CA JCLCheck™ Workload Automation

Installation and Configuration Guide

Version 12.0.00 Second Edition

CA Technologies Product References

This document references the following CA Technologies products:

- CA Chorus™ Software Manager (CA MSM)
- CA Workload Automation CA 7® Edition (CA WA CA 7 Edition)
- CA Workload Automation Restart Option for z/OS Schedulers (CA WA Restart Option for z/OS Schedulers)
- CA Scheduler® Job Management (CA Scheduler)
- CA Workload Automation ESP Edition (CA WA ESP Edition)
- CA Endevor® Software Change Manager (CA Endevor SCM)
- CA APCDOC™ Automated Job Documentation (CA APCDOC)
- CA APCDDS™ Automated Report Balancing (CA APCDDS)
- CA Dispatch™ (CA Dispatch)
- CA ACF2™ (CA ACF2)
- CA Top Secret® (CA Top Secret)
- CA ASM2® Backup and Restore (CA ASM2)
- CA 1® Tape Management (CA 1)
- CA TLMS® Tape Management (CA TLMS)
- CA Roscoe® Interactive Environment (CA Roscoe)
- CA Librarian® (CA Librarian)
- CA PanAPT® (CA PanAPT)
- CA Panvalet® (CA Panvalet)
- CA InterTest™ Batch (CA InterTest Batch)

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 last release of this documentation:

Documentation Fixes - Second Edition

- Removed chapter 5—Installing Your Product from Tape.
- Updated the installation file names.

Contents

Chapter 1: Overview

This guide describes how to install and implement CA JCLCheck.

This section contains the following topics:

Audience

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

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

Consult with the following personnel, as required:

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

Product Description

A job can fail with a JCL error after it has been running for a long time because z/OS only performs JCL syntax checking. Typically, this type of failure requires a costly rerun. CA JCLCheck validates z/OS JCL before it is submitted, which ensures more effective and efficient use of your system by detecting JCL errors that could cause your job to fail.

The following list describes the features of CA JCLCheck:

- Performs complete syntax checks on the JCL, including JES2 or JES3 control statements. CA JCLCheck reads, validates, and interprets IDCAMS, IEHPROGM, IEBCOPY, IEBGENER, SORT, ADRDSSU, and XCOM control statements that perform JCL functions.

- Simulates allocation and termination conditions to identify common errors such as: misspelled data set names, incorrect disposition, and incorrect volume serial numbers. The system checks for execution-time errors that result in system ABENDs. JCL problems (including missing or invalid programs, missing data sets, and incorrect DCB information) cause system ABENDs. CA JCLCheck makes special checks for other common errors, such as, incorrect order of cataloged procedure overrides.
- Performs extensive pre-validation checking of the security environment before the job executes. Pre-validation of the security environment is available for CA ACF2, CA Top Secret, and any SAF compatible product (for example, IBM RACF). CA JCLCheck derives the DF/SMS classes and storage groups that SMS assigns (including the DF/SMS ACS routines).
- Checks and validates STORCLAS, DATACLAS, or MGMTCLAS overrides that you provide and SMS defines.
- Supports all z/OS environments including more recently added features such as, in-stream data in procedures and JOBRC.
- Supports variable pre-resolution of scheduling product variables in ISPF and batch processing. Scheduling products that are supported are CA WA CA 7 Edition, CA WA ESP Edition, Tivoli Workload Scheduler by IBM, and Control-M by BMC.
- Contains the JCLNeat component that reformats JCL according to user-defined specifications. This component provides a methodology to standardize JCL without affecting the efficiency and creativity of the original coder.
- Performs complete syntax validation checking for IBM DB2 commands and runtime validation with the IBM DB2 catalog.
- Performs pre-validation of IBM IMS PSB and DBD definitions.
- Validates UNIX System Services (USS) path directories, files, and disposition specifications.
- Provides complete documentation of a single job or an entire production system with 11 different types of reports. These reports include a flow diagram that displays the flow of the JCL that CA JCLCheck has processed. Also provided is DATACLAS, STORCLAS, and MGMTCLAS information.
- Job Control Standards (JCS), an online menu-driven feature of CA JCLCheck, allows an administrator to define standard rules for JCL checking. The administrator compiles and links these standards using a SAMPJCL member, which then produces a module that is linked into the same load library as CA JCLCheck.
- The REXX interface provides standard rules definition capabilities using a REXX EXEC. Standards are coded as a regular REXX routine. The REXX interface is not dependent on the version. This independence provides greater freedom and mobility among products. You can use the REXX interface and the Job Control Standards facility independently or jointly.

- CA JCLCheck is easy to install; it adapts itself automatically to unit and volume conventions in your installation, without requiring any operating system modifications. You can tailor it to check for (and enforce) installation accounting and JCL standards, and submit error-free jobs to JES2 or JES3 for execution.
- CA JCLCheck is easy to use online, in batch mode, from TSO, or from CA Roscoe. You do not need to use control statements. The only CA JCLCheck input is an unmodified JCL job stream. CA JCLCheck is designed so that you can interpret both its reports and error messages.
- The following CA Technologies products use CA JCLCheck for JCL syntax, runtime validations, or both:
  - CA WA CA 7 Edition
  - CA WA ESP Edition
  - CA Scheduler
  - CA Endevor
  - CA Deliver
  - CA InterTest Batch
  - CA PanAPT
  - CA APCDOC
  - CA Roscoe

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—Optionally creates a new CSI environment and runs the RECEIVE, APPLY and ACCEPT steps. The software is untailored.
- Deployment—Copies the target libraries to another system or LPAR.
- Configuration—Creates customized load modules, bringing the software to an executable state.

CA MSM provides a web-based interface to make the standardized installation process easier. Using CA MSM, someone with limited knowledge of JCL and SMP/E can install a product.

Note: If you do not have CA MSM, you can download it from the Download Center at the CA Support Online website. Follow the installation instructions in the CA Mainframe Software Manager documentation bookshelf on the CA Mainframe Software Manager product page. The standardized installation process can also be completed manually.

To install your product, do the following:

1. Prepare for the installation by confirming that your site meets all installation requirements (see page 15).
2. Use one of the following methods to acquire the product:
   - Download the software from CSO using CA MSM (see page 27).
   - Download the software from CSO using Pax-Enhanced Electronic Software Delivery (ESD) (see page 33).
3. Perform an SMP/E installation using one of the following methods:
   - If you used CA MSM to acquire the product, start the SMP/E step from the SMP/E Environments tab in CA MSM.
   - If you used ESD to acquire the product, you can install the product manually or use the Insert New Product option in CA MSM to complete the SMP/E install.

   Note: If a CA Recommended Service (CA RS) package is published for your product, install it before continuing with deployment.
4. Deploy the target libraries using one of the following methods:
   - If you are using CA MSM, deployment is required; it is a prerequisite for configuration.
   - If you are using a manual process, deployment is an optional step.

   Note: Deployment is considered part of starting your product.
5. Configure your product using CA MSM or manually.

   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 JCLCheck operates under all levels of the z/OS operating systems that IBM supports.

CA JCLCheck does not modify the operating system in any way. CA JCLCheck automatically adapts itself to the local generic and esoteric unit names. CA JCLCheck loads the system device name and mask, and eligible device tables, or CA JCLCheck interfaces with the documented scheduler service routines. Other than these tasks, the operating system and other local dependencies are specified entirely in the CA JCLCheck tables. CA JCLCheck is re-entrant; therefore, it can be placed in the Link Pack Area (LPA) of z/OS systems.

CA JCLCheck and its modules must be in authorized libraries when any of the following features are selected: AUTOPROC, CA ASM2, automatic destination checking, RESOLVE scheduling package variable resolution, HCD, or Security Interface support. Certain exceptions apply, as noted elsewhere in this document.

To run CA JCLCheck, allocate a virtual region size of at least 512 KB.

Software Requirements

The following software is required for CA JCLCheck:

- IBM supported release of z/OS
- SMP/E

TSO and ISPF Version Compatibility

For CA JCLCheck to run authorized in the TSO environment, TSO/E is required.

ISPF Interface

CA JCLCheck includes a Dialog Manager interface for use with IBM ISPF/PDF under z/OS. You can use this interface to run CA JCLCheck in the foreground or to submit a job to execute CA JCLCheck in the background.

If executed in the foreground, the user can:

- Request error-free jobs to submit for execution
- Review the output using BROWSE
- Generate hardcopy output using the PRINTOFF command

You can use menus to select runtime options and save them in your user profiles for subsequent requests. You can also access a HELP function using PF keys.

Compatibility with other CA Technologies Products

The following table lists the CA Technologies products that you can use with CA JCLCheck.

Product	Minimum Release Level
CA Roscoe	Release 6.0
CA Librarian	Release 4.3
CA PanAPT	Release 3.1
CA Panvalet	Release 14.5
CA 1	Release 12.6
CA ASM2	Release 4.2
CA TLMS	Release 12.6
CA WA Restart Option for z/OS Schedulers	Release 3.0
CA WA CA 7 Edition	Release 11
CA WA ESP Edition	Release 11.4
CA APCDOC	Release 1.3
CA Dispatch	Any currently supported release
CA Scheduler	Release 11.0
CA ACF2	Release 9.0

Product	Minimum Release Level
CA Top Secret	Release 9.0

Common Services for z/OS

Common Services for z/OS must be installed into an APF-authorized library. LINKLIST is not required; however, we recommend it as a best practice. If you choose not to use the LINKLIST, STEPLIB statements are required.

CA Common Services Requirements

The following CA Common Services are used with CA JCLCheck:

- CAIRIM
- CAICCI
- CA LMP

Note: If other CA Technologies products are installed at your site, some of these services are already installed.

CAIRIM

CAI Resource Initialization Manager (CAIRIM), is the common driver for a collection of dynamic initialization routines. The routines eliminate the need for user SVCs, SMF exits, subsystems, and other installation requirements that are commonly encountered when installing system software. The routines are grouped under the CA Technologies z/OS dynamic service code CAS9C00, which is the CAIRIM component. CAIRIM allows you to:

- Obtain SMF data
- Verify proper software installation
- Install z/OS interfaces
- Automatically startup CA Technologies and other vendor products
- Properly time and order initialization

CA JCLCheck requires CAIRIM; it is an initialization program that prepares your operating system environment for your CA Technologies products and then starts them. CAIRIM is required for product licensing authorization. Optionally, you can use CAIRIM for the security option or to install SUBCHEK.

The following sample is a control statement:

```
//CAIRIM   EXEC  PGM=CAIRIM
//STEPLIB  DD    DISP=SHR,DSN=CAI..JCLCHECK.CAZ2LOAD
//         DD    DISP=SHR,DSN=CAI.CAS90S.LINKLIB
//KEYS     DD  *
        PROD(J6) ……
//PARMLIB  DD  *
        PRODUCT(CA JCLCHECK) VERSION(Z1C0) INIT(Z1C0INIT)
//
```

Note: For more information, see the *CA Common Services for z/OS Getting Started Guide*.

CAIRIM must be installed into an APF-authorized library. Linklist is not required; however, we recommend it to help prevent errors that modules cause when they are not available. CAIRIM is installed in the common target library CAW0LOAD.

CAIRIM must be installed at the generation level that is required to support LMP. See your maintenance or installation cover letter for this information. If you have already installed CAIRIM with another CA Technologies product, ensure that it is at this maintenance level or above. If it is not, you must RECEIVE and APPLY the SYSMOD again. Alternatively, apply the appropriate maintenance with the CA Common Services for z/OS tape.

CAIRIM requires approximately 16 KB of ECSA and 4 KB of CSA for a z/OS environment. Additional CSA requirements for resident modules vary with each component or product.

The CAIRIM component and CAISSF (CAI Standard Security Interface) must be installed before using the CA JCLCheck security interface.

CA JCLCheck must be authorized if you use the security interface to SAF (for example, IBM RACF). CAZ2LOAD must also be authorized if you use a SAF compatible product with CAISSF.

LMP Support

The CA License Management Program (CA LMP) portion of CAIRIM services interfaces with CA JCLCheck to determine licensing authorization. CAIRIM must be at the requisite generation level to support CA LMP. See your maintenance or installation cover letter for this information.

CA JCLCheck requires CA LMP, one of the CA Common Services for z/OS, to initialize correctly. CA LMP also provides a standardized and automated approach to the tracking of licensed software. Examine the CA LMP key certificate that you received with your CA JCLCheck installation or maintenance tape. That certificate contains the following information:

Product Name

Defines the trademark or registered name of the CA JCLCheck licensed for the designated site and CPUs.

Product Code

Defines a two-character code that corresponds to CA JCLCheck.

Supplement

Defines the reference number of your license for the particular CA JCLCheck, in the format *nnnnnn - nnn*.

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

CPU ID

Defines the code that identifies the specific CPU for the installation of your CA JCLCheck that is valid.

Execution Key

Defines an encrypted code that CA LMP requires for the CA JCLCheck installation.

During installation, it is referred to as the LMP Code.

Expiration Date

Defines the date your CA JCLCheck license expires, for example, *ddmmmyy* as in 15JAN07.

Technical Contact

Defines the name of the technical contact at your site that is responsible for installation and maintenance of the designated CA JCLCheck.

This contact is the person to whom CA Technologies addresses all CA LMP correspondence.

MIS Director

Defines the name of the Director of MIS, or the person who performs that function at the site.

If the title, but not the name of the individual is indicated on the certificate, supply the actual name when correcting and verifying the certificate.

CPU Location

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

CA LMP is provided as part of CAIRIM, another one of the CA Common Services for z/OS. Once CAIRIM has been installed or maintained at Service Level A5 or higher, CA LMP support is available for all CA LMP-supported CA Technologies software products.

The CA LMP execution key, provided on the key certificate, must be added to the CAIRIM parameters to ensure proper initialization of the CA Technologies software product. To define a CA LMP execution key to the CAIRIM parameters, modify the member KEYS in CAI.CAW0OPTN.

The parameter structure for the member KEYS is:

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

pp

Indicates the two-character product code. This code agrees with the product code already in use by the CAIRIM initialization parameters for earlier genlevels of CA JCLCheck. (Required)

ddmmmyy

Indicates the CA LMP licensing agreement expiration date. (Required)

tttt-mmmm

Indicates the CPU type and model (for example, 3090-600) on which CA LMP is to run. If the CPU type or model requires less than four characters, blank spaces are inserted for the unused characters. (Required)

ssssss

Indicates the serial number of the CPU on which CA LMP is to run. (Required)

kkkkkkkkkkkkkkkk

Indicates the execution key that is required to run CA LMP. This CA LMP execution key is provided on the key certificate that is shipped with each CA LMP software solution. (Required)

The following example is a control statement for the CA LMP execution software parameter. Although this example uses CA SORT to represent CA JCLCheck, the CA LMP execution key value is invalid and is provided as an example only.

```
PROD(SL) DATE (01JAN07) CPU(3090-600 /370703) LMPCODE(52H2K07030Z7RZD6)
```

Note: For more information about defining the CA LMP execution key to the CAIRIM parameters, see the *CA Common Services for z/OS Getting Started*.

