opentext™

ChangeMan ZMF

Migration Guide

8.3

Table of Contents

1. About this Guide

This guide provides guidance for upgrading ChangeMan ZMF.

Audience and Scope

This manual is intended for system administrators and systems programmers responsible for installing, administering, and maintaining ChangeMan ZMF.

Navigating this Book

This manual is organized as follows:

- Chapter 1 Contains information about this guide.
- Chapter 2 describes issues you must consider. It presents possible strategies you can employ to arrive at a migration plan that fits your circumstances. It describes a model high-level plan and presents a sample conversion-day script.
- Chapter 3 discusses the steps you should include in your plan to migrate a ZMF 7.x instance to ZMF 8.3.
- Chapter 4 discusses the steps you should include in your plan to migrate a ZMF 8.x instance to ZMF 8.3.
- Chapters 5 and 6 provide simplified instructions for configuring and using the ZMF Merge and Reconcile (M+R) Option to apply existing customization to the new release.

The following sections provide basic information about ChangeMan ZMF documentation.

ChangeMan ZMF Documentation Suite

Manual	Description
Administrator's Guide	Describes ChangeMan ZMF features and functions with instructions for choosing options and configuring global and application administration parameters.
Customization Guide	Provides information about ChangeMan ZMF skeletons, exits, and utility programs that will help you to customize the base product to fit your needs.

Manual	Description
Db2 Option Getting Started Guide	Describes how to install and use the Db2 Option of ChangeMan ZMF to manage changes to Db2 components.
ERO Concepts	Discusses the concepts of the Enterprise Release Option (ERO) of ChangeMan ZMF for managing
releases containing change packages.	
ERO Getting Started Guide	Explains how to install and use ChangeMan ZMF ERO to manage releases containing change packages.
ERO Messages	Describes system messages and codes produced by ChangeMan ZMF ERO.
ERO XML Services User's Guide	Documents ERO functions and services available for general customer use. These services are also known as the "green" services and provide mostly search and query functions.
High-Level Language Exits Getting Started Guide	Explains how to configure and call the high-level language exits.
IMS Option Getting Started Guide	Provides instructions for implementing and using the IMS™ Option of ChangeMan ZMF to manage changes to IMS components.
INFO Option Getting Started Guide	Describes two methods by which ChangeMan ZMF can communicate with other applications: Through a VSAM interface file and through the Tivoli® Information Management for z/OS product from IBM®.
Installation Guide	Provides step-by-step instructions for initial installation of ChangeMan ZMF. Assumes that no prior version is installed or that the installation will overlay the existing version.
Java / zFS Getting Started Guide	Provides information about using ZMF to manage application components stored in USS file systems, especially Java® application components.
Load Balancing Option Getting Started Guide	Explains how to install and use the Load Balancing Option of ChangeMan ZMF to connect to a ZMF instance from another CPU or MVS™ image.
M+R Getting Started Guide	Explains how to install and use the M+R Option of ChangeMan ZMF to consolidate multiple versions of source code and other text components.
M+R Quick Reference	Provides a summary of M+R Option commands in a handy pamphlet format.
Messages	Explains messages issued by ChangeMan ZMF, SERNET, and System Software Manager (SSM) used for the Staging Versions feature of ZMF.
Migration Guide	Gives guidance for upgrading ChangeMan ZMF from versions 7.x and 8.x to version 8.2 Patch 6.

Manual	Description
Online Forms Manager (OFM) Option Getting Started Guide	Explains how to install and use the OFM option of
ChangeMan ZMF.	
REST Services Getting Started Guide	Getting Started Guide for ZMF REST Services.
SER10TY User's Guide	Gives instructions for applying licenses to enable ChangeMan ZMF and its selectable options.
User's Guide	Describes how to use ChangeMan ZMF features and functions to manage changes to application components.
XML Services User's Guide	Documents the most commonly used features of the XML Services application programming interface to ChangeMan ZMF.
ZMF Quick Reference	Provides a summary of the commands you use to perform the major functions in the ChangeMan ZMF package life cycle.
ZMF Web Services User's Guide	Documents the Web Services application programming interface to ChangeMan ZMF.

Using the Manuals

Use Adobe® Reader® to view ChangeMan ZMF PDF files. Download the Reader for free at get.adobe.com/reader/.

This section highlights some of the main Reader features. For more detailed information, see the Adobe Reader online help system.

The PDF manuals include the following features:

- **Bookmarks**. All of the manuals contain predefined bookmarks that make it easy for you to quickly jump to a specific topic. By default, the bookmarks appear to the left of each online manual.

- **Links**. Cross-reference links within a manual enable you to jump to other sections within the manual with a single mouse click. These links appear in blue.

- **Comments**. All PDF documentation files that Serena delivers with ChangeMan ZMF have enabled commenting with Adobe Reader. Adobe Reader version 7 and higher has commenting features that enable you to post comments to and modify the contents of PDF documents. You access these features through the Comments item on the menu bar of the Adobe Reader.

- **Printing**. While viewing a manual, you can print the current page, a range of pages, or the entire manual.

- **Advanced search**. Starting with version 6, Adobe Reader includes an advanced search feature that enables you to search across multiple PDF files in a specified directory.

Searching the ChangeMan ZMF Documentation Suite

There is no cross-book index for the ChangeMan ZMF documentation suite. You can use the Advanced Search facility in Adobe Acrobat Reader to search the entire ZMF book set for information that you want. The following steps require Adobe Reader 6 or higher.

1. Download the ZMF All Documents Bundle ZIP file and the ZMF Readme to your workstation from the My Downloads tab on the Serena Support website.

2. Unzip the PDF files in the ZMF All Documents Bundle into an empty folder. Add the ZMF Readme to the folder.

3. In Adobe Reader, select Edit | Advanced Search (or press Shift+Ctrl+F).

4. Select the **All PDF Documents** in option and use **Browse for Location** in the drop down menu to select the folder containing the ZMF documentation suite.

5. In the text box, enter the word or phrase that you want to find.

6. Optionally, select one or more of the additional search options, such as **Whole words only** and **Case-Sensitive**.

7. Click **Search**.

8. In the **Results**, expand a listed document to see all occurrences of the search argument in that PDF.

9. Click on any listed occurrence to open the PDF document to the found word or phrase.

Choosing the Right Installation/Upgrade Manual

Choose the manual that fits your situation when installing or upgrading ChangeMan ZMF.

Your task	Manual to use
Installing ChangeMan ZMF for the first time	ChangeMan ZMF Installation Guide
Building a new ChangeMan ZMF 8.2 Patch 6 instance from scratch	ChangeMan ZMF Installation Guide

Your task	Manual to use
Upgrading from ChangeMan ZMF 7.x or 8.x to version 8.2 Patch 6	ChangeMan ZMF Migration Guide (this manual)

> 🔥 **Important**
>
> Always see the most current Readme for your ChangeMan ZMF release in case it contains documentation updates for the installation/upgrade manual you use.

ChangeMan ZMF Release Notes

High-level descriptions of the enhancements that are delivered in the ChangeMan ZMF 8.2 major version release and in all subsequent ZMF 8.2.x maintenance and patch releases are included in the "Features and Fixes" section of the latest ChangeMan ZMF 8.2 Patch 6 Readme.

Online Help

Online help is the primary source of information about ChangeMan ZMF. Online help is available as a tutorial, through help panels, and in ISPF error messages.

Online tutorial

ChangeMan ZMF includes an online tutorial that provides information about features and operations, from high-level descriptions of concepts to detailed descriptions of panel fields.

To view the tutorial table of contents, select option T from the Primary Option Menu, or jump to it from anywhere in ChangeMan ZMF by typing =T and pressing ENTER.

Press PF1 from anywhere in the Tutorial for a complete list of Tutorial navigation commands and PF keys.

Online Help Panels

If you have questions about how a ChangeMan ZMF panel works, you can view a help panel by pressing PF1 from anywhere on the panel.

Online Error Messages

If you make an invalid entry on a ChangeMan ZMF panel, or if you make an invalid request for a function, a short error message is displayed in the upper right corner of the panel.

Press PF1 to display a longer error message that provides details about the error condition.

Remember that the long message does not display automatically. Request the long message by pressing PF1.

Typographical Conventions

The following typographical conventions are used in the online manuals and online help. These typographical conventions are used to assist you when using the documentation; they are not meant to contradict or change any standard use of typographical conventions in the various product components or the host operating system.

Convention	Explanation
italics	Introduces new terms that you may not be familiar with and occasionally indicates emphasis.
bold	Indicates panel titles, field names, and emphasizes important information.
UPPERCASE	Indicates keys or key combinations that you can use. For example, press ENTER.
`monospace`	Indicates syntax examples, values that you specify, or results that you receive.
`monospace italics`	Indicates names that are placeholders for values you specify; for example, `filename`.
`monospace bold`	Indicates the results of an executed command.
vertical rule	Separates menus and their associated commands. For example, select File

> ♀ **Note**
>
> IBM® Sterling Connect:Direct® is a point-to-point file transfer software product that can be used to transfer files between two ChangeMan ZMF instances. The original name of the product was Network Data Mover (NDM). The NDM mnemonic persists, embedded in Connect:Direct and ChangeMan ZMF component names, options, and JCL examples.

2. Migration Strategy, Plans, and Issues

This chapter describes strategies for migrating a ChangeMan ZMF instance to version 8.3. It presents a model migration plan that you can use as a template for your migration plan, and it presents sample scripts you can use as templates for your own conversion-day script.

The relevant chapter for your migration has a table of contents for your migration which can also be used as a high level checklist (The entries are listed as steps).

- Assumptions
- Migration Strategy
- Model Migration Plan
- Sample Conversion Day Script

Assumptions

The following assumptions are implicit in this migration guide:

- You have read Choosing the Right Installation/Upgrade Manual to verify that this migration guide is the correct manual for your instance of ChangeMan ZMF.

- Before you begin planning your migration, you have read the entire ChangeMan ZMF Installation Guide and this current migration guide.

- You can build multiple ChangeMan ZMF test environments that model key features of your current production ZMF architecture and customization.

- You manage ChangeMan ZMF custom components with a ChangeMan ZMF instance. Custom components include skeletons, panels, exit programs, started procedures, housekeeping JCL, and so on. The release level of the ZMF instance you use to manage custom components is not important.

Migration Strategy

This section:

- Lists factors that limit how you can migrate to ChangeMan ZMF 8.3.

- Outlines measures you can take to reduce risk in your migration to the new version.

- Discusses the advantages of release libraries.

- Outlines a strategy you can use to convert P (production) instances before you migrate your ChangeMan ZMF D (development) or DP (development-production) instance.

Constraints

All package and component functions begun under pre-8.3 ChangeMan ZMF releases can be continued under version 8.3, including promotion/demotion and install/backout.

However, there are constraints on the migration to ChangeMan ZMF 8.3:

- There is no process to deconvert from ChangeMan ZMF 8.3 back to a pre-8.3 instance.

  - Because of the changes to the structure of ChangeMan ZMF files, there is no process to deconvert data stores that are processed through conversion programs that prepare them for version 8.3.

  - Installation jobs that are file tailored under ChangeMan ZMF 8.3 will not run successfully under pre-8.3 versions.

- Only one type of phased migration to version 8.3 is supported: Convert P instances to version 8.3 before you convert D and DP instances.

  You cannot convert DP instances before P instances because installation jobs that are file tailored under ChangeMan ZMF 8.3 will not run successfully under pre-8.3 versions.

- If you back out a conversion to ChangeMan ZMF 8.3, you must restore all converted ZMF master files to their pre-conversion state. If you back out the conversion after developers have performed work under ZMF 8.3, you will have these problems:

  - There will be no record of the changes that developers made to components between the conversion and the conversion backout. Such changes remain in staging libraries, but these changes are unknown to ChangeMan ZMF, so they will cause audit errors.

  - Staging versions recorded between conversion and conversion backout are lost.

  - Packages and staging libraries that were created between conversion and conversion backout are unknown to ChangeMan ZMF. However, staging libraries created during that time will remain on DASD and in the catalog, and they will interfere with the allocation of staging libraries as developers try to recreate packages that they created under 8.3.

- Installation jobs that are file tailored between conversion and conversion backout will not run.

Installation JCL Issues

Installation jobs that are file tailored on a pre-8.3 instance can run on a version 8.3 instance. When ChangeMan ZMF 8.3 installation programs process an installation Package data set created under a pre-8.3 version, the Package records are converted to version 8.3 before they are inserted into the ZMF 8.3 Package Master.

The challenge is that the installation JCL created on a pre-8.3 ChangeMan ZMF instance must execute with version 8.3 libraries to install on a version 8.3 instance.

Customers may have devised a solution to this issue when they upgraded ChangeMan ZMF to their current release. Here are two solutions, one solution for customers with a simple ZMF environment, and one solution for customers with a more complex environment:

- If all of your ChangeMan ZMF instances use the same production load library, you can convert all ZMF instances at the same time, replacing pre-8.3 components in the LOAD library with version 8.3 components.

- If you have a complex environment and use release libraries (libraries with a unique DSN for each ChangeMan ZMF release), you can use INCLUDE groups for JOBLIB and STEPLIB statements in installation skeletons. By changing the libraries in the INCLUDE group stored in a PROCLIB, you effectively change the JOBLIB libraries in the installation jobs at run time.

Strategies to Reduce Risk

Use these strategies to reduce risk in the process of migrating a pre-8.3 ChangeMan ZMF instance to version 8.3.

- Segregate delivered vendor versions of components from custom components; store them in separate libraries. Preserve the vendor version of all ChangeMan ZMF 8.3 components so you can revert to the original versions if your modifications do not work as expected.

- Create new physical libraries for ChangeMan ZMF 8.3 custom and vendor components. Do not update libraries containing pre-8.3 components with version 8.3 components. Not all pre-8.3 components are brought forward into version 8.3.

- Analyze the ChangeMan ZMF 8.3 components that you can customize, and customize these components as appropriate for your installation and your ChangeMan ZMF version 8.3 instance.

- Use the ChangeMan ZMF M+R Option to locate and analyze modifications you made to pre-8.3 components. If you do this, keep in mind that customizations you made to components for an

earlier ZMF release may no longer apply to components in the new ZMF release distribution libraries. The M+R Option is included in ChangeMan ZMF distribution libraries. Any version of the M+R Option is suitable for this task. If you do not already license the M+R Option, contact your Micro Focus account representative.

• Preserve ISPF statistics on vendor versions of ChangeMan ZMF components. Micro Focus Customer Care may need the USER and the CHANGED date from the delivered version of a component to help you diagnose problems.

• Create two sets of change packages for your customized ChangeMan ZMF 8.3 components:

  • Packages with components that are installed into ChangeMan ZMF 8.3 libraries in production, such as exit programs, ISPF messages, panels, and skeletons.

  • Packages with components that are installed into system libraries in production, such as JCL, cataloged procedures, and CLISTs.

ChangeMan ZMF 8.3 libraries can be populated before the actual conversion. Unless you make the name of your started procedures and log-on CLISTS version specific, you cannot update system libraries until the day a pre-8.3 instance is converted to version 8.3.

• Test all data conversions in the migration process with copies of your pre-8.3 production data to determine how much DASD is required for intermediate and final files and to determine how long each process will run on conversion day.

• Create a conversion plan, write a test plan, and build a detailed script for the day of conversion. Build a script for reversing the upgrade in case there are serious problems after conversion that cannot be fixed.

• Execute a dry-run conversion before you execute the actual conversion. If you have problems, correct the problems and execute the dry-run conversion again until it runs smoothly.

• Ensure that your conversion scripts contain file backups that create a pre-conversion snapshot that you can reload if you must back out the upgrade of a ChangeMan ZMF instance. Also verify that your backups can be restored. Ensure that your conversion scripts contain post-conversion backups that create a snapshot that is included in your standard disaster recovery backups.

• Keep all job logs in their entirety from your final conversion activities.

Libraries By Release

There are significant benefits to creating a new set of uniquely named vendor and custom production libraries for each release of ChangeMan ZMF. Unique production libraries allow:

- Phased migration where there are multiple ChangeMan ZMF instances. Two ZMF releases can coexist in the production environment.

- A simpler conversion-day process, because most components in the new release can be installed into production before conversion without interfering with the old release.

When you manage custom ChangeMan ZMF components with a ChangeMan ZMF instance, implement unique production libraries for a release by creating a new application for each new ZMF release.

Strategies for Sites with Multiple ChangeMan ZMF Instances

If you have many ChangeMan ZMF instances, you may not be able to upgrade all of those instances to version 8.3 at the same time. If you have uniquely named vendor and custom production libraries for each release of ZMF, you may be able to convert instances individually or in groups:

- You can convert A (ALL) instances one at a time or in groups.

- You can convert P instances before you convert the D or DP instances that communicate with them.

- You can convert a D or DP instance and all of the P instances it communicates with.

You cannot convert a D or DP instance before you convert the P instances that they communicate with. Convert P instances first.

A Release Format Version Field has been Added to the Package Master

A format version field (vvpp, for example, 8300 for ZMF version 8.3) has been added to the global 0 record of the Package Master file. If there is a mismatch between the actual file version format and the file version format that the startup program CMNSTART expects, the Sernet started task will incur a U320 abend with a message in the log that indicates which of the following three master files has the error:

- Package Master

- Component Master

- Long Name Component Master

This abend will happen if you have not run a conversion program on that file to update the format to the correct level for the release that you are installing. To resolve the problem, run the correct conversion JCL, and restart the task. The CMN990 report will show the current versions. The enhancement is intended to prevent you from running a ChangeMan ZMF version with a Package, Component, or Long Name Component Master file that is incompatible with the ChangeMan ZMF version that is being installed.

Prevention of multiple versions of ZMF running in a single ISPF session (U0044)

Different versions of ChangeMan ZMF are incompatible in ISPF split screen mode. User abend 44 blocks users from opening a second session with a different version and can also signify that incomplete SEREX006 exit information has been supplied when it occurs within the ZMF Server started task.

Model Migration Plan

A migration plan lists the tasks that you will perform over a period of weeks or months to prepare to migrate a ChangeMan ZMF instance to version 8.3. This section outlines a model migration plan.

The hour-by-hour tasks that you execute on the day you convert a ChangeMan ZMF instance to version 8.3 are listed in a conversion-day script. The section Sample Conversion-Day Script shows two sample conversion-day scripts.

> 💡 **Note**
>
> The model plan in this section and the model conversion-day scripts in the next section assume a relatively complex ChangeMan ZMF environment where INCLUDE groups are used to manage installation JCL issues. Your environment will probably be different, so your plan and scripts will be different in their details. However, these models can help you think about issues so you can devise solutions, plans, and scripts that fit your particular situation.

Review this model migration plan and write a plan that fits your implementation of ChangeMan ZMF:

1. Write a test plan for standard ChangeMan ZMF 8.3 functions, and write a test plan for functions that you have customized.

2. Install changes to pre-8.3 skeletons to use INCLUDE groups to set JOBLIB and STEPLIB library names in installation jobs. Install the corresponding INCLUDE groups into production PROCLIB libraries.

3. Create change packages for custom ChangeMan ZMF 8.3 components:

- Components that will be installed in ChangeMan ZMF 8.3 custom libraries.
- Components that will be installed in system libraries, such as PROCLIB, CLIST, and JCLLIB.

4. Follow the instructions in the ChangeMan ZMF Installation Guide to build a test ChangeMan ZMF 8.3 instance. Perform the minimum customization described in that manual to get your test instance running.

5. If you use P instances for production installations, built a test P instance of ChangeMan ZMF 8.3.

6. Test standard (not customized) ChangeMan ZMF 8.3 functions to verify that they work as they are supposed to and to gain an understanding of new and changed features in the base product.

7. Forward fit your customizations into ChangeMan ZMF 8.3 components, if and as appropriate. If possible, use the ChangeMan ZMF M+R Option for this task. Any version of the M+R Option is suitable.

8. Test your customizations on the ChangeMan ZMF 8.3 test instances.

9. Use the guidelines given in this manual to convert an existing pre-8.3 ZMF test instance to version 8.3.

10. Write conversion-day scripts based on your experience converting pre-8.3 test instances to ChangeMan ZMF 8.3.

11. Refine conversion JCL so that it is production-ready. Run volume tests to validate space allocations and obtain run-time estimates.

12. Modify existing housekeeping jobs to fit ChangeMan ZMF 8.3.

13. Test housekeeping jobs with production volume input to validate space allocations and obtain run-time estimates.

14. Execute dry-run tests of your conversion-day scripts. Repeat until they run smoothly.

15. Install the change package that contains the customized components that are stored in ChangeMan ZMF 8.3 production libraries.

