opentext™

ChangeMan ZMF

User Guide

8.3

Table of Contents

1. About this Guide

ChangeMan® ZMF is a comprehensive and fully integrated solution for Software Change Management systems in z/OS environments. It provides reliable and streamlined implementation of software changes from development into production. ChangeMan ZMF manages and automates the application life cycle, protects the integrity of the code migration process, and results in higher quality delivered code to any test environment and to the production environment.

Before You Begin

See the ReadMe file for the latest updates and corrections for this manual.

Objective

The ChangeMan ZMF User's Guide provides instructions for using functions and facilities of ChangeMan ZMF to manage changes to application software.

Audience

This document is intended for developers, project managers, testing coordinators, and change managers who will use ChangeMan ZMF to manage software changes in their organization.

Navigating this Book

This guide describes the ISPF user interface to ChangeMan ZMF and explains how to perform ChangeMan ZMF functions.

Guide to ChangeMan ZMF Documentation

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.
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.

Manual	Description
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.
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.

Online Help

Online help is the primary source of information about ChangeMan ZMF. Online help is available as a tutorial, through Help screens, 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 Screens

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

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.
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`* .
vertical rule	Separates menus and their associated commands. For example, select File

2. Introduction

ChangeMan® ZMF is a comprehensive system that provides reliable and streamlined implementation of software changes in z/OS environments. It manages and automates the migration of software changes from a development environment to any test environment and to the production environment.

Features of ChangeMan ZMF

- Change package architecture guarantees the coordination of changed elements
- Automated job builds, installation scheduling, and rollback simplify implementation
- Version control detects discrepancies and out-of-sync component relationships
- Concurrent development is managed to serialize change
- Merge-and-reconcile option converges parallel lines of development
- Automated notifications keep developers and managers informed of change actions
- Built-in approval process ensures accountability
- Automated business rules enforce change policies at a controlled level of flexibility
- Library management supports CA Librarian®, CA Panvalet®, or IBM® PDS, PDSE and PDSE version 2 data sets
- ZMF 8.1.4 running under z/OS 2.3 allows sending of certain job completion messages to a TSO user's RACF defined email address. Refer to IBM documentation on z/OS 2.3 for the details on how to implement this feature.

- Database management options automate generation of IBM Db2® and IMS™ elements
- Customizable user exits and custom user variables add flexibility
- XML Services interface supports customized data interchange and product integrations
- Integrated network facility coordinates change across multiple servers & sites
- Full suite of online and batch query and reporting capabilities track software in motion
- Historical information is maintained in a single repository
- High Level Language Functional Exits (HLLX) to implement user exits written in Language Environment (LE)-compliant languages (such as COBOL and PL/1) and REXX to enforce local business rules in ZMF functions

What is a Change Package?

The heart of ChangeMan ZMF is the change package. A *change package* is a unit of work within a software development or maintenance project. It is managed as a unit through a defined lifecycle while software changes are in motion.

A change package consists of descriptive information, control parameters, and history information stored in VSAM files. It also includes a set of libraries that belong exclusively to the change package and which contain the software components being changed. A change package is a secure development environment for project components, with access managed by ChangeMan ZMF using rules stored in your security system.

A change package and the components in the package are guided through your system development lifecycle by the ChangeMan ZMF package lifecycle.

Package Lifecycle

The change package lifecycle is a rule-based process consisting of actions that you perform and actions automatically initiated by ChangeMan ZMF. These actions guarantee the integrity of changes you make to your production application system. These actions begin with Create Package and usually end with Baseline Ripple. Authority to perform any of these actions is controlled by ChangeMan ZMF and defined in your security system.

The following actions are defined in the package lifecycle:

- **Create Package** is the first step in the package lifecycle. A series of ISPF input panels request information that describes the change package and sets control parameters that determine how the package behaves during the rest of the package lifecycle.

- **Checkout Component** copies components from a baseline or promotion library into a staging library allocated exclusively to your package. You can also check out components to a personal library, which is tracked by ChangeMan ZMF.
- **Stage Component** is where you edit and build package components to meet project requirements. Source components are processed through predefined build processes to create executables and build listings. You can also stage components into your package from libraries outside of ChangeMan ZMF to bring those components under the control of ChangeMan ZMF.
- **Package Audit** detects problems that will occur in production if you install your package now in its current condition. Audit detects synchronization problems in relationships between components in your change package and synchronization problems between package components, components in participating packages, components in promotion, and baseline components.
- **Freeze Package** locks package information and package components to prevent further changes and to ensure that the components you install into production are the same as the components you tested. You can selectively unfreeze, change, audit, and refreeze components to fix problems found in testing.
- **Promote Package** copies package components from staging libraries into test libraries. As a package is promoted from one testing level to the next, package components are removed from libraries in the prior level and copied from staging libraries into test libraries for the next level. Demote Package removes package components from test libraries.
- **Approve Package** allows predefined approvers to review package information, components, and test results and approve or reject the package for install. An approver who rejects the package must enter text Reject Reasons.
- **Revert Package** removes all previously entered approvals, unlocks package information and components, and reopens the package for development.
- **Distribute Package** starts automatically when all required package approvals are received for a package that is scheduled for install at a remote site. The package is transmitted to the remote site, where package records are added to a ChangeMan ZMF P instance running there. Package staging libraries are allocated and populated, then the package is added to the P instance internal scheduler.
- **Install Package** starts automatically, either when the package install date and time arrive, or when the last approval is entered. If the application has production libraries that are separate from baseline libraries, current production modules are backed up and new versions are copied from package staging libraries into the production libraries.
- **Baseline Package** starts automatically after a package is installed. This process ripples current and prior versions of package components down in the stack of prior baseline versions, then copies package components into the baseline libraries as the new current version.

- **Backout Package** removes package components from production libraries and restores the backups made during package installation. Package components that are the current version in baseline libraries are removed, and components are reverse rippled up the stack of prior baseline versions to restore the old current version. This process includes components that have been scratched.

Inside ChangeMan ZMF Development

Behind the displayed ChangeMan ZMF panels, there are jobs being performed that ensure the smooth flow of enhancements to each application maintained by development analysts.

Create

Create is the first step of the ChangeMan ZMF lifecycle. After you create a change package, ChangeMan ZMF allocates staging libraries as needed. The data set names of the staging libraries reflect the application mnemonic chosen for your application, the package number assigned for this change, and the type of components placed in the library; for example, demo.cmnstage.#000023.src. The Global Administrator decides on the format of the data set name. The package information is recorded on the package master along with the TSO ID of the creator. A record of this event (package creation) is placed on the log.

Checkout Process

Checkout is the process of copying components from the baseline library (any level back or from promotion libraries) to a staging library or a development area outside of ChangeMan ZMF for modification in a change package. You can check out online or in a batch job. If you check out in batch mode, ChangeMan ZMF asks you to verify (initially type or update) the jobcard statements needed to perform the batch job.

When you check out a component, the standard ISPF statistics are carried forward and the version number (the *vv* portion of *vv.mm*) is incremented. ChangeMan ZMF adds the checkout information to the statistics that make up the component history. A record of this event (checkout component) is placed on the *Activity Log*. Anyone can browse this log for information not only on checkout actions, but also other ChangeMan ZMF activities.

When you associate the checkout to a valid change package ID, the component name is added to the package staging list. This means that when you select the stage option from the Build Options Menu and select to stage from the Package Driven option, the component is already listed with a checkout status.

Staging Process

Staging introduces components into ChangeMan ZMF by copying them from development or personal libraries into ChangeMan ZMF staging libraries. All staging library components must be associated with pre-defined change packages.

Depending on how your administrator configures staging parameters for your site, you can either stage any newly created application component into any change package, or only components previously associated with (that is, checked out to) change packages.

For instance, your administrator may want to restrict new development on an application and designate that only existing components be maintained. The administrator can restrict the staging process so that only components previously associated with change packages can be staged back into the change cycle.

Before staging, verify that your administrator has:

Assigned compile procedures for each language type you intend to stage.

- Assigned appropriate compilers during installation of ChangeMan ZMF.

Staging libraries contain components of the same type. The following table lists component types that ChangeMan ZMF recognizes and considers when staging.

Type	Description
SRC	Source modules
LOD	Load modules
JCL	JCL
DOC	Documentation
CPY	Copybooks
LCT	Linkage Control Cards
LIKE SRC, LIKE LOD, LIKE CPY	Assign this type to SRC, LOD, or CPY components when you want to stage components of same type into separate staging libraries.
LIKE N	Like-NCAL: NCAL load subroutines. Once staged they are concatenated in the SYSLIB for links within the same package (if the lib type is present in the package).
LIKE O	Like-Object: Object code subroutines. Once staged they are concatenated in the SYSLIB for links within the same package (if the lib type is present in the package).
OTHER	Assign this type to components when you want to customize processing of a component.
PRC	Compiling procedures

You can stage components online, or stage them in batch.

Staging Type	Enables User to Do the Following:
Online	Use *confirmation panels* to review relevant parameters and compile procedures prior to staging a component.
	Use *language assumption* (a feature that automatically assigns language types) and designated compile procedures when staging source components.
Batch	Stage multiple components simultaneously.
	Stage complete libraries of components.
	By-pass confirmation panels to stage components faster.
	Use the *language assumption* feature.

Auditing

When you audit a package's staging libraries, ChangeMan ZMF analyzes and reports on every module contained in your change package with respect to the baseline versions. The Audit function also validates all copies and program calls, producing a report listing all duplicates and out-of-sync conditions. (Audit also includes copybook promotion libraries when generating the hash token table.)

Freezing Packages

When you are ready to freeze the package for promotion (optional) and approval (required), ChangeMan ZMF checks two things:

- Are all components in an Active status?

  During the stage process, if the component is successfully copied into the appropriate staging library and if source components have compiled, link/edited, and their bind has completed successfully, then ChangeMan ZMF changes the status of the component to Active.

- Did the package pass the audit?

  The audit level selected by the application's administrator must not be exceeded.