Security Requirements

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

- SUPERUSER authority—if you need to mount a zFS or HFS file system.

CAISSF

CAI Standard Security Facility (CAISSF), allows CA Technologies software to offer standardized security interfaces regardless of the underlying access control software. CAISSF offers user authentication and resource access validation facilities. CAISSF can interface with CA Technologies security products CA ACF2 and CA Top Secret, or compatible non-CA Technologies security products. CAISSF is a subservice that is contained within the CA Technologies z/OS service code, CAS9 (CAIRIM). For CA Technologies security products, some of the CAISSF features include:

- A single security mechanism
- Isolation of CA Technologies enterprise solutions from CA Technologies or vendor mechanisms

For security products not belonging to CA Technologies, some of the CAISSF features include:

- Resource class translation
- Access level translation
- Selective logging of requests
- Request type control
- Message support for failed access

Storage Requirements

Ensure that you have the following storage available:

- If installing with ESD, 40 cylinders for the downloaded files
- For installation and setup:
  - Installation = 180 cylinders
  - SMP/E temporary libraries = 40 cylinders

Target Libraries

The following table shows the minimum storage requirements for the SMP/E target libraries that are required to execute CA JCLCheck. The file space requirements are in terms of tracks for 3390 type DASD.

Library Name	Blksize	Tracks (prim, sec)	Dir Blks	Description
CAI.CAZ2CLS0	27920	38,4	12	Common CLIST library
CAI.CAZ2DBRM	27920	3,1	12	DB2 database request module library
CAI.CAZ2JCL	27920	44,2	40	Sample JCL library
CAI.CAZ2LOAD	32760	469,81	288	Common Load library
CAI.CAZ2MAC	27920	56,6	62	Common Macro library
CAI.CAZ2OPTN	27920	31,2	24	Source for optional modules
CAI.CAZ2PNL0	27920	133,8	143	Common ISPF panel library
CAI.CAZ2PROC	27920	23,2	24	Supplied procedures
CAI.CAZ2MSG0	27920	14,1	14	Common ISPF message library
CAI.CAZ2RPF	27920	20,3	12	Roscoe procedure facility library
CAI.CAZ2SAMP	27920	54,3	48	Sample program source library
CAI.CAZ2SCST	27920	26,2	24	Common MSM SCS template library
CAI.CAZ2SKL0	27920	14,1	12	Common ISPF skeleton library
CAI.CAZ2SRC	27920	89,9	49	Source library

Library Name	Blksize	Tracks (prim, sec)	Dir Blks	Description
CAI.CAZ2TBL0	27920	11,1	12	Common ISPF table library
CAI.CAZ2XML	27998	86,10	24	Common MSM product XML library

Allocate the following table for each user if it is not already established because CA JCLCheck uses the ISPF Table Facility in its ISPF dialog. Typically, the ISPF data set profile of the user is sufficient. The following data set is for ISPF use:

DDname	Description	RECFM	LRECL	BLKSIZE
ISPTABL	Table output library	FB	80	See note.

Note: The block size must be a multiple of 80.

Distribution Libraries

The following table lists the amount of disk space that is required to install the distribution libraries:

Library Name	Blksize	Tracks (prim,sec)	Dir Blks	Description
CAI.AAZ2CLS0	27920	38,4	12	CA JCLCheck CLIST library
CAI.AAZ2DBRM	27920	3,1	12	CA JCLCheck common component database request module library
CAI.AAZ2JCL	27920	44,2	40	CA JCLCheck sample JCL library
CAI.AAZ2MAC	27920	56,6	62	CA JCLCheck macro library
CAI.AAZ2MOD0	32760	126,18	114	CA JCLCheck module library
CAI.AAZ2MSG0	27920	13,1	13	CA JCLCheck ISPF message library
CAI.AAZ2OPTN	27920	31,2	24	CA JCLCheck source for optional modules
CAI.AAZ2PNL0	27920	133,8	143	CA JCLCheck ISPF panel library

Library Name	Blksize	Tracks (prim,sec)	Dir Blks	Description
CAI.AAZ2PROC	27920	23,2	24	CA JCLCheck supplied procedures
CAI.AAZ2RPF	27920	20,3	12	CA JCLCheck Roscoe library
CAI.AAZ2SAMP	27920	54,3	48	CA JCLCheck sample program source library
CAI.AAZ2SCST	27920	26,2	24	CA JCLCheck MSM SCS template library
CAI.AAZ2SKL0	27920	14,1	12	CA JCLCheck ISPF skeleton library
CAI.AAZ2SRC	27920	89,9	49	CA JCLCheck source library
CAI.AAZ2TBL0	27920	11,1	12	CA JCLCheck ISPF table library
CAI.AAZ2XML	27998	86,10	24	CA JCLCheck MSM product XML library

Note: For information about the CA General Transaction Server (GTS) distribution libraries that are required for CA JCLCheck, see the *CA GTS User Guide*.

The product installation SYSMODS are:

Required Common Component SYSMODs

CAZ2C00 Common CA JCLCheck components

Required Component SYSMODs

CAZ1C00 Base-level function

Optional Component SYSMODs

CAZ1C01 CA Roscoe RPF support

CD51C00 General Transaction Server – CA GTS (required for ChekPlex)

CAZ1C02 CA 1 support

CAZ1C03 CA TLMS support

Other Requirements

The following requirements assist you in preparing for installation of CA JCLCheck:

- JCL Procedures (see page 25)
- System Requirements for JCLNeat (see page 25)
- Documentation (see page 25)

JCL Procedures

The JCL procedures CA JCLCheck uses are copied into the JCLCheck procedure library, CAZ2PROC, during the product installation. The procedures are customized during the product customization. CA Technologies recommend that these procedures be made available for use through your system PROCLIB concatenation. This can be accomplished by copying the contents of CAZ2PROC to a common procedure library that currently exists in your PROCLIB concatenation. You can also add the CAZ2PROC library to your system PROCLIB concatenation. Choose the method that conforms to your site installation standards.

System Requirements for JCLNeat

The JCLNeat ISPF interface, JCKNSPF, invokes JCLNeat from an ISPF edit session to reformat the JCL in the current edit workspace. This feature allows JCLNeat to run as an edit macro under the IBM Interactive Systems Productivity Feature (ISPF/PDF) product. With this feature, you can invoke JCLNeat while you are in the ISPF editor by pressing a PF key or entering the JCKNSPF command.

Because JCKNSPF runs as an ISPF edit macro, the options that are set in panel JCKN01 are saved in the ISRPROF member of the ISPF profile data set. This makes the JCKN01 options unique (not shared) from the JCLNEAT options set in panel JCK0205. The options set in panel JCK0205 are saved in the CAZ1PROF member of the ISPF profile data set.

Documentation

CA Technologies provide the following documentation to help you install, customize, maintain, and use CA JCLCheck:

- *Installation and Configuration Guide*
- *Programming Guide*
- *Tutorial*
- *Best Practices Guide*
- *Command Reference Guide*

- *Message Reference Guide*
- *Common Component Installation and Configuration Guide*
- *CA General Transaction Server User Guide*
- *CA General Transaction Server Message Reference Guide*

Concurrent Releases

You can install this release of CA JCLCheck 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:

- 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 MSM 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 MSM..

- 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 Product Using CA MSM

These topics provide information to get you started managing your product using CA MSM. You can use the online help included in CA MSM to get additional information.

Before using these topics, you must already have CA MSM installed at your site. If you do not have CA MSM installed, you can download it from the Download Center at the CA Support Online website, which also contains links to the complete documentation for CA MSM.

How to Use CA MSM: Scenarios

Imagine that your organization has started using CA MSM to simplify the installation of CA Technologies products and unify their management. You have also licensed a new CA Technologies product. In addition, you have a number of existing CSIs from previously installed CA Technologies products.

You can use the following scenarios to guide you through the process:

1. Acquire the new product (see page 27).
2. Install the new product (see page 28).
3. Maintain products already installed in your environment (see page 29).
4. Deploy the product to your target systems (see page 30).
5. Configure the deployed product to your target systems (see page 31).

How to Acquire a Product

The *Product Acquisition Service (PAS)* facilitates the acquisition of mainframe products and the service for those products, such as program temporary fixes (PTFs). PAS retrieves information about products to which your site is entitled. Then it records these entitlements in a software inventory that is maintained on your driving system.

You can use the PAS component of CA MSM to acquire a CA Technologies product.

Follow these steps:

1. Set up a CA Support Online account.

   To use CA MSM to acquire or download a product, you must have a CA Support Online account. If you do not have an account, you can create one on the CA Support Online website.

2. Determine the CA MSM URL for your site.

   To access CA MSM (see page 32), you require its URL. You can get the URL from your site's CA MSM administrator and log in using your z/OS credentials. When you log in for the first time, you are prompted to create a CA MSM account with your credentials for the CA Support Online website. This account enables you to download product packages.

3. Log in to CA MSM and go to the Software Catalog page to locate the product that you want to manage.

   After you log in to CA MSM, you can see the products to which your organization is entitled on the Software Catalog tab.

   If you cannot find the product you want to acquire, update the catalog. CA MSM refreshes the catalog through the CA Support Online website using the site IDs associated with your credentials for the CA Support Online website.

4. Download the product installation packages.

   After you find your product in the catalog, you can download the product installation packages.

   CA MSM downloads (acquires) the packages (including any maintenance packages) from the CA FTP site.

After the acquisition process completes, the product is ready for you to install or maintain.

How to Install a Product

The *Software Installation Service (SIS)* facilitates the installation and maintenance of mainframe products in the software inventory of the driving system. This facilitation includes browsing downloaded software packages, managing SMP/E consolidated software inventories on the driving system, and automating installation tasks.

You can use the SIS component of CA MSM to install a CA Technologies product.

Follow these steps:

1. Initiate product installation and review product information.
2. Select an installation type.
3. Review installation prerequisites if any are presented.

4. Take *one* of the following steps to select an SMP/E environment:
   - Create an SMP/E environment:
     a. Set up the global zone.
     b. Create a target zone.
     c. Create a distribution zone.
   - Use an existing SMP/E environment from your working set:
     a. Update the global zone.
     b. Set up the target zone: Either create a target zone or use an existing target zone.
     c. Set up the distribution zone: Either create a distribution zone or use an existing distribution zone.

     Note: If you install a product or its components into an existing target or distribution zone, older versions are deleted from the zone and associated data sets. We recommend that you use new target and distribution zones for this installation so that you can apply maintenance to your current version, if necessary.
5. Review the installation summary and start the installation.

After the installation process completes, check for and install available product maintenance. The product is ready for you to deploy. Sometimes there are other steps to perform manually outside of CA MSM before beginning the deployment process.

How to Maintain Existing Products

If you have existing CSIs, you can bring those CSIs into CA MSM so that you can maintain all your installed products in a unified way from a single web-based interface.

You can use the PAS and SIS to maintain a CA Technologies product.

Follow these steps:

1. Migrate the CSI to CA MSM to maintain an existing CSI in CA MSM.

   During the migration, CA MSM stores information about the CSI in the database.
2. Download the latest maintenance for the installed product releases from the Software Catalog tab.

   If you cannot find a release (for example, because the release is old), you can add the release to the catalog manually and then update the release to download the maintenance.

3. Apply the maintenance.

Note: You can also install maintenance to a particular CSI from the SMP/E Environments tab.

After the maintenance process completes, the product is ready for you to deploy. You may have to perform other steps manually outside of CA MSM before beginning the deployment process.

How to Deploy a Product

The *Software Deployment Service (SDS)* facilitates the mainframe product deployment from the software inventory of the driving system to the target system. This facilitation includes deploying installed products that are policy-driven with a set of appropriate transport mechanisms across a known topology.

You can use the SDS component of CA MSM to deploy a CA Technologies product that you have already acquired and installed.

Follow these steps:

1. Set up the system registry:
   a. Determine the systems you have at your enterprise.
   b. Set up remote credentials for those systems.
   c. Set up the target systems (non-sysplex, sysplex or monoplex, shared DASD cluster, and staging), and validate them.
   d. Add network information, including data destination information, to each system registry entry.
2. Set up methodologies.
3. Create the deployment, which includes completing each step in the New Deployment wizard.

   After creating the deployment, you can save it and change it later by adding and editing systems, products, custom data sets, and methodologies, or you can deploy directly from the wizard.

   Note: If you must deploy other products to the previously defined systems using the same methodologies, you must create a separate deployment.
4. Deploy the product, which includes taking a snapshot, transmitting to target, and deploying (unpacking) to your mainframe environment.

After the deployment process completes, the product is ready for you to configure. You may have to perform other steps manually outside of CA MSM before beginning the configuration process.

How to Configure a Product

The *Software Configuration Service (SCS)* facilitates the mainframe product configuration from the software inventory of the driving system to targeted z/OS operating systems.

You can use the SCS component of CA MSM to configure a CA Technologies product that you have already acquired, installed, and deployed.

Follow these steps:

1. Select a deployed product to configure from the Deployments tab to open the Create Configuration wizard.
2. Create the configuration, which includes completing each step in the Create Configuration wizard, including the following:
    a. Define a configuration name and select a target system.
    b. Select configuration functions and options.
    c. Define system preferences.
    d. Create target settings.
    e. Select and edit resources.
3. Build the configuration. The last step of the Create Configuration wizard lets you build the configuration.
4. Implement the configuration. The implementation process in CA MSM is a step-by-step process that carefully guides you and provides detailed instructions to start, stop, and manage the steps of the implementation process.

After the configuration process completes, the product is ready for you to use. You may have to perform other steps manually outside of CA MSM.

Note: You cannot use CA MSM to configure a product to a staging system.

Access CA MSM Using the Web-Based Interface

You access CA MSM using the web-based interface. Obtain the URL of CA MSM from the CA MSM administrator.

Follow these steps:

1. Start your web browser, and enter the access URL.

   The login page appears.

   Note: If the Notice and Consent Banner appears, read and confirm the provided information.

2. Enter your z/OS login user name and password, and click the Log in button.

   The initial page appears. If you log in for the first time, you are prompted to define your account on the CA Support Online website.