16. Revise customer-written user documentation, administrator procedures, and training materials. Alert ChangeMan ZMF users to processing changes delivered in version 8.3.

17. Follow your company procedures to schedule the conversion, provide notice to users, and schedule support.

18. Take appropriate actions to rebuild package installation JCL that does not set JOBLIB and STEPLIB libraries with INCLUDE group statements (for example, Freeze Option 5 or Approve Option 5 to rebuild the installation JCL).

19. For customers who license the ChangeMan ZMF Db2 Option and/or ERO Option:

ChangeMan ZMF 8.1.1 contained significant changes if you license the Db2 Option and/or the ERO Option. Therefore if you are migrating from an earlier version than 8.1.1, you must also refer to one of the following sections, as appropriate, for details:

- If you are migrating a ZMF 7.1.x release to 8.3, see Step 9: Implement Changes for the Db2 Option and Step 10: Upgrade ERO.

- If you are migrating an 8.1.0 or 8.1.0.01 release to 8.3, see Step 9: Implement Changes for the Db2 Option and Step 10: Upgrade ERO.

> ♀ **Note**
>
> ChangeMan ZMF Db2 programs are precompiled with VERSION(AUTO) so you can have multiple versions of a package in the Db2 catalog.

20. Execute conversion-day scripts to migrate your production ChangeMan ZMF environment to version 8.3.

21. As time permits after conversion, log on to each migrated ChangeMan ZMF 8.3 instance and make post conversion administration changes.

Sample Conversion Day Script

A conversion-day script lists all of the tasks that you must complete on the day that you convert a ChangeMan ZMF instance to version 8.3.

The conversion-day script is one of the last steps in your migration plan. See Model Migration Plan for an example of a migration plan.

> ♀ **Note**
>
> The model conversion-day scripts shown in this section are meant to be embedded within the model plan in the previous section. There are important conversion steps that are in the plan rather than the script because those steps can be executed before the actual day of conversion.

Conduct dry-run tests of your script with copies of production data to validate space allocations and to establish expected run times for each step.

The following list of tasks is an example of the kind of tasks that you should include in your conversion-day script for upgrading a ChangeMan ZMF instance to version 8.3. Detailed instructions for these tasks are provided in one of the following chapters, as appropriate to the release that you are upgrading:

- Converting ChangeMan ZMF 7.1.x Instances to Version 8.3.

- Converting ChangeMan ZMF 8.1.x and 8.2 Instances to Version 8.3.

Review this sample conversion-day script and write scripts that fit your implementation of ChangeMan ZMF:

1. Verify that vendor libraries for ChangeMan ZMF 8.3 are installed in production.

2. Verify that the change package containing the following version 8.3 custom components is installed in ChangeMan ZMF 8.3 production libraries:

   • Load modules for customized exit programs.

   • Customized messages.

   • Customized panels.

   • Customized REXX programs.

   • Customized skeletons.

   • Customized and generated tables (including CMNKEYS).

   • Recompile HLL exits that are written in an LE language, ensuring the recompile uses the new versions of copybooks.

3. Stop the started task for the ChangeMan ZMF version that you are migrating.

4. Use existing pre-conversion housekeeping jobs or other JCL to back up the following VSAM files in your production environment:

   a. Package Master

   b. Component Master

   c. Impact Analysis File

   d. Delay File

   e. Staging Versions VSAM files

5. Use existing pre-conversion housekeeping jobs or other JCL to back up and clear the following log and recovery files in your production environment:

   a. Activity Log File

   b. Recovery File

   c. Impact Analysis Recovery Log

   d. Component Master Recovery Log if applicable (7.1.2.01 or before)

6. For customers who license the ChangeMan ZMF ERO Option, unload or image copy the release management Db2 tables.

7. Execute the procedure to convert your Package Master, Component Master, (and Saved Staging Version) data sets.

8. Execute the procedure to create and load your ChangeMan ZMF XMLSPACE LDS.

9. For customers who license the ChangeMan ZMF ERO Option, perform the following actions as required:

    a. Execute the procedure to expand ERO release audit and release management Db2 tables.

    b. Unload or image copy the release management tables, and bind ERO DBRMs in your production environment.

10. Install the change package containing the following components into system libraries:

    a. Sernet started procedure.

    b. JCL for started procedure CMNxADSP and/or the procedures you created for:

        • Installation (X.node) JCL build.

        • HLLX started task procedure if used.

        • Batch component build.

        • Promotion JCL build.

        • Other jobs that are file tailored.

    c. Log-on CLIST.

    d. INCLUDE group updated with ChangeMan ZMF 8.3 libraries for installation jobs that were file tailored under the ZMF release that you are migrating.

    e. Housekeeping JCL and cataloged procedures.

11. Use version 8.3 housekeeping jobs to back up the following converted VSAM files in your production environment:

    a. Package Master.

    b. Short Name Component Master.

    c. Long Name Component Master.

12. Bring up the new Sernet started task that you defined for ChangeMan ZMF 8.3.

13. If you change library type definitions or baseline library names in application administration, execute the procedure to synchronize the impact analysis data.

14. Run post-conversion test scripts to validate version 8.3 functions, custom modifications, and administration entries prior to releasing the system to developers for use.

3. Converting ZMF 7.1.x to v.8.3

This chapter provides detailed information about the tasks that you need to perform to convert any ChangeMan ZMF 7.1.x instance to version 8.3.

Read the assumptions, constraints, and strategies in Migration Strategy, Plans, and Issues before you start work on the tasks in this chapter. Have a comprehensive migration plan in place before you execute conversion tasks. Also ensure you have reliable backups before you execute conversion tasks.

- Conversion Task Dependencies
- Step 1: Apply ChangeMan ZMF License
- Step 2: Build JCL INCLUDEs for JOBLIBs and STEPLIBs
- Step 3: Update Started Procedures and Log-on CLIST
- Step 4: Define Keylists for ISPF Expand
- Step 5: Back Up Version 7.1.x Files
- Step 6: Convert Package Master, Component Master, and SSV Data Sets
- Step 7: Reload XMLSPACE VSAM LDS
- Step 8: Update Sernet Keyword Options
- Step 9: Implement Changes for the Db2 Option
- Step 10: Upgrade ERO
- Step 11: Customize SERLCSEC
- Step 12: Customize Other Version 8.3 Components
- Step 13: Replace somnode with Variables
- Step 14: Update RACF PassTicket Support in Sernet
- Step 15: Synchronize Impact Analysis Data
- Step 16: Customize User Exit SEREX006
- Step 17: Perform Post-Migration Checks
- Step 18: Submit Job CMNRPKSN if You License ERO
- Step 19: Configure Support for IBM Health Checker
- Step 20: Configure Activity Log Reporting Procedure

Conversion Task Dependencies

The steps that you execute in this chapter and the specific actions within each step may depend on the level of the last ChangeMan ZMF maintenance release, patch release, or hotfix that you applied to your system.

- Read the instructions in each step carefully to determine if a specific action applies to you.
- In some steps, such as bringing customization forward and upgrading housekeeping jobs, assess the scope of the action you must take by comparing the vendor libraries at your current release level to the vendor libraries for ChangeMan ZMF 8.3.

If you have questions about whether a step or action applies to you, contact Micro Focus Customer Care.

Step 1: Apply ChangeMan ZMF License

If you are upgrading from a ZMF 7.1.x release, you need to obtain a new ChangeMan ZMF license.

If you also license the ZMF Client Pack, your license file should contain a member called LIC00213. If the license file does not have this member, you need to request a new license key.

The SER10TY User's Guide provides instructions for applying those licenses.

Step 2: Build JCL INCLUDEs for JOBLIBs and STEPLIBs

You may have a number of jobs already generated (for example, for frozen packages awaiting install) that need to run successfully once you have completed your conversion from version 7.1.x to version 8.3. If these jobs reference skeletons (such as CMN$$JBL) that have hard-coded JOBLIB/STEPLIB concatenations pointing back to version 7.1.x software, they will fail when they run against the upgraded version 8.3 instance.

One way to avoid this failure is to keep the same execution library names and replace the contents with the version 8.3 software.

However, it is safer (and recommended) to keep different versions of ZMF software in different named libraries.

This can also be implemented by the use of aliases for the datasets involved.

A method for addressing this situation by the use of INCLUDE JCL members follows:

1. Ensure you have separate execution libraries for the two different releases of ZMF.

2. Set up two JCL members to be INCLUDEd into CMN$$JBL for each ZMF instance: one for JOBLIB and one for STEPLIB. Prior to any conversion these members will contain version 7.1.x execution libraries.

3. Modify skeleton CMN$$JBL to reference these INCLUDE members based on the subsystem ID of the ZMF for which the job is being generated.

4. Make the modified CMN$$JBL skeleton live in your ZMF instance(s).

> 💡 **Note**
>
> You have to take these steps well in advance of any conversion so that all frozen package jobs will be using the modified CMN$$JBL at the time of the conversion to version 8.3.

5. As part of the upgrade process for each ZMF instance, update the relevant INCLUDE member so that the JOBLIB and STEPLIBs point to version 8.3 execution libraries.

Update Started Procedures and Log-on CLIST

This section tells you how to modify the started procedures and CLIST for the version that you are migrating for use with version 8.3. You can compare the following members between the libraries of the version that you are migrating from, and ZMF 8.3 libraries to see what is changed.

- SERVER in the SERCOMC.CNTL library
- STARTJCL in the CMNZMF.CNTL library
- CMNADSP in the CMNZMF.CNTL library
- SERDRT in the SERCOMC.CNTL library
- CMN in the CMNZMF.CLIST library

> 🔥 **Important**
>
> When you create a new DD statement or add a file or library, change the placeholder somnode for the high-level qualifier in the examples shown here to make the library or data set name fit the naming conventions that you use for ChangeMan ZMF.

Started Procedure for ALL, DP, and D Instances

Follow these steps to create a started task procedure for ChangeMan ZMF 8.3:

1. JCL member ACTLOG of the SERCOMC.CNTL distribution library is shipped with ZMF 8.3. Run this job to delete and define a Sernet Activity Log, which is new in ZMF 8.3. This Activity Log data set is then referenced in the following DD statement in the SERVER member of the SERCOMC.CNTL distribution library:

    ```
    //SERALOG DD DISP=SHR,DSN=somnode.ACTLOG
    ```

    > 💡 **Note**
    >
    > Each instance of the started task requires its own SERALOG DD statement; attempts to share a single data set between multiple started tasks will produce unpredictable results.
    >
    > If the DD statement is missing, the started task will abend with code U4000.

2. In the custom CNTL library that you create for version 8.3, append the JCL in member STARTJCL of the CMNZMF CNTL distribution library to the JCL in member SERVER of the SERCOMC CNTL distribution library. You must do this to enable ChangeMan ZMF to run under the Serena Network (Sernet).

3. Follow the comments in the JCL to customize the started task JCL for version 8.3 as appropriate for your installation standards. You can use the customizations that your made to the started task JCL for the 7.1.x version that you are upgrading as a guide. However, some of the customizations that you made for version 7.1.x may no longer be appropriate for version 8.3.

4. We recommend that you make the SYSMDUMP data set a GDG as in the supplied SERVER example to prevent diagnostic information from being overwritten when the Sernet instance is restarted after an abend.

5. If not already present, create a GDG base for the dump capture data set name in the SYSMDUMP DD statement.

6. Copy the customized procedure to the appropriate system PROCLIB library for testing.

Started Procedure for P-Site Instances

Follow similar steps to create the Sernet/ChangeMan ZMF started procedure for the 7.1.x P-site instances that you are upgrading to run version 8.3.

ChangeMan ZMF 7.1.3 and later P-site instances do not require Component Master and Impact Analysis files.

File Tailoring Procedures - CMNADSP

The started procedures defined on panel Global Parameters - Part 2 of 8 (CMNGGP02) are based on JCL sample CMNADSP in the CMNZMF.CNTL library.

Follow similar steps to create your 8.3 file tailoring started procedures.

File Recall Procedure - SERDRT

The started procedure used to recall an ML2 dataset when requested by the Started Task are based on JCL sample SERDRT in the SERCOMC.CNTL library. The number of these procedures is defined by the ASIDS4RECALL parameter in the SERSYSIN DD of the Started Task. The default if omitted is 4, the range is from 2 to 16. The limit can be changed by an operator command. See the Installation Guide for more information on operator commands.

Log-On CLIST

Follow these steps to create the log-on CLIST for ChangeMan ZMF 8.3 instance:

1. Copy the log-on CLIST CMN from the vendor CMNZMF.CLIST library to your custom version 8.3 CMNZMF.CLIST library.

2. In the ALTLIB statement, ensure the somnode values are pointing to your new 8.3 libraries, retain the custom libraries in the order they are in.

3. In the LIBDEF statements, ensure the somnode values are pointing to your new 8.3 libraries, retain the custom libraries in the order they are in. You will allocate and populate the TABLES (ISPTLIB) libraries later in this chapter.

4. Copy the updated CLIST to the CLIST library that is allocated to `//SYSPROC` in your ZMF users' TSO log-on procedure for testing.

Step 4: Define Keylists for ISPF Expand

In ChangeMan ZMF 7.1 and later releases, component names and file names are expanded to accept long zFS component names and path names. The logical unit name for sites is expanded to accept longer transmission vehicle node names. The Db2 Option in ZMF 7.1 and later releases processes the long Db2 object names that are available in Db2 V8 and later.

In the ISPF interface for ChangeMan ZMF 7.1 and later releases, fields for input and display of this longer data are extended to the limit of the 80 character panel width. If the data is longer than the available space, you can use the PF11 and PF10 keys to scroll right and left respectively in the panel fields.

The panels also respond to the ISPF Expand function to open a pop-up for input or display of the entire long name field. To execute the Expand function from the keyboard, the panels in ChangeMan ZMF 7.1 and later releases call a new keylist that assigns PF4 to the Expand function.

The job in member CMNDTL of the CMNZMF.CNTL distribution library builds keylist CMNKEYS in the TABLES library you added to the ChangeMan ZMF started procedures and to the connect CLIST.

Follow these steps to build ISPF keylist CMNKEYS in a new ISPF TABLES library.

1. Copy member CMNDTL from the version 8.3 CMNZMF.CNTL distribution library to your custom ZMF 8.3 CMNZMF.CNTL library. Edit CMNDTL in your custom CMNZMF.CNTL library.

2. Code your JOB statement at the top.

3. Change DD statement DTLGML to point to the CMNZMF SAMPLES distribution library.

4. Change all remaining occurrences of placeholder somnode to the high level qualifier you use for ISPF system libraries.

5. Submit job CMNDTL.

6. Examine job output for error messages. The ISPLOG should report 0 error(s) found. Error message ISPC506E in the SYSTSPRT likely means that the member in the SAMPLIB library was not found.

Example of ISPLOG from Job Step DTL

```
Time            *** ISPF transaction log ***        Userid: USER015    Date: 19/02/20      Page: 1

21:49   Start of ISPF Log - - - - Session # 1 --------------------------------------------------------
21:49      TSO - Command -      - ISPDTLC CMNELSK (PROFDDN=* CUASUPP REPLACE)
21:49                           - ISPC028AI: Total of 2 message(s) have been suppressed.
21:49                           - ISPC029AI: Total of 2 warning(s) and 0 error(s) found.
21:49   End of ISPF Log - - - - Session # 1 ----------------------------------------------------------
Time            *** ISPF transaction log ***      Userid: USER015    Date: 19/02/20      Page: 1

21:49   Start of ISPF Log - - - - Session # 2 --------------------------------------------------------
21:49      TSO - Command -      - ISPDTLC CMNEROK (PROFDDN=* CUASUPP REPLACE)
21:49                           - ISPC028AI: Total of 2 message(s) have been suppressed.
21:49                           - ISPC029AI: Total of 2 warning(s) and 0 error(s) found.
21:49   End of ISPF Log - - - - Session # 2 ----------------------------------------------------------
Time            *** ISPF transaction log ***      Userid: USER015    Date: 19/02/20      Page: 1

21:49   Start of ISPF Log - - - - Session # 3 --------------------------------------------------------
21:49      TSO - Command -      - ISPDTLC CMNMREK (PROFDDN=* CUASUPP REPLACE)
21:49                           - ISPC028AI: Total of 2 message(s) have been suppressed.
21:49                           - ISPC029AI: Total of 2 warning(s) and 0 error(s) found.
21:49   End of ISPF Log - - - - Session # 3 ----------------------------------------------------------
```

Example of SYSTSPRT from job step DTL

```
READY
    ISPSTART CMD(ISPDTLC CMNELSK (PROFDDN=* CUASUPP REPLACE))
ISPC021I: ISPDTLC ISPF 7.3 Dialog Tag Language Conversion Utility
ISPC020AI: ISPF Dialog Tag Language Conversion Utility for ISPF 7.3. Current
    APAR level: PID. Current PTF number: 3 Sep 2017.
ISPC022I: Converting source file "CMNTP.CMN820.SAMPLES(CMNELSK)"...
ISPC028I: 2 message(s) have been suppressed.
ISPC029I: 2 warning(s) and 0 error(s) found.
ISPD117
The initially invoked CLIST ended with a return code = 1
    USER015.USER0154.J0281166.D0000103.? was preallocated (no free was done).
READY
    ISPSTART CMD(ISPDTLC CMNEROK (PROFDDN=* CUASUPP REPLACE))
ISPC021I: ISPDTLC ISPF 7.3 Dialog Tag Language Conversion Utility
ISPC020AI: ISPF Dialog Tag Language Conversion Utility for ISPF 7.3. Current
    APAR level: PID. Current PTF number: 3 Sep 2017.
ISPC022I: Converting source file "CMNTP.CMN820.SAMPLES(CMNEROK)"...
ISPC028I: 2 message(s) have been suppressed.
ISPC029I: 2 warning(s) and 0 error(s) found.
ISPD117
The initially invoked CLIST ended with a return code = 1
    USER015.USER0154.J0281166.D0000103.? was preallocated (no free was done).
READY
    ISPSTART CMD(ISPDTLC CMNMREK (PROFDDN=* CUASUPP REPLACE))
ISPC021I: ISPDTLC ISPF 7.3 Dialog Tag Language Conversion Utility
ISPC020AI: ISPF Dialog Tag Language Conversion Utility for ISPF 7.3. Current
    APAR level: PID. Current PTF number: 3 Sep 2017.
ISPC022I: Converting source file "CMNTP.CMN820.SAMPLES(CMNMREK)"...
ISPC028I: 2 message(s) have been suppressed.
ISPC029I: 2 warning(s) and 0 error(s) found.
ISPD117
The initially invoked CLIST ended with a return code = 1
    USER015.USER0154.J0281166.D0000103.? was preallocated (no free was done).
READY
END
```

Step 5: Back Up Version 8.1.x or 8.2.x Files

Use your existing preconversion housekeeping jobs or other JCL to back up ChangeMan ZMF files and clear log files for the 8.1.x or 8.2.x release that you are migrating from, before you begin converting files for version 8.3.

The following table lists the files you should back up. To accomplish this task, use the sample JCL members in the vendor CMNZMF CNTL library that was distributed with the ZMF 8.1.x or 8.2.x instance that you are migrating from.

Last DSN Node	Description	Type	Vendor CMNZMF.CNTL Job from Preconversion Instance	Notes
CMNPMAST/ CMNCMPNL/ CMNCMPNT	Package Master	VSAM	BACKUP	Note 1

Last DSN Node	Description	Type	Vendor CMNZMF.CNTL Job from Preconversion Instance	Notes
CMNELCTX/ CMNCRSnn	Component Control/ Component Record Store	LDS	ELSUNLD	Note 2
CMNRECV	Package Master Forward Recovery	VSAM	CLEARRCV	Note 3
CMNELLOG	Component Forward Recovery Log	QSAM	ELSBKUP	Note 4
CMNLOG	Activity Log	VSAM	CLEARLOG	
CMNDELAY	Delay	VSAM		Note 5
IADSP	Impact Analysis	LDS	LDSBKUP	
IALOG	Impact Analysis Forward Recovery Log	QSAM	LDSLOGR	
CMNSVSM	Staging Versions Site Master	VSAM	SVBKUP	
CMNSVOM	Staging Versions Object Master	VSAM	SVBKUP	
CMNSVDM	Staging Versions Delta Master	VSAM	SVBKUP	

- Note 1: The BACKUP member of the ZMF 7.1.3 CMNZMF.CNTL library will back up the CMNPMAST, CMNCMPNL, and CMNCMPNT data sets. The BACKUP member of the ZMF 7.1.x CMNZMF.CNTL library prior to ZMF 7.1.3 will back up the CMNCMPNT data set.