  When the package is successfully frozen, the status of the package changes from DEV to FRZ, which locks out anyone from staging into the package libraries. A record of this event (freeze package) is placed in the Activity Log.

Promoting Packages and Components

The Promotion facility allows you to set up intermediate environments or promotion levels where you can perform quality assurance, unit, and system tests on packages and components.

Promoting involves migrating change packages or components through these intermediate environments. Demoting is the deleting of components logically or physically from these environments.

Before using the Promotion facility, your application administrator must first set up:

- Promotion levels. You can have one or more levels of promotion, each level having one or more libraries associated with it.
- Promotion process. You can promote packages and components online or in batch.
- Promotion authorization. Each promotion level can be secured. Your administrator can build rules in ChangeMan ZMF and your security system that designates which users can promote a package to a specific level.

  Generally, you promote packages from staging libraries to specified promotion levels.

  The following functional characteristics of the Promotion facility may affect decisions you make about when and how to promote and demote packages and components:
- After components are copied from package staging libraries, they still reside in the staging libraries. This implies that you should only include executable libraries in your promotion environment. Source modules do not have to be promoted because they will be retained in the package libraries.
- Promotion from level to level may be a logical copy or a logical move; that is, the components may remain in the previous environment, or they may be deleted from the previous environment upon promotion.
- Each time you promote (or demote), ChangeMan ZMF updates the statistics constituting the component history. A record of this event (promote package) is placed in the Activity Log.
- Staging skeletons for source components may reference promotion copybook libraries as part of the copybook concatenation. Therefore, if copybooks are promoted, they may be made available to source compilation of other packages.
- ChangeMan ZMF does not require the use of promotion, even if it has been set up by an administrator. Moreover, upon completion of the approval process, the package is distributed (and installed) regardless of the level of promotion reached. This gives you the flexibility to alter the path of migration of each package. However, if you do want to require a promotion path, you can administratively link your promotion security to your approval security. This technique allows a promoter to submit approval of a package once it has been successfully promoted and tested.

Approving Packages

When a person accesses the ChangeMan ZMF panels, that person's TSO ID is passed along and used to determine which functions are available. Approval may be performed only by those TSO IDs associated to the entity names that the application administrator specified as approvers.

- The approval process consists of browsing the package information and staging libraries for quality control and standards and selecting to Approve (or Reject) the package.
- A record of this event (Package Approval) is placed in the Activity Log. The package status is changed from FRZ to APR.
- All approvals for a package must be gathered before ChangeMan ZMF Installs a package. In fact, the final approval of a package actually initiates or schedules the package Installation.
- A change package must be in frozen (FRZ) status to be approved or rejected.
- In general, a package's components cannot be modified while in frozen status. This implies that a package's components cannot be modified while approvals are being gathered. Components can be selectively unfrozen, modified, and refrozen while the package is still in frozen status.
- There can be multiple levels of approvals. ChangeMan ZMF requires at least one approval, but allows administrators to set up more than one level.
- Multiple levels of approval can be set up in a hierarchy. ChangeMan ZMF enforces an order of approvals, and does not allow approvals to be gathered out of order.
- More than one User ID can be authorized to satisfy a given approval level. This is set up in your security system.
- Your application administrator may have set up approval notifications. Each approval level can be configured with multiple User ID notifications. The User IDs that are notified may or may not coincide with the User IDs that can actually satisfy the approval.
- Different packages may have different approvals. ChangeMan ZMF allows administrators to set up separate approval Lists by application and by time of day. ChangeMan ZMF attaches an *abbreviated* approval list to unplanned packages created outside of normal business hours, and a *complete* approval list to all other packages. Your administrator may have tailored a user exit to customize approvals lists further.
- ChangeMan ZMF provides special processing for packages with an abbreviated approval list attached. These approvals must be gathered before the package can be Installed. Once installed, the package continues to be available for approval or rejection by approvers on the *complete* approval list. This allows for a post-installation approval strategy.
- Packages can be promoted and demoted while approvals are being gathered. The final approval of a package Installs it, *regardless of the promotion status*. Therefore, the final approver of a package should be sensitive to the promotion activities of packages.

- If a package is rejected, it must be *reverted* if it is to be updated to conform to the reject reasons. A package revert action resets the rejection and places the package in development status. The package must then be frozen again to reinitiate approvals.
- If a package was promoted before it was rejected, it must be demoted before it can be reverted.
- Package revert resets any gathered approvals. This is true regardless of whether or not the package is first rejected.

Installation

Installation depends on whether or not an internal scheduler is set up by the global administrator, or if the Install job JCL has been modified. There are four variations on Installation:

- If no scheduling system is specified, the package goes through the installation process immediately.
- If the Install job JCL is set up with a TYPRUN=HOLD, the user releases the job when they are ready to install.
- If a scheduling system other than the ChangeMan ZMF internal scheduler is specified, then ChangeMan ZMF performs a batch interface to add the install job to the scheduler's list. The operator, however, must still demand the job for the package to be installed.
- If ChangeMan ZMF is the scheduler, it checks the package master every few minutes for any packages that are ready and installs those that meet the criteria.

Backing Up

Backup is the first job to be performed when installation time arrives. This job copies the production libraries (only those components that are about to be overlaid with updates) to a backup set in case they are needed to back out the incoming enhancement. Next, the contents of the change package staging libraries are copied into production libraries. A record of this event (package installation) is placed on the Activity Log. This occurs each time the package is Installed at one of the remote sites.

Once the package is verified as Installed in all requested sites, the following steps are performed:

1. The package status is changed from DIS to INS (or from APR to INS if there are no remote sites).
2. A job is sent to the development center to clear out the last level of promotion reached, and to ripple the baseline libraries for that application.
3. The package status is changed from INS to BAS.
4. A record of the baseline ripple is placed in the log.

> **Note**
>
> Only the various versions of changed software components are updated; ChangeMan ZMF ripples the changes through the versions of an application's baseline libraries.

Assume that the following is true:

- An application maintains up to three versions of its baseline library software: current(0), -1, and -2.
- You want to update the baseline libraries with a change package in which component A is changed, component B is scratched, and component C is added.
- There already is a -1 and -2 version of component A. Thus, the baseline library is updated as follows:
- The -1 version of component A is copied to overlay the -2 version of component A.
- The 0 version of component A is copied to overlay the -1 version of component A.
- The newly-installed version of component A is copied from the production staging libraries to overlay the baseline library 0 version of A.
- Component B is scratched.
- The newly installed version of component C is copied from staging libraries and added to the baseline 0 libraries.

Backing Out Packages or Components

If there is a problem with the change package after it has been Installed, the change package is backed out by deleting the updated component in production, and then retrieving the previous version of application software from the Backup library. This option is selected by an authorized user in the production environment (usually an operator or production analyst).

ChangeMan ZMF backs out the entire package by copying the components from the backup libraries to overlay production, including components that have been scratched. The package status is changed from BAS to BAK. A record of the package backout is placed in the Activity Log.

A job is submitted in the development area to reverse ripple the baseline library. A record of the baseline reverse ripple is placed in the Activity Log.

Temporary Change Cycle

When a temporary change package is created, the user must type the number of days the change is to remain in the temporary (override) environment (if your global administrator has setup this option). The installation process is different from other package types because the contents of the temporary change package staging libraries are copied into temporary libraries (which are concatenated ahead of production libraries). Because the production library components are not touched, ChangeMan ZMF does not perform the hot backup.

The components are never rippled into the baseline library. After the package is Installed, ChangeMan ZMF begins the aging process at each site selected to receive the temporary change. The components in the temporary library are deleted when the number of elapsed days is met. If you use a scheduler, the job automatically runs. If you use a manual scheduling method, the job is submitted on hold, and must be released when the duration of days is met. After the package is deleted from all sites, its status is changed from INS to TCC, and a record of this event (Temporary Change Cycle Completed) is placed on the Activity Log.

Distribution to Remote Sites

The next step after approval depends on the environment type configured for the site.

- If there are remote sites, then the package staging libraries, the installation JCL, and a copy of the package master records pertaining to this change are distributed (copied) to all the sites specified in the creation/update package process. A record of the package distribution is placed in the Activity Log, and a distribution acknowledgment is sent back to the development center. The package status is changed from APR to DIS.
- If remote sites exist, the package is ready for installation. For further information, see Installation

Distributing and Installing Components at Remote Sites

Remote sites are additional CPUs where ChangeMan ZMF installs components. An additional CPU can be:

- A separate computer in another building
- A separate computer in the same building
- A logical CPU on the same machine as part of an LPAR (logical partition) without shared DASD

  Any of these remote site configurations enables you to develop components on one CPU, and distribute and install production-level components on a different CPU.

Remote sites act only as receivers of production level components. The only time developers interact with remote sites is when they select which remote site to distribute and install production level components.

ChangeMan ZMF Library Environment

Checkout

Checkout enables you to reintroduce components that reside in baseline or promotion libraries to the change cycle. Generally, production-level components are checked out for modification. You can check out any previous version of a baseline component that exists.

Depending on the way your administrator configured ChangeMan ZMF, you can check out components:

- To personal libraries.
- To staging libraries.
- Only if they are associated with change packages.
- In batch.
- Online.
- Concurrently with other developers.

  If your site has applications that require parallel development, you can configure ChangeMan ZMF to allow concurrent checkout of components. ChangeMan ZMF has an automated process for managing this concurrent development. As part of this process, ChangeMan ZMF ensures that each owner of a version is aware of the actions of the other owners.

  After you check out components and make necessary modifications, ChangeMan ZMF records the components and the associated change package for further impact analysis. This ensures that your developers are always working with the proper version of a component.

Impact Analysis

To analyze the impact of changes, many organizations rely on data from a variety of sources, such as batch library scans and cross reference files. This method makes it difficult to maintain all sources of data and ensure that they are current. ChangeMan ZMF provides a comprehensive facility to capture, query, and *enforce* relationships between components.