   Note: For more information about the interface, click the online help link at the top right corner of the page.

3. Click New.

   You are prompted for the credentials to use on the CA Support Online website.

   Important! The account to which the credentials apply *must* have the Product Display Options set to BRANDED PRODUCTS. You can view and update your account preferences by logging in to the CA Support Online website and clicking My Account. You need the correct setting to use CA MSM to download product information and packages.

4. Specify the credentials, click OK, and then click Next.

   You are prompted to review your user settings.

   Note: These settings are available on the User Settings page.

5. Change the settings or keep the defaults, and then click Finish.

   A dialog shows the progress of the configuration task. You can click Show Results to view the details of the actions in a finished task.

Important! If your site uses proxies, review your proxy credentials on the User Settings, Software Acquisition page.

Chapter 4: Installing Your Product from Pax-Enhanced ESD

This section contains the following topics:

How to Install a Product Using Pax-Enhanced ESD

This section describes the Pax-Enhanced ESD process. We recommend that you read this overview and follow the entire procedure the first time you complete a Pax-Enhanced ESD installation. For experienced UNIX users, the *Pax-Enhanced ESD Quick Reference Guide* has sufficient information for subsequent installations.

Important! Downloading pax files for the SMP/E installation as part of the Pax-Enhanced ESD process requires write authority to the UNIX System Services (USS) directories that are used for the ESD process.

If you prefer not to involve all CA Technologies product installers with z/OS UNIX System Services, assign a group familiar with USS to perform Steps 1 through 4 and provide the list of the unpacked MVS data sets to the product installer. USS is not required for the actual SMP/E RECEIVE of the product or for any of the remaining installation steps.

To install files using Pax-Enhanced ESD, use the following process:

1. Allocate and mount the file system. This 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 Pax-Enhanced ESD and create the directory in this file system. Ensure that all users who will be working with pax files have write authority to the directory.

2. Copy the product pax files into your USS directory. To download files, choose one of the following options:

   - Download a zip file from CA Support Online to your PC, unzip the file, and then upload the product pax files to your USS file system.
   - FTP the pax files from CA Support Online directly to your USS directory.

   Note: Perform Steps 3 through 6 for each pax file that you upload to your USS directory.

3. Create a product directory from the pax file. 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*

4. Use the SMP/E GIMUNZIP utility to create z/OS installation data sets. The file UNZIPJCL in the directory that the pax command created in Step 3 contains a sample JCL to GIMUNZIP the installation package. Edit and submit the UNZIPJCL JCL.

5. Receive the SMP/E package. Use the data sets that GIMUNZIP created in Step 4. Perform a standard SMP/E RECEIVE using the SMPPTFIN and SMPHOLD (if applicable) DASD data sets. Also, specify the high-level qualifier for the RELFILEs on the RFPREFIX parameter of the RECEIVE command.

6. Proceed with product installation. Consult product-specific documentation, including AREADME files and installation notes to complete the product installation.

7. (Optional) Clean up the USS directory. Delete the pax file, the directory that the pax command created, all of the files in it, and the SMP/E RELFILEs, SMPMCS, and HOLDDATA data sets.

More Information:

Allocate and Mount a File System (see page 38)
Create a Product Directory from the Pax File (see page 41)

How the Pax-Enhanced ESD Download Works

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

Use the following process to download files using Pax-Enhanced ESD:

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

   The CA Support Online web page appears.

2. Under Download Center, select Products from the first drop-down list, and specify the product, release, and genlevel (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 or mainframe. If you download a zip file, you must unzip it before continuing.

   For both options, The ESD Product Download Window (see page 35) topic explains how the download interface works.

   Note: For traditional installation downloads, see the *Traditional ESD User Guide*. Go to https://support.ca.com/, log in, and click Download Center. A link to the guide appears under the Download Help heading.

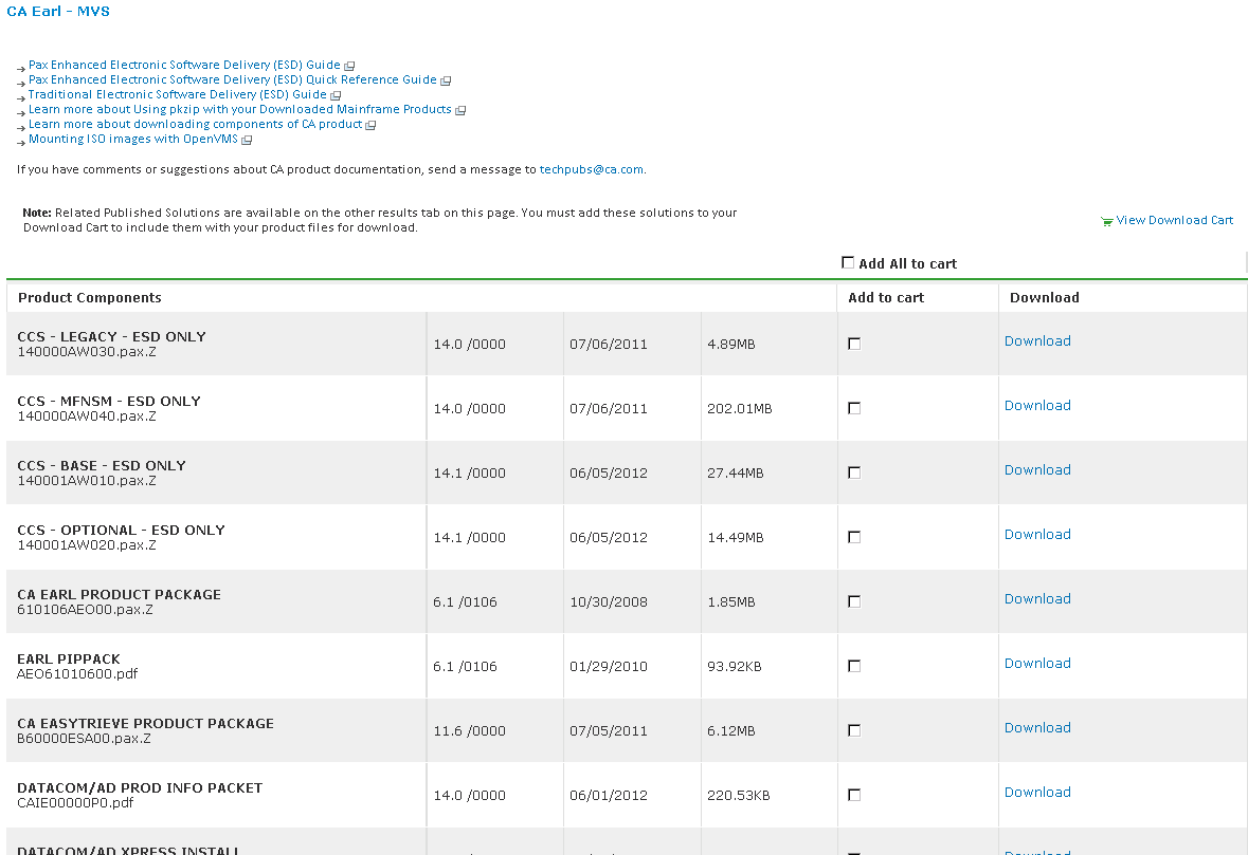
4. Perform the steps to install the product based on the product-specific steps.

   The product is installed on the mainframe.

ESD Product Download Window

You can download CA Technologies product ESD packages multiple ways. Your choices depend on the size of the individual files and the number of files that you want to download. You can download the complete product with all components, or you can select individual pax and documentation files for your product or component.

The following illustration shows sample product files. The illustration lists all components of the product. You can use the Download Cart by selecting one or more components that you need, or selecting the check box for Add All to cart. If you prefer to immediately download a component, click the Download link.



CA Earl - MVS

- Pax Enhanced Electronic Software Delivery (ESD) Guide
- Pax Enhanced Electronic Software Delivery (ESD) Quick Reference Guide
- Traditional Electronic Software Delivery (ESD) Guide
- Learn more about Using pkzip with your Downloaded Mainframe Products
- Learn more about downloading components of CA product
- Mounting ISO images with OpenVMS

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

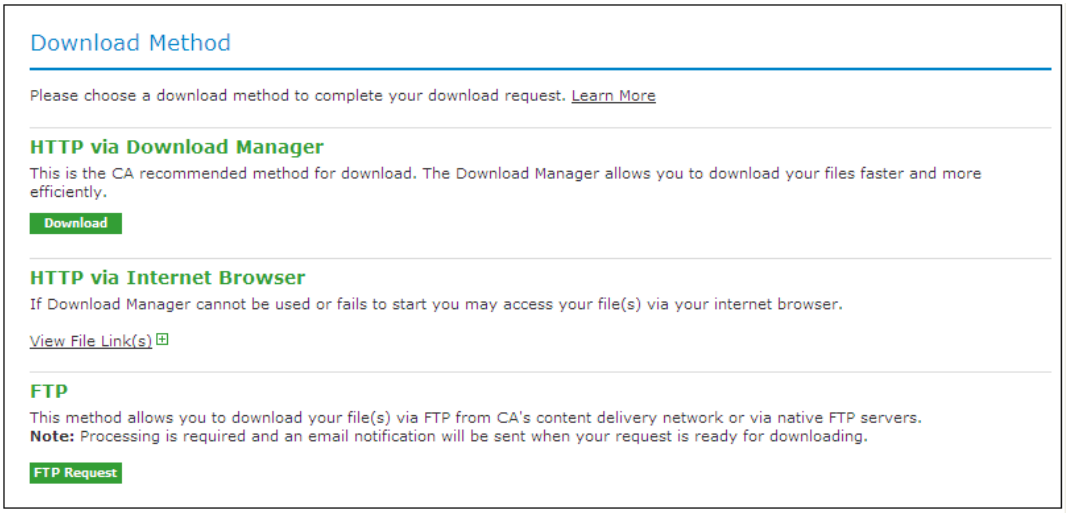
Note: Related Published Solutions are available on the other results tab on this page. You must add these solutions to your Download Cart to include them with your product files for download.

View Download Cart

Add All to cart

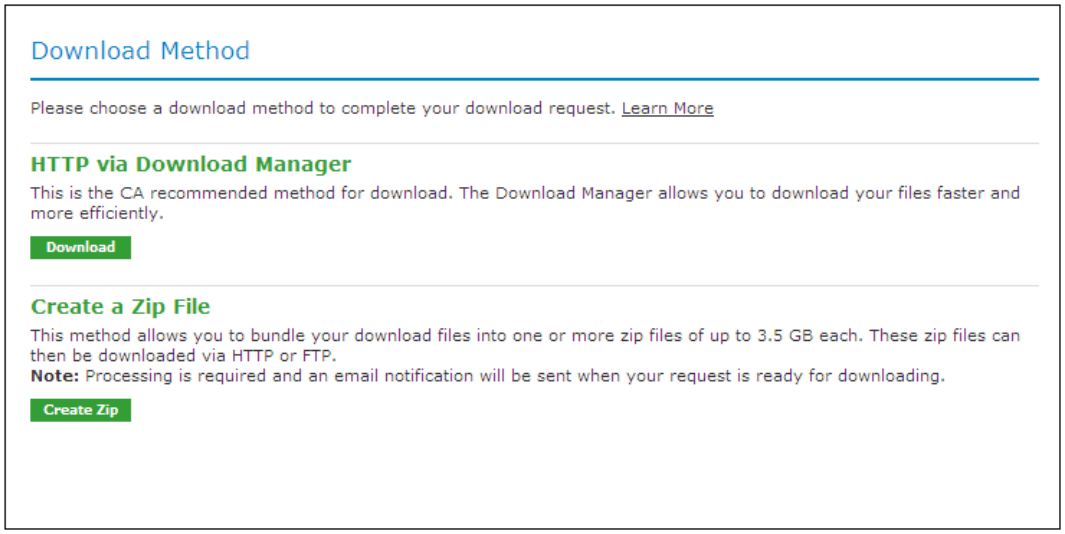
Product Components				Add to cart	Download
CCS - LEGACY - ESD ONLY 140000AW030.pax.Z	14.0 /0000	07/06/2011	4.89MB	☐	Download
CCS - MFNSM - ESD ONLY 140000AW040.pax.Z	14.0 /0000	07/06/2011	202.01MB	☐	Download
CCS - BASE - ESD ONLY 140001AW010.pax.Z	14.1 /0000	06/05/2012	27.44MB	☐	Download
CCS - OPTIONAL - ESD ONLY 140001AW020.pax.Z	14.1 /0000	06/05/2012	14.49MB	☐	Download
CA EARL PRODUCT PACKAGE 610106AEO00.pax.Z	6.1 /0106	10/30/2008	1.85MB	☐	Download
EARL PIPPACK AEO61010600.pdf	6.1 /0106	01/29/2010	93.92KB	☐	Download
CA EASYTRIEVE PRODUCT PACKAGE B60000ESA00.pax.Z	11.6 /0000	07/05/2011	6.12MB	☐	Download
DATACOM/AD PROD INFO PACKET CAIE00000P0.pdf	14.0 /0000	06/01/2012	220.53KB	☐	Download
DATACOM/AD XPRESS INSTALL				☐	

Clicking the link for an individual component takes you to the Download Method page.



Depending on the size and quantity of ordered product files, the Download Method screen could also have these options:

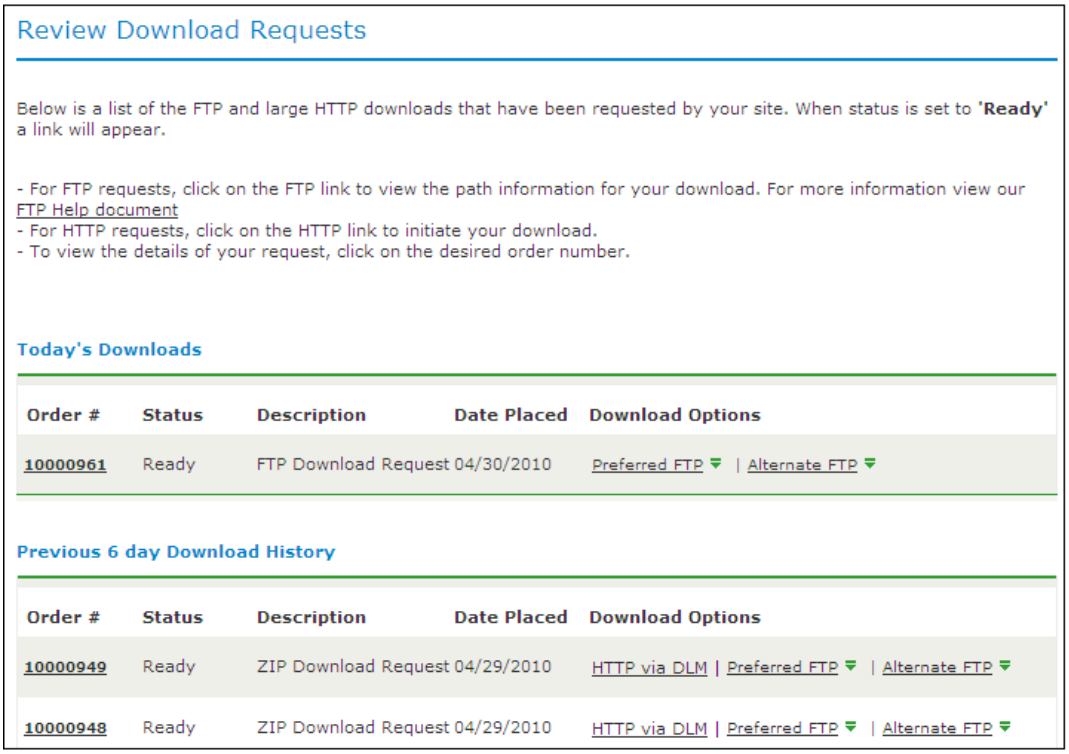
Note: For mainframe downloads using this HTTP method, click the Learn More link.