- Note 2: The CMNELCTX and CMNCRSnn data sets are obsolete in ZMF 7.1.3 and later releases. They are only used in releases where the Component Master is implemented as a VSAM LDS (in releases 7.1.0 through 7.1.2.01).

- Note 3: The format of records in the CMNRECV recovery file are changed in ZMF 7.1.3 and later releases. Job CLEARRCV accomplishes the file conversion, but a new CMNRECV file is explicitly defined and initialized in "Step 6: Convert Package Master, Component Master, and SSV Data Sets".

- Note 4: The CMNELLOG component recovery log file is obsolete in ZMF 7.1.3 and later releases. Use sample job ELSBKUP (DD statements INDDL/OUTDD) to backup CMNELLOG.

- Note 5: There is no sample job for backing up the CMNDELAY file. Use IDCAMS with output QSAM file attributes RECFM=VB, LRECL=5000, BLKSIZE=32700.

Step 6: Convert Master Files, Component Master, and SSV Data Sets

The format of the Package Master changed in ZMF 8.1.2. Read the CMNZMF CNTL distribution library member $$$INDEX for a short description of all sample JCL.

In ChangeMan ZMF 7.1.3 and later releases, long name tokens are removed from the Package Master, and long names are stored in VARCHAR fields. Component data is split into two VSAM KSDS files, one for components with names of 8 or fewer characters, including zFS components with short names, and the other for components with names longer than 8 characters. A new format is required for forward recovery records to match the converted Package and Component Masters.

If you use Saved Staging Version (SSV) data sets, these data sets need to be converted to a format that ChangeMan ZMF 8.3 can use.

Execute the following jobs, as appropriate, to convert your Package Master, Component Master, and SSV data sets to a format that is suitable for ChangeMan ZMF 8.3 and create a new recovery file for ZMF 8.3:

ZMF 7.1.3 thru to 7.1.3.02 to ZMF 8.3

Job Name	Description
CMVCONV8	For non-P sites. Converts the ZMF v7.1.3 thru 7.1.3.02 package, component and SSV master files to a format compatible with v8.3.
CMVCONP8	For P-sites. Converts a ZMF v7.1.3 thru 7.1.3.02 package master file for P-sites to a format compatible with v8.3.
ACTLOG	Deletes and defines the SerNet activity log, required from ZMF 8.2 Patch 2 onwards. The resulting dataset is allocated to the server started task under the SERALOG DD statement. See Started Procedure for ALL, DP, and D Instances.

ZMF 7.1.0 thru to 7.1.2.01 to ZMF 8.3

Job Name	Description
CMVCONV7	For non-P sites. Converts the ZMF v7.1.0 thru 7.1.2.01 package, component and SSV master files to a format compatible with v8.3.
CMVCONP7	For P-sites. Converts the ZMF v7.1.0 thru 7.1.2.01 package master file for P-sites to a format compatible with v8.3.

Job Name	Description
ACTLOG	Deletes and defines the SerNet activity log, required from ZMF 8.2 Patch 2 onwards. The resulting dataset is allocated to the server started task under the SERALOG DD statement. See Started Procedure for ALL, DP, and D Instances.

Step 7: Reload XMLSPACE VSAM LDS

The contents of sequential file MAPDATA and VSAM LDS XMLSPACE have changed in ChangeMan ZMF 8.3.

Follow this procedure to define and populate the MAPDATA sequential file and the XMLSPACE VSAM LDS:

1. Copy member XMLLOAD from the SERCOMC CNTL distribution library to your ZMF 8.3 custom SERCOMC CNTL library. Edit XMLLOAD in your ZMF 8.3 custom SERCOMC.CNTL library.

2. Alter the somnode entries to ensure that libraries point to the new ZMF 8.3 library names.

3. Submit job XMLLOAD.

4. Examine the job output for problems:

   a. Examine the output of job step RECEIVE in SYSTSPRT.

   b. Examine the job's JESMSGLG to verify that job step LOAD completed with RC=00.

   c. Browse member XMLDATA in your SERCOMC CNTL library. Note the date and time in this line of text near the top of the data set:

   ```
   SERNET XML Dsect Cross Reference. Created: dd mmm yyyy hh:min:ss
   ```

   d. Browse the new MAPDATA file and look for the same text at the top of that file. The date and time should be the same as in the XMLDATA member.

Step 8: Update Sernet Keyword Options

Sernet keyword options are input to Sernet through one of these methods:

- In the PARM= parameter for program SERVER in the started procedure (SERVER member of the SERCOMC CNTL library).
- In the library at the DDname specified in keyword parameter DDNAME= in the PARM=parameter for program SERVER in the started procedure.

Sernet keyword options are described in an appendix of the ChangeMan ZMF Installation Guide.

Update Sernet keyword options as appropriate.

Step 9: Implement Changes for the Db2 Option

Steps required to migrate the ZMF Db2 option vary depending on which configuration you wish to implement and on which other, Db2-reliant, ZMF features are to be used (e.g. ERO).

The option to host the Impact Analysis (I/A) table on Db2 has been reinstated in the ZMF 8.3 Release. The specific steps required to migrate your existing ZMF Db2 environment to 8.3 depend on the configuration selected from the following list:

Combinations of ZMF features and configurations

- No ZMF Db2 Option License but Using Db2 to Host the Baseline I/A Table
- Full Function ZMF Db2 Option, without ERO, Not Using Db2 to Host the Baseline I/A
- Full Function ZMF Db2 Option, without ERO, Using Db2 to Host the Baseline I/A Table
- Core ZMF Db2 Option, without ERO, Not Using Db2 to Host the Baseline I/A Table
- Core ZMF Db2 Option, without ERO, Using Db2 to Host the Baseline I/A Table
- Full Function ZMF Db2 Option, with ERO, Not Using Db2 to Host the Baseline I/A Table
- Full Function ZMF Db2 Option, with ERO, Using Db2 to Host the Baseline I/A Table
- Core ZMF Db2 Option, with ERO, Not Using Db2 to Host the Baseline I/A Table
- Core ZMF Db2 Option, with ERO, Using Db2 to Host the Baseline I/A Table

Please refer to Migrating ZMF Db2 Options for a comprehensive list of migration steps required for your particular mix.

> ♀ **Notes**
>
> - You must be on a supported version of Db2.
> - The Database Request Modules (DBRMs) that are delivered in the CMNZMF DBRM distribution library have been generated under Db2 version 12. Contact Micro Focus Customer Care if a DBRM is incompatible with the version of Db2 that you use.
> - If you license the Db2 option, refer to the following publication for more information: *ChangeMan ZMF Db2 Option Getting Started Guide*

Step 10: Upgrade ERO

If you license the ChangeMan ZMF Enterprise Release Option (ERO), you must bind the packages and plans associated with ERO Db2 programs. Information about ERO components are provided in the ChangeMan ZMF ERO Getting Started Guide.

This table shows what sample JCL members in the ZMF 8.3 vendor CMNZMF CNTL library you must run to migrate any ZMF 7.1.x release to version 8.3.

Sample JCL Member	Run converting from	Table	Change
AUDALTR2	v7.1.0 - v7.1.1.x	CMNCLODOBJ	Add columns to area audit table to improve ERO audit auto resolve for object dependencies.
CIMALTR4	v7.1.0 - v7.1.1.x	CMNRLSCIM	Add columns to improve detection of version regression (ERR0417).
CIMALTR5	v7.1.0 - v7.1.1.x - V7.1.2.x	CMNRLSCIM	Alter the existing DTM_CHKIN column to make it a WITH TIME ZONE column (requires Db2 v10+).
CIMALTR6	V7.1.0 - V7.1.1.x - V7.1.2.x - V7.1.3.x	CMNRLSCIM	Add column TXT_CHGID to record the userid that last updated the physical component.
CMNLDERT	v7.1.0 - v7.1.1.x - V7.1.2.x	CMNERRORTYPES	Add column TXT_ERROR_AUTO to area audit table to allow suppression of auto resolve for an error number.
HSTALTR4	v7.1.0 - v7.1.1.x	CMNRLSHST	Add columns to release management table to improve detection of version regression (ERR0417).
HSTALTR5	V7.1.0 - V7.1.1.x - V7.1.2.x - V7.1.3.x	CMNRLSHST	Add the following columns to the CMNRLSHST (release management)table: TXT_COMP_TSOI (the userid associated with the last state change) and TXT_COMP_CHGID (the userid that last updated the physical component).

Follow these steps to expand ERO Db2 tables and bind the needed packages and plans:

1. Use existing pre-conversion housekeeping jobs or other JCL to back up ChangeMan ZMF 7.1.x ERO Db2 tables before you begin converting files to version 8.3. An image copy is recommended.

2. Copy the sample JCL members from the ZMF 8.3 vendor CMNZMF CNTL library to your custom CMNZMF CNTL library. Edit the sample JCL members in your custom library.

   a. Code your JOB statement at the top.

   b. Change each of the following to fit the Db2 subsystem where ERO tables reside and to match the Db2 version running on that subsystem:

   - *dsnvrm*.SDSNLOAD in the JOBLIB statement

   - SYSTEM(ssss), where ssss is the Db2 subsystem ID

   - PLAN(DSNTIAvr) for utility DSNTIAD

   - DSNvrm.RUNLIB.LOAD

   - SET CURRENT SQLID = 'auth-id', where auth-id is a user ID that has been granted sufficient authority to execute the Db2 commands in this job

   - CMNx, where x is the one-character subsystem ID that identifies the ChangeMan ZMF instance where the ERO option runs. For example, change CMNx to CMN5 for the ZMF instance with subsystem ID of 5.

3. Submit the first four jobs in the following order:

   a. AUDALTR2

   b. CIMALTR4

   c. CIMALTR5

   d. CIMALTR6

4. When each job completes, examine the job output for problems.

5. Submit job HSTALTR4. When the job completes, examine the job output for problems.

6. Submit job HSTALTR5. When the job completes, examine the job output for problems.

7. Submit job CMNLDERT.

> ⚠️ **Caution**
>
> If you have customized your ERO Audit error messages or return codes in your customized CMNZMF.ERR(ERRORTYP) member, merge these changes into the ERRORTYP member of the CMNZMF.ERR distribution library before you submit job CMNLDERT to reload the CMNERRORTYPES table.

   When the job completes, examine the job output for problems.

8. Follow the instructions in topic "Bind ERO DBRM" in the ChangeMan ZMF ERO Getting Started Guide to bind the SQL in programs CMNDB2RQ, CMNRARTM, CMNRARPT, and CMNRA905 into Db2 packages. Those instructions explain how to customize and execute sample JCL members in the version 8.3 CMNZMF.CNTL library.

9. ERO customers who are upgrading from a 7.1.2 or 7.1.3 instance to 8.3, who are running utility jobs CMNRACHK and CMNRAHAR, must run the BINDPKG step in CMNRAUTB to rebind the Db2 packages associated with these utility programs. There is no need to run the BINDPLN step unless you are changing the PKLIST for Db2 plan CMNUTIL.

   ERO customers who are upgrading from a 7.1.0 or 7.1.1 instance to 8.3 should please contact Micro Focus Customer Care for instructions to bind the CMNRACHK and CMNRAHAR Db2 packages.

   All ChangeMan ZMF Db2 programs are precompiled with VERSION(AUTO) so you can bind SQL before you actually install the load module for the new version of the program.

10. When all jobs have executed successfully, take an image copy of the expanded ERO Db2 tables.

Step 11: Customize SERLCSEC

Before you start this step, look at the information in the Install Guide as the functionality of SERLCSEC is enhanced to now also protect XMLSERV and XML invocations.

Follow these steps:

1. Compare the version of SERLCSEC that is distributed in the ZMF 8.3 SERCOMC ASMSRC library with your current version of SERLCSEC.

2. Copy the SERLCSEC that is distributed in the ZMF 8.3 SERCOMC ASMSRC library to your new 8.3 customized source library.

3. Edit the copy of SERLCSEC in your customized source library as appropriate, and reassemble the program.

Step 12: Customize Other Version 8.3 Components

Find and analyze the components that you customized in the ChangeMan ZMF 7.1.x version that you are upgrading, and decide whether to bring those modifications forward into version 8.3. You may decide that some of your modifications are not appropriate for version 8.3 because:

• ChangeMan ZMF 8.3 includes functions that can replace your custom code.

• Your company does not need the custom function anymore.

• Your custom code may not work with revised functions in version 8.3.

If your custom code will not work with revised functions in version 8.3, consult with Micro Focus Customer Care if you want to explore alternatives.

When you analyze the modifications in the version from which you are upgrading and apply customization to version 8.3, follow these recommendations:

- Before you begin, review Strategies to Reduce Risk.
- Verify each change to be sure you do not mistake customized code for changes that ZMF developers made between versions.
- Thoroughly test customized components on a test ChangeMan ZMF 8.3 instance.

General Procedures For Reapplying Customization

Follow these steps to update your custom libraries:

1. Allocate a new set of the desired custom libraries. Copy any custom components that are all original code into your new custom libraries.

2. If you use the ChangeMan ZMF M+R Option:

    a. On the Versions (SERCDFCM) panel, define your base and derivative libraries as follows:

       Base: Vendor library for old version Drv1: Custom library for old version Drv2: Vendor library for new version

    b. On the Consolidation Workplace panel, reconcile your old customization with the new version 8.3 components.

    c. When you have reconciled all of your customizations, export reconciled components to your new custom library.

    For abbreviated instructions for installing and using the M+R Option, see ChangeMan® M+R Option Quick Start.

3. If you do not use the ChangeMan ZMF M+R Option:

    a. Determine what components are changed in version 8.3 by comparing vendor libraries in the version 7.1.x release that you are upgrading with the components in the corresponding version 8.3 distribution libraries.

    b. Find components in your custom libraries that are changed in version 8.3.

    c. Determine whether your modifications will be valid in version 8.3.

    d. Copy the desired version 8.3 components from the vendor distribution libraries to a new custom library and reapply your modifications.

4. Assemble any custom source members into your custom load library.

Panel Changes

Starting with ChangeMan ZMF 8.1, you can code and invoke exit processing routines in REXX and in any Language Environment (LE)-compliant programming language, such as COBOL and PL/I. Any client that can connect to ZMF, such as ChangeMan ZDD or ChangeMan ZMF for Eclipse, can call these user-written exits. The same, customersupplied, exit code will be executed regardless of the client that is being used.

If an appropriate high-level language exit point is available, custom processing currently being executed through ISPF modification (for example, the addition of ISPF panel exits) can be moved to the high-level language exit. In turn, this will simplify upgrading to new ChangeMan ZMF releases in the future.

ChangeMan ZMF 8.3 provides exit processing routines for the following ZMF functions:

- Build (including Stage, Recompile, and Relink)
- Package Create
- Package Update
- File Tailoring
- Checkout
- Promote/Demote
- Audit
- Freeze
- Approve/Reject
- Revert

Refer to the ChangeMan ZMF High Level Language Functional Exits Getting Started Guide for details.

Most ZMF panels have been changed in version 8.1 and later in the effort to make all ZMF panels follow the IBM guidelines for Common User Access (CUA). CUA guidelines were only partially implemented in the panels of previous ZMF releases.

Exit Changes

Exit changes can result from changes to:

- Exit copybooks
- Exit program source

Follow these steps:

1. Compare the exit copybook and program source modules that are distributed in the ZMF 8.3 CMNZMF ASMCPY and CMNZMF ASMSRC libraries respectively.
2. Copy the needed exits to your customized ZMF 8.3 source library.
3. Edit the exit modules in your customized ZMF 8.3 source library as appropriate, and reassemble them.

Step 13: Replace *somnode* with Variables

The CMN$$DSN member of the CMNZMF SKELS library defines and initializes the data set names for system libraries, Sernet and ChangeMan JOBLIBs and STEPLIBs, and 3rd-party product data sets.

It replaces the *somnode* high-level qualifier definitions, which were embedded in skeletons in previous releases, with variables.

Customize the variable definitions in CMN$$DSN to reflect the data set naming conventions used in your installation.

Step 14: Implement PassTicket Support in Sernet

Background

In ChangeMan ZMF 8.1, PassTickets were implemented for mainframe clients connecting through TCP/IP. The actions detailed below are required for customers that have mainframe users or batch jobs that:

1. Connect to ZMF from LPARs other than the one that hosts the Server started task (that is, those connections that use the ChangeMan ZMF Load Balancing Option (LBO)).
2. Explicitly request TCP/IP connection. For example:

   a. Those customers specifying CON(T) in their connection REXX or CLIST.
   b. Those customers specifying `<method>T</method>` in XML Service requests.

    c. Those customers using the `CONNECT=T` JCL parameter (for example, on program CMNVSRVC execution).

> 💡 **Note**
>
> TCP/IP connections will also be attempted if an XMS (Cross Memory Services) connection attempt fails.

We recommend that you enable PassTicket Support even if you are not explicitly requesting TCP/IP connections.

Required RACF Administration

To enable RACF PassTickets, customers must:

1. Activate the PTKTDATA class by entering:

   ```
   SETROPTS CLASSACT(PTKTDATA) RACLIST(PTKTDATA)
   ```

2. Refresh the PTKTDATA class by entering:

   ```
   SETROPTS RACLIST (PTKTDATA) REFRESH
   ```

3. Create a profile in the PTKTDATA class by entering:

   ```
   RDEFINE PTKTDATA SERNET SSIGNON(user_must_choose) APPLDATA('NO REPLAY PROTECTION')
   ```

The value `SERNET` in the above `RDEFINE` command is mandatory and should not be altered. The `SSIGNON` specification value is a 64-bit Data Encryption Standard (DES) key that must be uniquely assigned by each customer. Refer to the *RACF Command Language Reference Manual* for details.

The PassTicket support information given here is for RACF customers. If you use CA ACF2™ or CA Top Secret®, check the appropriate ACF2 or Top Secret documentation for instructions on implementing PassTicket support. Use the provided RACF instructions as a guide.

Role of the SERSET Utility

Support for PassTickets in Sernet is provided through the execution of the SERSET utility.

Each time the ChangeMan ZMF started task is brought up, the SERSET utility is invoked during the initialization process. This addresses the case where clients connect to a started task running on the same LPAR.

However, the SERSET utility can also be executed as a batch utility, provided in the SERSET member of the SERCOMC CNTL distribution library. If users are signing on to the started task from other LPARs, you need to run the SERSET batch job on all of these other LPARs to provide support for the PassTickets.

The SERSET utility must be executed on all LPARs hosting ChangeMan ZMF users or batch jobs before TCP/IP connection will be possible.

PassTicket Generation

Sernet generates PassTickets when SERCLIEN calls SERXPTIK. The generation process requires authorization (key zero), so SERXPTIK executes as a PC routine to obtain authorization (that is, supervisor state). The sole purpose of the SERSET utility is to implement this routine.

SERCLIEN only needs to know the PC number associated with SERXPTIK to generate a PassTicket. To find this number, SERCLIEN retrieves two system-level tokens, as follows:

Token	Description
SerNet.PTickTok	Contains the `SerNet.PTickX` value.
SerNet.PTickX	Contains the PC number.
	Note: The X in this token name corresponds directly to the TOKEN= value established when SERSET runs.

Failures in PassTicket Generation

SERXPTIK calls the routine anchored in field RCVTPTGN of the RACF CVT. Errors will be returned to the caller of SERCLIEN, as follows:

```
SER6035E Passticket generation failed, RCVTPTGN RC=nnnn
```

RCVTPTGN and its accompanying return codes are documented in the section titled "Using the service to generate a passticket" in the RACF Macros and Interfaces manual.

For more information about PassTicket support in Sernet, refer to the *ChangeMan ZMF Installation Guide*.

Step 15: Synchronize Impact Analysis Data

If you change any library type definitions or baseline library definitions while migrating to version 8.3, you must synchronize your impact analysis data with your current package and component information. It is recommended that this is done during the migration process even if you don't change these definitions.

Synchronizing impact analysis data is a two step process:

1. Execute impact analysis data extract.
2. Load data to the impact analysis LDS.

See topic "Generating and Updating Impact Analysis Data" in the ChangeMan ZMF Administrator's Guide for instructions.

Step 16: Customize User Exit SEREX006

A customized Sernet user exit, SEREX006, is required in ZMF 8.2 Patch 2 and later releases. A sample version of the user exit is provided as member SEREX006 of the SERCOMC ASMSRC distribution library.