These relationships include not only the traditional ones, such as a source and executable relationships, but other relationships based on common references to copybooks, SQL Include components, CA Panvalet ++INCLUDE components, CA Librarian - INC components, called subroutines, and JCL fields such as program name, filename, or data set name.

Staging

Staging is the process of introducing newly-developed or previously-developed components into the ChangeMan ZMF change cycle for modification or enhancement, and packaging them with related change package components. When you *stage* a component, ChangeMan ZMF recognizes the type of component that you are staging and copies it into a *staging* library of the corresponding type (source, load, JCL, documentation, copybook, and so on.). Staged components are also associated with a pre-defined change package, the vehicle ChangeMan ZMF uses to move components through the change cycle and track the history of change management activities for each staged component.

In change management systems other than ChangeMan ZMF, staging libraries are merely pre-production holding areas shared by one or more application groups. After components are tested in development libraries, they are copied into staging libraries prior to production implementation.

ChangeMan ZMF staging libraries, however, are more than pre-production holding libraries. Components can be modified and tested *in* protected ChangeMan ZMF staging libraries. When you stage source components, they are compiled, and the resulting load modules are identified, helping you to maintain the integrity of source-to-load relationships.

ChangeMan ZMF maintains up-to-date records of *all* staging activities for packages and components. For example, when you stage a source component, ChangeMan ZMF records the time that the component was staged, the name of any associated load modules or copybooks, and the compiling procedures and linkage *parameters* used during the compile. This information is kept in the ChangeMan ZMF master file (the package master). You can view this component and package information by using the *query* function.

ChangeMan ZMF further extends the concept of staging by isolating components from other changes in progress. This prevents uncontrolled and unknown copybooks and subroutines from being inadvertently referenced, allowing parallel or concurrent development without the risk of accidental overlays. The stable coexistence of multiple versions of a single component simplifies the blending of changes.

Audit

The ChangeMan ZMF audit process ensures correct synchronization of components and procedures. Because of the range of features offered by the package master and the impact analysis database, ChangeMan ZMF maintains control of current and past modifications and component versions. Potential production problems can be identified before they impact production.

The audit function inspects the staging library contents of an evolving change package (in the DEV/FRZ status) with respect to baseline library contents. The inspection looks for situations such as a package that shows no change from the baseline library, or a package that contains a LOD component that does not match its SRC component. Recognizing these situations (called *out-of-sync* components), ChangeMan ZMF helps you to detect code that is inconsistent with your development procedure and other code problems.

Examples of out-of-sync situations include:

- Copybooks that were changed after a source program was compiled.
- Source programs that must be recompiled due to a copybook change.
- Called subroutines that were changed *after* a referencing source program was compiled and linked.

You can specify if you want an audit, and if so, whether or not you want to correct or ignore uncovered problems.

Recompile and Relink

You can use audit to analyze the staging library contents of an evolving change package with respect to baseline contents, for the purpose of finding any out-of-sync situations.

The recompile function resolves certain types of *out-of-sync* conditions found during the audit. The allowable audit return code is determined during global and application parameter generation, and you are not allowed to freeze the change package without passing the audit return value entered for the application.

Using the relink option, you can relink load components without associating them with source code.

The relink process is similar to compile because you select a component from a baseline list. A new load component is produced and copied into the package staging library.

Use the delete function to remove recompiled or relinked components that do not have associated source in the package. You can also delete the resulting LST file and any other non-load components that were associated with it through the CMNBAT90 service. (See your administrator for details on this service.)

The component history is picked up from the history record for that component in the package master. For example, the relink picks up the user options on CMNUSR01 that were there when the program was last relinked.

When relinking, you can include LCT cards that contain the link control cards from staging or baseline libraries, or you can dynamically generate them if there is no LCT component available. You do this if you:

- Do not have source code for a component, but make a change to a subroutine.
- Must perform a composite link where the resulting load component name does not have accompanying source.

Freeze

Another unique ChangeMan ZMF feature is the ability to freeze change packages. When the change package is ready for the next phase of the change implementation lifecycle, a freeze is performed to prevent further modifications. The freeze also positions the change package for promotion or approval. Traditional methods accomplish this function by moving components from the development libraries to a separate set of libraries or, in some cases, separate environments. With ChangeMan ZMF, however, the ChangeMan ZMF instance controls your updates in conjunction with your security system, so component movement is no longer necessary.

If further modifications are required, you can unfreeze a change package, and the approval process is reset.

Promotion

ChangeMan ZMF can promote change packages through multiple, shared, pseudo- production promotion environments. These promotion environments are secured as if they are production, and ChangeMan ZMF controls all updates.

ChangeMan ZMF considers shared promotion environments to be places where full integrated system testing can be performed. When the time comes for a full system or an integrated system test, authorized approvers promote the acceptable components into the promotion environments.

When testing is complete and the change package is approved, ChangeMan ZMF (optionally) removes the components from the promotion environments. All production installation occurs from the change package staging environment. With ChangeMan ZMF, you define your testing methodology and the number of testing levels that are required.

Approve

Approvals for change package installation are performed online, eliminating the requirement for manual approval processes. During the ChangeMan ZMF approval process, authorized approvers can indicate that the change package is acceptable for production implementation, or they can reject or review the change and generate a checklist of questionable or unclear items.

ChangeMan ZMF relies on your security system. ChangeMan ZMF does not use internal personnel tables. Approval lists of specific User IDs or approving entities are defined to your security system so that electronic signatures can be collected.

For each application, multiple approvers can be included in an approver list. Separate approval lists can be created for scheduled planned changes or unplanned emergency changes, or you can choose to use an approval hierarchy. With ChangeMan ZMF, you have the flexibility to make these choices.

Production Installation

ChangeMan ZMF is involved in the management and control of production component installation. Component installation can be automated through the ChangeMan ZMF internal scheduling system, or through a direct interface with a job scheduling system. In addition to component movement, ChangeMan ZMF performs other production installation activities such as Db2 Plan binding.

ChangeMan ZMF also has a change quantity threshold facility that allows you to control the number of changes that occur in a time period. For example, you may want to limit the number of change packages that are installed during month-end processing.

Baseline Libraries and Delta Decks

ChangeMan ZMF recognizes that your software components are important business assets. ChangeMan ZMF gives you the ability to store your production source components in a structure that works for your organization. Components can be stored in PDSs, CA-Librarian files, or CA-Panvalet files. Components can be segregated by application, or by categories, such as batch versus online. Equally, applications can share libraries.

ChangeMan ZMF automatically stores prior versions of components. These versions can be stored as full copies (inherent for load components), or as *delta decks*. ChangeMan ZMF uses a reverse base/delta technique known as *stacked reverse deltas*. With this technique, the current version of the component is the base, and delta decks are created to backtrack to previous versions.

Backout Management Facilities

Comprehensive backout management requires more than simply *backing up* the components of a change.

ChangeMan ZMF has comprehensive backout management facilities. In addition to source components, the prior functioning executable components are backed up. If a backout becomes necessary, ChangeMan ZMF automatically restores these executable components to production. ChangeMan ZMF also performs all necessary Db2 Plan rebinding automatically.

Because ChangeMan ZMF is package driven, it backs out all the components of a change automatically, including components that have been scratched.

Emergency Changes

Critical abends can occur at inopportune times, and require immediate attention. Because ChangeMan ZMF can create unplanned change packages, and because it maintains a separate list of approvers for unplanned changes, emergency changes are safe, fast, and easy to perform. ChangeMan ZMF notifies developers affected by the change so that the emergency fix can be incorporated globally into all change packages.

ChangeMan ZMF does not impede the emergency *change process* by requiring that the component be released, reassigned, or renamed by the original owner.

Storage Name Considerations

CA Panvalet allows ten-character names, which ChangeMan ZMF does not recognize, because it looks for eight-character names. References in this manual assume PDS naming is the convention.

Build Processing Controls

Change management best practices require consistent, repeatable build processes.

ChangeMan ZMF offers a variety methods to restrict build processing to provide administrators with the level of level consistency they want, and the level of flexibility they want to offer to application developers.

Build Processing Consistency	ChangeMan ZMF Processes and Configuration
Minimum	The compile procedure and build options that a developer enters on ChangeMan ZMF panels are recorded in package component records. These values are shown on build process panels the next time a build process is initiated for the component in the package, but a developer can change the information. When the package is installed and the component is baselined, build process information stored in package component records is written to the Component History File. When the component is checked out to another package, the compile procedure and build options are copied from component history to package component records. These values are shown on build process panels when a build process is initiated for the component in the package, but a developer can change the information.
Medium	Some compile and link edit options are hard coded in compile procedure skeletons. Some compile and link edit options are prohibited, and if a developer uses them, the package cannot be frozen and the component cannot be selectively refrozen. (Exit program CMNEX025)

Build Processing Consistency	ChangeMan ZMF Processes and Configuration
High	Application administrator defines a designated compile procedure with a Force Level 1 for single component or a group of components. The designated compile procedure specifies the compile procedure skeleton and build options that must be used the last time a component is built before the package is frozen.
Maximum	Application administrator defines a designated compile procedure with a Force Level 2 for single component or a group of components. The designated compile procedure specifies the compile procedure skeleton and build options that must always be used in build processing for the component.

Designated Compile Procedures

A designated compile procedure imposes consistency in build processing for a component before it is installed into production. A designated compile procedure can eliminate all variation in build processing for a component throughout the development life cycle.

When your administrator defines a designated compile procedure in application administration, the following build process information is specified for a component name, or for a name pattern, in a library type, for an application.

Field	Description
Language	The coding language of the like-source member. Valid language names are defined in application administration
Compile Procedure	The name of the main ISPF skeleton that is file tailored to create build job JCL. Valid compile procedure names are associated with a language name in application administration.
Compile Parms	A 34-byte field where users set compile options that are not set by: System defaults for the compiler. Options that are hard coded in Compile Procedure skeletons. User Options that set variables used by file tailoring to set compile options
Link Edit parms	A 34-byte field where users set linkage editor or binder options that are not set by: System defaults for the linkage editor or binder. Options that are hard coded in Compile Procedure skeletons. User Options that set variables used by file tailoring to set link edit or binder options
Db2 Precompile Indicator	An indicator that determines whether a Db2 precompile step is included in the build job JCL.
Force Level	Determines when the compile procedure and other build options in the designated procedure must be used.
	1. The compile procedure and build options in the designated procedure must be used in the last build before package freeze or component refreeze.