The HTTP method lets you start downloading immediately. The FTP method takes you to the Review Orders page that displays your order, first in a Pending status changing to Ready when your order has been processed.

Preferred FTP uses the new content delivery network (CDN). Alternate FTP uses the CA Technologies New York-based FTP servers.

The Create a Zip File option first creates the zip, and when ready, offers the options that the Zip Download Request examples show in the next illustration.



Review Download Requests

Below is a list of the FTP and large HTTP downloads that have been requested by your site. When status is set to **'Ready'** a link will appear.

- For FTP requests, click on the FTP link to view the path information for your download. For more information view our FTP Help document
- For HTTP requests, click on the HTTP link to initiate your download.
- To view the details of your request, click on the desired order number.

Today's Downloads

Order #	Status	Description	Date Placed	Download Options
10000961	Ready	FTP Download Request	04/30/2010	Preferred FTP | Alternate FTP

Previous 6 day Download History

Order #	Status	Description	Date Placed	Download Options
10000949	Ready	ZIP Download Request	04/29/2010	HTTP via DLM | Preferred FTP | Alternate FTP
10000948	Ready	ZIP Download Request	04/29/2010	HTTP via DLM | Preferred FTP | Alternate FTP

Allocate and Mount a File System

You can use the zSeries File System (zFS) or hierarchical file system (HFS) for ESD 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
//CAESD  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/CAESD directory in an existing directory, /u/maint. From the TSO OMVS shell, enter the following commands:

   ```
   cd /u/maint/
   mkdir CA
   cd CA
   mkdir CAESD
   ```

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

   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('yourUSSESDdirectory')
           TYPE(ZFS)  MODE(RDWR)
           PARM(AGGRGROW)
     ```

   - On an HFS, use the following sample:

     ```
     MOUNT FILESYSTEM('your_HFS_data_set_name')
           MOUNTPOINT('yourUSSESDdirectory')
           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 ESD directory and its files. For example, to allow write access to the ESD directory for other users in your USS group, from the TSO OMVS shell, enter the following command:

   ```
   chmod -R 775 /yourUSSESDdirectory/
   ```

   Write access is granted.

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

Create a Product Directory from the Pax File

Use the sample job attached to the PDF file as Unpackage.txt to extract the product pax file into a product installation directory.

Important! To simplify the Pax-Enhanced ESD process, the PDF version of this guide includes a sample JCL job that you can copy directly to the mainframe. To access this job, click the paper clip icon at the left of the PDF reader. A window displaying attachments opens. Double-click the file to view the sample JCL.

Follow these steps:

1. Supply a valid JOB statement.
2. Replace *yourUSSESDdirectory* with the name of the USS directory that you use for ESD 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 runs and 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.

Sample Job to Execute the Pax Command (Unpackage.txt)

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

```
//ESDUNPAX JOB (ACCOUNTNO),'UNPAX 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 "yourUSSESDdirectory" with the name of the USS         *
//*    directory used on your system for 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, make  *
//*       sure the 'X' continuation character is in column 72.        *
//*********************************************************************
//UNPAXDIR EXEC PGM=BPXBATCH,
// PARM='sh cd /yourUSSESDdirectory/; pax -rvf paxfile.pax.Z'
//*UNPAXDIR EXEC PGM=BPXBATCH,
//* PARM='sh cd /yourUSSESDdirectory/; 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)*.

How to Install Products Using Native SMP/E JCL

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

1. Allocate product data sets and SMP/E data sets.
2. Create SMP/E CSI.
3. Receive base functions.
4. Apply base functions.
5. Accept base functions.
6. Configure the product according to your site requirements.

Prepare the SMP/E Environment for Pax Installation

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

Set the NULLFILE HLQ and use it for the IMS, HSSR, and CICS DDDEF specifications if valid libraries do not exist in your environment.

Follow these steps:

1. Customize the macro AJ6SEDIT 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 AJ6SEDIT 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* for the unzip to DASD ESD JCL.

   Note: The following steps include instructions to execute the AJ6SEDIT macro each time you open a new SAMPJCL member. To edit all SAMPJCL members simultaneously, read and follow the instructions in the AJ6EDALL member.

2. Open the SAMPJCL member AJ62ALL in an edit session and execute the AJ6SEDIT macro from the command line.

   AJ62ALL is customized.

3. Submit AJ62ALL.

   This job produces the following results:

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

4. Open the SAMPJCL member AJ63CSI in an edit session and execute the AJ6SEDIT macro from the command line.

   AJ63CSI is customized.

5. Submit AJ63CSI.

   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.

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.

Follow these steps:

1. Open the SAMPJCL member AJ64RECD in an edit session, and execute the AJ6SEDIT macro from the command line.

   Note: Comment out any unwanted FMIDs.

   AJ64RECD is customized.

2. Submit the *yourHLQ*.SAMPJCL member AJ64RECD to receive SMP/E base functions.

   CA JCLCheck is received and now resides in the global zone.

3. Open the SAMPJCL member AJ65APP in an edit session, and execute the AJ6SEDIT macro from the command line.

   Note: Comment out any unwanted FMIDs.

   AJ65APP is customized.

4. Submit the *yourHLQ*.SAMPJCL member AJ65APP to apply SMP/E base functions.

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

5. Open the SAMPJCL member AJ66ACC in an edit session, and execute the AJ6SEDIT macro from the command line.

   Note: Comment out any unwanted FMIDs.

   AJ66ACC is customized.

6. Submit the *yourHLQ*.SAMPJCL member AJ66ACC to accept SMP/E base functions.

   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-Enhanced 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 Maintenance

CA Support Online has maintenance and HOLDDATA published since the installation data was created. After the maintenance process completes, the product is ready to deploy.

Follow these steps:

1. Check CA Support Online and download any PTFs and HOLDDATA published since this release was created. If the base release was created recently, no PTFs or HOLDATA will have been published yet.
2. Transfer the downloaded files to two separate FB 80 sequential data sets. Use one data set to contain the PTFs and the other to contain the HOLDDATA.

   The PTFs and HOLDDATA become accessible to the *yourHLQ*.SAMPJCL maintenance members.
3. The AJ6SEDIT macro was customized in the installation steps. Verify that you still have the values from the base installation.
4. Open the SAMPJCL member AJ67RECP in an edit session and execute the AJ6SEDIT macro from the command line.

   AJ67RECP is customized with your JOB statement, CSI location, and zone names.
5. Customize the AJ67RECP SMPPTFIN and SMPHOLD DD statements to reference the FB 80 data sets for the PTFs and HOLDDATA.
6. Submit AJ67RECP.

   The PTFs and HOLDDATA are received.
7. Open the SAMPJCL member AJ68APYP in an edit session and execute the AJ6SEDIT macro from the command line.

   AJ68APYP is customized.
8. Submit AJ68APYP.

   The PTFs are applied.
9. (Optional) Open the SAMPJCL member AJ69ACCP in an edit session and execute the AJ6SEDIT macro from the command line.

   AJ69ACCP is customized.
10. (Optional) Submit *yourHLQ*.SAMPJCL member AJ69ACCP.

    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.

DB2BIND

Indicates that DBRMs have changed and packages need to be rebound.

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.

EXIT

Indicates that changes delivered by this SYSMOD require reassembly of user exits.

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.

MULTSYS

Apply this SYSMOD to multiple systems for either pre-conditioning, coexistence, or exploitation.

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. The HOLDDATA is commonly used for SYSMODs that have been distributed and later are discovered to cause problems.

Download the external HOLDDATA from CA 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.

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.

A special HOLDDATA class that is called ERREL exists. We have determined that the problem fixed by the SYSMOD is more important than the one that it causes. We recommend that you apply these SYSMODs.

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.

Chapter 5: How to Configure Without CA MSM

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

Note: A worksheet for the steps in this section can be found in Appendix A: Preparation Worksheets (see page 83).

Summary of Configuration Steps

The following list summarizes the steps involved in configuring CA JCLCheck. Review this list before you begin, and then use it as a checklist during the configuration process.

1. Tailor the CA JCLCheck JCL
2. (Optional) Install Parameter CSECT Modifications
3. Establish CA JCLCheck as APF-Authorized
4. Install ISPF Support
5. Install JCLNeat ISPF Interface
6. Install TSO Support
7. (Optional) Install SUBCHEK
8. (Optional) Install EDCHEK
9. (Optional) Install ChekPlex
10. (Optional) Install CA APCDOC interface
11. (Optional) Install CA JCLCheck/CA Roscoe Monitor
    a. (Optional) Install RPF programs
    b. (Optional) Install CA JCLCheck/CA Roscoe ISPF Support
12. (Optional) Install Printer Support
13. Tailor the CA Common Services for z/OS Initialization Procedure
14. (Optional) Install security interface
15. (Optional) Install Support for Job Control Standards
16. (Optional) Install Support for the REXX Interface
17. (Optional) Install Support for the DB2 Interface

18. Verify the Installation
19. (Optional) Install CA TLMS and CA 1 Support
20. (Optional) Install User Exits
21. (Optional) Install CA JCLCheck Tables for Modification
22. (Optional) Install JCLNeat Tables for Modification
23. Save Installation Materials

Tailor the CA JCLCheck JCL

The CAI Common Procedure library (DD=CAZ2PROC) contains all the procedures relevant to CA JCLCheck. These procedures were placed there during SMP APPLY processing.

Edit each JCL procedure to conform to your installation standards and the previously completed worksheet.

After you complete these modifications, you can copy the procedures into a PROCLIB of your choice. Alternatively, you can add the CAI Common Procedure library (CAZ2PROC) to the system PROCLIB concatenations.

The following procedures are supplied with CA JCLCheck:

Procedure	Description
CAZ1ASM	Assembles source for CA JCLCheck user exits.
CAZ1JCCM	Resolves Control-M auto-edit variables before executing CA JCLCheck.
CAZ1JCC7	Resolves CA WA CA 7 Edition variables before executing CA JCLCheck.
CAZ1JCES	Resolves CA WA ESP Edition variables before executing CA JCLCheck.
CAZ1JCHK	Executes JCLCHECK program.
CAZ1JCKT	Executes JCLCHECK as a batch TSO session.
CAZ1JCS	Executes CAZ1EVAL program to compile Job Control Standards.
CAZ1JCTW	Resolves Tivoli Workload Scheduler variables before executing CA JCLCheck.
CAZ1LNK	Links assembled exits or options into the target load library, CAZ2LOAD for CA JCLCheck.
CAZ1NEAT	Executes JCLNeat program.

Procedure	Description
CAZ1OPCA	Activates the OPC/ESA interface.
CAZ1RLST	Executes the CAZ1RLST program to report on JCS rules.
CAZ1SMFP	Executes JSMFANAL program
CAZ2ASM	Assembles source for CA JCLCheck user exits (contains the MACLIB statements for the TMS or TLMS source modules).
CAZ2LNK	Links assembled exits or options into the target load library, CAZ2LOAD for Common Component.

Important! If your site uses CA TLMS support, add the DD statement CAIVMFI to the procedure CAZ1JCHK. Additionally, specify the data set name of the CA TLMS Volume Master File (VMF).

Review the following execution JCL members in the CAZ2JCL library and modify to reflect your system environment. These members are provided in the CAI.CAZ2JCL library for executing some of the procedures on the previous page.

Member	Description
AZ1GCHK	Executes CAZ1JCHK with graphic SYSOUT
AZ1JCHK	Executes CAZ1JCHK
AZ1NEAT	Executes CAZ1NEAT
AZ1SMFA	Executes CAZ1SMFP

Note: Member AZ1SMFA performs SMF analysis. This program estimates the cost of the JCL errors if CA JCLCheck did not detect them. For more information about modifying input control statements, see Special Usage Considerations in the *Programming Guide*.

(Optional) Install Parameter CSECT Modifications

Note: We recommend that you install the parameter exits after you install CA JCLCheck.

You can replace or add any parameter CSECT in CA JCLCheck by assembling and linking your own exit, then changing the appropriate JCLTABLE to load your exit.

To install parameter CSECT modifications, do the following steps:

1. Modify the exit source code from member CAZ1XPAR in CAZ2SRC. Be sure to change the $JCLPRTN macro if substituting a CA Technologies supplied exit.
2. Execute the member MZ1C033 in CAI.CAZ2JCL to assemble the exit as a stand-alone load module. You can create one load module per parameter exit.

3. Change the JCLTABLE MVS4XTBL to load your exit.
4. See the install table for modifications step in this chapter.

Note: For information about modifying parameter CSECTs, see Installation Options in the *Programming Guide*.

Since you can implement this step after installation, JCLCHECK and its aliases can exist in other libraries. Make sure that you execute member AJ6CAPF in CAZ2JCL to replace them with the updated copy. For information, see Establish CA JCLCheck as APF-Authorized (see page 54).

Important! If you write an exit for a parameter that already includes an exit defined as part of the standard $VTBs, be sure to invoke the supplied exit (with all registers reset) after your exit has completed. Failure to invoke a supplied exit generates unpredictable results, since many of these exits are vital to the successful execution of CA JCLCheck. Invoke the supplied exit after your routine, because many of the supplied exits alter the registers, before returning to CA JCLCheck. The $JCLPRTN macro is provided to assist you in meeting this requirement.

Establish CA JCLCheck as APF Authorized

An increasing number of CA JCLCheck options require authorization to accomplish their purpose. We recommend that you place CA JCLCheck modules in a common load library with other CA Technologies products and that the library is APF authorized. This way, you do not have to concern yourself whether options need authorization. You insulate yourself from future changes that could require authorization.

Note: For more information about using these options, see Runtime Options in the *Command Reference Guide*.

The following options require APF authorization:

Option	Qualifier
ASM	All options
AUTOPROC	All options
DESTCHK(J)	(J) Options only
EASYRDR	All options
HCD	All options
JCLLIB	All Options (Explicit or Implied)
PVTCAT(AUTH)	(AUTH) option only
REMOTE	All options

Option	Qualifier
RESOLVE	All options
SCHENV	All options
SIGNON	All options
SECURITY	All options

CA ASM2, AUTOPROC, HCD, and in certain cases DESTCHK(J), require CA JCLCheck to run authorized. CA ASM2 support allows CA JCLCheck to identify those data sets that are under CA ASM2. AUTOPROC is the automatic PROCLIB recognition option. DESTCHK(J) permits automatic destination checking for JES2/JES3 shops (JES2 shops before MVS/ESA Version 3.0 do not need authorization, or if you are running the SECURITY option with the SAF interface). HCD activates hardware configuration definition support for dynamically defined devices. If you select any one of these options, authorize CA JCLCheck modules (link with AC=1) and put them in an Authorized Program Facility (APF) library. Also, if CA JCLCheck is to run authorized in a TSO environment, TSO/E and a TSO authorization update is required.

To accomplish this APF requirement, do one of the following tasks:

- Copy CA JCLCheck load modules to an APF library or a linklist library
- Add CAI.CAZ2LOAD to either the APF list (IEAPF*xx*) or the linklist (LNKLST*xx*).

  We recommend that you use a linklist library that is also named in the APF list of the system. Be aware when non-authorized libraries are concatenated with authorized libraries, authorization is turned off for all of the libraries in the concatenation.

Important! CA JCLCheck load modules have numerous ALIAS names. Do not attempt to copy CA JCLCheck load modules using an online utility from one library to another. Instead, if you must copy the load modules, always tailor then run the CAZ2JCL member AJ6CAPF in batch. If you do not copy the base module with all of its aliases in one operation, then you create multiple load modules. These modules could cause execution and maintenance problems.

Establish CA JCLCheck as APF Authorized in TSO

When running within the TSO/E environment, there are other requirements for proper authorization, for example, updating the appropriate TSO authorization table entries.

To authorize CA JCLCheck within TSO/E, update the Authorized TSO Service Facility table.

If you plan to run CA JCLCheck as a TSO Command Processor, for example, 'TSO JCLCHECK …', update the Authorized Command table.

If you plan to run CA JCLCheck as an Authorized Program with CA Endevor, update the Authorized Program table.

Here are the entries that are required for each of the tables:

AUTHTSF—Authorized TSO Service Facility	Add ➔	CAZ2CTSO
AUTHCMD—Authorized Command	Add ➔	JCLCHECK
AUTHPGM—Authorized Program	Add ➔	JCLCHECK

Entries for AUTHTSF, AUTHCMD, and AUTHPGM are specified in the IKJTSOxx member of SYS1.PARMLIB. You can update the IKJTSO00 tables without an IPL by using the IBM TSO PARMLIB command.

If the PARMLIB command is used, TSO/ISPF users that are currently logged on have to log out and log back in again before their TSO session uses the updated AUTHTSF definition.

TSO/E option

When running under ISPF (which is not authorized), CA JCLCheck automatically recognizes that TSO/E is installed and uses the TSOLNK TSO service facility to obtain authority. You do not have to specify this option. This option is ignored when running CA JCLCheck in batch. When running under an ISPF environment, be aware that the TSOLNK service is strictly a TSO service, ISPF does not provide it. Therefore, this service does not recognize any modules in an ISPLLIB concatenation.

For ISPF Version 2.2 and above

Using CA JCLCheck as a command processor creates more implications if you are using the ISPF interface and the default options have been changed in the ISPF ISPTCM module. You use this module to define TSO commands and certain options including a security check. If a TSO command is not found in this table, the default flag setting is taken. The default flag setting is set at X'70' by IBM, with the X'20' bit indicating that you should make a security check before calling a command as authorized. Either enable this bit or define JCLCHECK in this table, setting this flag on, for an authorized option to work.

Per IBM, SUBMIT should not be run as an authorized command. When SUBMIT is in AUTHCMD, two ENQs are issued on SYSZTSOE TCBA causing the session to go into a WAIT state. Therefore, if SUB and SUBMIT are in the authorized command table in SYS1.PARMLIB member IKJTSO*xx*, remove them.

CAZ1JSUB is linked with AC=1, which does not cause a problem as long as the commands are *not* in the authorized command table. Link authorize SUBCHEK to support the options that require authorization, such as AUTOPROC.

Install ISPF Support

CA JCLCheck supports a series of formatted ISPF online panels under TSO. These panels let you perform database maintenance and job monitoring functions.

To install the ISPF panel support, do the following steps:

1. Construct your own user CLIST library or modify your logon CLIST to concatenate the CAI ISPF libraries with your system libraries as follows:

```
ALLOC F(SYSPROC)   DA('CAI.CAZ2CLS0'-
ALLOC F(ISPPLIB)   DA('CAI.CAZ2PNL0'-
ALLOC F(ISPMLIB)   DA('CAI.CAZ2MSG0'-
ALLOC F(ISPSLIB)   DA('CAI.CAZ2SKL0'-
ALLOC F(ISPTLIB)   DA('CAI.CAZ2TBL0'-
```

   Alternatively, you can use the LIBDEF method of allocating your system libraries by modifying the JCKSPF CLIST as follows:

```
PROC 0 CONLIST SYMLIST
GLOBAL ZMAIN TSOE AUTOP GTLOAD JCLOAD PRNT
ALTLIB ACTIVATE APPLICATION(CLIST) DATASET('SYS3.JCLCHECK.CAICLS0')
ALTLIB ACTIVATE APPLICATION(EXEC) DATASET('SYS4.JCLCHECK.RULES')
ISPEXEC LIBDEF ISPMLIB DATASET ID('CAI.CAZ2MSG0')
ISPEXEC LIBDEF ISPPLIB DATASET ID('CAI.CAZ2PNL0')
ISPEXEC LIBDEF ISPSLIB DATASET ID('CAI.CAZ2SKL0')
ISPEXEC LIBDEF ISPTLIB DATASET ID('CAI.CAZ2TBL0')
ISPEXEC SELECT CMD(%JCKMAIN &CONLIST &SYMLIST) NEWAPPL(CAZ1) PASSLIB
ISPEXEC LIBDEF ISPMLIB
ISPEXEC LIBDEF ISPPLIB
ISPEXEC LIBDEF ISPSLIB
ISPEXEC LIBDEF ISPTLIB
ALTLIB DEACTIVATE APPLICATION(CLIST)
ALTLIB DEACTIVATE APPLICATION(EXEC)
END
```

   Important! Do not use ISPLLIB to contain CA JCLCheck modules if CA JCLCheck must run authorized. TSO authorization can only be obtained when the load modules are loaded from LINKLIST or a STEPLIB with only authorized libraries.

Since this product uses ISPF Table Facilities, the following additional allocation is required for each user:

```
ALLOC F(ISPTABL)   DA('user.file')
```

' *user.file* ' files can be the user's profile data set (ISPPROF) or any other PDS that has an LRECL of 80 and record format of fixed blocks. Generally, most environments already have ISPTABL defined. This is only required if ISPTABL is not currently allocated.

2. Modify the CLIST JCKUSR in CAI.CAZ2CLS0 as follows. Since this is an SMPE controlled library, you may want to copy JCKUSR to another CLIST library before altering.

   a. Make sure that the assignment for the variable &JCLOAD points to the JCLCHECK load library.

   b. Delete the assignment statement for the variable ZMAIN that is not correct for your Job Entry subsystem.

   c. If you are running TSO/E, change the statement SET TSOE=N to SET TSOE=Y. If you want to use AUTOPROC, change the statement SET AUTOP=N to SET AUTOP=Y.

      Note: AUTOPROC requires CA JCLCheck be authorized.

   d. If you are not going to use AUTOPROC, define the default PROCLIBs of your installation in the ALLOCATE statement for the JCHKPLIB file. The default PROCLIBs are the PROCLIBs defined in your JES procedure: ddname PROC00 for JES2, ddname IATPLBST for JES3.

      Code additional ALLOCATE statements for each set of alternate procedure libraries. For example, if you define PROC01 and PROC02 in addition to PROC00 in the JES procedure, you could tailor JCKUSR as follows:

```
ALLOC FI(JCHKPLIB)  DA('SYS1.PROCLIB' +
                           *
                           *
                           *
                    'SYS3.PROCLIB') SHR REUSE
ALLOC FI(PROC01) DA('SYS4.PROCLIB' +
                           *
                           *
                           *
                    'SYS7.PROCLIB') SHR REUSE
ALLOC FI(PROC02) DA('SYS8.PROCLIB' +
                           *
                           *
                           *
                    'USER.PROCLIB') SHR REUSE
```

Code a FREE statement for every PROCLIB DD you allocate, in addition to the default JCHKPLIB in the CLIST. Code the FREE statements after the line in JCKUSR that reads:

```
'FREE FI(JCHKPLIB)'
```

For example, if you code allocations for PROC01 and PROC02 in JCKUSR as previously shown, you can free these allocations by coding the following statements:

```
IF &TSOE = N THEN -
     FREE FI(JCHKPLIB
FREE FI(PROC01 PROC02)  <++++ ADDITIONAL FREE
```

3. Delete the assignment statement of the variable PRNT for the printer facility that does *not* apply to your site.

4. If you use the TLMS option, add an ALLOCATE and FREE statement for the CAIVMFI file. See item d. in item 2 shown previously for format.

5. Tailor the skeleton file JCKSKEL in file CAZ2SKL0 to define the default PROCLIBs in your installation for the ddname JCHKPLIB. Use the same definition for the CLIST described in Step 1.

   Note: If you are using AUTOPROC, comment out the JCHKPLIB DD statement in JCKSKEL. If you use the TLMS option, remove the comment from the CAIVMFI DD statement and tailor it, if necessary.

6. Place CA JCLCheck modules in a library that is allocated to JOBLIB, STEPLIB, or the LINKLIST when you are using options that do not require authorization. Authorize CA JCLCheck modules when you use an option requiring authorization; therefore, they come from an authorized library. For more information, see Establish CA JCLCheck as APF-Authorized (see page 54). The suggested location is LINKLIST or STEPLIB. According to IBM, never authorize ISPF libraries; therefore, the use of the ISPLLIB is not supported.

7. Add a JCLCHECK entry to one of your menu panels. You can accomplish this step in several ways. Here is one method:

   a. Copy your system ISPF/PDF Primary Option Menu, ISR@PRIM, into CAI.CAZ2PNL0 and rename the copy to CAY@PRIM.

   b. Add the CA JCLCheck option to the CAY@PRIM selection list:

      ```
      %  J + CA JCLCheck - CA JCLCheck ISPF interface
      ```

   c. In the process section ,)PROC , of CAY@PRIM, add the additional statement that you need for the new option in the following way:

      ```
      J,'CMD(%JCKSPF)'
      ```

   d. JCKSPF in CAZ2CLS0 specifies NEWAPPL(CAZ1). You can modify JCKSPF to specify the APPL ID of your choice.

e. To test the new menu, enter the following command from native TSO:

```
ISPSTART PANEL(CAY@PRIM)
```

f. To make this new primary panel available to the user community, rename it to ISR@PRIM and place it in the first panel library concatenated to ISPPLIB.

Note: If you have CA Scheduler or CA WA CA 7 Edition, update members CAZ1SFJC and CAZ17FJC in the skeleton JCL library to reflect your environment.

As noted in these members, add your procedure libraries. The JCL executes procedures that are installed with the scheduling system. Refer to the appropriate scheduling installation manual to make sure that the procedure is installed and to see if additional tailoring is necessary.

Install JCLNeat ISPF Interface

JCKNSPF is an edit macro in CAZ2CLS0 used to reformat JCL while in ISPF edit mode.

- To use this feature, those modules that are prefixed with CAZ1N must be in a library allocated to JOBLIB/STEPLIB or the linklist. Step 13, item 1, documents the libraries that you must allocate for JCKNSPF. As indicated in Step 13, item 1, you should allocate these libraries.
- For ease of use, set a PF key to the characters JCKNSPF and press that key while in edit mode on a JCL data set or member, or enter **JCKNSPF** on the command line to display the JCLNeat Options panel.

Change the options you want to change and press Enter. When all options pass verification checking, CA JCLCheck passes control to JCLNeat to reformat the JCL. If JCLNeat encounters any errors, it inserts a CAY6*xxx* message in the original edit work file, otherwise the next panel JCLNeat displays contains reformatted JCL.

All messages inserted into the edit work file due to error conditions are explained in the JCLNeat section of the *Message Reference Guide*.

Parameters and options not applicable to the reformatting of an SPF Edit Work File have been omitted from the option panel, Inlib, Lines, Members, or Outlib report.

JCKNSPF will allocate and build a KEYWORDS file based on the input supplied on the JCK0215 Keyword Ordering screen.

Although it is possible to pre-allocate //CHANGES, //ADDCARDS, and //GSRCNTL and have JCLNeat honor these requests, we do not recommend this since ISPF has the editing capabilities at hand.

Install TSO Support

In addition to using the ISPF panels, you can invoke communications from a TSO session to CA JCLCheck by using the supplied CLIST member CAZ1TJCK.

To use this member, you must first have CAI.CAZ2CLS0 allocated to SYSPROC or copy CAZ1TJCK to one of the CLIST libraries that is allocated to SYSPROC (that is, ALLOC F(SYSPROC) DA(CAI.CAZ2CLS0)).

To install TSO support, do the following:

1. Copy CAZ1TJCK to a library concatenated with SYSPROC and tailor the data set name to your system standards.
2. If CA TLMS support is installed, modify CAZ1TJCK by adding an allocation and FREE command for the CA TLMS Volume Master File (VMF).

(Optional) Install SUBCHEK

SUBCHEK is the option that CA JCLCheck uses to give control to the TSO SUBMIT command.

Note: For more information, see Installation Options in the *Programming Guide*.

Important! Per IBM, SUBMIT should not be run as an authorized command. When SUBMIT is in AUTHCMD, two ENQs are issued on SYSZTSOE TCBA causing the session to go into a WAIT state; therefore, if SUB and SUBMIT are in the authorized command table in SYS1.PARMLIB member IKJTSOxx, you should remove them.

SUBCHEK is linked with AC=1, which does not cause a problem as long as the commands are not in the authorized command table. SUBCHEK needs to be linked authorized to support the options that require authorization, such as AUTOPROC.

Installation of SUBCHEK requires the following steps:

1. To the CAIRIM input parameter in CAZ2OPTN member Z1LIST, add the following to the end of the "PRODUCT....." parameter preceded by a blank:

   ```
   PARM(SUBCHEK)
   ```

   The statement is as follows:

   ```
   //CAIRIM   EXEC  PGM=CAIRIM
   //STEPLIB  DD    DISP=SHR,DSN=CAI..JCLCHECK.CAZ2LOAD
   //           DD    DISP=SHR,DSN=CAI.CAS90S.LINKLIB
   //KEYS       DD  *
       PROD(J6) ……
   //PARMLIB  DD  *
       PRODUCT(CA JCLCHECK) VERSION(Z1C0) INIT(Z1C0INIT) PARM(SUBCHEK,¢ddname=)
   //
   ```

The preceding statement is in a fixed format and the "¢*ddname*=" is replaced with the appropriate ddname if used.

a. Add SYS1.CMDLIB or the data set containing the SUBMIT program to the STEPLIB concatenation.

Important! You can optionally specify an 8-byte ddname as a sub-parameter of the PARM parameter to be used by SUBCHEK. The ddname "DISABLE" is used to deactivate the SUBCHEK hook.

2. If your installation does not want to let users bypass SUBCHEK by freeing the control file, or if you want to allocate the control file through the program, examine the control file allocation exit CAZ1JSDX in CAI.CAZ2SRC and modify it as required. USERMOD MZ1C022 in CAZ2JCL should be used to assemble and link this change for SUBCHEK. Also execute AJ6CAPF, if necessary, to authorize the file.

   Note: If you are using CAZ1JSDX SUBCHEK exit, the product initialization parameter looks like the following:

```
//CAIRIM   EXEC  PGM=CAIRIM
//STEPLIB  DD    DISP=SHR,DSN=CAI..JCLCHECK.CAZ2LOAD
//           DD    DISP=SHR,DSN=CAI.CAS90S.LINKLIB
//KEYS       DD  *
    PROD(J6) ……
//PARMLIB  DD  *
    PRODUCT(CA JCLCHECK) VERSION(Z1C0) INIT(Z1C0INIT) PARM(SUBCHKX,¢ddname=)
//
```

3. Tailor a SUBCHEK control statement file for your installation. The following examples are supplied in CAI.CAZ2OPTN:

Member in CAZ2OPTN	Function
CAZ1SSUP	Control statement file without AUTOPROC
CAZ1SSUA	Control statement file with AUTOPROC
CAZ1SSUT	Control statement file with CA TLMS but not AUTOPROC
CAZ1SSUM	Control statement file with CA TLMS and AUTOPROC

Note: CAZ1SSUP does not use AUTOPROC; CAZ1SSUA uses AUTOPROC. Specify the one most appropriate to CAI.CAZ2CLS0 member SUBCHK, or to any user CLIST member. Tailor it to reflect proper options and the correct PROCLIBs, if applicable.

4. Add the SUBCHEK control statement file to the LOGON procedures of the TSO users who intend to use SUBCHEK, reflecting the library and member name created in Step 3. The CAZ2CLS0 member SUBCHK illustrates this use.

   For example:

   ```
   //SUBCHKDD DD DSN=CAI.CAZ2OPTN(CAZ1SSUP),DISP=SHR
   ```

   Note: You can also perform this allocation in a LOGON CLIST, or by using an allocate command prior to using a SUBMIT command.

5. Make the CA JCLCheck program modules resident in a linklist library, the link pack area, or a library that is allocated to your TSO LOGON procedure, see Establish CA JCLCheck as APF-Authorized (see page 54). We recommend that a systems programmer perform this step.

6. If you are upgrading from a prior version, review the description of the ERROR option in the *Command Reference Guide*.

(Optional) Install EDCHEK

The EDCHEK feature, installed by SYSMOD CAZ1C00, provides input and output services in the ISREDIT environment. EDCHEK should be installed unless you do not run ISPF in your environment.

Note: For an overview of the EDCHEK features, see the *Programming Guide*.

1. Modify an EDCHEK control statement file for your installation, you will find examples in CAI.CAZ2OPTN. For EDCHEK, LIST is the standard default runtime option. If report generation is not preferred, you should add NOL (NOLIST) to the EDCHEK control file.

Member in CAZ2OPTN	Function
CAZ1SEDP	Control statement file without AUTOPROC
CAZ1SEDA	Control statement file with AUTOPROC
CAZ1SEDL	Control statement file using report files
CAZ1SEDT	Control statement file with CA TLMS but NOT AUTOPROC
CAZ1SEDM	Control statement file with CA TLMS and AUTOPROC

2. Add the EDCHEK control statement file to the LOGON procedures of all TSO users who intend to use EDCHEK, reflecting the library and member name defined on the worksheet. You can also add an ALLOCATE in your TSO allocation CLIST.

   For example:

   ```
   //EDCHKDD DSN=CAI.CAZ2OPTN(CAZ1SEDP),DISP=SHR
   ```

3. If you use report files as in sample CAZ1SEDL, allocate a list file that matches the LIST option specified. An example CLIST called CAZ1SEDC is in CAZ2OPTN.

4. Place the CAZ1JCKM and JCLCHECK program modules with their aliases in a library that is accessible to ISPF using the LINK macro instruction. You can use member AJ6CAPF to copy these modules if you do not want to make CAZ2LOAD accessible to ISPF. Such libraries include:

   - SYS1.LINKLIB or another library in the linklist (as recommended).
   - A library that is part of the STEPLIB allocation in your TSO LOGON procedures.

     Important! CA JCLCheck load modules have numerous ALIAS names. Do not attempt to copy CA JCLCheck load modules from one library to another using an online utility. Instead, if you must copy the load modules, always tailor, and then run the CAZ2JCL member AJ6CAPF in batch. If you do not copy the base module with all of its aliases in one operation, you create multiple load modules, which cause execution and maintenance problems.

   If you are upgrading from a prior version, review the description of the ERROR option in the *Command Reference Guide*.

(Optional) Install ChekPlex

The ChekPlex feature enables you to issue JCL validation requests between systems that XCF or TCP/IP connects. ChekPlex uses the CA General Transaction Server (GTS) to route JCL and procedures to a remote system for validation. The results are returned to the original local system.

ChekPlex is available through the CA JCLCheck ISPF panels, by using ISPF edit with the CA JCLCheck EDCHEK feature, or a CA JCLCheck batch job.

SUBCHEK cannot use the ChekPlex feature.

ChekPlex uses CA GTS to communicate to other systems and requires additional steps be taken to configure the network architecture of CA GTS.

No additional steps are required at this time. Complete the CA JCLCheck installation and verify the basic product operation before configuring ChekPlex for your environment.

For information about setting up and using the ChekPlex feature, see Installation Options in the *Programming Guide.*

(Optional) Install CA APCDOC Interface

This interface lets CA JCLCheck read or update the CA APCDOC Cross-Reference database.

Starting with CA APCDOC Version 1.3, database allocations are handled using the DBHLQ option. This option is explained in the *CA APCDOC Installation Guide* and is a modification to the ISPF Primary panel. This modification provides a user interface to CA APCDOC using online ISPF panel functions.

For CA APCDOC Version 1.3, contact CA Support for a list of APCDOC PTFs required when used with CA JCLCheck Version 11.

(Optional) Install CA JCLCheck/CA Roscoe Monitor

To install CA JCLCheck/CA Roscoe monitor support:

1. Transfer JCLCHECK to an authorized load library accessible to CA Roscoe using AJ6CAPF in CAZ2JCL as a model. All CA Roscoe monitor routines must run authorized.
2. Add two DD statements to the online step in the CA Roscoe procedure:

   JCHKPLIB

   Use this statement to point to your default JCL procedure libraries (similar to the SYSPROC DD for CA JCLCheck batch execution). The AUTOPROC option is available in CA Roscoe Version 5.7 and above.

   JCHKDUMP

   Use this SYSOUT data set for diagnostic output if the monitor fails.

   Note: If you want CA TLMS support, add an allocation for the Volume Master File (VMF). The ddname for this file is CAIVMFI.
3. Add the following control statement to the CA Roscoe SYSIN data set:

   ```
   RUN=(JCK(S))
   ```

   The preceding control statement lets CA Roscoe translate any lowercase JCL characters to uppercase during processing. This translation prohibits JCLCHECK from detecting and reporting lowercase entries as JCL errors.

   You need a second option on this control statement to instruct CA Roscoe not to translate any lowercase JCL characters to uppercase. To specify this option, change the control statement to the following:

   ```
   RUN=(JCK(SX))
   ```

(Optional) Install RPF Programs

Important! You must APPLY SYSMOD CAZ1C01.

Note: We recommend that you perform the following steps to a CA Roscoe test account first. After testing the RPFs, repeat the same steps to the execution account to make it available to the rest of the user community.

To install the RPF program:

1. One of the RPFs uses the JCK monitor. If you have not already installed the monitor, complete Install CA JCLCheck/CA Roscoe Monitor (see page 65).

   Note: Authorize CAI.CAZ2LOAD properly and specify it in the CA Roscoe STEPLIB concatenation to run the monitor.

2. RPF installation requires ETSO for JCL reformatting and for full reporting of scanned JCL; therefore, you need to define JCLCHECK and JCLNeat in the Eligible Program List (EPL), pointed to by the CA Roscoe initialization parameter ETSOPGMS. (Typically, this member is RO.ETSOPGMS.) Add the following lines to the ETSOPGMS member (in ascending sequence):

   ```
   JCLCHECK 010 5000 001024 001024 001024 001024 N Y
   JCLNeat  010 5000 001024 001024 001024 001024 N N
   ```

   The program name starts in column 1, the 010 in column 10. (These lines are displayed in version 1 EPL format.)

   Note: These values are suggested values and should be customized based on the site requirements. For more information, see the *CA Roscoe Systems Programmer Guide*.

3. If your CA Roscoe ETSO initialization parameter, ETSSRCH, uses its default value or is set to ETSSRCH=DD, you need to define CAI.CAZ2LOAD to the //ETSOLIB DD statement in the JCL you use to execute CA Roscoe. ETSSRCH=ALL finds the programs it needs in the STEPLIB concatenation.

4. Sign on to a CA Roscoe account (preferably the Execution library). The Execution library is defined by the EXECPFX RPF Initialization Parameter defined in the JCL you use to execute CA Roscoe.

5. SYSMOD CAZ1C01 creates a target CAZ2RPF containing RPF programs. Import the member JCKINST from the target CAZ2RPF into the CA Roscoe account. Save the RPFs into the CA Roscoe library as JCKINST.

6. Browse through the JCKINST RPF. This RPF creates or replaces members in whatever CA Roscoe account you are using. You want to be sure that you do not inadvertently destroy some identically named member. You can do this by issuing the command INCL 'UPD'.

7. Execute the JCKINST RPF and answer the questions. When asked to provide the DSN of the CAIPDS containing RPFs, specify the name of the target CAZ2RPF that contains the RPF source programs.

8. After JCKINST RPF completes, execute the RPF JCKRPF. JCKRPF is the main execution procedure you use to check JCL for runtime errors. Be sure to do the Batch Model option. If you are executing from the account assigned as the Execution library, you will set the batch model for your entire site. Continue testing JCKRPF.
9. Execute the RPF JCKNRPF. JCKNRPF is the main procedure for reformatting JCL. If you are executing from the Execution library account, you can set Default Options for the entire shop. Be careful not to choose the Hide Panel option from the Execution library account.

(Optional) Install CA JCLCheck/CA Roscoe ISPF Support

Parts of CA JCLCheck were developed to execute solely as an ISPF/PDF dialog, (for example, job control standards). If you must run an ISPF dialog, run it under CA Roscoe by running ISPF as an ETSO task. You can do this by using the procedures listed following. We recommend that you use a test CA Roscoe region until you are certain that you are able to execute CA JCLCheck successfully. When you are ready to install CA JCLCheck into production, make the same changes and use the same procedure as you used in the test region.

CA JCLCheck only recognizes valid TSO user IDs (maximum of seven characters) that the ISPF dialogs use. If you use ISPF under CA Roscoe already, you may have valid TSO user IDs, but if you do not have valid user IDs, you have to provide programming that converts 22-character CA Roscoe user IDs to seven-character user IDs used by ISPF dialogs. When the first seven characters of the current CA Roscoe account codes are acceptable, you can use the macro CAZ1$ZVA as follows:

1. Obtain the source for CLLEXIT, provided on the CA Roscoe installation tape.
2. Find the assembler tag 'TSOCPPL'.
3. Place a line invoking the macro CAZ1$ZVA immediately after this tag.
4. Assemble and link the CLLEXIT as instructed by CA Roscoe.
5. Transfer CLLEXIT to a library accessible to CA Roscoe.

The CAZ1$ZVA macro uses the current account number to build the TSO UPTPREFX (&ZPREFIX) and TSO PSCBUSER (&ZUSER) fields before starting the ISPF dialog. If you do not modify these fields, ETSO derives a user ID consisting of the user's CA Roscoe prefix concatenated with the character string 'ETSO'. This derived user ID is also used as the first DSN level when ISPF attempts to dynamically allocate files.

As per CA Roscoe instructions, assemble the module ETSSUB and link it into a library accessible to CA Roscoe. The load module must be given the name of SUB.

ETSO is required to run ISPF/PDF under CA Roscoe. For a CA JCLCheck dialog to start, define ISPF in the Eligible Program List (EPL). The EPL is found in the control account, in a member (usually RO.ETSOPGMS) in the CA Roscoe initialization parameter with the name ETSOPGMS. Add the following two lines to the ETSOPGMS member (in ascending sequence):