Its job is to provide the necessary parameters for Sernet to perform dynamic allocation of user-specific log data sets. Sernet started tasks will abend with code S806-4 if the SEREX006 user exit is missing or with code U0044 if the SEREX006 user exit does not provide all required information.

> ♀ **Note**
>
> The Sernet started tasks will need ALTER access (in RACF) to the high-level qualifier specified.

Step 17: Perform Post-Migration Checks

Perform some checks such as the following after you bring up your migrated ChangeMan ZMF 8.3 instance:

- Check for messages such as the following in your task output to ensure the release numbers reflect ZMF 8.3:

```
CMN7308I CMNSTART ZMF version: 8.3
CMN7300I CMNSTART Initialization in progress; @TCA=198FC000
SER9776I Invoking SERALREP for activity log monthly reporting
SER7301I CMNSTART Loading CMN services; @ADR=00008000
SER6414I SERXMLDB Header: SERNET XML Dsect Cross Reference. Created:
18 Feb 2021 08:42:01. Version: 8205
SER6415I SERXMLDB Version: 8205
```

- Check for obsolete start-up keywords in the task output, such as: SER0804W Obsolete parameter ignored: ECLIPSE

Step 18: Submit Job CMNRPKSN if You License ERO

Customize and submit the job in member CMNRPKSN of the CMNZMF.CNTL library with input of `RLS=*` after all other upgrade activities are complete and your 8.2 Patch 6 task has been started. Failure to do so will result in the following message if you attempt to access a release package: `CMR6504I - No information found for Package search request`.

This utility creates a new Package Master record for each package in an ERO release.

Step 19: Configure Support for IBM Health Checker

Some features of the IBM Health Checker are being supported under ChangeMan ZMF 8.3. If you wish to configure and turn this support on in your ChangeMan ZMF 8.3 instance, follow the instructions in the ChangeMan ZMF Installation Guide. This step is optional.

Step 20: Configure Activity Log Reporting Procedure

You must configure support for the Activity Log Reporting Procedure, default name is SERALREP. To do this follow the instructions in the ChangeMan ZMF Installation Guide. This step is required.

4. Converting ZMF 8.1.x and 8.2 to v.8.3

This chapter provides detailed information about the tasks that you need to perform to convert ChangeMan instances to version 8.3.

Read the assumptions, constraints, and strategies in Migration Strategy, Plans, and Issues before you start work on the tasks in this chapter. Have a comprehensive migration plan in place before you execute conversion tasks. Also ensure you have reliable backups before you execute conversion tasks.

- Conversion Task Dependencies
- Step 1: Apply ChangeMan ZMF License
- Step 2: Build JCL INCLUDEs for JOBLIBs and STEPLIBs
- Step 3: Update Started Procedures and Log-on CLIST
- Step 4: Define Keylists for ISPF Expand
- Step 5: Back Up Version 8.1.x or 8.2 Files
- Step 6: Convert Package Master, Component Master, and SSV Data Sets
- Step 7: Reload XMLSPACE VSAM LDS
- Step 8: Update Sernet Keyword Options
- Step 9: Implement Changes for the Db2 Option
- Step 10: Upgrade ERO
- Step 11: Customize SERLCSEC
- Step 12: Customize Other Version 8.3 Components
- Step 13: Replace somnode with Variables
- Step 14: Update RACF PassTicket Support in Sernet
- Step 15: Synchronize Impact Analysis Data
- Step 16: Customize User Exit SEREX006
- Step 17: Perform Post-Migration Checks
- Step 18: Submit Job CMNRPKSN if You License ERO
- Step 19: Configure Support for IBM Health Checker
- Step 20: Configure Activity Log Reporting Procedure

Conversion Task Dependencies

The steps that you execute in this chapter and the specific actions within each step may depend on the level of the last ChangeMan ZMF maintenance release, patch release, or hotfix that you applied to your system.

- Read the instructions in each step carefully to determine if a specific action applies to you.
- In some steps, such as bringing customization forward and upgrading housekeeping jobs, assess the scope of the action you must take by comparing the vendor libraries at your current release level to the vendor libraries for ChangeMan ZMF 8.3.

If you have questions about whether a step or action applies to you, contact Micro Focus Customer Care.

> ♀ **Note**
>
> If you are migrating from Patch 2 or Patch 3 and are using REST services then the .war files and administration has changed. Refer to the REST Services Getting Started Guide.

Step 1: Apply ChangeMan ZMF License

If you license the ZMF Client Pack, you may need to obtain a new license key to install ZMF 8.3. The new license key provides a number for the ZMF Client Pack that identifies the ChangeMan ZDD and ChangeMan ZMF for ECL products as a single product.

If your license file does not contain a member called LIC00213, you need to request a new license key.

The SER10TY User's Guide provides instructions for applying those licenses.

Step 2: Build JCL INCLUDEs for JOBLIBs and STEPLIBs

You may have a number of jobs already generated (for example, for frozen packages awaiting install) that need to run successfully once you have completed your conversion from version 8.1.x to version 8.3. If these jobs reference skeletons (such as CMN$$JBL) that have hard-coded JOBLIB/STEPLIB concatenations pointing back to version 7.1.x software, they will fail when they run against the upgraded version 8.3 instance.

One way to avoid this failure is to keep the same execution library names and replace the contents with the version 8.3 software.

However, it is safer (and recommended) to keep different versions of ZMF software in different named libraries.

This can also be implemented by the use of aliases for the datasets involved.

A method for addressing this situation by the use of INCLUDE JCL members follows:

1. Ensure you have separate execution libraries for the two different releases of ZMF.

2. Set up two JCL members to be INCLUDEd into CMN$$JBL for each ZMF instance: one for JOBLIB and one for STEPLIB. Prior to any conversion these members will contain version 8.1.x execution libraries.

3. Modify skeleton CMN$$JBL to reference these INCLUDE members based on the subsystem ID of the ZMF for which the job is being generated.

4. Make the modified CMN$$JBL skeleton live in your ZMF instance(s).

> 💡 **Note**
>
> You have to take these steps well in advance of any conversion so that all frozen package jobs will be using the modified CMN$$JBL at the time of the conversion to version 8.3.

5. As part of the upgrade process for each ZMF instance, update the relevant INCLUDE member so that the JOBLIB and STEPLIBs point to version 8.3 execution libraries.

Update Started Procedures and Log-on CLIST

This section tells you how to modify the started procedures and CLIST for the version that you are migrating for use with version 8.3. You can compare the following members between the libraries of the version that you are migrating from, and ZMF 8.3 libraries to see what is changed.

- **SERVER** in the SERCOMC.CNTL library
- **STARTJCL** in the CMNZMF.CNTL library
- **CMNADSP** in the CMNZMF.CNTL library
- **CMN** in the CMNZMF.CLIST library

> 🔥 **Important**
>
> When you create a new DD statement or add a file or library, change the placeholder somnode for the high-level qualifier in the examples shown here to make the library or data set name fit the naming conventions that you use for ChangeMan ZMF.

Started Procedure for ALL, DP, and D Instances

Follow these steps to create a started task procedure for ChangeMan ZMF 8.3:

1. JCL member ACTLOG of the SERCOMC.CNTL distribution library is shipped with ZMF 8.3. Run this job to delete and define a Sernet Activity Log, which is new in ZMF 8.3. This Activity Log data set is then referenced in the following DD statement in the SERVER member of the SERCOMC.CNTL distribution library:

   ```
   //SERALOG DD DISP=SHR,DSN=somnode.ACTLOG .
   ```

   > 💡 **Note**
   >
   > Each instance of the started task requires its own SERALOG DD statement; attempts to share a single data set between multiple started tasks will produce unpredictable results.
   >
   > If the DD statement is missing, the started task will abend with code U4000.

2. In the custom CNTL library that you create for version 8.3, append the JCL in member STARTJCL of the CMNZMF CNTL distribution library to the JCL in member SERVER of the SERCOMC CNTL distribution library. You must do this to enable ChangeMan ZMF to run under the Serena Network (Sernet).

3. Follow the comments in the JCL to customize the started task JCL for version 8.3 as appropriate for your installation standards. You can use the customizations that you made to the started task JCL for the 8.1.x or 8.2.x version that you are upgrading as a guide. However, some of the customizations that you made for version 8.1.x or 8.2.x may no longer be appropriate for version 8.3.

4. We recommend that you make the SYSMDUMP data set a GDG as in the supplied SERVER example to prevent diagnostic information from being overwritten when the Sernet instance is restarted after an abend.

5. If not already present, create a GDG base for the dump capture data set name in the SYSMDUMP DD statement.

6. Copy the customized procedure to the appropriate system PROCLIB library for testing.

Started Procedure for P-Site Instances

Follow similar steps to create the Sernet/ChangeMan ZMF started procedure for the 8.1.x P-site instances that you are upgrading to run version 8.3.

File Tailoring Procedures - CMNADSP

The started procedures defined on panel Global Parameters - Part 2 of 8 (CMNGGP02) are based on JCL sample CMNADSP in the CMNZMF.CNTL library.

Follow similar steps to create your 8.3 file tailoring started procedures.

File Recall Procedure - SERDRT

The started procedure used to recall an ML2 dataset when requested by the Started Task are based on JCL sample SERDRT in the SERCOMC.CNTL library. The number of these procedures is defined by the ASIDS4RECALL parameter in the SERSYSIN DD of the Started Task. The default if omitted is 4, the range is from 2 to 16. The limit can be changed by an operator command. See the Installation Guide for more information on operator commands.

Log-On CLIST

Follow these steps to create the log-on CLIST for ChangeMan ZMF 8.3 instance:

1. Copy the log-on CLIST CMN from the vendor CMNZMF.CLIST library to your custom version 8.3 CMNZMF.CLIST library.

2. In the ALTLIB statement, ensure the somnode values are pointing to your new 8.3 libraries, retain the custom libraries in the order they are in.

3. In the LIBDEF statements, ensure the somnode values are pointing to your new 8.3 libraries, retain the custom libraries in the order they are in. You will allocate and populate the TABLES (ISPTLIB) libraries later in this chapter.

4. Copy the updated CLIST to the CLIST library that is allocated to `//SYSPROC` in your ZMF users' TSO log-on procedure for testing.

Step 4: Define Keylists for ISPF Expand

In ChangeMan ZMF 7.1 and later releases, component names and file names are expanded to accept long zFS component names and path names. The logical unit name for sites is expanded to accept longer transmission vehicle node names. The Db2 Option in ZMF 7.1 and later releases processes the long Db2 object names that are available in Db2 V8 and later.

In the ISPF interface for ChangeMan ZMF 7.1 and later releases, fields for input and display of this longer data are extended to the limit of the 80 character panel width. If the data is longer than the available space, you can use the PF11 and PF10 keys to scroll right and left respectively in the panel fields.

The panels also respond to the ISPF Expand function to open a pop-up for input or display of the entire long name field. To execute the Expand function from the keyboard, the panels in ChangeMan ZMF 7.1 and later releases call a new keylist that assigns PF4 to the Expand function.

The job in member CMNDTL of the CMNZMF.CNTL distribution library builds keylist CMNKEYS in the TABLES library you added to the ChangeMan ZMF started procedures and to the connect CLIST.

Follow these steps to build ISPF keylist CMNKEYS in a new ISPF TABLES library.

1. Copy member CMNDTL from the version 8.3 CMNZMF.CNTL distribution library to your custom ZMF 8.3 CMNZMF.CNTL library. Edit CMNDTL in your custom CMNZMF.CNTL library.

2. Code your JOB statement at the top.

3. Change DD statement DTLGML to point to the CMNZMF SAMPLES distribution library.

4. Change all remaining occurrences of placeholder somnode to the high level qualifier you use for ISPF system libraries.

5. Submit job CMNDTL.

6. Examine job output for error messages. The ISPLOG should report 0 error(s) found. Error message ISPC506E in the SYSTSPRT likely means that the member in the SAMPLIB library was not found.

Example of ISPLOG from Job Step DTL

```
Time          *** ISPF transaction log ***       Userid: USER015    Date: 19/02/20     Page: 1

21:49   Start of ISPF Log - - - - Session # 1 ------------------------------------------------------
21:49      TSO - Command -     - ISPDTLC CMNELSK (PROFDDN=* CUASUPP REPLACE)
21:49                          - ISPC028AI: Total of 2 message(s) have been suppressed.
21:49                          - ISPC029AI: Total of 2 warning(s) and 0 error(s) found.
21:49   End of ISPF Log - - - - Session # 1 ------------------------------------------------------
Time          *** ISPF transaction log ***       Userid: USER015    Date: 19/02/20     Page: 1

21:49   Start of ISPF Log - - - - Session # 2 ------------------------------------------------------
21:49      TSO - Command -     - ISPDTLC CMNEROK (PROFDDN=* CUASUPP REPLACE)
21:49                          - ISPC028AI: Total of 2 message(s) have been suppressed.
21:49                          - ISPC029AI: Total of 2 warning(s) and 0 error(s) found.
21:49   End of ISPF Log - - - - Session # 2 ------------------------------------------------------
Time          *** ISPF transaction log ***       Userid: USER015    Date: 19/02/20     Page: 1

21:49   Start of ISPF Log - - - - Session # 3 ------------------------------------------------------
21:49      TSO - Command -     - ISPDTLC CMNMREK (PROFDDN=* CUASUPP REPLACE)
21:49                          - ISPC028AI: Total of 2 message(s) have been suppressed.
21:49                          - ISPC029AI: Total of 2 warning(s) and 0 error(s) found.
21:49   End of ISPF Log - - - - Session # 3 ------------------------------------------------------
```

Example of SYSTSPRT from job step DTL

```
READY
    ISPSTART CMD(ISPDTLC CMNELSK (PROFDDN=* CUASUPP REPLACE))
ISPC021I: ISPDTLC ISPF 7.3 Dialog Tag Language Conversion Utility
ISPC020AI: ISPF Dialog Tag Language Conversion Utility for ISPF 7.3. Current
    APAR level: PID. Current PTF number: 3 Sep 2017.
ISPC022I: Converting source file "CMNTP.CMN820.SAMPLES(CMNELSK)"...
ISPC028I: 2 message(s) have been suppressed.
ISPC029I: 2 warning(s) and 0 error(s) found.
ISPD117
The initially invoked CLIST ended with a return code = 1
    USER015.USER0154.J0281166.D0000103.? was preallocated (no free was done).
READY
    ISPSTART CMD(ISPDTLC CMNEROK (PROFDDN=* CUASUPP REPLACE))
ISPC021I: ISPDTLC ISPF 7.3 Dialog Tag Language Conversion Utility
ISPC020AI: ISPF Dialog Tag Language Conversion Utility for ISPF 7.3. Current
    APAR level: PID. Current PTF number: 3 Sep 2017.
ISPC022I: Converting source file "CMNTP.CMN820.SAMPLES(CMNEROK)"...
ISPC028I: 2 message(s) have been suppressed.
ISPC029I: 2 warning(s) and 0 error(s) found.
ISPD117
The initially invoked CLIST ended with a return code = 1
    USER015.USER0154.J0281166.D0000103.? was preallocated (no free was done).
READY
    ISPSTART CMD(ISPDTLC CMNMREK (PROFDDN=* CUASUPP REPLACE))
ISPC021I: ISPDTLC ISPF 7.3 Dialog Tag Language Conversion Utility
ISPC020AI: ISPF Dialog Tag Language Conversion Utility for ISPF 7.3. Current
    APAR level: PID. Current PTF number: 3 Sep 2017.
ISPC022I: Converting source file "CMNTP.CMN820.SAMPLES(CMNMREK)"...
ISPC028I: 2 message(s) have been suppressed.
ISPC029I: 2 warning(s) and 0 error(s) found.
ISPD117
The initially invoked CLIST ended with a return code = 1
    USER015.USER0154.J0281166.D0000103.? was preallocated (no free was done).
READY
END
```

Step 5: Back Up Version 8.1.x or 8.2 Files

Use your existing preconversion housekeeping jobs or other JCL to back up ChangeMan ZMF files and clear log files for the 8.1.x or 8.2.x release that you are migrating from, before you begin converting files for version 8.3.

The following table lists the files you should back up. To accomplish this task, use the sample JCL members in the vendor CMNZMF CNTL library that was distributed with the ZMF 8.1.x or 8.2.x instance that you are migrating from.

Last DSN Node of the File to be Backed Up	Description	Type	Vendor CMNZMF.CNTL Job of Your Existing, Preconversion Instance
CMNRECV	Package Master, Short-Name Component Master, and Long-Name Component Master Forward Recovery File	VSAM	CLEARRCV Note 1
CMNLOG	Activity Log	VSAM	CLEARLOG

Last DSN Node of the File to be Backed Up	Description	Type	Vendor CMNZMF.CNTL Job of Your Existing, Preconversion Instance
CMNDELAY	Delay file	VSAM	See Note 3
IADSP	Impact Analysis	LDS	LDSBKUP
IALOG	Impact Analysis Forward Recovery Log	QSAM	LDSLOGR
CMNSVSM	Staging Versions Site Master	VSAM	SVBKUP Note 4
CMNSVOM	Staging Versions Object Master	VSAM	SVBKUP Note 4
CMNSVDM	Staging Versions Delta Master	VSAM	SVBKUP
CMNSVAM	Staging Versions Archive Master	VSAM	SVBKUP

- Note 1: Please read the comments in the supplied sample JCL members.

- Note 2: The BACKUP sample job will do the files listed, and optionally has a step to take an image copy of SQL Process Admin tables if so desired.

- Note 3: There is no sample job for backing up the CMNDELAY file. Use IDCAMS with output QSAM file attributes RECFM=VB, LRECL=5000, BLKSIZE=32700.

- Note 4: These jobs do the same work as the second step of the BACKUP job. You may use either.

Step 6: Convert Package Master and Create a NewForward Recovery File

The format of the Package Master changed in ZMF 8.1.2. Read the CMNZMF CNTL distribution library member $$$INDEX for a short description of all sample JCL.

Customize and execute the following jobs from the CMNZMF.CNTL and SERCOMC.CNTL distribution libraries, as appropriate. There are steps to do the various conversions from version to version. See the comments in the supplied sample JCL:

ZMF 8.2 Patch 4 thru 8.2 Patch 7 to ZMF 8.3

Job Name	Description
CMVCONVG	For non-P sites. Run the job in member CMVCONVG to convert master files from releases 8.2 Patch 4 thru 8.2 Patch 7 to release 8.3.
CMVCONPG	For P sites. Run the job in member CMVCONVG to convert master files from releases 8.2 Patch 4 thru 8.2 Patch 7 to release 8.3.

ZMF 8.2 Patch 2 / Patch 3 to ZMF 8.3

Job Name	Description
CMVCONVF	For non-P sites. Run the job in member CMNCONVF to convert master files from release 8.2.0.02/03 to release 8.3.
CMVCONPF	For P-sites. Run the job in member CMNCONPF to convert master files from release 8.2.0.02/03 to release 8.3.

ZMF 8.2 Patch 1 to ZMF 8.3

Job Name	Description
CMVCONVE	For non-P sites. Run the job in member CMNCONVE to convert master files from release 8.2.0.01 to release 8.3.
CMVCONPE	For P sites. Run the job in member CMNCONPE to convert master files from release 8.2.0.01 to release 8.3.
ACTLOG	This job deletes and defines the SerNet activity log, required from ZMF 8.2 Patch 2 onwards. The resulting dataset is allocated to the server started task under the SERALOG DD statement. See Started Procedure for ALL, DP, and D Instances.

ZMF 8.2 to ZMF 8.3

Job Name	Description
CMVCONVD	For non-P sites.Run the job in member CMNCONVD to convert the ZMF 8.2 Package Master files to a format that supports ZMF 8.3.
CMVCONPD	For P-sites. Run the job in member CMNCONVD to convert the ZMF 8.2 Package Master file for a P-site to a format that supports ZMF 8.3.

Job Name	Description
ACTLOG	Deletes and defines the SerNet activity log, required from ZMF 8.2 Patch 2 onwards. The resulting dataset is allocated to the server started task under the SERALOG DD statement. See Started Procedure for ALL, DP, and D Instances.

ZMF 8.1.2 thru 8.1.4 Patch 1 to ZMF 8.3

Job Name	Description
CMVCONVC	For non-P sites. Run the job in CMVCONVC to convert the ZMF v8.1.2 thru v8.1.4 Patch 1 Package Master files that supports ZMF 8.3.
CMVCONPC	For P-sites. Run the job in CMVCONPC to convert the ZMF v8.1.2 through v8.1.4 Patch 1 Package Master file for a P-site that supports ZMF 8.3.
ACTLOG	Deletes and defines the SerNet activity log, required from ZMF 8.2 Patch 2 onwards. The resulting dataset is allocated to the server started task under the SERALOG DD statement. See Started Procedure for ALL, DP, and D Instances.