Field	Description
	2. The compile procedure and build options in the designated procedure must always be used.
Additional options	Fifty-eight fields ranging in length from 1-72 bytes that ChangeMan ZMF installers and administrators can define for variables or indicators used to file tailor JCL for build processing.

See the **ChangeMan/ZMF Administrator's Guide** - *Setting up Application Administration* for more details.

Build Information Search Order

Even when there is no designated compile procedure to impose consistency in build processing, ChangeMan ZMF encourages consistency by populating build processing panel fields with values used previously to build the component.

Change Man ZMF uses this search order to obtain values for the compile procedure skeleton and build options for a component. Component history is keyed by component name within library type.

Search Sequence	Location of Build Information	Comment
1	Force Level 2 Designated Compile Procedure	Fields on build panels are set to the values in the designated procedure, and the fields are displayed in read-only mode.
2	Stage: Mass Build panel Batch Mass Recompile Job Information panel	If the Suppress History field is Y on these panels, panel values are used in build processing for all selected components.
3	Package Component Records	These records contain the values used the last time you performed a build for this component/library type in your package. These records are deleted if you delete the component from your package.
4	Component History	This history contains the values used to build the component/library type that was last baselined.
5	Designated Compile Procedure with Force Level 1	Fields on build panels are set to the values in the designated procedure, but you can overtype the values.
6	Build process ISPF panel.	All required build panel fields must be filled to provide a default values in case they are needed for a new component.

3. Using the ISPF Interface

The ChangeMan ZMF ISPF client is a standard ISPF dialog manager application. It will look familiar to users who are experienced in ISPF.

You can display online help for ChangeMan ZMF panels by pressing **PF1**. *PF1 Help is your friend. (Anonymous)*

Start a Session

You can initiate a ChangeMan ZMF session in ISPF in one of two ways:

- Select an option from an ISPF menu that executes a logon CLIST, or
- Execute the CLIST directly.

Ask your ChangeMan ZMF administrator how to connect to ChangeMan ZMF in your environment.

When you start your session

The initialization panel is displayed:

```
+----------------------------------------------------------------------+
|                                                                      |
|              ChangeMan(R) ZMF                                        |
|              Version = 8.2.0                                         |
|                                                                      |
|              Initialization in progress                              |
|                                                                      |
| Copyright (C) 1985 - 2018 Serena Software, Inc., a Micro Focus company. |
| Licensed material. All rights reserved.                              |
| ChangeMan is a registered trademark of                               |
| Serena(R) Software Inc.                                              |
|                                                                      |
+----------------------------------------------------------------------+
```

When initialization is complete, the ChangeMan ZMF **Primary Option Menu** appears.

ChangeMan ZMF Primary Option Menu and Build Options Menu

ChangeMan ZMF functions are arranged in a hierarchy of menus, submenus, and functional panels. Most functions in the ChangeMan ZMF change package lifecycle are accessed from the **Serena ChangeMan ZMF Primary Option Menu** or from the **Build Options** submenu.

Primary Option Menu

The **Primary Option Menu** (CMN@PRIM) is the first panel displayed in the ChangeMan ZMF ISPF client.

```
CMN@PRIM Serena(R) ChangeMan(R) ZMF Primary Option Menu SYS(6)

Option ===>----------------------------------------------------
0   Settings    User parameters
1   Build       Create, update and review package data
2   Freeze      Freeze or unfreeze a package
3   Promote     Promote or demote a package
4   Approve     Approve or reject a package
5   List        Display (to process) package list
6   Reports     Generate ChangeMan ZMF batch reports
7   Release     Extended Release Management
A   Admin       Perform administrative functions
B   Backout     Back out a package in production
C   M+R         Merge+Reconcile
D   Delete      Delete or undelete a package
L   Log         Browse the activity log
M   Monitor     Monitor internal scheduler or packages in limbo
N   Notify      Browse the Global Notification File
O   OFMlist     Online Forms package list
Q   Query       Query packages, components and relationships
R   Revert      Revert a package to DEV status
T   Tutorial    Display information about ChangeMan ZMF
X   Exit        Exit ChangeMan ZMF
```

The options displayed on the **Serena ChangeMan Primary Option Menu** are determined by the authorization of your TSO ID. When you connect to ChangeMan ZMF, it checks your security system to see if your TSO ID is associated with security entities that allow access to ChangeMan ZMF administrative functions. Then ChangeMan ZMF dynamically builds the **ChangeMan Primary Option Menu** with only those options that you are authorized to use.

Your **Primary Option Menu** may not include the following options, which require special authorization in your security system:

```
A Admin - Perform administrative functions

B Backout - Back out a package in production

M Monitor - Monitor internal scheduler or packages in limbo

R Revert - Revert a package to DEV status
```

Three other items displayed on the Primary Option Menu shown above are included only if your organization licenses the ChangeMan ZMF option.

```
7 Release - Extended Release Management

M+R       - Merge+Reconcile (formerly CDF)

O OFMlist - Online Forms package list
```

From the **Primary Option Menu**, you can select functions directly or select submenus that lead to the ChangeMan ZMF functions you want to execute.

You can return to the **Primary Option Menu** from almost anywhere in ChangeMan ZMF by typing RETURN and pressing ENTER.

The **Primary Option Menu** (CMN@PRIM) is the first panel displayed in the ChangeMan ZMF ISPF client.

```
CMN@PRIM     Serena(R) ChangeMan(R) ZMF Primary Option Menu  SYS
Option ===> -----------------------------------------------------
0   Settings        User parameters
1   Build           Create, update and review package data
2   Freeze          Freeze or unfreeze a package
3   Promote         Promote or demote a package
4   Approve         Approve or reject a package
5   List            Display (to process) package list
6   Reports         Generate ChangeMan ZMF batch reports
7   Release         Extended Release Management
A   Admin           Perform administrative functions
B   Backout         Back out a package in production
C   M+R             Merge+Reconcile
D   Delete          Delete or undelete a package
L   Log             Browse the activity log
M   Monitor         Monitor internal scheduler or packages in limbo
N   Notify          Browse the Global Notification File
O   OFMlist         Online Forms package list
Q   Query           Query packages, components and relationships
R   Revert          Revert a package to DEV status
T   Tutorial        Display information about ChangeMan ZMF
X   Exit            Exit ChangeMan ZMF
```

Settings Submenu

To access the **Settings** submenu, select **0 Settings** from the **Primary Option Menu**.

```
CMNUSET                    User settings
Command ===>------------------------------------

Enter "/" to select option
 - Save user sort parameters
 - Panel fields SKIP(ON)
```

Save user sort parameters

Select to remember how item lists are sorted. If selected, item lists are sorted according to the last SORT command entered by the user for each type of item list. If not selected, items are displayed according to a default sort order.

For example, if this option is selected and "SORT INSTALL" is entered to sort packages by install date in option 5 (package list), then the package list will be display in install date order in the current and future sessions. If this option is not selected, the package list will revert to being sorted by package name the next time it is displayed.

Panel fields SKIP(ON)

Select to automatically skip to the next input field when a character is entered into the last character location of an input field.

Navigating Panels

There are several ways to navigate the hierarchy of ChangeMan ZMF menus, submenus, and functional panels to access the function you want to use.

Using the Menu Hierarchy

You can access every function in ChangeMan ZMF by following a path through menus and submenus.

Starting on the **Primary Option Menu**, select an option by typing the option number or letter in the **Option** field, then press **Enter**. If a submenu is displayed, select an option on that menu, repeating the process until you reach the function you want to execute.

For example, to access the **Audit Change Package** panel:

1. On the **Primary Option Menu**, type **1** in the **Option** field and press **Enter** to display the **Build Options** menu.
2. On the **Build Options** menu, type **7** in the **Option** field and press **Enter** to display the **Audit Change Package** panel.

   To return to a previous menu or panel, type **End** in the **Option** or **Command** field and press **Enter**, or press **PF3**. To return to the **Primary Option Menu**, press **PF4**.

   Use panel-by-panel navigation to learn ChangeMan ZMF. When you become familiar with the functions in ChangeMan ZMF and how they are arranged in the panel hierarchy, you can use other methods of navigation.

Accessing Panels Directly

You can access ChangeMan ZMF submenus and functional panels directly from other panels in the panel hierarchy.

In the **Option** field or **Command** field of the panel you are on, first type an equal (**=**) or plus (**+**) sign. This is followed by a sequence of numbers and letters separated by periods (**.**) that represent the sequence of menu options you would use to reach the desired function from the **Primary Option Panel** using the menu hierarchy.

For example, to reach the **Audit Change Package** panel from another panel in ChangeMan ZMF, type one of the following in the **Option** or **Command** field and press **Enter**:

```
=1.7
or
+1.7
```

You can often type a direct panel access path in other fields on a panel besides the

Option or **Command** field, which makes direct panel access very efficient.

When you press **PF3** from a panel you accessed directly, you are usually returned to the **Primary Option Menu**. In addition, you can always return to the **Primary Option Menu** by pressing **PF4**.

Accessing Panels Using the Change Package List

Almost all package functions in ChangeMan ZMF are available on the **List - Display (to process) package list**, which is Option **5** on the **Primary Option Menu**. In the Change Package List, the hierarchy of ISPF menus and submenus is restructured into a list of two- character options that you can type as line commands on a package list panel. If you typically work with a single package or a group of related packages, the Change Package List is an especially efficient way to work in ChangeMan ZMF.

The Change Package List is described in Managing Packages With The Package List.

Online Help and Online Tutorial

ChangeMan ZMF includes online help and an online tutorial.

Accessing Online Help

Press **PF1** from anywhere in ChangeMan ZMF to display an online help panel. Online help explains the function of the ChangeMan ZMF panel where you pressed **PF1** and describes the panel fields. Online help is not context sensitive, so you may have to page down through the help to find information about a particular panel field.