```
ISPF     010 1550 000512 000350 005000 003000 D N CP  ISPF/PDF
ISPSTART 010 1550 000512 000350 005000 003000 N N CP  ISPF/PDF
```

The program name starts in column 1, the 010 in column 10. (These lines are displayed in Version 2 EPL format.)

If your CA Roscoe ETSO initialization parameter ETSSRCH uses the default value or is set to ETSSRCH=DD, you need to define CAI.CAZ2LOAD to the //ETSOLIB DD statement in the JCL used to execute CA Roscoe. ETSSRCH=ALL finds the programs needed in the STEPLIB concatenation.

Import the member JCKRISPF from the library CAZ2RPF into the CA Roscoe account. Modify this RPF to properly allocate the files and libraries needed for the CA JCLCheck dialog. Save this member in the execution account as JCKRISPF. At this point, you may want to add a line to execute the CA JCLCheck RPF into an existing menu RPF, if you have one. You may also want to move this RPF to the execution library after testing. The execution library is defined by the EXECPFX CA Roscoe initialization parameter.

Modify the ISR@PRIM panel to invoke CA JCLCheck.

You can execute the RPF JCKRISPF at this point. This RPF starts an ISPF dialog that can invoke CA JCLCheck.

From here on, there are only slight differences in the CA JCLCheck dialog, otherwise it is identical to the dialog run under TSO/E.

The differences are:

- The way the attention key responds.
- The way you clear errors (by using the Clear key under CA Roscoe).
- ROSC is an internal option indicating the CA Roscoe environment.

Tailor the CA Common Services for z/OS Initialization Procedure

The CA Technologies initialization procedure, CAS9, is required for most CA Technologies mainframe products including CA JCLCheck.

Submit CAS9 after every IPL to execute the module CAIRIM, which sets up the running environment for those CA Technologies products that require it. To automate this start-up of the initialization procedure, add a START command for CAIRIM to the appropriate COMMNDxx member in SYS1.PARMLIB. For example, COM='START CAS9'. The execution of this command occurs automatically after JES2 is started.

CAIRIM accesses data sets for information that each product requires. The data set ddnames are:

AUTOCMDS

Contains the commands that CAIRIM automatically issues during start-up of a product.

PARMLIB

Contains information that is passed to a product during start-up.

KEYS

Contains the execution keys that each product requires for license verification and management.

Note: For more information about CAIRIM, see CAIRIM (see page 17).

1. Find the CAS9 procedure in the appropriate system PROCLIB.
2. Identify the common CAW0OPTN data sets that AUTOCMDS DD, the KEYS DD, and the RIMPARM DD use.
3. Identify the names of the members that are referenced in each DD.
4. Insert the data that CA JCLCheck requires into the CAIRIM CAW0OPTN data sets.
5. Merge the CA JCLCheck CAZ2OPTN member Z1LIST into the appropriate RIMPARM PARMLIB member.

   CA JCLCheck does not use AUTOCMDS, but optionally for documentation a member with at least one comment line (containing an asterisk '*' in column 1) can be created.
6. Supply your LMP code and add it to the KEYS member exactly as printed on the License Management Program (LMP) certificate accompanying CA JCLCheck.

(Optional) Install Security Interface

You can run the CA JCLCheck security interface with CA ACF2, CA Top Secret, or any SAF-compatible products.

CA JCLCheck must run authorized for the SECURITY option. Make the necessary changes as documented elsewhere in this guide, particularly Establish CA JCLCheck as APF-Authorized (see page 54).

If you have other CA Technologies products, then you may already have CAISSF implemented. If not, perform the following steps.

To support the security interface, initialize CAISSF.

Follow these steps:

1. Initialize CAISSF (CA Top Secret and RACF).
2. Install the CAISSF SAF interface.

   For details on customizing the CAISSF step, see the *CA Common Services for z/OS Getting Started.*
3. Add the CAISSF CAIRIM input initialization statement to the member pointed to by ddname PARMLIB in the CAS9 procedure.

   The default value for this member is CARIMPRM in the CA Common Services CAW0OPTN. This step is documented in the *CA Common Services for z/OS Getting Started,* in the topic Optional Placement of the CAISSF Routines in CSA.

This statement is as follows for service level A5 of CA Common Services for z/OS:

```
PRODUCT(CAIRIM) VERSION(CAS9) INIT(CAS9INIT)
```

Note: For any changes, see the documentation *CA Common Services for z/OS*.

(Optional) Install Support for Job Control Standards

If you are not currently using the Job Control Standards (JCS) table, CA Technologies recommends that you use a STDREXX REXX program instead to control standards. STDREXX provides more robust validation of the JCL standards than the JCS tables allows.

This procedure describes how to install Job Control Standard ISPF dialog support.

Follow these steps:

1. Modify the CAZ2PROC member for batch invocation.

   Edit member CAZ1JCHK in data set CAI.CAZ2PROC, remove the comment, and change the STDRULE DD data set name to reflect your installation default library.

   You do not need to create a standards database, when you are upgrading from prior versions of CA JCLCheck and one exists. A conversion is not required. If you do not have a standards database, member AZ1SSTD in the CAZ2JCL library allocates and initializes the CA JCLCheck/STANDARDS database. Edit this member to conform to your system standards. Tailor the IDCAMS control statements and the JCL PROC variables to coincide with each other.

2. Submit the job and review the output to verify that the job ran successfully.
3. Add an allocation for the rules database defined in Step 1 to the TSO sessions of those users authorized to access the rules database. You can accomplish this task by adding an STDRULE DD statement to the TSO logon procedure, changing the allocate command of the JCKSTD CLIST, or by adding a TSO allocate command to a customized CLIST. The following sample is of the CLIST command:

   ```
   ALLOC DD(STDRULE) DS('RULE.DATABASE.NAME') SHR
   ```

(Optional) Install Support for the REXX Interface

This procedure describes how to install REXX Interface support in the following environments: Batch Invocation and TSO-ISPF Invocation.

Batch Invocation

Follow these steps:

1. Modify the CAZ2PROC member to use the STDREXX JCLCheck Option.
2. Edit member CAZ1JCHK in data set CAI.CAZ2PROC:

   - Remove the comment, and change the SYSEXEC DD data set name to reflect your installation default library where the REXX EXECs are stored.

3. Edit member CAZ1NEAT in data set CAI.CAZ2PROC if you are using the REXXMEM option in JCLNeat:
   - Remove the comment, and change the SYSEXEC DD data set name to reflect your installation default library where the REXX EXECs are stored.

TSO-ISPF Invocation

Follow these steps:

1. Tailor the skeleton JCL for CA JCLCheck ISPF dialog.
2. Edit members JCKSKEL, CAZ1SFJC, and CAZ17FJC in data set CAI.CAZ2SKL0. You may want to move these to a non-SMPE library before changing.
   - Remove the comment, and change the SYSEXEC DD data set name to reflect your installation default library where the REXX EXECs are stored.
3. Update the TSO logon Procedure or logon CLIST:
   - Add the allocation of the SYSEXEC ddname that contains the CA JCLCheck or JCLNeat REXX EXECs to the TSO logon procedure.

   If your installation uses a logon CLIST, add the SYSEXEC DD allocation to that CLIST.

   Note: If your installation uses only SYSPROC to store REXX EXECs, add the REXX members to that library.

(Optional) Install Support for the DB2 Interface

To install DB2 interface support, you must be at DB2 Version 2 or above.

The interface is dynamic and uses the standard DB2 Call Attach Facility. CA JCLCheck provides DB2 support using plan name CAZ2JCAF (supplied in the CAZ2DBRM library).

This procedure describes how to install DB2 Interface support.

Follow these steps:

1. Update the JCL—Depending on how DB2 was installed at your site, it can be necessary to add the DB2 DSNLOAD load library to the CA JCLCheck STEPLIB DD. Edit member CAZ1JCHK in CAI.CAZ2PROC and add the name of your DB2 DSNLOAD to the STEPLIB DD.

   Note: If DB2 is installed into the linklist or LPA, you do not have to update the JCL.

2. Change the TSO logon procedure to include DD statements for the DB2 DSNLOAD libraries in the STEPLIB DD concatenations if these libraries are not in the linklist or in LPA.

   This step is required if CA JCLCheck is used under TSO, ISPF, EDCHEK, or SUBCHEK.

3. Bind the Plan—Issue the following DB2 command through batch using IKJEFT01 or online under SPUFI. Operands in square brackets are optional. Do not specify the square brackets in the BIND commands you submit. Specify the appropriate values for the operand values shown in lowercase.

   ```
   DSN SYSTEM(your_db2_subsys)
   BIND PACKAGE(CAZ2JCAF) [OWNER(owner)] -
        MEMBER(CAZ2JCAF) -
        LIBRARY('cai.CAZ2DBRM') -
        SQLERROR(NOPACKAGE) VALIDATE(BIND) ISOLATION(CS) -
        RELEASE(COMMIT) EXPLAIN(NO) ACTION(REPLACE) CURRENTDATA(NO)
   BIND PLAN(CAZ2JCAF) [OWNER(owner)] -
        PKLIST(CAZ2JCAF.CAZ2JCAF) -
        VALIDATE(BIND) ISOLATION(CS) [CACHESIZE(cachesize)] -
        RELEASE(COMMIT) ACQUIRE(USE) [ENCODING(encoding)] -
        EXPLAIN(NO) ACTION(REPLACE) RETAIN CURRENTDATA(NO)
   END
   ```

   CAZ2JCAF is the CA JCLCheck default plan name. If you want to have multiple CA JCLCheck releases communicating with a single DB2 subsystem, a unique plan name is required for each CA JCLCheck release. The suggested unique plan name for r12.0 is CAZ2J6C0; however, you can use any plan name as site standards require.

4. Grant Access—Issue the following DB2 commands through either batch SPUFI or online SPUFI:

   ```
   GRANT SELECT ON TABLE SYSIBM.SYSPLANDEP TO CAZ2JCAF
   GRANT SELECT ON TABLE SYSIBM.SYSVIEWDEP TO CAZ2JCAF
   GRANT EXECUTE ON PLAN CAZ2JCAF TO PUBLIC
   ```

   Note: You can change the GRANT EXECUTE on plan CAZ2JCAF from PUBLIC to a list of one or more specific authorized users. See the *IBM publication SC6-4374* or the *IBM Database 2 Administration Guide WC II*.

   Once these steps are complete, you can specify the DB2 option to request interface support. The default of the DB2 option is to use either the subsystem ID coded on DB2 control cards or the first DB2 subsystem found on the system. You can change the DB2 subsystem using the DB2 option with the SSID parameter where the ID is a one to four character DB2 subsystem identifier.

   Note: For information about the DB2 runtime option, see the *Command Reference Guide*.

5. DB2 V7 introduces the ENCODING sub-parameter in the BIND command. The ENCODING specification can be a numeric CCSID. By default, CCSID 37 (U.S. English) will be valid. If other CCSIDs are required, install the following USERMOD (MZ2C044).

   Examples:

   a. Support U.S. English CCSID 37 and Germany CCSID 273

   ```
   $J6PCB 37      CCSID - US ENGLISH
   $J6PCB 273     CCSID - GERMANY
   ```

   b. Support Germany CCSID 273 Only

   ```
   $J6PCB 273     CCSID - Germany
   ```

Verify the Installation

Member AZ1JCHK in CAZ2JCL invokes CA JCLCheck in batch. Use sample JCL containing errors such as those in member AZ1SAMP in CAZ2JCL to demonstrate the basic functions of JCL validation.

Member AZ1NEAT in CAZ2JCL is supplied to invoke JCLNeat in batch. Use sample JCL in member AZ1NDTI in CAZ2JCL to test the installation. The reformatted output is placed in CAZ1NDTO of CAZ2JCL.

Do the following:

1. Modify job AZ1JCHK.
2. Execute job AZ1JCHK.
3. Modify job AZ1NEAT.
4. Execute job AZ1NEAT.

(Optional) Install CA TLMS and CA 1 Support

The following topics discuss installation considerations for CA 1 and CA TLMS:

CA 1

If you are sharing the same SMP CSI Target zone with CA JCLCheck and CA 1:

CAZ2JCL Procedure	Description
UZ1C002	Apply this UCLIN to the SMP Target Zone used to install CA 1.
MZ1C048	Apply during the install of CA JCLCheck.

If you are not sharing the same SMP CSI Target zone with CA JCLCheck and CA 1:

CAZ2JCL Procedure	Description
MZ1C048	Apply during the install of CA JCLCheck.
MZ1C048	Apply with the SMP REDO option each time CA 1 maintenance is applied.

UZ1C002 for CA JCLCheck supports CA 1 r5.2, r11, and higher.

The SMP APPLY process for MZ1C048 will install CAZ1TMS5 into the CAZ2LOAD for CA JCLCheck. If you are planning on running CA JCLCheck from another library, you must copy CAZ1TMS5 to the other library.

Note: For more information on CA 1 support, see the TMS option in the *Command Reference Guide*.

CA TLMS

If you are sharing the same SMP CSI Target zone with CA JCLCheck and CA TLMS:

CAZ2JCL Procedure	Description
UZ1C003	Apply this UCLIN to the SMP Target Zone used to install CA TLMS.
MZ1C047	Apply during the install of CA JCLCheck.

If you are not sharing the same SMP CSI Target zone with CA JCLCheck and CA TLMS:

CAZ2JCL Procedure	Description
MZ1C047	Apply during the install of CA JCLCheck.
MZ1C047	Apply with the SMP REDO option each time CA TLMS maintenance is applied.

UZ1C003 for CA JCLCheck supports CA TLMS r5.5, r11, and higher. CA JCLCheck no longer supports any releases of CA TLMS before r5.5.

The SMP APPLY process for MZ1C047 installs CAZ1TLMS into the CAZ2LOAD for CA JCLCheck. If you are planning on running CA JCLCheck from another library, copy CAZ1TLMS to the other library.

Note: For more information on CA TLMS support, see the TLMS option in the Command Reference Guide.

(Optional) Install User Exits

Important! We recommend that you install the user exits when you finish installing CA JCLCheck. When installing a new version, we recommend that you reassemble and compile all exits to access all revised macros.

Make sure that you execute member AJ6CAPF to replace the user exits with the updated copy after you install CA JCLCheck.

The following table contains all the exits that you can modify. The code, as supplied, contains examples of how to use the exit but each exit returns to its caller immediately. This behavior is new in CA JCLCheck Version 12.0. In order for the exit to execute a function, you are required to change the code.

Install these user exits only with careful review.

CAZ2JCL Member Name	Source in CAZ2SRC	Function
MZ1C032	CAZ1XJOB **	Install JOB exit support.
MZ1C031	CAZ1XEXE **	Install EXEC exit support.
MZ1C027	CAZ1XDD **	Install DD exit support.
MZ1C035	CAZ1XSTE **	Install STEP END exit support.
MZ1C028	CAZ1XEOF **	Install EOF exit support.
MZ1C030	CAZ1XERR **	Install ERROR exit support.
MZ1C033	CAZ1XPAR	Install PARAMETER exit support.
MZ1C029	CAZ1XEOJ **	Install End of JOB exit support.
AZ2XCOB *	CAZ2XCOB ***	Install COBOL exit support.
AZ2XCB2 *	CAZ2XCOB ***	Install COBOL II exit support.
AZ2XCEE *	CAZ2XCOB ***	Install COBOL II exit support in LE/370 environment.
MZ1C023	CAZ1SSFX	Install SIGNON exit for the security interface.
MZ1C043	CAZ1XSEC **	Install exit to bypass existence checking for selected DSNs.
MZ1C026	CAZ1XCTL **	Install CONTROL CARD exit.
MZ1C036	CAZ1XGET **	Install SYSIN input record support

* One asterisk specifies the CAZ2JCL member is in non-SMP format due to a requirement for a COBOL compile.

** For information about activating these exits, see the DYNEXITS parameter option in the Command Reference Guide.

*** Three asterisk specifies the source member resides in CAZ2SAMP, not in CAZ2SRC.

Note: The parameter field has been lengthened, see User Exits in the *Programming Guide*.

Important! Before you execute any of the CAZ2JCL members, move the "./ CHANGE" and "./ END IEBUPDTE commands to column 1 to avoid a JCL error. This move is documented in the CAZ2JCL members.

(Optional) Install CA JCLCheck Tables for Modification

Since you can implement this step after installation, CA JCLCheck and its aliases may already exist in other libraries. Make sure that you execute member AJ6CAPF in CAZ2JCL to replace them with the updated copy.

This list contains all the tables that can be modified. These tables are optional and only need to be installed or modified if you are using that feature.

To install CA JCLCheck tables for modification, do the following:

1. View the source code in CAZ2SRC for members that you can modify.
2. Modify the source code.

   Important! You must ensure that the editor you use to tailor these members does not change the line number. For ISPF, set NUM OFF. Future maintenance (in the form of IEBUPDTE statements) does not apply correctly if the integrity of the sequence numbers is not preserved.

3. Execute the member in CAZ2JCL to assemble and link the module to CA JCLCheck. You can do this under SMP control or outside of it.

   Important! If you have modified any of the user-customizable tables, you must ensure that the appropriate jobs are executed to re-assemble these tables. Changes may have been made to internal macros at release levels. Failure to re-assemble the tables may result in improper operation and abnormal termination of CA JCLCheck.

CAZ2JCL Member Name	Source in CAZ2SRC	Function
MZ1C024	JCLDFLT *	Modify the default CA JCLCheck execution options. Important! It is a best practice to perform this customization to ensure the default options are consistent across all environments.
MZ2C012	MV4XTBL *	Install support for z/OS and MVS (Version 4 and above).
MZ2C013	CAZ2DTBL *	Install destination support.
MZ2C014	CAZ2JAC2 *	Install CA ACF2 JCL support.
MZ2C019	CAZ2JAJ3 *	Install JES3 and CA ACF2 JCL support.
MZ2C020	JOBCARD *	Modify default JOBCARD.
MZ2C021	CAZ2J3TB *	Install JES3 support.
MZ2C023	JCLMSG *	CA JCLCheck messages.
MZ2C024	SSCTLTBL	Install JES2 subsystem JCL support.
MZ2C025	JCLUPT *	Support Utility Program.
MZ2C042	CAZ2JAPT *	Install IMS, CA APCDDS, and alternate program support.
MZ2C044	JCLUTIL *	Support for modifications to utility verbs for Utility Program Table.
MZ2C045	CAZ2UTAB*	Unit device equivalency for runtime checking.
MZ2C049	CAZ2JOEL *	CA JCLCheck Option Exclusion List

Note: If you want CA JCLCheck to update the CA APCDOC Cross-Reference database, select the XREFAL option.

* These modules are already included in the base product and you only need to assemble and link them to modify them.

CAZ2UTAB is used to determine non-DASD device types for the DD statement UNIT=keyword when the NORUNTIM option is used. This prevents message CAY6083E (SPACE PARAMETER NOT PROVIDED FOR NEW DIRECT ACCESS DATASET) from being issued for non-DASD devices.

NORUNTIM is used by the CA JCLCheck common component or by specification using the full product.

To add or remove unit names from this list, apply the SMP USERMOD MZ2C045.

Important! Before you execute any of the CAZ2JCL members, move the "./ CHANGE" and "./ END IEBUPDTE commands to column 1 to avoid a JCL error. This move is documented in the CAZ2JCL members.

(Optional) Install JCLNeat Tables for Modification

You can implement this step after installation.

Perform the following steps when modifying the following modules:

1. View the source code in CAZ2SRC for members you want to modify.
2. Modify the source code.
3. Execute the member in CAZ2JCL to assemble and link the module to JCLCHECK. You can do this under SMP control or outside of it.

This list contains all the tables that you modify for JCLNeat. Each table is a stand-alone module.

CAZ2JCL Member Name	Source in CAZ2SRC	Function
MZ1C010	CAZ1NCCM	Modify operator command support.
MZ1C011	CAZ1NCCN	Modify default keyword order for PRINTCNTL statement.
MZ1C012	CAZ1NCDC	Modify default keyword order for DCB parameter.
MZ1C013	CAZ1NCDD	Modify default keyword order for DD statement.

CAZ2JCL Member Name	Source in CAZ2SRC	Function
MZ1C014	CAZ1NCEX	Modify default keyword order for EXEC statement.
MZ1C015	CAZ1NCJB	Modify default keyword order for JOB statement.
MZ1C016	CAZ1NCOU	Modify default keyword order for OUTPUT statement.
MZ1C017	CAZ1NCPD	Modify default keyword order for PRINTDEV statement.
MZ1C018	CAZ1NCPR	Modify default keyword order for PROC statement.
MZ1C019	CAZ1NCJ3	Modify JES3 command support.
MZ1C020	CAZ1NDEF	Modify default option table for JCLNeat.

Important! Before you execute any of the CAZ2JCL members, move the "./ CHANGE" and "./ END IEBUPDTE commands to column 1 to avoid a JCL error. This move is documented in the CAZ2JCL members.

Save Installation Materials

Be sure to save all of your installation materials and all output from the installation process. This material is essential for CA Technologies to provide timely, accurate maintenance support of the product.

Post-Installation Considerations

Now that you have successfully installed CA JCLCheck, consider reviewing these guides:

- For information on how to use CA JCLCheck, see the *Tutorial*.
- For information on customizing CA JCLCheck, see the *Programming Guide*.
- For a summary of the options available for CA JCLCheck, see the *Command Reference Guide*.

Chapter 6: Migration Information

This section contains the following topics:

Migration Considerations

When migrating to CA JCLCheck Version 12.0, there are a few items to review.

- The JCKUSR CLIST has new variables that require specification. CA JCLCheck uses this CLIST to define data sets and options that are run under ISPF. Move in the updated version and specify the information requested in the CLIST.
- Review the new capabilities that are available in Version 12, if you are using REXX for standards checking or JCLNEAT for JCL formatting. In JCLNeat, there are many new statement calls available and some functions that previously could only be performed in RAW processing can now be performed with ease in statement processing. Additionally, a change in the CA JCLCheck Standards REXX process improves the information available to your REXX program.
- In Version 11, MZ2B051 could be used to add limited support for SORT control cards. This requirement is no longer needed in Version 12, since the support for SORT is included in the base product.
- CA Technologies recommends that you reassemble and compile all exits to access all revised macros. This process is applicable if you have existing exits that are installed for CA JCLCheck.

Appendix A: Preparation Worksheets

You can print out the worksheets in this section to record the values that are needed for your site when installing the product.

Non-MSM Customization Worksheet

The CA JCLCheck non-MSM customization worksheet helps you define the requirements for your site. This worksheet presents all of the choices you have during the customization phase of the installation. We recommend that you review the *Programming Guide* before answering these questions.

(Optional) Install parameter CSECT modifications