ZMF 8.1.1 thru ZMF 8.1.1.01 to ZMF 8.3

Job Name	Description
CMVCONVB	For non-P sites.ZMF 8.1.1 thru 8.1.1.01 Package and Component Master files to a format that supports ZMF 8.3.
CMVCONPB	For P-sites. Run the job in member CMVCONPB to convert the ZMF 8.1.1 thru 8.1.1.01 Package Master file for a P-site to a format that supports ZMF 8.3.
ACTLOG	Deletes and defines the SerNet activity log, required from ZMF 8.2 Patch 2 onwards. The resulting dataset is allocated to the server started task under the SERALOG DD statement. See Started Procedure for ALL, DP, and D Instances.

ZMF 8.1.0 thru 8.1.0.01 to ZMF 8.3

Job Name	Description
CMVCONVA	For non-P sites.This job converts the ZMF 8.1.0 thru 8.1.0.01 package and component master files to a format that supports v8.3.
CMVCONPA	For P-sites. This job converts the ZMF 8.1.0 thru 8.1.0.01 package master file to a format that can be used to support v8.3.

Job Name	Description
ACTLOG	Deletes and defines the SerNet activity log, required from ZMF 8.2 Patch 2 onwards. The resulting dataset is allocated to the server started task under the SERALOG DD statement. See Started Procedure for ALL, DP, and D Instances.

> 💡 **Note**
>
> The CMNSVAM SSV archive master RECORDSIZE changed in ZMF 8.3. The conversion jobs create a new, empty CMNSVAM and initialize it. Reporting and recovery may be performed on backups of older archived data by restoring to a temporary archive VSAM data set defined with RECORDSIZE(3700 22576) and running the SVARCLST and SVARCREC jobs against the temporary archive VSAM data set.

Step 7: Reload XMLSPACE VSAM LDS

The contents of sequential file MAPDATA and VSAM LDS XMLSPACE have changed in ChangeMan ZMF 8.3.

Follow this procedure to define and populate the MAPDATA sequential file and the XMLSPACE VSAM LDS:

1. Copy member XMLLOAD from the SERCOMC CNTL distribution library to your ZMF 8.3 custom SERCOMC CNTL library. Edit XMLLOAD in your ZMF 8.3 custom SERCOMC.CNTL library.
2. Replace ZMF 8.1.* library names with ZMF 8.3 library names.
3. Submit job XMLLOAD.
4. Examine the job output for problems:

    a. Examine the output of job step RECEIVE in SYSTSPRT.

    b. Examine the job's JESMSGLG to verify that job step LOAD completed with RC=00.

    c. Browse member XMLDATA in your SERCOMC CNTL library. Note the date and time in this line of text near the top of the data set:

    ```
    SERNET XML Dsect Cross Reference. Created: dd mmm yyyy hh:min:ss
    ```

    d. Browse the new MAPDATA file and look for the same text at the top of that file. The date and time should be the same as in the XMLDATA member.

Step 8: Update Sernet Keyword Options

Sernet keyword options are input to Sernet through one of these methods:

- In the PARM= parameter for program SERVER in the started procedure (SERVER member of the SERCOMC CNTL library).
- In the library at the DDname specified in keyword parameter DDNAME= in the PARM=parameter for program SERVER in the started procedure.

Sernet keyword options are described in an appendix of the *ChangeMan ZMF Installation Guide*.

Update Sernet keyword options as appropriate.

Step 9: Implement Changes for the Db2 Option

Steps required to migrate the ZMF Db2 option vary depending on which configuration you wish to implement and on which other, Db2-reliant, ZMF features are to be used (e.g. ERO).

The option to host the Impact Analysis (I/A) table on Db2 has been reinstated in the ZMF 8.3 Release. The specific steps required to migrate your existing ZMF Db2 environment to 8.3 depend on the configuration selected from the following list:

Combinations of ZMF features and configurations

- No ZMF Db2 Option License but Using Db2 to Host the Baseline I/A Table
- Full Function ZMF Db2 Option, without ERO, Not Using Db2 to Host the Baseline I/A
- Full Function ZMF Db2 Option, without ERO, Using Db2 to Host the Baseline I/A Table
- Core ZMF Db2 Option, without ERO, Not Using Db2 to Host the Baseline I/A Table
- Core ZMF Db2 Option, without ERO, Using Db2 to Host the Baseline I/A Table
- Full Function ZMF Db2 Option, with ERO, Not Using Db2 to Host the Baseline I/A Table
- Full Function ZMF Db2 Option, with ERO, Using Db2 to Host the Baseline I/A Table
- Core ZMF Db2 Option, with ERO, Not Using Db2 to Host the Baseline I/A Table
- Core ZMF Db2 Option, with ERO, Using Db2 to Host the Baseline I/A Table

Please refer to Migrating ZMF Db2 Options for a comprehensive list of migration steps required for your particular mix.

> ♡ **Notes**
>
> - You must be on a supported version of Db2.
>
> - The Database Request Modules (DBRMs) that are delivered in the CMNZMF DBRM distribution library have been generated under Db2 version 12. Contact Micro Focus Customer Care if a DBRM is incompatible with the version of Db2 that you use.
>
> - If you license the Db2 option, refer to the following publication for more information: *ChangeMan ZMF Db2 Option Getting Started Guide*

Step 10: Upgrade ERO

If you license the ChangeMan ZMF Enterprise Release Option (ERO), you must bind the packages and plans associated with ERO Db2 programs. Information about ERO components are provided in the ChangeMan ZMF ERO Getting Started Guide.

1. Use existing pre-conversion housekeeping jobs or other JCL to back up ChangeMan ZMF 8.1.x ERO Db2 tables before you begin converting files to version 8.3. An image copy is recommended.

2. ERO customers who are upgrading from a pre-8.1.3 version of ChangeMan ZMF must run the job in member CIMALTR5 of the CMNZMF.CNTL distribution library to avoid having to recycle ChangeMan ZMF during the Autumn time change.

3. ERO sites must run the job in member CIMALTR6 of the version 8.3 CMNZMF.CNTL distribution library. This job adds the column TXT_CHGID to the CMNRLSCIM table. The TXT_CHGID column identifies the userid that last updated the physical component. This job updates the CMNRLSCIM table that exists in releases 8.1.2 through 8.2.0 to support release 8.2 Patch 2+.

4. ERO customers who are upgrading from an 8.1.x or 8.2 instance to 8.3 must run the job in member HSTALTR5 of the 8.3 CMNZMF.CNTL distribution library. This job will alter the ERO Release History table (CMNRLSHST) to add column TXT_COMP_CHGID. The new column will record the userid that performed the last state change to the physical component.

5. Submit job CMNLDERT.

> ⚠ **Caution**
>
> If you have customized your ERO Audit error messages or return codes in your customized CMNZMF.ERR(ERRORTYP) member, merge these changes into the ERRORTYP member of the CMNZMF.ERR distribution library before you submit job CMNLDERT to reload the CMNERRORTYPES table.

When the job completes, examine the job output for problems.

6. Follow the instructions in topic "Bind ERO DBRM" in the ChangeMan ZMF ERO Getting Started Guide to bind the SQL in programs MNDB2RQ, CMNRARTM, CMNRARPT, and CMNRA905 into Db2 packages. Those instructions explain how to customize and execute sample JCL members in the CMNZMF CNTL library.

7. ERO customers who are upgrading from an 8.1.x or 8.2 instance to 8.3, who are running utility jobs CMNRACHK and CMNRAHAR, must run the BINDPKG step in CMNRAUTB to rebind the Db2 packages associated with these utility programs. There is no need to run the BINDPLN step unless you are changing the PKLIST for Db2 plan CMNUTIL.

   All ChangeMan ZMF Db2 programs are precompiled with VERSION(AUTO) so you can bind SQL before you actually install the load module for the new version of the program..

8. When all jobs have executed successfully, take an image copy of the expanded ERO Db2 tables.

9. If you have questions about whether a step or action applies to you, contact Micro Focus Customer Care.

Step 11: Customize SERLCSEC

Before you start this step, look at the information in the Install Guide as the functionality of SERLCSEC is enhanced to now also protect XMLSERV and XML invocations.

Follow these steps:

1. Compare the version of SERLCSEC that is distributed in the ZMF 8.3 SERCOMC ASMSRC library with your current version of SERLCSEC.

2. Copy the SERLCSEC that is distributed in the ZMF 8.3 SERCOMC ASMSRC library to your customized source library.

3. Edit the copy of SERLCSEC in your customized source library as appropriate, and reassemble the program.

Step 12: Customize Other Version 8.3 Components

Find and analyze the components that you customized in the ChangeMan ZMF version that you are upgrading, and decide whether to bring those modifications forward into version 8.3. You may decide that some of your modifications are not appropriate for version 8.3 because:

ChangeMan ZMF 8.3 includes functions that can replace your custom code.

Your company does not need the custom function anymore.

Your custom code may not work with revised functions in version 8.3.

If your custom code will not work with revised functions in version 8.3, consult with Micro Focus Customer Care if you want to explore alternatives.

When you analyze the modifications in the version from which you are upgrading and apply customization to version 8.3, follow these recommendations:

Before you begin, review Strategies to Reduce Risk.

Verify each change to be sure you do not mistake customized code for changes that ZMF developers made between versions.

Thoroughly test customized components on a test ChangeMan ZMF 8.3 instance.

General Procedures For Reapplying Customization

Follow these steps to update your custom libraries:

1. Allocate a new set of the desired custom libraries. Copy any custom components that are all original code into your new custom libraries.

2. If you use the ChangeMan ZMF M+R Option:

   a. On the Versions (SERCDFCM) panel, define your base and derivative libraries as follows:

   Base: Vendor library for old version **Drv1:** Custom library for old version **Drv2:** Vendor library for new version

   b. On the Consolidation Workplace panel, reconcile your old customization with the new version 8.3 components.

   c. When you have reconciled all of your customizations, export reconciled components to your new custom library.

   For abbreviated instructions for installing and using the M+R Option, see Appendix A, ChangeMan® M+R Option Quick Start.

3. If you do not use the ChangeMan ZMF M+R Option:

a. Determine what components are changed in version 8.3 by comparing vendor libraries in the version 7.1.x release that you are upgrading with the components in the corresponding version 8.3 distribution libraries.

b. Find components in your custom libraries that are changed in version 8.3.

c. Determine whether your modifications will be valid in version 8.3.

d. Copy the desired version 8.3 components from the vendor distribution libraries to a new custom library and reapply your modifications.

4. Assemble any custom source members into your custom load library.

Panel Changes

Starting with ChangeMan ZMF 8.1, you can code and invoke exit processing routines in REXX and in any Language Environment (LE)-compliant programming language, such as COBOL and PL/I. Any client that can connect to ZMF, such as ChangeMan ZDD or ChangeMan ZMF for Eclipse, can call these user-written exits. The same, customersupplied, exit code will be executed regardless of the client that is being used.

If an appropriate high-level language exit point is available, custom processing currently being executed through ISPF modification (for example, the addition of ISPF panel exits) can be moved to the high-level language exit. In turn, this will simplify upgrading to new ChangeMan ZMF releases in the future.

ChangeMan ZMF 8.3 addresses requirements in the following ZMF functions:

- Build (Including checkin, build, recompile, relink, delete)
- Package Create (Initial create of a package)
- Package Update (Subsequent update of package attributes)
- File Tailoring (Define customized ISPF variables for file tailoring)
- Checkout (Component Checkout from baseline/promotion)
- Promote/Demote (Promotion and demotion of components)
- Audit (Audit job submission and audit process)
- Freeze (Package freeze and selective unfreeze/refreeze)
- Approve/Reject (Package approve and reject)
- Revert (Package revert)
- Package Syslib(Package syslib list service)
- Scratch/Rename (Utility functions)

Refer to the ChangeMan ZMF High Level Language Functional Exits Getting Started Guide for details.

Most ZMF panels were changed in the effort to make all ZMF panels follow the IBM guidelines for Common User Access (CUA). CUA guidelines were only partially implemented in the panels of previous ZMF releases.

Exit Changes

Exit changes can result from changes to:

- Exit copybooks
- Exit program source

Follow these steps:

1. Compare the exit copybook and program source members in your current libraries to the members that are distributed in the ZMF 8.3 CMNZMF ASMCPY and CMNZMF ASMSRC libraries respectively.
2. Copy the needed exits to your customized ZMF 8.3 source library.
3. Edit the exit modules in your customized ZMF 8.3 source library as appropriate, and reassemble them.

Step 13: Replace *somnode* with Variables

The CMN$$DSN member of the CMNZMF SKELS library defines and initializes the data set names for system libraries, Sernet and ChangeMan JOBLIBs and STEPLIBs, and 3rd-party product data sets.

It replaces the *somnode* high-level qualifier definitions, which were embedded in skeletons in previous releases, with variables.

Customize the variable definitions in CMN$$DSN to reflect the data set naming conventions used in your installation.

Step 14: Update RACF PassTicket Support in Sernet

Refreshing SERXPTIK

You will need to refresh SERXPTIK. SERSET loads SERXPTIK into common storage. By design, SERXPTIK is a very small piece of code that should rarely change. On occasion, however, it may need to be refreshed without an IPL.

To do this, rerun SERSET ensuring it will pick up the new version of SERXPTIK from STEPLIB and specifying a different TOKEN= value. For example, specifying TOKEN=B will result in the following message sequence:

```
SER1704I CSVDYLPA loaded SERXPTIK @ xxxxxxxx
SER1708I SerNet.PtickB token created
SER1708I SerNet.PtickTok token created
SER1709I Passticket support enabled
```

Once SERSET has executed, SerNet.PTickTok will point at SerNet.PTickB. This will cause SERCLIEN to invoke the new version of SERXPTIK with a different PC number.

The TOKEN= parameter will accept any value from A-Z and 0-9 but any use beyond A and, rarely, B would be highly unusual.

Step 15: Synchronize Impact Analysis Data

If you change any library type definitions or baseline library definitions while migrating to version 8.3, you must synchronize your impact analysis data with your current package and component information. It is recommended that this is done during the migration process even if you don't change these definitions.

Synchronizing impact analysis data is a two step process:

1. Execute impact analysis data extract.

2. Load data to the impact analysis LDS.

See topic "Generating and Updating Impact Analysis Data" in the ChangeMan ZMF Administrator's Guide for instructions.

Step 16: Customize User Exit SEREX006

A customized Sernet user exit, SEREX006, has been required since ZMF 8.1.1. You should reassemble your source in 8.3 to be sure that you use the new SEREX006 copybook. A sample version of the user exit is provided as member SEREX006 of the SERCOMC ASMSRC distribution library.

SEREX006 provides the necessary parameters for Sernet to perform dynamic allocation of user-specific log data sets. Sernet started tasks will abend with code S806-4 if the SEREX006 user exit is missing or with code U0044 if the SEREX006 user exit does not provide all required information.

> 💡 **Note**
>
> The Sernet started tasks will need ALTER access (in RACF) to the high-level qualifier specified.

Step 17: Perform Post-Migration Checks

Perform some checks such as the following after you bring up your migrated ChangeMan ZMF 8.3 instance:

- Check for messages such as the following in your task output to ensure the release numbers reflect ZMF 8.3:

```
CMN7308I CMNSTART ZMF version: 8.3
CMN7300I CMNSTART Initialization in progress; @TCA=198FC000
SER9776I Invoking SERALREP for activity log monthly reporting
SER7301I CMNSTART Loading CMN services; @ADR=00008000
SER6414I SERXMLDB Header: SERNET XML Dsect Cross Reference. Created:
18 Feb 2021 08:42:01. Version: 8205
SER6415I SERXMLDB Version: 8205
```

- Check for obsolete start-up keywords in the task output, such as:
```
SER0804W Obsolete parameter ignored: ECLIPSE
```

Step 18: Submit Job CMNRPKSN if You License ERO

Customize and submit the job in member CMNRPKSN of the CMNZMF.CNTL library with input of RLS=* after all other upgrade activities are complete and your 8.2 Patch 6 task has been started. Failure to do so will result in the following message if you attempt to access a release package:

`CMR6505I - No information found for Package search request`.

This utility creates a new Package Master record for each package in an ERO release.

Step 19: Configure Support for IBM Health Checker

Some features of the IBM Health Checker are being supported under ChangeMan ZMF 8.3. If you wish to configure and turn this support on in your ChangeMan ZMF 8.3 instance, follow the instructions in the ChangeMan ZMF Installation Guide. This step is optional.

Step 20: Configure Activity Log Reporting Procedure

You must configure support for the Activity Log Reporting Procedure, default name is SERALREP. To do this follow the instructions in the ChangeMan ZMF Installation Guide. This step is required.

5. Migrating ZMF Db2 Options

These instructions include the installation or migration tasks required to achieve the target configuration. They are organized according to the type of license you hold.

> 💡 **Note**
>
> These tasks only relate to the various Db2 related ZMF configurations and are to be executed in addition to anything required by the base product ZMF install and migration.

The ZMF Db2 option can now be configured so that only "core" features are available, i.e. package bind automation (see the Db2 getting started guide for further information). This allows customers to avoid having to create and maintain the extra ZMF admin tables required to support the "full function" Db2 option.

The new option to allow Db2 to host the baseline Impact Analysis database will be required by customers whose I/A data volumes exceed the physical limit of the dataspace. A specialist RDBMS is better placed to manage such large volumes of data. If you think you are in such a position (as regards data volumes), first be sure to discard all irrelevant relationship data by the judicious use of CMNEX022. (Ask support for help with this if necessary).

> 💡 **Note**
>
> As with any collection of Db2 data, it is important to help Db2 choose the optimal path for access to the data. To this end, it is important to use the RUNSTATS utility, with the INDEX clause, to update catalog statistics followed by a re-binding of the affected packages on a regular basis, but especially following on from an initial load of data to the relevant tables.

No ZMF Db2 Option License but Using Db2 to Host the Baseline I/A Table

New Install

Note that the following steps 1,2, and 3 are part of the base ZMF install process. You can save time by following these instructions during that part of your install. However, you can also execute the "Follow-on step:" on page 61 at some later time if you didn't follow steps 1,2,3 during the base product installation.

1. INITIAL1—This sample JCL member is customized and runs as part of the creation of the vsam master files, which support a new installation of ZMF. For this task, you should remove the step which allocates the I/A dataspace LDS (stepname IADSP) prior to running INITIAL1.

2. INITIAL2—This is the job run after INITIAL1. For this task, you need to remove the step which initializes the I/A dataspace LDS (stepname CMNIAIN) prior to running INITIAL2.

3. STARTJCL—This sample member forms the basis for your ZMF started task procedure. For this task, you need to edit your ZMF proc to remove the CMNIALOG and CMNIMPCT ddnames prior to attempting to start the ZMF stc for the first time.

4. CMNDB2IA—This sample JCL member creates the Db2 tables, binds the packages, and binds the plan to enable the baseline I/A table to be hosted by Db2.

5. Global parms update for Db2 for I/A—The ZMF subsystem must be started up and the global parms updated to indicate that you wish to use Db2 for baseline I/A. The global parm for this is located (via ISPF option A.G.1) on this panel:

```
CMNGGP03                 Global Parameters - Part 3 of 8
Command ===>
Enter "/" to select option
        Baselines
        / Stacked Reverse Delta
        / Panvalet
        User defined
        / Librarian
        Librarian Access Method (LAM)
        Notification Vehicles
        / Email
        / Batch
        Other options
        / Use primary Db2 subsystem for I/A: D10L
        Require CR number
        Require Department
        / Disable installation calendar
        / Allow temporary packages
        / Process participating packages by install date
        / Hierarchical approval process
        / Use global notification file
        / Allow application update to file
        Force display of global notification file
Global notification file . . CMNDEV.CMNSYS.U900DP.GNFFILE
```

This setting takes effect only after a restart of the ZMF started task.

6. IMPACTD2—This job runs the mass data extraction process for an installation where Db2 has been used for the I/A database. It produces the usual three flat files (i.e. BUNSPACE, CMPSPACE, and RELSPACE).

7. CMNDB2IL—This job loads the I/A data from the three flat files into the set of Db2 tables that are hosting the I/A database.