> **Tip**
>
> In ISPF-based Help, press Enter to go to the next page.

If you have questions about navigating the online help, press **PF1** at any online help panel to display a description of help navigation commands. Press **Enter** to exit this information and return to online help.

Press **PF3** to exit help and return to the ChangeMan ZMF panel where you started.

Accessing the Online Tutorial

ChangeMan ZMF includes an online tutorial to help you learn about ChangeMan ZMF functions, panels, and data in the context of the ISPF client. It is organized as a standard ISPF tutorial arranged in a hierarchy that reflects the hierarchy of menus and panels in the ChangeMan ZMF ISPF client.

To view the tutorial, select **T Tutorial** on the **Primary Option Menu**.

If you have questions about navigating in the ChangeMan ZMF tutorial, press **PF1** from inside the tutorial to display a description of navigation commands. Press **Enter** to exit this information and return to the tutorial.

Press **PF3** to exit the tutorial and return to the **Primary Option Menu** where you started.

Working With ChangeMan ZMF ISPF Panels

ChangeMan ZMF uses the ISPF Dialog Manager to simplify data entry, validate the data you enter, and provide information when an error occurs.

Data Entry Fields and Display-Only Fields

Data entry fields differ from display-only fields on ChangeMan ZMF panels as follows:

- If you have a color monitor, data entry fields display in a different color than display- only fields.
- If you have a monochrome screen, the intensity of data entry fields differs from display-only fields.
- Data entry field labels are followed by ===> or a series of leader dots. Display-only field labels are followed by a series of leader dots.
- If the field attributes for a panel change dynamically from data entry-enabled to display-only, the field colors change and BROWSE ONLY appears in the ISPF short message, at the upper right of the panel.

Panel Error Messages

If you enter invalid data in a field, or if you make an invalid request for a function, ChangeMan ZMF displays a short error message in the upper right corner of the panel. Press **PF1** to display a long error message that provides details about the error condition.

> **Note**
>
> The long message does not display automatically. You must request the long message by pressing PF1.

Press **PF1** again to display **online help** for the panel where you made the error.

Finding Valid Values for Fields, Commands, and Line Commands

When you do not know what value to use in a field, command, or line command, use one of the following techniques.

Press PF1 for Online Help

Press **PF1** to display an online help panel that explains the purpose of the panel you are using, describes panel fields, and lists valid values for commands, line commands, and panel fields.

Intentionally Trigger an Error Message

To get an error message that may tell you what values are valid for a field, command, or line command, follow these steps:

1. Type **?** (or any invalid value) in the field, command line, or line command and press **Enter**. A short ISPF message appears in the upper right corner of the screen, indicating that you have typed an invalid value.
2. Press **PF1** to see the long version of the error message. The long message will usually tell you what values are valid for a line command.
3. Press **Enter** to clear the long message so you can type a valid value in the field.

> **Note**
>
> If you attempt this technique on the Change Package List or the Stage: package Components panel, a panel is immediately displayed that lists all of the valid line commands with a short description of each.

Trigger a Selection List

If you do not know what value to use in a field, leave the field blank and ChangeMan ZMF may display a selection list of valid values.

Canceling Changes on a Panel

To cancel information on a panel without executing the panel, type **Cancel** at the **Command** or **Option** line and press **Enter**.

Typing **End** and pressing **Enter**, or pressing **PF3**, may either cancel the panel or cause the information on the panel to be processed.

Read the instructions on each panel to see which method to use.

Working With Lists

ChangeMan ZMF makes extensive use of lists to help you work efficiently with packages, components, and the data required on ISPF panels.

Common List Commands

Common list commands for ChangeMan ZMF are itemized below. To use one of these commands, type it in the **Command** line of a list panel and press **Enter**. You may be able to abbreviate the command by typing the first one or two characters.

- REFRESH reads ChangeMan ZMF database and updates the displayed data.
- CANCEL leaves the list without processing any line commands you have entered.
- SORT [heading] sorts the list on the data under the column heading you specify.
- LOCATE finds the first occurrence of a value you specify in the column you last sorted. If you have not sorted the list, LOCATE looks in the first column.

Building Lists Using Patterns

You can use a pattern with wildcard characters in some data entry fields to select a subset of list items. The subset displays in a new list panel.

> **Important**
>
> Assume that ISPF panel fields are case sensitive when you enter patterns to build selection lists. Support for zFS requires case sensitive fields.

Wildcard Characters

Wildcard characters have these functions in patterns in ChangeMan ZMF.

Wildcard	Function
\	Represents any string of characters from this position to the end.
?	Represents any character in this one position.

Package ID Pattern Examples

A package ID is composed of a three- or four-character application mnemonic and a six- character package number. When a three-character application mnemonic is used, a blank is valid in the fourth character of the package ID. On panels where a pattern can be used for package ID, patterns operate independently on the two parts of the package ID.

Starting with the following group of package IDs:

```
ABC 000001
ABC 000002
ABCD000001
ABCD000002
ABCE000001
ABCE000002
AXYD000001
AXYD000002
WXYZ000021
WXYZ000022
```

The following table shows the package IDs that are selected with each sample pattern.

Pattern	Package IDs Selected
ABC* or ABC?	ABC 000001
	ABC 000002
	ABCD000001
	ABCD000002
	ABCE000001

Pattern	Package IDs Selected
	ABCE000002
A??? or A\	ABC 000001
	ABC 000002
	ABCD000001
	ABCD000002
	ABCE000001
	ABCE000002
	AXYD000001
	AXYD000002
ABC*000001	ABC 000001
	ABCD000001
	ABCE000001
?XY?00002\	WXYZ000021
	WXYZ000022
?XY?0000?1	AXYD000001
	WXYZ000021

Note

If you omit the leading zeros in the package number, ChangeMan ZMF often assumes that zeros precede the package number. For example, if you type ABCD1 in some package ID fields, package ABCD000001 is selected or displayed. When you use a pattern, zero fill does not apply.

Using Selection Lists

When you leave a panel field blank, ChangeMan ZMF may display a selection list from which you can choose a value to fill the field. When you execute a ChangeMan ZMF function and there are multiple objects that are eligible for action, ChangeMan ZMF may display a list for you to select one or more of the objects.

There are three types of selection lists in ChangeMan ZMF: single-item selection lists, multiple-item selection lists, and extended multiple-item selection lists.

Single-Item Selection

Only one selected item is processed from a single-item selection list. To select an item, type **S** in the line command for the selected item, then press **Enter**. If you select multiple items, only the first item is processed. After you press **Enter**, processing continues with the panel where the selection list was called.

Example:

```
CMNSTG07 ------------------ LANGUAGE SELECTION LIST ---------- Row 1 to 5 of 5
COMMAND ===>--------------------------------------- SCROLL ===> HALF
LANGUAGE
_ ASM
_ COBOL
_ COBOL2
_ PLI
_ SQL
*********************** Bottom of data*****************************
```

Multiple-Item Selection List

You can select multiple items for processing from a multiple-item selection list. To select one or more items, type **S** in the line command for each of the items you want, then press **Enter**. If the list is longer than one panel, you can usually page down and up to make additional selections before you press **Enter**.

On some multiple-item selection lists, the selected items are processed and the list is redisplayed with a status shown for each selected item. In other cases, processing continues with the panel where the selection list was called.

Example:

```
CMNSTG01        STAGE: ACTP000001 Components       Row 61 to 68 of 68
Command ===>---------------------------------------Scroll ===> PAGE

Name      + Type Status     Changed         Procname User    Request
__ACPSRC99  SRC ACTIVE      20141119 030722 CMNCOB2  JPRIMER LOCKED
__PLICPYX0  CPY ACTIVE      20141118 190724          JPRIMER LOCKED
__PLICPY01  CPY ACTIVE      20141118 190727          JPRIMER LOCKED
__PLICPY02  CPY ACTIVE      20141118 190731          JPRIMER LOCKED
__PLIPGM01  SRC ACTIVE      20141119 030701 CMNPLIE  JPRIMER
__PLIPGM1   SRC ACTIVE      20141119 030633 CMNPLIE  JPRIMER
__SAMCPY1A  CPY ACTIVE      20141118 190734          JPRIMER LOCKED
__SAMCPY1B  CPY ACTIVE      20141118 190738          JPRIMER LOCKED
************************** Bottom of data **************************
```

Extended Multiple-Item Selection

With an extended multiple-item selection list, you select one or more items, then exit the list to initiate processing for the selected items.

To select one or more items, type **S** in the line command of each item, then press **Enter**. If the selection list panel contains a Status (or Request) column, *SELECT* displays in the Status field for each selected item. Deselect items by typing **D** in the line command of selected items and press **Enter**. *DE-SEL* may display in the Status (or Request) column for each deselected item. You can select and deselect items from multiple pages of the list by scrolling down and up.

To initiate processing for the selected items, exit the panel by typing **End** on the Command line and pressing **Enter**, or by pressing **PF3**. Processing usually continues with the panel where the selection list was called.

Example:

```
CMNRPM05 PROMOTE TO: SERT6 Lvl 10 (S6P1UT) Row 1 to 1 of 1
Command ===>------------------------------ Scroll ===> PAGE
       Package: ACTP000002 Status: DEV Install date: 20141231
Name   +   Type Status Promotion      Date   Time  User    REQUEST
__TESTC001 LOD         0 STAGING 2014/11/19  13:41 USER001 *SELECT*
__TESTC002 LOD         0 STAGING 2014/11/19  13:41 USER001 *SELECT*
__TESTC003 LOD         0 STAGING 2014/11/19  13:41 USER001 *SELECT*
******************** Bottom of data*******************************
```

Scrolling Through Lists

When ChangeMan ZMF builds a selection that is too long to display on one panel, use the **UP** and **DOWN** keys on your keyboard — usually **PF7** and **PF8** — to scroll up and down through the list. The scroll amount for **UP** and **DOWN** is specified in the **SCROLL** field in the upper right of the list panel. Options are **PAGE**, **HALF**, and **CSR** (cursor).

