEHLQ.CBTDPROC	SDSHLQ.CBTDPROC	RTHLQ.CBTDPROC
SMPEHLQ.CBTDSAMP	SDSHLQ.CBTDSAMP	RTHLQ.CBTDSAMP
SMPEHLQ.CBTDXML	SDSHLQ.CBTDXML	RTHLQ.CBTDXML
SMPEHLQ.CSI		
SMPEHLQ.CSI.DATA		
SMPEHLQ.CSI.INDEX		
SMPEHLQ.SMPLOG		
SMPEHLQ.SMPLOGA		
SMPEHLQ.SMPLTS		
SMPEHLQ.SMPMTS		
SMPEHLQ.SMPPTS		

POST SIS	POST SDS	POST SCS
SMPEHLQ.SMPSCDS		
SMPEHLQ.SMPSTS		
		RTHLQ.MIMSCNTL(MIAPROC) (MIASYN) (MIAINIT) (MIACMNDS) (MIASYNCH) (MIAUNITS) (MIIPROC) (MIISYN) (MIIINIT) (MIISYNCH) (MIICMNDS) (EDIPARMS) (GDIEXMPT) (MIIQNAME)
		RTHLQ.MIACF00
		RTHLQ.MIACF01
		RTHLQ.MIICF00
		RTHLQ.MIICF01

Index