8. You should take your usual measures to ensure the best access paths are chosen by Db2 for this new data. This would usually include using the RUNSTATS utility, with the INDEX clause, to update the stats for the tablespaces involved (i.e. those created when you ran CMNDB2IA) and then rebinding the packages involved (those you originally bound using sample JCL member CMNDB2IA). As with any collection of Db2 table data, using the RUNSTATS/rebind mechanism is something you should do on a regular basis.

Follow-on step:

If you didn't follow the instructions in steps 1, 2, and 3 during the initial product install, you can now delete the redundant I/A LDS and log dataset and remove references to them from your ZMF stc procedure.

Migration from a prior version of ZMF

If you're migrating from a prior version of ZMF, you most likely have the I/A dataspace in place and will be making a conscious decision to convert to using Db2. We recommend moving to Db2 some time after you have successfully migrated to this latest release of ZMF. However, if you must do this at the same time, perform the following steps:

1. STARTJCL—This sample member forms the basis for your ZMF started task procedure. For this task, you need to edit your ZMF proc to remove the CMNIALOG and CMNIMPCT ddnames prior to attempting to start the migrated ZMF stc for the first time.

2. CMNDB2IA—This sample JCL member creates the Db2 tables, binds the packages, and binds the plan to enable the baseline I/A table to be hosted by Db2.

3. LDSUNLD or IMPACTD2—LDSUNLD is a sample job that unloads your existing I/A dataspace to flat files. IMPACTD2 creates these flat files by running the batch data extract (CMNIA000). Choose one of these in order to create the flat files to be used to load the new Db2 database. Run IMPACTD2 only if you have made application admin changes (e.g. new appls, new baselines) as part of the conversion (this is not recommended). Either job produces the usual three flat files (i.e. BUNSPACE, CMPSPACE, and RELSPACE).

4. CMNDB2IL—This job loads the I/A data from the three flat files into the set of Db2 tables which are hosting the I/A database.

5. You should take your usual measures to ensure the best access paths are chosen by Db2 for this new data. This would usually include using the RUNSTATS utility, with the INDEX clause, to update the stats for the tablespaces involved (i.e. those created when you ran CMNDB2IA) and then rebinding the packages involved (those you originally bound using sample JCL member CMNDB2IA). As with any collection of Db2 table data, using the RUNSTATS/rebind mechanism is something you should do on a regular basis.

6. Global parms update for Db2 for I/A—The ZMF subsystem must be started up and the global parms updated to indicate that you want to use Db2 for baseline I/A. The global parm for this is located (via ISPF option A.G.1) on this panel:

```
CMNGGP03                 Global Parameters - Part 3 of 8
Command ===>
Enter "/" to select option
    Baselines
    / Stacked Reverse Delta
    / Panvalet
      User defined
    / Librarian
      Librarian Access Method (LAM)
    Notification Vehicles
    / Email
    / Batch
    Other options
    / Use primary Db2 subsystem for I/A: D10L
      Require CR number
      Require Department
    / Disable installation calendar
    / Allow temporary packages
    / Process participating packages by install date
    / Hierarchical approval process
    / Use global notification file
    / Allow application update to file
      Force display of global notification file
Global notification file . . CMNDEV.CMNSYS.U900DP.GNFFILE
```

This setting is in effect only after a restart of the ZMF started task.

After you are satisfied that all is well, you can delete the now redundant I/A LDS and log dataset.

Full Function ZMF Db2 Option, without ERO, Not Using Db2 to Host the Baseline I/A

New Install

1 DB2OPTN/R—This sample job creates the full function ZMF Db2 option admin tables, binds the related packages, and binds the plan for full function ZMF Db2 option support. DB2OPTNR is for P-sites, and DB2OPTN is for all other types of sites.

Migration from a prior version of ZMF

If you are migrating from a prior version, you most likely have the ZMF Db2 option tables in place (depending on which release of ZMF they are coming from).

1. DB2OPTN/R—This sample job creates the full function ZMF Db2 option admin tables, binds the related packages, and binds the plan for full function ZMF Db2 option support. DB2OPTNR is for P-sites, and DB2OPTN is for all other types of sites. If you already have the tables in place, remove the TABLES step from the job (i.e., do not drop/create your tables).

2. DB2ADMGT—This sample JCL shows how to reformat an existing CMNADMIN_GENERAL table to support ZMF version from 8.2.0 or greater. You need run it only if you are migrating from a version earlier than 8.2.0.

3. DB2ALTER—This sample JCL shows how to add the columns to an existing CMNADMIN_NAMED table to support ZMF version 8.2 patch 4 or greater. You need to run it only if you are migrating from a version earlier than 8.2 patch 4.

Full Function ZMF Db2 Option, without ERO, Using Db2 to Host the Baseline I/A Table

New Install

Note that steps 1, 2, and 3 are part of the base ZMF install process. You can save time by following these instructions during that part of your install. However, if you didn't follow steps 1, 2, and 3 during the base product installation, you can also execute the follow-on step (see below) at some later time.

1. INITIAL1—This sample JCL member is customized and run as part of the creation of the vsam master files that will support a new installation of ZMF. For this task, you should remove the step that allocates the I/A dataspace LDS (stepname IADSP) prior to running INITIAL1.

2. INITIAL2—This is the job run after INITIAL1. For this task, you need to remove the step which initializes the I/A dataspace LDS (stepname CMNIAIN) prior to running INITIAL2.

3. STARTJCL—This sample member forms the basis for your ZMF started task procedure. For this task, you need to edit your ZMF proc to remove the CMNIALOG and CMNIMPCT ddnames prior to attempting to start the ZMF stc for the first time.

4. DB2OPTN/R—This sample job creates the full function ZMF Db2 option admin tables, binds the related packages, and binds the plan for full function ZMF Db2 option support. DB2OPTNR is for P-sites, and DB2OPTN is for all other types of sites.

5. CMNDB2IA—This sample JCL member creates the Db2 tables, binds the packages, and binds the plan to enable the baseline I/A table to be hosted by Db2.

6. Global parms update for Db2 for I/A—The ZMF subsystem must be started up. You must ensure that the default Db2 subsystem (i.e. the one which will host the I/A database) is set as the first in the list of physical subsystems defined to Global ZMF Db2 option admin, i.e. via ISPF option A.G.O.2.1 (D10L in the following case):

```
CMNGD2S0 Db2 Physical Subsystems - Part 1 of 2      Row 1 to 3 of 3
Command ===>                                        Scroll ===> CSR

    Db2
    subsys  Site    Db2 System Load Library
    D10L            DSN1210.SDSNLOAD
    D10L    LOCAL   DSN1210.SDSNLOAD
    D10K    TEST    DSN1110.SDSNLOAD
```

Then update the Global parms to indicate that you wish to use Db2 for baseline I/A. The parm for this is located (via ISPF option A.G.1) on this panel:

```
CMNGGP03                 Global Parameters - Part 3 of 8
Command ===>

Enter "/" to select option
    Baselines
    / Stacked Reverse Delta
    / Panvalet
    User defined
    / Librarian
    Librarian Access Method (LAM)
    Notification Vehicles
    / Email
    / Batch
    Other options
    / Use primary Db2 subsystem for I/A: D10L
        Require CR number
        Require Department
    / Disable installation calendar
    / Allow temporary packages
    / Process participating packages by install date
    / Hierarchical approval process
    / Use global notification file
    / Allow application update to file
    Force display of global notification file
Global notification file . . CMNDEV.CMNSYS.U900DP.GNFFILE
```

This setting is in effect only after a restart of the ZMF started task.

7. IMPACTD2—This job runs the mass data extraction process for an installation where Db2 has been used for the I/A database. It produces the usual three flat files (i.e. BUNSPACE, CMPSPACE, and RELSPACE).

8. CMNDB2IL—This job loads the I/A data from the three flat files into the set of Db2 tables that are hosting the I/A database.

9. You should take your usual measures to ensure the best access paths are chosen by Db2 for this new data. This would usually include using the RUNSTATS utility, with the INDEX clause, to update the stats for the tablespaces involved (i.e. those created when you ran CMNDB2IA) and

then rebinding the packages involved (those you originally bound using sample JCL member CMNDB2IA). As with any collection of Db2 table data, using the RUNSTATS/rebind mechanism is something you should do on a regular basis.

Follow-on step:

If you didn't follow the instructions in steps 1, 2, and 3 during the initial product install, you can now delete the redundant I/A LDS and log dataset and remove references to them from your ZMF stc procedure.

Migration from a prior version of ZMF

If you are migrating from a prior version, you will have the I/A dataspace in place and will be making a conscious decision to convert to using Db2 at the same time as migrating to this latest release of ZMF.

You will most likely already have the ZMF Db2 option tables in place (depending on which release of ZMF you are migrating from).

You will also have a default Db2 subsystem already defined as the first in the list of subsystems in global admin for the ZMF Db2 option, physical subsystems.

1. STARTJCL—This sample member forms the basis for your ZMF started task procedure. For this task, you need to edit your ZMF proc to remove the CMNIALOG and CMNIMPCT ddnames prior to attempting to start the migrated ZMF stc for the first time.

2. DB2OPTN/R—This sample job creates the full function ZMF Db2 option admin tables, binds the related packages, and binds the plan for full function ZMF Db2 option support. DB2OPTNR is for P-sites, and DB2OPTN is for all other types of sites. If you already have the tables in place, remove the TABLES step from the job (i.e., do not drop/create your tables).

3. DB2ADMGT—This sample JCL shows how to reformat an existing CMNADMIN_GENERAL table to support ZMF version 8.2.0 or greater. You need to run it only if you are migrating from a version earlier than 8.2.0.

4. DB2ALTER—This sample JCL shows how to add the columns to an existing CMNADMIN_NAMED table to support ZMF version 8.2 patch 4 or greater. You need to run it only if you are migrating from a version earlier than 8.2 patch 4.

5. CMNDB2IA—This sample JCL member creates the Db2 tables, binds the packages, and binds the plan to enable the baseline I/A table to be hosted by Db2.

6. LDSUNLD or IMPACTD2—LDSUNLD is a sample job that unloads your existing I/A dataspace to flat files. IMPACTD2 creates these flat files by running the batch data extract (CMNIA000). Choose one of these in order to create the flat files to be used to load the new Db2 database. Run IMPACTD2 only if you have made application admin changes (e.g. new appls, new

baselines) as part of the conversion (this is not recommended). Either job produces the usual three flat files (i.e. BUNSPACE, CMPSPACE, and RELSPACE).

7. CMNDB2IL—This job loads the I/A data from the three flat files into the set of Db2 tables that are hosting the I/A database.

8. You should take your usual measures to ensure the best access paths are chosen by Db2 for this new data. This would usually include using the RUNSTATS utility, with the INDEX clause, to update the stats for the tablespaces involved (i.e. those created when you ran CMNDB2IA) and then rebinding the packages involved (those you originally bound using sample JCL member CMNDB2IA). As with any collection of Db2 table data, using the RUNSTATS/rebind mechanism is something you should do on a regular basis.

9. Global parms update for Db2 for I/A—The ZMF subsystem must be started up and the global parms updated to indicate that you want to use Db2 for baseline I/A. The global parm for this is located (via ISPF option A.G.1) on this panel:

```
CMNGGP03                Global Parameters - Part 3 of 8
Command ===>

Enter "/" to select option
    Baselines
    / Stacked Reverse Delta
    / Panvalet
      User defined
    / Librarian
      Librarian Access Method (LAM)
    Notification Vehicles
    / Email
    / Batch
    Other options
    / Use primary Db2 subsystem for I/A: D10L
      Require CR number
      Require Department
    / Disable installation calendar
    / Allow temporary packages
    / Process participating packages by install date
    / Hierarchical approval process
    / Use global notification file
    / Allow application update to file
      Force display of global notification file
Global notification file . . CMNDEV.CMNSYS.U900DP.GNFFILE
```

This setting takes effect only after a restart of the ZMF started task.

After you are satisfied, you can delete the now redundant I/A LDS and log dataset.

Core ZMF Db2 Option, without ERO, Not Using Db2 to Host the Baseline I/A Table

New Install

1. DB2CORE—This sample JCL binds the package and plan required for the core ZMF Db2 option support. It should be run instead of DB2OPTN or DB2OPTNR.

2. Global Db2 general parms update for Core ZMF Db2 option—The ZMF subsystem must be started up and the global Db2 general parms updated to indicate that you want to use only core Db2 functions. This is located (via ISPF option A.G.O.2.G) on this panel:

```
CMNGDPM0                Global Db2 General Parameters
Command ===>

Enter "/" to select option
    / Use Core Db2 Option Functions Only
      Use Package Name in Db2 PC version
      Force Pkg Name in Db2 PC version
```

This setting takes effect immediately.

Migration from a prior version of ZMF

1. DB2CORE—This sample JCL binds the package and plan required for the core ZMF Db2 option support. It should be run instead of DB2OPTN.

2. Global Db2 general parms update for Core ZMF Db2 option—The ZMF subsystem must be started up and the global Db2 general parms updated to indicate that you want to use only core Db2 functions. This is located (via ISPF option A.G.O.2.G) on this panel: This setting takes effect immediately.

```
CMNGDPM0                Global Db2 General Parameters
Command ===>

Enter "/" to select option
    / Use Core Db2 Option Functions Only
      Use Package Name in Db2 PC version
      Force Pkg Name in Db2 PC version
```

This setting takes effect immediately.

After you are satisfied with the core Db2 option functionality, you can drop the now redundant Db2 option admin tables and free the relevant packages using sample job DB2DRPNC.

Core ZMF Db2 Option, without ERO, Using Db2 to Host the Baseline I/A Table

New Install

Note that steps 1, 2, and 3 are part of the base ZMF install process. You can save time by following these instructions during that part of your install. However, you can also execute the follow-on step (see below) at some time later if you didn't follow steps 1, 2, and 3 during the base product installation.

1. INITIAL1—This sample JCL member is customized and run as part of the creation of the vsam master files that support a new installation of ZMF. For this task you should remove the step which allocates the I/A dataspace LDS (stepname IADSP) prior to running INITIAL1.

2. INITIAL2—This is the job run after INITIAL1. Before running INITIAL2, you need to remove the step that initializes the I/A dataspace LDS (stepname CMNIAIN).

3. STARTJCL—This sample member forms the basis for your ZMF started task procedure. For this task, you need to edit your ZMF proc to remove the CMNIALOG and CMNIMPCT ddnames prior to attempting to start the ZMF stc for the first time.

4. CMNDB2IA—This sample JCL member creates the Db2 tables, binds the packages, and binds the plan to enable the baseline I/A table to be hosted by Db2.

5. Global parms update—The ZMF subsystem must be up and running.

   You must ensure that the default Db2 subsystem (i.e. the one that will host the I/A database) is set as the first item in the list of physical subsystems defined for Global ZMF Db2 option admin. This is done via ISPF option A.G.O.2.1 (D10L in the following case):

   ```
   CMNGD2S0 Db2 Physical Subsystems - Part 1 of 2        Row 1 to 3 of 3
   Command ===>                                          Scroll ===> CSR
       Db2
       subsys  Site    Db2 System Load Library
       D10L            DSN1210.SDSNLOAD
       D10L    LOCAL   DSN1210.SDSNLOAD
       D10K    TEST    DSN1110.SDSNLOAD
   ```

   Then update the Global parms to indicate that you want to use Db2 for baseline I/A. The global parm for this is located (via ISPF option A.G.1) on this panel:

```
CMNGGP03 Global Parameters - Part 3 of 8
Command ===>

Enter "/" to select option
Baselines
    / Stacked Reverse Delta
    / Panvalet
    User defined
    / Librarian
      Librarian Access Method (LAM)
      Notification Vehicles
    / Email
    / Batch
    Other options
    / Use primary Db2 subsystem for I/A: D10L
      Require CR number
      Require Department
    / Disable installation calendar
    / Allow temporary packages
    / Process participating packages by install date
    / Hierarchical approval process
    / Use global notification file
    / Allow application update to file
      Force display of global notification file
Global notification file . . CMNDEV.CMNSYS.U900DP.GNFFILE
```

This setting takes effect only after a restart of the ZMF started task. Finally, update global Db2 general parms to indicate that you want to use only core Db2 functions. This is located (via ISPF option A.G.O.2.G) on this panel:

```
CMNGDPM0                      Global Db2 General Parameters
Command ===>
Enter "/" to select option
    / Use Core Db2 Option Functions Only
    Use Package Name in Db2 PC version
    Force Pkg Name in Db2 PC version
```

This setting takes effect immediately.

6. IMPACTD2—This job runs the mass data extraction process for an installation where Db2 has been used for the I/A database. It produces the usual three flat files (i.e., BUNSPACE, CMPSPACE, and RELSPACE.)

7. CMNDB2IL—This job loads the I/A data from the three flat files into the set of Db2 tables that are hosting the I/A database.

8. You should take your usual measures to ensure the best access paths are chosen by Db2 for this new data. This would usually include using the RUNSTATS utility, with the INDEX clause, to update the stats for the tablespaces involved (i.e. those created when you ran CMNDB2IA) and then rebinding the packages involved (those you originally bound using sample JCL member CMNDB2IA). As with any collection of Db2 table data, using the RUNSTATS/rebind mechanism is something you should do on a regular basis.

Follow-on step:

If you didn't follow the instructions in steps 1, 2, and 3 during the initial product install, you can now delete the redundant I/A LDS and log dataset and remove references to them from your ZMF stc procedure.

Migration from a prior version of ZMF

If you are migrating from a prior version, you will most likely have the I/A dataspace in place and will be making a conscious decision to convert to using Db2. We recommend that you move to Db2 some time after you have successfully migrated to this latest release of ZMF. However, if you must do this at the same time, then run these tasks.

1. STARTJCL—This sample member forms the basis for your ZMF started task procedure. For this task, you need to edit your ZMF proc to remove the CMNIALOG and CMNIMPCT ddnames prior to attempting to start the migrated ZMF stc for the first time.

2. CMNDB2IA—This sample JCL member creates the Db2 tables, binds the packages, and binds the plan to enable the baseline I/A table to be hosted by Db2.

3. LDSUNLD or IMPACTD2—LDSUNLD is a sample job that unloads your existing I/A dataspace to flat files. IMPACTD2 creates these flat files by running the batch data extract (CMNIA000). Choose one of these in order to create the flat files to be used to load the new Db2 database. Run IMPACTD2 only if you have made application admin changes (e.g. new appls, new baselines) as part of the conversion (this is not recommended). Either job produces the usual three flat files (i.e. BUNSPACE, CMPSPACE, and RELSPACE.)

4. CMNDB2IL—This job loads the I/A data from the three flat files into the set of Db2 tables that are hosting the I/A database.

5. You should take your usual measures to ensure the best access paths are chosen by Db2 for this new data. This would usually include using the RUNSTATS utility, with the INDEX clause, to update the stats for the tablespaces involved (i.e. those created when you ran CMNDB2IA) and then rebinding the packages involved (those you originally bound using sample JCL member CMNDB2IA). As with any collection of Db2 table data, using the RUNSTATS/rebind mechanism is something you should do on a regular basis.

6. Global parms update—The ZMF subsystem must be up and running. You must ensure that the default Db2 subsystem (i.e. the one which will host the I/A database) is set as the first in the list of physical subsystems defined to Global ZMF Db2 option admin. (You have most likely already done this if you have been using a system with a ZMF Db2 option license.)

   Update the Global parms to indicate that you want to use Db2 for baseline I/A. The global parm for this is located (via ISPF option A.G.1) on the following panel.

```
CMNGGP03              Global Parameters - Part 3 of 8
Command ===>
Enter "/" to select option
    Baselines
    / Stacked Reverse Delta
    / Panvalet
      User defined
    / Librarian
      Librarian Access Method (LAM)
    Notification Vehicles
    / Email
    / Batch
    Other options
    / Use primary Db2 subsystem for I/A: D10L
      Require CR number
      Require Department
    / Disable installation calendar
    / Allow temporary packages
    / Process participating packages by install date
    / Hierarchical approval process
    / Use global notification file
    / Allow application update to file
      Force display of global notification file
Global notification file . . CMNDEV.CMNSYS.U900DP.GNFFILE
```

This setting takes effect only after a restart of the ZMF started task.

Update global Db2 general parms to indicate that you wish to use only core Db2 functions.
This is located (via ISPF option A.G.O.2.G) on this panel:

```
CMNGDPM0              Global Db2 General Parameters
Command ===>
Enter "/" to select option
    / Use Core Db2 Option Functions Only
      Use Package Name in Db2 PC version
      Force Pkg Name in Db2 PC version
```

This setting takes effect immediately.