To move up or down a specific number of lines, type a **number of lines** on the Command line and press **PF7** to go up that number of lines, or **PF8** to go down that number of lines.

To move to the top or bottom of the list, type the command **TOP** or **BOTTOM** on the **Command** line and press **Enter**. You can also type **M** (maximum) in the **Command** line and press **PF7** or **PF8**.

Working with Long Fields

Component names in USS file systems can be 256 characters long, and path names can be 1024 characters long.

In the ChangeMan ZMF ISPF interface, there are three ways users can see a long component or path name on a panel whose total width is limited to 80 characters.

- Scroll in the panel field
- Zoom in on the field with EXPAND

- Display an alternate panel

In this section, example package ACTP000052 contains these components:

Staging Library (Path) Name	Component Name
/cmntp/s6/ACTP/#000052/d/JVS	hellow.java
/cmntp/s6/ACTP/#000052/d/JCT	hellow.jct
/cmntp/s6/ACTP/#000052/d/JVS	org/dom/xpath/jhfhth30long.java
/cmntp/s6/ACTP/#000052/d/JTH	org/jdom/xpath/jhfhth30long.jth
/cmntp/s6/ACTP/#000052/d/JTH	testfile.jth

Important

In USS file systems, a "component name" may include a partial directory path because the actual name of the file is unique only within a hierarchy that includes it.

When line command **S2** is entered by package JZFS000004 on the **Change Package List**

panel, the **STAGE: package COMPONENTS** panel is displayed in this format.

```
CMNSTG01     STAGE: ACTP000052 Components        Row 1 to 5 of 5
Command ===>___________________________________ Scroll ===> CSR
    Name              + Type Status Changed       Procname User Request
____hellow.java         JVS ACTIVE 20150305 145828 CMNJAVA USER016
____hellow.jct          JCT ACTIVE 20150309 184115 CMNJAR  USER016
____org/dom/xpath/jhfh  JVS ACTIVE 20150309 200446 CMNJAVA USER016
____org/jdom/xpath/jhf  JTH ACTIVE 20150309 192827         USER016
____testfile.hth        JTH ACTIVE 20150309 192103         USER016

*********************** Bottom of data ****************************
```

When a data field is *longer* than the panel field used to display it, a **+** (plus) is shown to the right of the panel field. If the panel field is in a list, the + is shown over the panel field column, as shown in the example above.

Notice that the full names of the first two and last components fit in the **Name** field. However, the names of the third and fourth components appear to overflow the Name field.

Scrolling LEFT and RIGHT

To see more of a long name that is truncated by a short panel field, you can scroll to the right, and then scroll back to the left.

- To scroll to the *right* in a field, place your cursor in the field and press **PF11**. (You can also type **RIGHT** in the **Command** line, place your cursor in the field, and press **ENTER**.)
- To scroll to the *left* in a field, place your cursor in the field and press **PF10**. (You can also type **LEFT** in the **Command** line, place your cursor in the field, and press **ENTER**.)

This panel shows the **Name** field after scrolling right one time.

```
CMNSTG01    STAGE: ACTP000052 Components        Row 1 to 5 of 5
Command ===>__________________________________ Scroll ===> CSR

Name        -+  Type Status   Changed          Procname User  Request
__              JVS ACTIVE   20150305 145828  CMNJAVA  USER016
__              JCT ACTIVE   20150309 184115  CMNJAR   USER016
fhth30long.java JVS ACTIVE   20150309 200446  CMNJAVA  USER016
hfhth30long.jth JTH ACTIVE   20150309 192827           USER016
                JTH ACTIVE   20150309 192103           USER016

*********************** Bottom of data*****************************
```

Rules for scrolling in long panel fields:

- The **SCROLL** amount at the upper right of the panel determines the scroll amount for long panel fields.
- You might have to scroll right more than once to see the end of a long field.
- When you have scrolled to right end of a long field, a **-** (minus) replaces the **+** (plus) to indicate that you can only scroll left from that point
- Both **-+** (minus plus) are displayed when you can scroll both right and left from your current position in a long field.
- If you type **M** (MAXIMUM) on the **Command** line before you position your cursor in a long field, when you press **PF11** or **PF10** the field scrolls all the way to the end or the beginning respectively.

Long Field Zoom - EXPAND

Rather than pressing **PF11** or **PF10** multiple times to scroll through a long field, you can zoom in on the field by placing your cursor over the field and pressing **PF4** to execute the ISPF **EXPAND** command. EXPAND displays the entire field in a pop-up panel.

If you place the cursor on the **NAME** field of the fourth component listed on the **STAGE: package COMPONENTS** (CMNSTG01) panel above and you press **PF4**, the pop-up (ISPEXPND) panel is displayed.

```
+----------------------------- CMPNAME+0 -----------------------------+
| ISPEXPND                                        Line 1 of 4          |
| Command ===> _______________________________    Scroll ===> CSR      |
| org/jdom/xpath/jhfhth30long.jth                                      |
|                                                                      |
|                                                                      |
|                                                                      |
|                                                                      |
+----------------------------------------------------------------------+
```

Zoom panel rules:

- The pop-up panel has the same attributes as the original panel field. If the original panel field is display-only, then the pop-up panel is also display-only. If the original panel field is available for input, you can type on the pop-up panel.
- If the original panel field is available for input, you can type up to 256 characters on multiple pop-up panel lines for a component name, or 1024 characters on multiple lines for a path name.
- Press **PF3** to exit the pop-up long name panel and return to the original panel.

> **Note**
>
> Most ZMF panels in the ISPF interface follow this rule: "Press ENTER to Process; Enter END or CANCEL command to exit." However, the pop-up panel for expanded long name fields requires END or PF3 to save data entered or changed on the panel.

Clearing Long Names From Panel Input Fields

If a long name extends beyond the visible end of an input field on a ChangeMan ZMF panel, you must take care to:

- Clear the entire long name before pressing ENTER to display a selection list.
- Clear any parts of the old name that extend beyond a shorter name that you type over the original name.

If you only clear the visible part of a long name input field, the rest of the name remains in the ISPF variable, and it will interfere with the processing of subsequence input until you exit the panel. (This is an ISPF behavior, not a ZMF shortcoming.)

> **Tips**
>
> The quickest way to ensure that you have cleared a long name input field is to press PF4 to display the entire long name in a pop-up panel, erase all lines that contain parts of the long name, and press PF3 to return to the original panel.

Alternate Panel - LONG and XLONG

In many ChangeMan ZMF functions that display a list of components, you can invoke an alternate panel that displays the component name on a separate line so there is room to show more characters of a long component name.

For example, if you specify an zFS library type in a package, then Stage Package Components (CMNSTG01) is displayed.

```
CMNSTG01         STAGE: ACTP000057 Components       Row 1 to 3 of 3
Command ===>______________________________________ Scroll ===> CSR
Name                + Type Status Changed         Procname User    Request
__averylongnamegoesh  JVS  ACTIVE 20150315 161850 CMNJAVA  USER016
__hw001.java          JVS  ACTIVE 20150315 152357 CMNJAVA  USER016
__org/jdom/xpath/jhf  JVS  ACTIVE 20150315 165512 CMNJAVA  USER016

*********************** Bottom of data ********************************
```

Panel CMNSTG01 shows only 18 characters of each component name. The panel permits scrolling left and right via PF10 and PF 11, and Expand (PF4) in the Name field so you can see the rest of the component name.

However, if you type **LONG** on the **Command** line and press **ENTER**, the **STAGE: package Components** (CMNSTG14) panel is displayed, which shows each component zFS path, up to 44 characters, on a line below the rest of the component information.

```
CMNSTG14     STAGE: ACTP000057 Components      Row 1 to 3 of 3
Command ===>_________________________________ Scroll ===> CSR
Name               + Type Status Changed        Procname User    Request
                     Org  Input dataset name              + Target lib
__averylongnamegoesh JVS ACTIVE 20150315 161850 CMNJAVA USER016
                     JFS /cmntp/s6/ACTP/#000057/d/JVS            JVL
__hw001.java         JVS ACTIVE 20150315 152357 CMNJAVA USER016
                     JFS /cmntp/s6/ACTP/#000057/d/JVS            JVL
__org/jdom/xpath/jhf JVS ACTIVE 20150315 165512 CMNJAVA USER016
                     JFS /cmntp/s6/ACTP/#000057/d/JVS            JVL
*************************** Bottom of data****************************
```

This panel shows up to 18 characters of a long component name. This panel also offers scrolling and zoom in the Name field if you still cannot see the entire component name.

To return to the original component list panel, type **SHORT** on the **Command** line and press **ENTER**.

Prior to the release of ChangeMan ZMF 7.1 with long names, some panels already responded to the **LONG** command by displaying a panel with additional information. In some of these cases, you can type **XLONG** in the **Command** line and press **ENTER** to display a special panel for long names.

If you type **XLONG** on the **Command** line and press **ENTER**, panel CMNSTG24 is displayed. This panel shows each component name on a line above the component information, with the staging directory path on a third line.

```
CMNSTG24          STAGE: ACTP000057 Components       Row 1 to 3 of 3
Command ===>_______________________________________ Scroll ===> CSR
Name           + Type Status    Changed        Procname User    Request
                 Org Input dataset name                   + Target lib
averylongnamegoeshereforjava.java
                JVS  ACTIVE   20150315 161850 CMNJAVA USER016
                JFS /cmntp/s6/ACTP/#000057/d/JVS                 JVL
hw001.java
                JVS  ACTIVE   20150315 152357 CMNJAVA USER016
                JFS /cmntp/s6/ACTP/#000057/d/JVS                 JVL
org/jdom/xpath/jhfhth40long.java
                JVS  ACTIVE   20150315 165512 CMNJAVA USER016
                JFS /cmntp/s6/ACTP/#000057/d/JVS                 JVL
*********************** Bottom of data ********************************
```

This panel shows up to 75 characters of a long component name. This panel also offers scrolling and zoom in the **Name** field if you still cannot see the entire component name. You can scroll and zoom on the staging directory path.