```
1. Do you need support for CSECT modification?          Y/N
```

Establish CA JCLCheck as APF-authorized

```
1. Do you want to establish CA JCLCheck as
   APF-authorized?                                      Y/N

2. CA JCLCheck must reside in authorized
   library for options:

   Do you want to install AUTOPROC option
   (automatic PROCLIB recognition)?                     Y/N

   Do you want to install CA ASM2 support?              Y/N

   Are you using DESTCHK(J) with JES3 or JES2?          Y/N

   Are you using the SAF security interface?            Y/N

   Are you using HCD?                                   Y/N

3. Authorization Type-1 used:

   What is the name of the APF library containing
   <product name> modules?                              DSN=_____________
```

```
4. Authorization Type-2 used:

   What is the name of the linklist library
   with CA JCLCheck modules?                          DSN=_____________

5. Authorization Type-3 used:

   Do you want to add CAI.CAZ2LOAD library
   to the APF list (IEAPFxx member)?                  Y/N

6. Authorization Type-4 used:

   Do you want to add CAI.CAZ2LOAD library to LNK
   list (LNKLSTxx)?                                   Y/N

   Name of linklist library containing
   LNKLSTxx member.                                   DSN=_____________

   Is the linklist library in the system APF list?    Y/N

   What is the name of your system APF list?          DSN=_____________
```

Install ISPF support

```
1. Do you need ISPF support?                          Y/N
```

Install JCLNeat ISPF interface

```
1. Do you need JCLNeat ISPF support?                  Y/N
```

Install TSO support

```
1. Do you need TSO support?                           Y/N

2. Do you want native TSO support?                    Y/N

3. Do you want ISPF TSO support?                      Y/N

   Identify Authorized Security Tables used for
   TSO commands and TSO service facilities.

   Table: IKJEFTE2                                    Table name_____________

   Table: IKJEFTE8                                    Table name_____________

   Table: IKJEFTAP                                    Table name_____________

   Table: IKJTSOxx in SYS1.PARMLIB                    Table name_____________
```

(Optional) Install SUBCHECK

```
1. Do you need SUBCHEK support?                    Y/N
```

(Optional) Install EDCHEK

```
1. Do you need EDCHEK support?                     Y/N
```

(Optional) Install ChekPlex

```
1. Do you need ChekPlex support?                   Y/N
```

(Optional) Install CA APCDOC interface

```
1. Do you need CA APCDOC support?                  Y/N

2. What is the name of your CA APCDOC
   installation database?                          DSN=_____________
```

(Optional) Install CA JCLCheck CA Roscoe monitor support

```
1. Do you need CA JCLCheck/CA Roscoe
   monitor support?                                Y/N
```

(Optional) Install RPF programs

```
1. Do you need RPF programs support?               Y/N
```

(Optional) Install CA JCLCheck CA Roscoe ISPF support

```
1. Do you need CA JCLCheck/CA Roscoe
   ISPF support?                                   Y/N

2. What is the name of your CA Roscoe
   product macro library?                          DSN=_____________

3. What is the name of your product
   ONLINE procedure?                               PROC=____________

4. Is the CA Roscoe product version
   prior to 5.7?                                   Y/N

5. Do you want to install the RPF members
   available for CA Roscoe support?                Y/N

6. Do you need OPC/ESA support?                    Y/N

   What is the name of your OPC/ESA
   DSN used in CAZ1OPC?                            DSN=_____________
```

Tailor CA Common Services for z/OS initialization procedure

```
1. Do you want to install product component for
   linklist support?                                    Y/N

2. Obtain LMP information from the key certificate supplied with
   this product. The following data must be available to
   install CA JCLCheck:

                                                Execution key_____________

                                                Support code______________

                                                Expiration date___________
```

(Optional) Install security interface

```
1. Do you need security support?                        Y/N
```

(Optional) Install support for job control standards

```
1. Do you need job control standards support?           Y/N
```

(Optional) Install support for the REXX interface

```
1. Do you need REXX support?                            Y/N

2. What is the name of your REXX EXEC library
   for standards processing?                            DSN=_____________
```

(Optional) Install support for the DB2 interface

```
1. Do you need DB2 support?                             Y/N
```

(Optional) Install user exits

```
1. Do you want to install user exits support?           Y/N

   Install SIGNON exit modification?                    Y/N

   Install CONTROL CARD exit modification?              Y/N
```

```
   Install DD exit modification?                          Y/N

   Install EOF exit modification?                         Y/N

   Install END-OF-JOB exit modification?                  Y/N

   Install ERROR exit modification?                       Y/N

   Install EXEC exit modification?                        Y/N

   Install JOB exit modification?                         Y/N

   Install PARAMETER exit modification?                   Y/N

   Install STEP END exit modification?                    Y/N

   Install CHECK BYPASS exit?                             Y/N

   Install job statement?                                 Y/N

   Install CA APCDOC element selection exit?              Y/N

   Install COBOL exit modification?                       Y/N

   Install COBOL II exit modification?                    Y/N
```

(Optional) Install CA JCLCheck tables for modification

```
1. For MVS/ESA Version 4 and 5, OS/390 and Z/OS
   with or without SMS;
   Do you want to modify JCLTABLE: MVS4XTBL?            Y/N

2. Do you want to install destination support?          Y/N

3. Do you want to install CA ACF2
   JCL support?                                         Y/N

4. Do you want to install CA ACF2 JCL
   and JES3 support?                                    Y/N

5. Do you want to modify the default
   JOB statement generation?                            Y/N

6. Do you want to install JES3 support?                 Y/N
```

```
7. Do you want to modify any of the
    CA JCLCheck messages?                                  Y/N

8. Do you want to install JES2 subsystem
    JCL support?                                           Y/N

9. Do you want to support the Utility Program?             Y/N

10. Do you want to install support for IMS,
    CA APCDDS, Alias Tables or Alternate
    Programs?                                              Y/N

11. Do you want support for modifications to
    Utility verbs for the Utility program table? Y/N

12. Do you want support for unit device
    equivalency for runtime checking?                      Y/N

13. Do you want to modify CA JCLCheck
    execution options?                                     Y/N
```

(Optional) Install JCLNeat tables for modification

```
1. Do you want to modify operator commands to the
   operator command table?                                 Y/N

2. Do you want to modify the default keyword order
   for the PRINTCNTL statement?                            Y/N

3. Do you want to modify the default keyword order
   for the DCB statement?                                  Y/N

4. Do you want to modify the default keyword order
   for the DD statement?                                   Y/N

5. Do you want to modify the default keyword order
   for the EXEC statement?                                 Y/N

6. Do you want to modify the default keyword order
   for the JOB statement?                                  Y/N

7. Do you want to modify the default keyword order
   for the OUTPUT statement?                               Y/N

8. Do you want to modify the default keyword order
   for the PRINTDEV statement?                             Y/N
```

9. Do you want to modify the default keyword order
   for the PROC statement? Y/N

10. Do you want to define additional JES3
    commands to the JES3 command table? Y/N

11. Do you want to modify the default option
    table for JCLNeat? Y/N