After you are satisfied with the core Db2 option functionality, you can drop the now redundant Db2
option admin tables and free the relevant packages using sample job DB2DRPNC.

Full Function ZMF Db2 Option, with ERO, Not Using Db2 to Host the Baseline I/A Table

New Install

Run the following jobs.

1. CMNDB2RM—This sample job creates the main ERO tables.

2. CMNDB2RA—This sample job creates the ERO audit tables.

3. CMNDB2RP—This sample job binds the ERO packages.

4. CMNDB2RB—This sample job binds the ERO plans.

5. CMNDB2RG—This sample job grants the relevant Db2 authorities for ERO.

6. DB2OPTN/R—This sample job creates the full function ZMF Db2 option admin tables, binds the related packages, and binds the plan for full function ZMF Db2 option support. DB2OPTNR is for P-sites, and DB2OPTN is for all other types of sites.

Migration from a prior version of ZMF

You most likely already have the ZMF Db2 option tables in place (depending on which release of ZMF they are coming from).

1. CIMALTR4—This sample job alters the ERO CIM table for migration from versions prior to ZMF v7.1.2.

2. HSTALTR4—This sample job alters the ERO HST table for migration from versions prior to ZMF v7.1.2.

3. CIMALTR5—This sample job alters the ERO CIM table for migration from versions prior to ZMF v8.1.2. (It must be run after CIMALTR4, if applicable).

4. CIMALTR6—This sample job alters the ERO CIM table for migration from versions prior to ZMF v8.2 patch 1. (This must be run after CIMALTR4 & CIMALTR5, if applicable).

5. HSTALTR5—This sample job alters the ERO HST table for migration from versions prior to ZMF v8.2 patch 1. (This must be run after HSTALTR4, if applicable).

6. DB2OPTN/R—This sample job creates the full function ZMF Db2 option admin tables, binds the related packages, and binds the plan for full function ZMF Db2 option support. DB2OPTNR is for P-sites, and DB2OPTN for all other types of sites. If you already have the tables in place, remove the TABLES step from the job (i.e., do not drop/create your tables).

7. DB2ADMGT —This sample JCL shows how to reformat an existing CMNADMIN_GENERAL table to support ZMF versions 8.2.0 and greater. You don't need to run it unless you are migrating from a version earlier than 8.2.0.

8. DB2ALTER—This sample JCL shows how to add the columns to an existing CMNADMIN_NAMED table to support ZMF versions 8.2 patch 4 and greater. You don't need to run it unless you are migrating from a version earlier than 8.2 patch 4.

Full Function ZMF Db2 Option, with ERO, Using Db2 to Host the Baseline I/A Table

New Install

Note that steps 1,2, and 3 are part of the base ZMF install process. You can save time by following these instructions during that part of your install. However, you can also execute the follow-on step (see below) at some later time if you didn't follow steps 1, 2, and 3 during the base product installation.

1. INITIAL1—This sample JCL member is customized and run as part of the creation of the vsam master files that support a new installation of ZMF. Before running INITIAL1, remove the step which allocates the I/A dataspace LDS (stepname IADSP).

2. INITIAL2—This is the job run after INITIAL1. Before running INITIAL2, you need to remove the step which initializes the I/A dataspace LDS (stepname CMNIAIN).

3. STARTJC—This sample member forms the basis for your ZMF started task procedure. For this task, you need to edit your ZMF proc to remove the CMNIALOG and CMNIMPCT ddnames prior to attempting to start the ZMF stc for the first time.

4. CMNDB2RM—This sample job creates the main ERO tables.

5. CMNDB2RA—This sample job creates the ERO audit tables.

6. CMNDB2RP—This sample job binds the ERO packages.

7. CMNDB2RB—This sample job binds the ERO plans.

8. CMNDB2RG—This sample job grants the relevant Db2 authorities for ERO.

9. DB2OPTN/R—This sample job creates the full function ZMF Db2 option admin tables, binds the related packages, and binds the plan for full function ZMF Db2 option support. DB2OPTNR is for P-sites, and DB2OPTN is for all other types of sites.

10. CMNDB2IA—This sample JCL member creates the Db2 tables, binds the packages, and binds the plan to enable the baseline I/A table to be hosted by Db2.

11. Global parms update for Db2 for I/A—The ZMF subsystem must be started up. You must ensure that the default Db2 subsystem (i.e. the one that will host the I/A database) is set as the first in the list of physical subsystems defined to Global ZMF Db2 option admin via ISPF option A.G.O.2.1 (D10L in the following case):

```
CMNGD2S0 Db2 Physical Subsystems - Part 1 of 2      Row 1 to 3 of 3
Command ===>                                        Scroll ===> CSR
    Db2
  subsys   Site        Db2 System Load Library
   D10L                DSN1210.SDSNLOAD
   D10L    LOCAL       DSN1210.SDSNLOAD
   D10K    TEST        DSN1110.SDSNLOAD
```

Then update the Global parms to indicate that you wish to use Db2 for baseline I/A. The parm for this is located (via ISPF option A.G.1) on this panel:

```
CMNGGP03                         Global Parameters - Part 3 of 8
Command ===>
Enter "/" to select option
    Baselines
    / Stacked Reverse Delta
    / Panvalet
      User defined
    / Librarian
      Librarian Access Method (LAM)
    Notification Vehicles
    / Email
    / Batch
    Other options
    / Use primary Db2 subsystem for I/A: D10L
      Require CR number
      Require Department
    / Disable installation calendar
    / Allow temporary packages
    / Process participating packages by install date
    / Hierarchical approval process
    / Use global notification file
    / Allow application update to file
      Force display of global notification file
 Global notification file . . CMNDEV.CMNSYS.U900DP.GNFFILE
```

This setting takes effect only after a restart of the ZMF started task.

12. IMPACTD2—This job runs the mass data extraction process for an installation where Db2 has been used for the I/A database. It produces the usual three flat files (i.e. BUNSPACE, CMPSPACE, and RELSPACE.)

13. CMNDB2IL—This job loads the I/A data from the three flat files into the set of Db2 tables that are hosting the I/A database.

14. You should take your usual measures to ensure the best access paths are chosen by Db2 for this new data. This would usually include using the RUNSTATS utility, with the INDEX clause, to update the stats for the tablespaces involved (i.e. those created when you ran CMNDB2IA) and then rebinding the packages involved (those you originally bound using sample JCL member CMNDB2IA). As with any collection of Db2 table data, using the RUNSTATS/rebind mechanism is something you should do on a regular basis.

Follow-on step:

If you didn't follow the instructions in steps 1, 2, and 3 during the initial product install, you can now delete the redundant I/A LDS and log dataset and remove references to them from your ZMF stc procedure.

Migration from a prior version of ZMF

If you are migrating from a prior version, you will most likely have the I/A dataspace in place and will be making a conscious decision to convert to using Db2 at the same time as migrating to this latest release of ZMF.

Yo will also likely already have the ZMF Db2 option tables in place (depending on which release of ZMF you are migrating from).

And you will also have a default Db2 subsystem already defined as the first in the list of subsystems in global admin for the ZMF Db2 option, physical subsystems.

1. STARTJCL—This sample member forms the basis for your ZMF started task procedure. For this task, you need to edit your ZMF proc to remove the CMNIALOG and CMNIMPCT ddnames prior to attempting to start the migrated ZMF stc for the first time.

2. CIMALTR4—This sample job alters the ERO CIM table for migration from versions prior to ZMF v7.1.2.

3. HSTALTR4—This sample job to alter the ERO HST table for migration from versions prior to ZMF v7.1.2

4. CIMALTR5—This sample job alters the ERO CIM table for migration from versions prior to ZMF v8.1.2. (This must be run after CIMALTR4, if applicable.)

5. CIMALTR6—This sample job alters the ERO CIM table for migration from versions prior to ZMF v8.2 patch 1. (This must be run after CIMALTR4 & CIMALTR5, if applicable.)

6. HSTALTR5—This sample job alters the ERO HST table for migration from versions prior to ZMF v8.2 patch 1. (This must be run after HSTALTR4, if applicable.)

7. DB2OPTN/R—This sample job creates the full function ZMF Db2 option admin tables, binds the related packages, and binds the plan for full function ZMF Db2 option support. DB2OPTNR is for P-sites, and DB2OPTN is for all other types of sites. If you already have the tables in place, remove the TABLES step from the job (i.e., do not drop/create your tables).

8. DB2ADMGT—This sample JCL shows how to reformat an existing CMNADMIN_GENERAL table to support ZMF versions 8.2.0 or greater. You don't need to run it unless you are migrating from a version earlier than 8.2.0.

9. DB2ALTER—This sample JCL shows how to add the columns to an existing CMNADMIN_NAMED table to support ZMF version 8.2 patch 4 or greater. You don't need to run it unless you are migrating from a version earlier than 8.2 patch 4.

10. CMNDB2IA—This sample JCL member creates the Db2 tables, binds the packages, and binds the plan to enable the baseline I/A table to be hosted by Db2.

11. Global parms update for Db2 for I/A—The ZMF subsystem must be started up and the global parms updated to indicate that you wish to use Db2 for baseline I/A. The global parm for this is located (via ISPF option A.G.1) on this panel:

```
CMNGGP03                        Global Parameters - Part 3 of 8
Command ===>
    Enter "/" to select option
    Baselines
    / Stacked Reverse Delta
    / Panvalet
      User defined
    / Librarian
      Librarian Access Method (LAM)
    Notification Vehicles
    / Email
    / Batch
    Other options
    / Use primary Db2 subsystem for I/A: D10L
      Require CR number
      Require Department
    / Disable installation calendar
    / Allow temporary packages
    / Process participating packages by install date
    / Hierarchical approval process
    / Use global notification file
    / Allow application update to file
      Force display of global notification file
Global notification file . . CMNDEV.CMNSYS.U900DP.GNFFILE
```

```
This setting takes effect only after a restart of the ZMF started task.
```

12. LDSUNLD or IMPACTD2—LDSUNLD is a sample job that unloads your existing I/A dataspace to flat files. IMPACTD2 creates these flat files by running the batch data extract (CMNIA000). Choose one of these in order to create the flat files to be used to load the new Db2 database. Run IMPACTD2 only if you have made application admin changes (e.g. new appls, new baselines) as part of the conversion (not recommended). Either job produces the usual three flat files (i.e. BUNSPACE, CMPSPACE, and RELSPACE.)

13. CMNDB2IL—This job loads the I/A data from the three flat files into the set of Db2 tables which are hosting the I/A database.

14. You should take your usual measures to ensure the best access paths are chosen by Db2 for this new data. This would usually include using the RUNSTATS utility, with the INDEX clause, to update the stats for the tablespaces involved (i.e. those created when you ran CMNDB2IA) and then rebinding the packages involved (those you originally bound using sample JCL member CMNDB2IA). As with any collection of Db2 table data, using the RUNSTATS/rebind mechanism is something you should do on a regular basis.

After you are satisfied, you can delete the now redundant I/A LDS and log dataset.

Core ZMF Db2 Option, with ERO, Not Using Db2 to Host the Baseline I/A Table

New Install

1. CMNDB2RM—This sample job creates the main ERO tables.

2. CMNDB2RA—This sample job creates the ERO audit tables.

3. CMNDB2RP—This sample job binds the ERO packages.

4. CMNDB2RB—This sample job binds the ERO plans.

5. CMNDB2RG—This sample job grants the relevant Db2 authorities for ERO.

6. DB2CORE—This sample JCL binds the package and plan required for the core ZMF Db2 option support. It should be run instead of DB2OPTN or DB2OPTNR.

7. Global Db2 general parms update for Core ZMF Db2 option—The ZMF subsystem must be started up and the global Db2 general parms updated to indicate that you want to use only core Db2 functions. This is located (via ISPF option A.G.O.2.G) on this panel:

```
CMNGDPM0                      Global Db2 General Parameters
Command ===>
Enter "/" to select option
    / Use Core Db2 Option Functions Only
      Use Package Name in Db2 PC version
      Force Pkg Name in Db2 PC version
```

This setting takes effect immediately.

Migration from a prior version of ZMF

1. CIMALTR4—This sample job alters the ERO CIM table for migration from versions prior to ZMF v7.1.2.

2. HSTALTR4—This sample job alters the ERO HST table for migration from versions prior to ZMF v7.1.2.

3. CIMALTR5—This sample job alters the ERO CIM table for migration from versions prior to ZMF v8.1.2 (It must be run after CIMALTR4, if applicable.)

4. CIMALTR6—This sample job alters the ERO CIM table for migration from versions prior to ZMF v8.2 patch 1. (It must be run after CIMALTR4 & CIMALTR5, if applicable.)

5. HSTALTR5—This sample job alters the ERO HST table for migration from versions prior to ZMF v8.2 patch 1. (It must be run after HSTALTR4, if applicable.)

6. DB2CORE—This sample JCL binds the package and plan required for the core ZMF Db2 option support. It should be run instead of DB2OPTN.

7. Global Db2 general parms update for Core ZMF Db2 option—The ZMF subsystem must be started up and the global Db2 general parms updated to indicate that you wish to use only core Db2 functions. This is located (via ISPF option A.G.O.2.G) on this panel:

```
CMNGDPM0                    Global Db2 General Parameters
Command ===>
Enter "/" to select option
   / Use Core Db2 Option Functions Only
     Use Package Name in Db2 PC version
     Force Pkg Name in Db2 PC version
```

This setting takes effect immediately.

After you are satisfied with the core Db2 option functionality, you can drop the now redundant Db2 option admin tables and free the relevant packages using sample job DB2DRPNC.

Core ZMF Db2 Option, with ERO, Using Db2 to Host the Baseline I/A Table

New Install

Note that steps 1, 2, and 3 are part of the base ZMF install process. You can save time by following these instructions during that part of your install. However, you can also execute the follow-on step (see below) at some later time if you didn't follow steps 1, 2, and 3 during the base product installation.

1. INITIAL1—This sample JCL member is customized and run as part of the creation of the vsam master files which will support a new installation of ZMF. For this task, you should remove the step which allocates the I/A dataspace LDS (stepname IADSP) prior to running INITIAL1.

2. INITIAL2—This is the job run after INITIAL1. Before running INITIAL2, you need to remove the step which initializes the I/A dataspace LDS (stepname CMNIAIN).

3. STARTJCL—This sample member forms the basis for your ZMF started task procedure. For this task, you need to edit your ZMF proc to remove the CMNIALOG and CMNIMPCT ddnames prior to attempting to start the ZMF stc for the first time.

4. CMNDB2RM—This sample job creates the main ERO tables.

5. CMNDB2RA—This sample job creates the ERO audit tables.

6. CMNDB2RP—This sample job binds the ERO packages.

7. CMNDB2RB—This sample job binds the ERO plans.

8. CMNDB2RG—This sample job grants the relevant Db2 authorities for ERO.

9. CMNDB2IA—This sample JCL member creates the Db2 tables, binds the packages, and binds the plan to enable the baseline I/A table to be hosted by Db2.

10. Global parms update—The ZMF subsystem must be up and running. You must ensure that the default Db2 subsystem (i.e., the one that will host the I/A database) is set as the first in the list of physical subsystems defined to Global ZMF Db2 option admin. This is done via ISPF option A.G.O.2.1 (D10L in the following case):

```
CMNGD2S0             Db2 Physical Subsystems - Part 1 of 2 Row 1 to 3 of 3
Command ===>                                          Scroll ===> CSR

    Db2
subsys   Site    Db2 System Load Library
    D10L            DSN1210.SDSNLOAD
    D10L    LOCAL   DSN1210.SDSNLOAD
    D10K    TEST    DSN1110.SDSNLOAD
```

Then update the Global parms to indicate that you wish to use Db2 for baseline I/A. The global parm for this is located (via ISPF option A.G.1) on this panel:

```
CMNGGP03                   Global Parameters - Part 3 of 8
Command ===>

Enter "/" to select option
    Baselines
    / Stacked Reverse Delta
    / Panvalet
      User defined
/ Librarian
      Librarian Access Method (LAM)
Notification Vehicles
/ Email
/ Batch
Other options
/ Use primary Db2 subsystem for I/A: D10L
  Require CR number
  Require Department
/ Disable installation calendar
/ Allow temporary packages
/ Process participating packages by install date
/ Hierarchical approval process
/ Use global notification file
/ Allow application update to file
  Force display of global notification file
Global notification file . . CMNDEV.CMNSYS.U900DP.GNFFILE
```

This setting is in effect only after a restart of the ZMF started task. Finally, update global Db2 general parms to indicate that you want to use only core Db2 functions. This is located (via ISPF option A.G.O.2.G) on this panel:

```
CMNGDPM0               Global Db2 General Parameters
Command ===>
Enter "/" to select option
    / Use Core Db2 Option Functions Only
      Use Package Name in Db2 PC version
    Force Pkg Name in Db2 PC version
```

This setting takes effect immediately.

11. IMPACTD2—This job runs the mass data extraction process for an installation where Db2 has been used for the I/A database. It produces the usual three flat files (i.e., BUNSPACE, CMPSPACE, and RELSPACE.)

12. CMNDB2IL—This job loads the I/A data from the three flat files into the set of Db2 tables that are hosting the I/A database.

13. You should take your usual measures to ensure the best access paths are chosen by Db2 for this new data. This would usually include using the RUNSTATS utility, with the INDEX clause, to update the stats for the tablespaces involved (i.e. those created when you ran CMNDB2IA) and then rebinding the packages involved (those you originally bound using sample JCL member CMNDB2IA). As with any collection of Db2 table data, using the RUNSTATS/rebind mechanism is something you should do on a regular basis.

Follow-on step:

If you didn't follow the instructions in steps 1, 2, and 3 during the initial product install, you can now delete the redundant I/A LDS and log dataset and remove references to them from your ZMF stc procedure.

Migration from a prior version of ZMF

We recommended that you move to Db2 some time after you have successfully migrated to this latest release of ZMF. However, if you must do this at the same time, run the following jobs:

1. STARTJCL—This sample member forms the basis for your ZMF started task procedure. For this task, you need to edit your ZMF proc to remove the CMNIALOG and CMNIMPCT ddnames prior to attempting to start the migrated ZMF stc for the first time.

2. CIMALTR4—This sample job alters the ERO CIM table for migration from versions prior to ZMF v7.1.2.

3. HSTALTR4—This sample job alters the ERO HST table for migration from versions prior to ZMF v7.1.2.

4. CIMALTR5—This sample job alters the ERO CIM table for migration from versions prior to ZMF v8.1.2. (It must be run after CIMALTR4, if applicable.)

5. CIMALTR6—This sample job alters the ERO CIM table for migration from versions prior to ZMF v8.2patch1. (It must be run after CIMALTR4 & CIMALTR5, if applicable.)

6. HSTALTR5—This sample job alters the ERO HST table for migration from versions prior to ZMF v8.2patch1. (It must be run after HSTALTR4, if applicable.)

7. CMNDB2IA—This sample JCL member creates the Db2 tables, binds the packages, and binds the plan to enable the baseline I/A table to be hosted by Db2.

8. Global parms update—The ZMF subsystem must be up and running. You must ensure that the default Db2 subsystem (i.e. the one which will host the I/A database) is set as the first in the

list of physical subsystems defined to Global ZMF Db2 option admin (You have most likely set this in previous upgrades). Update the Global parms to indicate that you wish to use Db2 for baseline I/A. The global parm for this is located (via ISPF option A.G.1) on the following panel:

```
CMNGGP03                    Global Parameters - Part 3 of 8
Command ===>

Enter "/" to select option
Baselines
    / Stacked Reverse Delta
    / Panvalet
      User defined
    / Librarian
      Librarian Access Method (LAM)
      Notification Vehicles
    / Email
    / Batch
      Other options
    / Use primary Db2 subsystem for I/A: D10L
      Require CR number
      Require Department
    / Disable installation calendar
    / Allow temporary packages
    / Process participating packages by install date
    / Hierarchical approval process
    / Use global notification file
    / Allow application update to file
      Force display of global notification file
Global notification file . . CMNDEV.CMNSYS.U900DP.GNFFILE
```

This setting takes effect only after a restart of the ZMF started task. Update global Db2 general parms to indicate that you want to use only core Db2 functions. This is located (via ISPF option A.G.O.2.G) on this panel:

```
CMNGDPM0                  Global Db2 General Parameters
Command ===>

Enter "/" to select option
    / Use Core Db2 Option Functions Only
      Use Package Name in Db2 PC version
      Force Pkg Name in Db2 PC version
```

This setting takes effect immediately.