On this panel you can use the **LONG** command to display the CMNSTG14 panel or **SHORT** to display the CMNSTG01 panel.

You can use the DCD command on some panels to display the first line of the component general description in addition to the fields displayed on the SHORT, LONG and XLONG panels. This is available on package component lists in the Checkout, Stage and Query functions and on baseline/ promotion component lists in the Browse Baseline and Checkout functions.

Tip

If you want to see the component name on a separate line, try **XLONG** first. If short message INVALID SELECTION CODE is displayed, try command **LONG**.

Component Names That Contain A Path

In USS file systems, a "component name" may include a partial directory path because the actual name of the file is unique only within a hierarchy that includes it.

ChangeMan ZMF automatically handles component names that include a directory path. However, when you stage an zFS component from development, you must indicate whether you want to choose a file name from the specified directory or a path name and file name from that directory.

On the **Stage: From Development** (CMNSTG02) panel, the Expand zFS subdirectories field controls what is displayed on a component selection list. When you select this field, all files and paths below the subdirectory you specify in the **DSN** field are displayed on the component selection panel.

```
CMNSTG02           Stage from Development
Command ===>_________________________________
        Package: ACTP000050     Status: DEV     Install date: 20180318
ISPF Library:
Project . . . . USER015
Group . . . . . JAVA
Type . . . . .  SRC
Member . . . .  ___(Blank/pattern for list; * for all members)

Other partitioned, sequential or zFS dataset:
DSN . . . . . . /cmntp/s4/v710/base/jzfs/jav/lvl-0/ +
Org . . . . . .___ (PDS, Seq, PAN, LIB, Oth, zFS)

Library type . . . . . JVS (Blank for list)
Stage name . . . . . . __________________________+
Stage mode . . . . . . 1 (1-Online, 2-Batch)

Enter "/" to select option
/ Confirm request        __Expand zFS subdirectories
/ Lock component         __Display component user options
__Extract Stored Procedure from Db2 catalog
```

Press ENTER to display the **Stage from zFS file** (CMNSTG23) panel with the component selection list.

```
CMNSTG23              Stage from zFS file            Row 1 to 6 of 6
Command ===>_______________________________________ Scroll ===> CSR

Input filename
/cmntp/s4/v710/base//jzfs/jav/lvl-0/                              +
Name    +          Function Created     Changed            Size  User
__ jhfjav40.java            2010/07/20  2010/11/05 15:30  00126  SERT
__ jhfjav50.java            2010/11/05  2010/11/05 15:49  00126  SERT
__ jhfjav60.java            2010/11/05  2010/11/05 16:23  00126  SERT
__ org/jdom/xpath/jhfj      2010/07/20  2010/07/01 17:53  00151  SERT
__ org/jdom/xpath/jhfj      2010/07/20  2010/07/01 17:54  00155  SERT
__ org/jdom/xpath/jhfj      2010/07/20  2010/07/01 17:54  00154  SERT
```

Notice that after the first three files, three more components are listed that are in a path of subdirectories below the directory you specified on the **Stage: From Development** panel. If you select one of the files with path names to stage into your package, the component name in the package master and in the component master will include the subdirectories as well as the file name.

If option **Expand** is not selected the resulting **Stage from component file** (CMNSTG23) panel displays only the three components that are contained as files in the directory you specified on the **Stage: From Development** panel.

```
CMNSTG23                    Stage from zFS file        Row 1 to 3 of 3
Command ===>_________________________________________ Scroll ===> CSR
Input filename
/cmntp/s4/v710/base//jzfs/jav/lvl-0/                                +

Name          + Function Created    Changed    Size        User
__ jhfjav40.java         2010/07/20 2010/11/05 15:30 00126 SERT
__ jhfjav50.java         2010/11/05 2010/11/05 15:49 00126 SERT
__ jhfjav60.java         2010/11/05 2010/11/05 16:23 00126 SERT

******************************* Bottom of data *************************
```

Note

On the Stage: From Development panel (CMNSTG02), you cannot provide a STAGE NAME when the EXPAND field is set to YES.

Case Sensitive Fields

By default, all ISPF panel input fields are folded to upper case, regardless of the case you type. However, zFS path names and file names are case sensitive. For example, these are three different files:

- FirstJavaComponent.java
- firstjavacomponent.java
- FIRSTJAVACOMPONENT.java

ChangeMan ZMF uses two methods to control the case sensitivity of input fields on ISPF panels.

- **Data Set Type** in library type definitions
- **Mixed Case** parameter on component list

Data Set Type

The global and application **Library Types Part 2 of 2** panel includes the **Data Set Type** field, with valid values of **LIBRARY** for PDSE, **PDS** or blank for PDS or **zFS**. When you type a component name or directory path for a data set type zFS, case is preserved and stored in ZMF repositories. When component name or directory path are displayed for data set type zFS, the case that is stored in ZMF repositories is displayed unchanged on ISPF panels.

For example, when you type information on the **application/site - Promotion Libraries** (CMNLRPM3) panel in application administration, ChangeMan ZMF uses the library type to determine whether the data should default to upper case or be processed exactly as you enter it.

In this example, the library names and directory paths for promotion are all entered in lower case. Panel truncated to only show zFS libraries..

```
CMNLRPM3             ACTP/SERT6 - Promotion Libraries            Row 2 to 8 of 8
Command ===>___________________________________________________ Scroll ===> CSR

Promotion name: S6P1UT          Level: 10

        Syslib
     Lib exclude  Target libraries
_____JCL   Y      CMNTP.S6.V810.PROM.S6P1UT.JCL                     +  Shadow
                  CMNTP.S6.V810.PROM.S6P1UT.JCL                     + Library 1
                  ______________________________________________    + Library 2
                  ______________________________________________    + Library 3
_____JVS N        /cmntp/s6/actp/prom10/jvs                         + Shadow
                  /cmntp/s6/actp/prom10/jvs                         + Library 1
                  ______________________________________________    + Library 2
                  ______________________________________________    + Library 3
...
```

When you press **ENTER**, the data set names for library type JCL are changed to upper case and stored that way on the package master. However, the zFS path names for library type JVS (Java) are left exactly as you entered them, and they are stored in mixed case on the package master.

Mixed Case

On list parameter panels where you specify filter criteria for building a component list, you can control how case is used for input by selecting the Mixed Case field on the bottom of **Component List Parameters** (CMNSTG12) panel.

By default, all fields are folded to upper case. However, if 'Mixed Case' option is selected then the component name is left as is. Whichever case you enter will then be processed.

For example, package ACTP000050 contains these two components:

```
CMNSTG01                       STAGE: ACTP000050 Components            Row 1 to 2 of 2
Command ===>_________________________________________________________ Scroll ===> CSR

Name                  + Type Status Changed          Procname User    Request
___ Acpdoc60            HTH  ACTIVE 20150315 212025           USER016
___ ACPDOC60            DOC  ACTIVE 20150315 211855           USER016
******************************* Bottom of data *******************************
```

You can filter the components displayed on the **Stage: package Components** panel by first setting selection criteria on the **Component List Parameters** panel (CMNSTG12). If you type the **Component Name** field in lower case on the **Component List Parameters** panel and select **Mixed case** field, then the filter is case sensitive, and only the HTH component is listed.

Selection criteria:

```
CMNSTG12             Component List Parameters
Command ===>________________________________________________

       Package: ACTP000050       Status: DEV            Install date: 20180405
Work request:   Work Req WR2                               Department: Dept

Component name . . . . . . acpdoc60                                   +
Component type . . . . . . SRC
Language . . . . . . . . ._____
Component status . . . .  __Active __Checkout __Frozen __Inactive
                          __Incomp __Unfrozen
Changed from date . . . . . ________ (yyyymmdd)
time . . . . .              _________(hhmmss)
Changed to date . . . . .   _________(yyyymmdd)
time . . . . .              _________(hhmmss)
Compile procedure . . . . . ________
User . . . . . . . . . . .  ________
Display mode . . . . . . . S         (S-short, L-long, X-extra long)
Data Encoding . . . . . . .          (1-ASCII, 2-UTF-8)
Enter "/" to select option
/ Confirm component delete           /  Confirm other requests
/ Display component user options     __ Mixed case
/ Comparison report for edit         Text type . . . . $._________
__Ignore recompiled components
```

Result:

```
CMNSTG01               STAGE: ACTP000050 Components                Row 1 to 1 of 1
Command ===>_____________________________________________________ Scroll ===> CSR

Name            + Type Status          Changed            Procname  User     Request
__acpdoc60        HTH  ACTIVE          20170915 212025              USER016
******************************* Bottom of data ********************************
```

If you type the **Component Name** field in lower case and set the **Mixed Case** field is not selected, then the component name you typed is folded to upper case, and only the DOC component is listed.

Selection criteria:

```
CMNSTG12                     Component List Parameters
Command ===>______________________________________________________

        Package: ACTP000050         Status: DEV         Install date: 20180405
Work request: Work Req WR2                                Department: Dept

Component name . . . . . . acpdoc60                                             +
Component type . . . . . . SRC
Language . . . . . . . . . ___________
Component status . . . .   __Active  __Checkout __Frozen __Inactive
                           __Incomp  __Unfrozen
Changed from date . . . . . ___________     (yyyymmdd)
time . . . . .              ___________     (hhmmss)
Changed to date . . . . . . ___________     (yyyymmdd)
time . . . . .              ___________     (hhmmss)
Compile procedure . . . . . ________
User . . . . . . . . . . .  ________
Display mode . . . . . . . S                (S-short, L-long, X-extra long)
Data Encoding . . . . . . .                 (1-ASCII, 2-UTF-8)
Enter "/" to select option
/  Confirm component delete                    /  Confirm other requests
/  Display component user options              __ Mixed case
/  Comparison report for edit                  Text type . . . . $._______
__ Ignore recompiled components
```

Result:

```
CMNSTG01            STAGE: ACTP000050 Components            Row 1 to 1 of 1
Command ===>______________________________________________ Scroll ===> CSR
Name        + Type Status   Changed         Procname User       Request
__ACPDOC60    DOC  ACTIVE   20170915 211855          USER016
******************************* Bottom of data ********************************
```

Batch Job Statement Information

ChangeMan ZMF panels that execute batch functions contain a section for **Job Statement Information**.

```
JOB STATEMENT INFORMATION:
===> //JOBNAME JOB (ACCOUNT),'CHANGE MAN',            <=== CHANGE ACCORDINGLY_____
===> //            CLASS=?,                           <=== CHANGE ACCORDINGLY_____
===> //            NOTIFY=?,                          <=== CHANGE ACCORDINGLY_____
===> //            MSGCLASS=?                         <=== CHANGE ACCORDINGLY_____
```

The first time you use any panel in ChangeMan ZMF that requires this information, you must type JOB information that is appropriate for your environment.