9. LDSUNLD or IMPACTD2—LDSUNLD is a sample job that unloads your existing I/A dataspace to flat files. IMPACTD2 creates these flat files by running the batch data extract (CMNIA000). Choose one of these in order to create the flat files to be used to load the new Db2 database. Run IMPACTD2 only if you have made application admin changes (e.g. new appls, new baselines) as part of the conversion (this is not recommended). Either job produces the usual three flat files (i.e. BUNSPACE, CMPSPACE, and RELSPACE.)

10. CMNDB2IL—This job loads the I/A data from the three flat files into the set of Db2 tables that are hosting the I/A database. When you are satisfied with the core Db2 option functionality,

you can drop the now redundant Db2 option admin tables and free the relevant packages using sample job DB2DRPNC.

11. You should take your usual measures to ensure the best access paths are chosen by Db2 for this new data. This would usually include using the RUNSTATS utility, with the INDEX clause, to update the stats for the tablespaces involved (i.e. those created when you ran CMNDB2IA) and then rebinding the packages involved (those you originally bound using sample JCL member CMNDB2IA). As with any collection of Db2 table data, using the RUNSTATS/rebind mechanism is something you should do on a regular basis.

6. ChangeMan M+R Option Quick Start

This appendix provides abbreviated instructions for installing the ChangeMan ZMF Merge and Reconcile (M+R) Option and using the option to reapply your customizations to ChangeMan ZMF 8.3 components.

- What is the ChangeMan ZMF M+R Option
- Installing the M+R Option
- Using the M+R Option to Reapply Customization

What is the ChangeMan ZMF M+R Option

The ChangeMan ZMF M+R Option is an automated version reconciliation tool that is fully integrated with ChangeMan ZMF. The M+R Option can reconcile the components in up to eight separate libraries.

You can use the M+R Option to forward fit your ChangeMan ZMF modifications into components delivered in a new release. In a single operation, you can compare three versions of components:

- Current ChangeMan ZMF vendor components
- Current ChangeMan ZMF custom components
- Vendor ChangeMan ZMF components in the new release

Then, you can select your current modifications that are displayed in the context of the new release components and apply them, discard them, or apply them with edits.

If you do not license the M+R Option, contact your account representative. The M+R Option is included in ChangeMan ZMF libraries, and any version of the M+R Option is suitable for this task.

Installing the M+R Option

If you license the M+R Option, follow the steps to enable the option:

1. Allocate a permanent WALKTHRU PDS library with these characteristics:

   - Primary space allocation 80 Blocks

   - Secondary space allocation 100 Blocks

   - Directory Blocks 10

   - Record Format VB

   - Logical Record Length 4096

   - Block Size 4100

2. Copy these members from the vendor CMNZMF SAMPLES library to the WALKTHRU library:

   - WALKBAS

   - WALKDV1

   - WALKDV2

3. Allocate a permanent sequential PROJECTS data set with these characteristics:

   - Primary space allocation 20 Blocks

   - Secondary space allocation 400 Blocks

   - Record Format VB

   - Logical Record Length 80

   - Block Size 6000

4. Grant the following data set access to all potential users of the ChangeMan ZMF M+R Option:

   a. READ/WRITE authority to the PROJECTS file.

   b. READ authority to the vendor WALKTHRU library.

5. Copy panel SEROJECT from the ChangeMan ZMF vendor CMNZMF PANELS library to your custom panel library. Edit the)INIT section of the panel to assign values to these variables:

Variable	Value
&CDFPROJ	DSN of the PROJECTS data set allocated previously.
&CDFSAMPL	DSN of the WALKTHRU library allocated previously.

Variable	Value
&CDFUNIT	Generic unit for allocating user work data sets. Variable CDFUNIT is preset to SYSDA. Note: User work data sets are cataloged data sets that hold intermediate or final results from the M+R Option reconcile process. User work data sets include the WIS (Work In Suspense) file, the PRINT file, and the EXPORT file. The ChangeMan ZMF M+R Option can automatically allocate these files for each user.
&CDFDSNQ	High level qualifier(s) for user work data set names. The maximum length is 24 characters. Use literal strings and/or system variables like &ZPREFIX or &ZUSER. The ChangeMan ZMF M+R Option automatically inserts &ZUSER as a mid-level DSN qualifier unless variable CDFDSNQ is set to &ZUSER. CDFDSNQ is preset to CDFUSER, which is the log-on ID of the user.

6. Copy panel SER@PRIM from the vendor CMNZMF.PANELS library to your custom panels library. Edit the)INIT section of the panel to assign a value to variable &CDFTIME:

Variable	Value
&CDFTIME	Set to YES to check for a TIME parameter in the JOB card in generated batch jobs. Set to NO to suppress this check.

7. Stop and start the Sernet started task running ChangeMan ZMF.

8. If you use CA ACF2® for your security system, and if you use the ACF2 TSO Command Limiting feature to restrict execution access to TSO commands, add M+R program CDF$ISPF to the ACF2 Command Limiting Table:

   This is the format for the table entries:

```
TSOtable CSECT *    REGIONAL ACF2 TABLE
TSOCST ,            TSO RESTRICTED COMMANDS LIST
$TSOCMD CLS         CLEAR SCREEN
$TSOCMD CDF$ISPF    M+R ISPF DRIVER
$TSOCEND ,          INDICATE END OF LIST
END
```

> ⚠ **Caution**
>
> Do not add CDF$ISPF to the TSO Command Table IKJTSOnn in SYS1.PARMLIB. If you do and you attempt to use M+R, your session will freeze and this message is displayed:
>
> ISPS118L SERVICE NOT INVOKED. A VALID ISPF ENVIRONMENT DOES NOT EXIST.

9. Examine the primary, secondary, and directory allocations of your ISPF profile library (ISPPROF) to verify that there is sufficient space for ZMF and M+R to create profiles and save them after each session. This applies to all ISPF applications that run with ISPF NEWAPPL(xxxx).

Using the M+R Option to Reapply Customization

Execute this procedure to set up the ChangeMan ZMF M+R Option to merge your modifications from the ZMF release from which you are migrating into version 8.3 components.

Log in to ChangeMan ZMF. On the ChangeMan(R) Primary Option Menu (CMN@PRIM), select option C (M+R). The first time you select the M+R Option, the Reconcile (CDFPANEL) panel is displayed.

```
View All -----------------------RECONCILE -------------------------------
Command ===> _____Scroll ===> CSR
B:WALKTR#1 C:WALKTR#2 665
****************************** TOP OF DATA ******************************
000000 ********/* 2017/07/26 17:19:40 CMNTP.CMN814.C6.WALKTHRU
000001 WELCOME to Merge+Reconcile (M+R) from Serena Software, Inc.
000002
000003 The M+R walk-through consists of tutorial text and reconciliation
000004 actions that can be browsed and/or executed on the same panel so that
000005 you can immediately try any discussed primary or line command.
000006
000007 The Reconcile panel looks like the ISPF/PDF Edit panel and executes
000008 almost all ISPF/PDF commands. Both have a line command area (first 6
000009 characters on a line) and a data area which is scrollable.
000010 The 3rd line on both panels is reserved for the long text of a message
000011 in case you need it; meanwhile, M+R uses it to relate the nicknames
000012 of the derivative versions with their one-character IDs.
000013 The first record (with 000000 number) is created by M+R to register the
000014 date and time of the reconciliation, and the dataset name of the Base
000015 version.
000016
000017 The Reconcile panel displays the result of the comparison and merging
000018 of the versions, and marks the differences in the line command area:
000019
000020 -records COMMON in all the versions have their sequence numbers
000021 displayed in the command area;
000022 -records in the Base version that are MISSING in some derivatives are
000023 marked with the "less" symbol '<' and the IDs of those derivatives;
000024 -records in some derivatives that are ADDED (i.e. missing in the Base
000025 version) are marked with the "greater" symbol '>' and the IDs of
000026 those derivatives;
000027 -records in some derivatives that are REPLACING records of the Base
000028 version are marked with the "difference" symbol '<>' and the IDs of
000029 those derivatives; the corresponding Base records are marked with
000030 the "less" symbol '<' and keep their sequence number.
000031
000032 Each line on the Reconcile panel has a specific color depending on its
000033 origin. You can choose your own color scheme from the Profile panel.
000034 The default colors are: white for Base records; yellow for the first
000035 derivative records; pink for the second derivative records; etc.;
000036 red reversed for common records in two or more (up to 7) derivatives.
000037
```

2. Read the Reconcile tutorial for an overview of the M+R Option. Press PF8 to page down and PF7 to page up.

3. Further down on this same panel is an example (in color) of some of the results you might see.

```
View All --------------------------RECONCILE --------------------------ROW 39 TO
77 OF 173
Command ===> _____Scroll
===> CSR
B:WALKTR#1 C:WALKTR#2
000038 An example of the M+R graphics and coloring can be seen if you place
000039 the cursor on the next line and page down. Then page down again.
000040 // *
000041 // WSER37TN JOB (9602,848),'CMNBOW10',CLASS=E,
< 0042 //   TIME=(,15)
<>BC__ //   TIME=(,30)
000043 // STEPLIB DD DSN=WSER99.CMN416.LINKLIB,DISP=SHR
000044 // CMNBOW10 EXEC PGM=CMNBOW10,REGION=4M
000045 // CMNGBLFL DD DISP=(,CATLG,DELETE),
> B___ //   UNIT=SYSDA,
000046 //   SPACE=(CYL,(2,1)),
< _C__ //   VOL=SER=VOL333,
000048 //   DCB=(RECFM=VB,LRECL=5000,BLKSIZE=23476),
000049 //   DSN=&TSOID..&PKGMST..GLOBAL
000050 // CMNCMPFL DD DISP=(,CATLG,DELETE),
000051 //   UNIT=SYSDA,
< 0052 //   SPACE=(CYL,(15,5)),
<>B___ //   SPACE=(CYL,(20,10)),
<>_C__ //   SPACE=(CYL,(25,15)),
000053 //   DCB=(RECFM=VB,LRECL=5000,BLKSIZE=23476),
000054 //   DSN=&TSOID..&PKGMST..COMPONNT
000055 //
000056 // *
000057 Let's recap the differences between the 3 versions of the JCL jobdeck.
000058 The 42nd Base record is replaced in both derivatives with the same
000059 text "TIME=(,30)". Accordingly, the Base record is marked with "< 0042"
000060 and the replacing line with "<>BC__" where "B" and "C" are the IDs of
000061 the derivatives.
000062 One record is added in the derivative "B" after the 45th Base record:
000063 it is marked with "> B___".
000064 The 47th Base record is missing in the derivative "C"; therefore, this
000065 record is marked with "< _C__" instead of with its sequence number.
000066 The 52nd Base record is replaced in both derivatives but with different
000067 text: "SPACE=(CYL,(20,10))," vs. "SPACE=(CYL,(25,15)),".
000068
000069 #0 You can browse each of the derivatives separately by issuing "View B"
000070 or "View C". You can try this right now, then issue "Locate 41" to
000071 view, then issue "View All" (or "v *"), and then "Find #1" to continue.
000072 Note that the replacing records are shown together with the replaced
```

4. Press PF3, or type End and press Enter, to exit from the Reconcile tutorial and select M and press Enter. The Member Reconciliation Menu (SER@MAIN) is displayed.

```
SER@MAIN                    Member Reconciliation Menu
Option ===> _____
0 Profile Update user profile settings
1 Versions Specify base and derivative data sets and member names
2 Execute Execute (compare and merge) the specified versions of a file
3 Save Save the current reconciliation as "work in suspense" (wis)
4 Load Load prior "work in suspense" for edit now
5 Edit Resume editing the current reconciliation or just loaded wis
5P Print Print the last view (snapshot) of the reconciliation
6 Export Export a successful reconciliation effort to a file
6P Exp print Print a successful reconciliation effort
7 Analysis Analyze the progress and complexity of the reconciliation
7P Anl print Print analysis of the reconciliation progress and complexity
X Exit
```

Do not merge and reconcile at the member level. Instead, merge and reconcile your modifications at the library level so you can work with all members in the custom library at the same time.

4. Press PF3, or type End and press Enter, to exit from the Member Reconciliation Menu. The Merge+Reconcile main menu (SER@PRIM) is displayed.

```
SER@PRIM                         Merge+Reconcile
Option ===> _____
M Reconcile Up to 8 versions of a member or sequential data set
L Consolidate Up to 8 versions of a library -PDS, panvalet, librarian
S Synchronize Up to 8 versions of source and copy/include libraries
J Compare Up to 8 versions of a jcl lib after symbol substitution
C Changes From prior releases of merge+reconcile (m+r)
W Walk through The version reconciliation process with m+r
T Tutorial
X Exit
```

6. On the Merge+Reconcile main menu, choose option L -Consolidate up to 8 versions of a Library. The Library Consolidation Menu (SER@TEAM) is displayed.

```
SER@TEAM                    Library Consolidation Menu
Option ===> _____
0 Profile Update user profile settings
1 Versions Specify base and derivative library data set names
3 Workplace Display consolidation workplace listing version members
4 Batch job Batch execution (compare and merge) of version libraries
5 Project Create or select a consolidation project
X Exit
```

7. On the Library Consolidation Menu, choose option 0 -Profile. The Profile panel (SERPROFL) is displayed.

```
SERPROFL Profile Option ===>
0 Restore Back out all changes to the profile options and data sets
1 Compare Specify version comparison options
2 Colors Specify coloring scheme for versions codelines
3 Editing Select merged versions editing options
4 Export Select export options for the consolidated version
5 Data Sets Specify Work In Suspense, print and export data set names
6 Filtering Auto-delete certain merged code lines upon business rules
T Tutorial Information about setting up M+R reconciliation profile
X Exit
```

8. On the Profile panel, choose option 5 -Data Sets. The Profile Data Sets panel (SERDSNML) is displayed.

```
SERDSNML                        Profile Data Sets
Command ===>
WIS dsname . . . . . . . _____
Print dsname . . . . . . _____
Export dsname . . . . . _____
Export DSORG . . . . . . PDS___ (PDS, SEQ, PAN, LIB, or blank)
Export member mask . . . _____(Mask to rename Base members on export)
LIB module name . . . .  LIB____ (LIBrarian update module name: LIBR, LIBRARY)
```

9. Type names for three new data sets that are used by the M+R Option when you merge and consolidate libraries. Use data set names that comply with your standards for permanent personal data sets.

WIS dsname	userid.MR.WIS
Print dsname	userid.MR.PRINT
Export dsname	userid.MR.EXPORT

> 💡 **Note**
>
> Press PF1 to obtain help at any time.

5. Press **PF3**, or type **End** and press **Enter** to exit from the **Profile Data Sets** panel (SERDSNML). The M+R Option allocates the named data sets with the appropriate characteristics, and the Profile panel (SERPROFL) is displayed.

Press **PF3**, or type **End** and press **Enter**, to exit from the **Profile** panel. The **Library Consolidation Menu** is displayed. Using the M+R Option to Reapply Customization

7. On the Library Consolidation Menu (SER@TEAM), choose option 1 -Versions. The Versions (SERCDFCM) panel is displayed.

```
SERCDFCM                        Versions
Command ===> _____
    CMN Export package: _____Libtype:____  Export member:_____
                                                   -Member---Type Nickname Id
Base: _____  _____  ___  _____ __
                                    Last Member: _____
           or CMN appl/pkg: _____  Libtype: ____
Drv1: _____  _____  ___  _____ __
           or CMN appl/pkg: _____  Libtype: ____
Drv2: _____  _____  ___  _____ __
           or CMN appl/pkg: _____  Libtype: ____
Drv3: _____  _____  ___  _____ __
           or CMN appl/pkg: _____  Libtype: ____
Drv4: _____  _____  ___  _____ __
           or CMN appl/pkg: _____  Libtype: ____
Drv5: _____  _____  ___  _____ __
           or CMN appl/pkg: _____  Libtype: ____
Drv6: _____  _____  ___  _____ __
           or CMN appl/pkg: _____  Libtype: ____
Drv7: _____  _____  ___  _____ __
           or CMN appl/pkg: _____  Libtype: ____
```

8. On the Versions panel, type the data set names of libraries you want to compare. Define your Base and Derivative versions in this order:

Base	Vendor library for old version
Drv1	Custom library for old version
Drv2	Vendor library for new version

For the Base, Derivative 1, and Derivative 2 libraries, type an entry in the Type and Nickname fields. For the Derivative 1 and Derivative 2 libraries, type an entry in the ID fields. Do not type a Member name for any library.

```
SERCDFCM -----------------------VERSIONS -----------------------------
Command ===>

      CMN Export Package: _____ LibType: ___
                                               -Member---Type Nickname Id
Base: PROD99.CMNZMF.OLDREL.SKELS_____ _____ PDS BASE____  _
                                    Last Member: _____
          or CMN Appl/Pkg: _____ LibType: ___
Drv1: PROD99.CMNZMF.CUSTOM.SKELS_____ _____ PDS CUSTOM__ C
          or CMN Appl/Pkg: _____ LibType: ___
Drv2: PROD99.CMNZMF.NEWREL.SKELS_____ _____ PDS NEW_____ N
          or CMN Appl/Pkg: _____ LibType: ___
Drv3: _____ _____ ___ _____ _
          or CMN Appl/Pkg: _____ LibType: ___
```

Press **PF1** to display a help panel if you need assistance.

9. Press **PF3**, or type **End** and press **Enter**, to exit from the **Versions** panel. The **Library Consolidation Menu** (SER@TEAM) is displayed.

10. On the **Library Consolidation Menu**, choose option **3 -Workplace**, and the **Consolidation Workplace** panel (CDFPANEL) is displayed.

```
CDFPANEL ----------------Consolidation Workplace --------ROW 1 to 20 of 293 Command
===> Scroll ===> CSR Base: PROD99.CMNZMF.OLDREL.SKELS_
Difference  Base  Member Action  Compl  Custom   New  Derv  #3  Derv #4
@$$.....  _   $$$INDEX  $$$INDEX  $$$INDEX
@$$.....  _   #VARLIST  #VARLIST  #VARLIST
@.$.....  _   CMN$$ACB  CMN$$ACB
@$$.....  _   CMN$$ASM  CMN$$ASM  CMN$$ASM
@.=.....  _   CMN$$AUD  CMN$$AUD
@.=.....  _   CMN$$BBL  CMN$$BBL
@.=.....  _   CMN$$BDT  CMN$$BDT
@$=.....  _   CMN$$BLA  CMN$$BLA  CMN$$BLA
@.$.....  _   CMN$$BLG  CMN$$BLG
@.......  _   CMN$$BND
@.=.....  _   CMN$$BSN  CMN$$BSN
@.$.....  _   CMN$$CAS  CMN$$CAS
@.=.....  _   CMN$$CA7  CMN$$CA7
@.$.....  _   CMN$$CBL  CMN$$CBL
@.=.....  _   CMN$$CIC  CMN$$CIC
@.$.....  _   CMN$$CKA  CMN$$CKA
@.$.....  _   CMN$$CKO  CMN$$CKO
@.....  _   CMN$$CLN
```

11. Type `s` next to a component Base Member name to see a merged version of the member. Use instructions in the following resources to merge your custom modifications into the version 8.3 members:

   • *M+R Option Getting Started Guide*

   • *M+R Option Quick Reference*

   • PF1 Help from the Consolidation Workplace

   > 💡 **Note**
   >
   > Do not do anything to a component on the Consolidation Workplace panel (CDFPANEL) if no member from the custom library is displayed. Do not reconcile differences between two vendor components.

12. Type E next to a component Base Member name to copy the merged member from your WIS file into the Export File you specified in Step9.

13. Copy the members from your M+R Option Export file into your ChangeMan ZMF 8.3 custom library.

7. Legal Notice

For information about legal notices, trademarks, disclaimers, warranties, export and other use restrictions, U.S. Government rights, patent policy, and FIPS compliance, see https://www.microfocus.com/about/legal/.

The only warranties for products and services of Micro Focus and its affiliates and licensors ("Micro Focus") are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Micro Focus shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Third-Party Notices

Additional third-party notices, including copyrights and software license texts, can be found in a 'thirdpartynotices' file in the root directory of the software.

Specific notices

In accordance with the GNU General Public License version 2 with Classpath Exception, you are entitled to the complete OpenJDK source code that went into the JRE used by this product which includes the source code for 3 subclasses of that standard OpenJDK; MultipleGradientPaint, MultipleGradientPaintContext and TypeResolver. Please contact product support if you wish to obtain the source code. This source code will be available for 3 years from the general availability date for version 17.0 SP1.