The four lines of JOB statement information are ISPF panel fields, not JCL records. You change the information by over-typing or erasing it. You cannot insert, repeat, or delete lines, and you cannot insert characters without deleting characters first or erasing to end- of-line. If you do not use all four lines, make unused lines into JCL comments with //* in positions 1-3.

The information you type is stored in variables in your ISPF profile. It is used to initialize the next panel you display that shows a JOB Statement Information section. You can overtype this information to change it whenever you want.

This is an example of completed JOB Statement Information:

```
Job statement information:
//USER001 JOB (0000),'S6 V810',
//            CLASS=A,MSGCLASS=X,NOTIFY=USER001
//*
//*
```

Online and Batch Execution Modes

Several ChangeMan ZMF functions offer you a choice between execution online in ISPF or in a batch job:

- Checkout
- Stage From Development (including Mass Stage)
- Stage from the Stage: package Components panel (the "package list")
- Recompile Source
- Library Scan
- Compare

You choose between **O** (online) or **B** (batch) execution by setting **Mode** field on an ISPF panel, as shown on this **Library Scan Utility** (CMNSCN01) panel:

```
CMNSCN01                    Library Scan Utility
Command ===>_______________________________________________________

Package . . . . . . . . . TEST001852
Application . . . . . . . ____      (Blank for list)
Library type . . . . . .  JCL_      (Blank for list)
Scan mode . . . . . . . . 1_        (1-Online, 2-Batch, 3-Alt Batch)
Records to select . . . . 0_        (0 = all)

Enter "/" to select option
  __ List member names only
  __ Scan for dependencies
  __ Case sensitive

Displacement:
From . . . 0 (0=all)                To . . . . . 0 (max=80)
Components to scan:                 (Blank for all components)
Begin . . . ____________________    End . . . . __________________
Data strings:
cobol                                    OR_ (and/or)
______________________________________
______________________________________
```

In general, batch mode builds JCL and submits a batch job to execute a ChangeMan ZMF process that would otherwise be executed in your ISPF address space. There are two potential advantages to batch mode:

- Your terminal can be freed as soon as the batch job is submitted, allowing you to perform other ZMF tasks while the batch job executes.
- Output is available in JES.

These advantages can be illustrated with a Library Scan example:

- Online Scan - Your terminal is locked while the scan runs in your address space. When the scan is complete, the results are only available on a scrollable ISPF panel.
- Batch Scan - Your terminal is locked only for as long as it takes for ISPF file tailoring to build JCL for a batch scan job and to submit the job. When the batch scan job is complete, the results are available in SDSF or whatever tool you use to view JES job output.

When you checkout a component, stage a component that is not like-source, or execute the compare component function, the online and batch modes operate like the library scan example.

Online and batch mode work somewhat differently when you stage or recompile a like- source component. These build functions always execute in a batch job. The ISPF file tailoring that creates the build job JCL is always performed outside of your address space in a special started task initiated by the stage or recompile process. The difference between online and batch mode is in the initiation of the file tailoring started task:

- Online Stage or Recompile - The file tailoring started task is initiated from your ISPF address space.
- Batch Stage or Recompile - ZMF submits a batch job that initiates the file tailoring started task.

> **Note**
>
> Whether batch mode releases your terminal faster than online mode depends on:
>
> - Which function is being executed
> - How many components are being processed
> - How your z/OS resources are allocated Sometimes it may take longer to build JCL and submit a batch job than it takes to execute the function in your ISPF address space.

Editing Components in ChangeMan ZMF

As delivered by Serena, ChangeMan ZMF uses IBM's ISPF line editor when you edit a change package component. If you have questions about using the ISPF editor, press **PF1** for a tutorial. The tutorial includes information about the following subjects:

- General introduction
- Edit entry panel
- Display screen format
- Scrolling data
- Sequence numbering
- Display modes (CAPS/HEX/NULLS)
- Tabbing (hardware/software/logical)
- Edit profiles
- Edit line commands
- Edit primary commands
- Labels and line ranges
- Ending an edit session

Ending an Edit Session

When you edit a package component in ChangeMan ZMF, you actually work with a temporary ChangeMan ZMF utility data set that has been populated from the staging library member that you want to change.

When you exit from your edit session, ChangeMan ZMF takes you through several steps to make sure you want to commit the changes you made and replace the staging library member with your edited text.

1. The contents of your temporary edit file are compared to the contents of the original member in the staging library and a line-by-line comparison is displayed. If you cancel your changes after viewing the compare listing, the member in the staging library is not replaced. Alternatively, you can choose to keep your editing changes and go on to the next step.
2. If you decide to keep your edit changes, and if the component you edited is not included in any other active packages, the member in the staging library is replaced with your edited content.
3. If you decide to keep your editing changes, and if the component you edited is included in other active packages, ChangeMan ZMF shows you a list of those packages. The list includes the TSO ID of the last person who acted on the component in each package. You can then contact those developers to ensure that your work is coordinated with theirs. If you are satisfied that cross-package conflicts do not exist, you can choose to keep your editing changes. The member in the staging library is then replaced.

   Alternatively, you can cancel your changes. In the latter case, the staging library member is not replaced.

Automatic Edit Recovery

Because you do not edit a component directly in the staging library, the ISPF edit SAVE command is blocked in ChangeMan ZMF edit sessions. This prevents confusion about whether you are saving your changes to the staging library member.

If your ISPF session is interrupted by a system problem or by cancellation of your TSO session, and if you have RECOVERY ON in your ISPF Profile, ISPF sets a recovery pending indicator.

> **Note**
>
> If your Change Man Administrator sets EDIT STAGING RECOVERY MODE ON to Y in application administration, an initial edit macro sets RECOVERY ON in your ISPF profile when you start each edit session.

When you connect to ChangeMan ZMF, it looks for an ISPF recovery pending condition involving a ChangeMan ZMF temporary edit data set. If ChangeMan ZMF detects such a condition, the **Primary Option Menu** is skipped and the **Edit Recovery** panel is displayed.

```
CMNEDRC1 ------------------------- EDIT - RECOVERY ------------------------
COMMAND ===>______________________________________________________________
                ****************************************
                * EDIT AUTOMATIC RECOVERY              *
                ****************************************

The following dataset was being edited when a system failure or
task abend occurred:

DATASET: CMNTP.SERT6.BASE.ACTP.CPY(ACPCPY00)

INSTRUCTIONS:

Press ENTER key to continue editing dataset, or
Enter END or CANCEL command to cancel recovery of the dataset.
```

The recommended procedure is:

1. Press **Enter** to resume the edit session.
2. Immediately press **PF3** to end the edit session, which saves the edited content into the staging library.
3. The **Primary Option Menu** is then displayed.

   ChangeMan ZMF does not have the same information about a recovered edit as it does for an edit session you initiated from within ChangeMan ZMF. When you end a recovered edit session, ChangeMan ZMF does not display the compare listing or a concurrent development warning.

Manual Edit Recovery

If your ISPF session is not terminated, but the started task running ChangeMan ZMF is stopped or your ChangeMan ZMF session times out, the changes you made in a ChangeMan ZMF edit session are not saved to the staging library. Since ISPF did not end, there is no pending recovery indicator for ChangeMan ZMF to find when you connect again.

However, the utility data set containing your edits is not scratched, as it would be after a normal end to an edit session. You can use that utility data set to recover your editing changes. Follow this procedure:

1. Ask your ChangeMan ZMF Administrator what naming convention is used for edit session utility data sets. The default convention (starting with ChangeMan ZMF 5.5) is:

   `\&ZUSER.&ASID.\#ttttttt.\#ttttttt.xxx`

   where:

   - `\&ZUSER` is the TSO userid.
   - `\&ASID` is the user address space ID.

- `ttttttt...ttttttt` is the binary time-of-day to nearest microsecond.
- `xxx` is the ChangeMan ZMF library type.

2. Use the ISPF Data Set List Utility (=3.4) to list all of your ChangeMan ZMF temporary data sets in the catalog.
3. Edit the last temporary edit data set on the list, and verify that it contains the component you were editing when your ChangeMan ZMF session was terminated.
4. Cut all lines in the temporary edit data set.
5. Connect to ChangeMan ZMF, and edit the component you were working on when your session was terminated.
6. Delete all of the lines in the component, then paste the lines cut from the temporary edit data set.
7. Press **PF3** to end the edit session, and carefully examine the compare listing to make sure you have recovered your changes correctly.
8. Continue the normal procedure to end a ChangeMan ZMF edit session until the edited component is saved to the staging library.

Exiting ChangeMan ZMF

To exit ChangeMan ZMF, do one of the following:

- At the **Primary Option Menu**, select option **X Exit**.
- At the **Primary OptIon Menu**, type **End** on the **Option** line and press **Enter**.
- At the **Primary Option Menu**, press **PF3**.
- On almost any panel in ChangeMan ZMF, type **=X** on the **Command** or **Option** line and press **Enter**.

4. Creating a Change Package

Create Package is the first step in the change package life cycle.

About Creating a Change Package

Using a series of ISPF panels, you enter information that describes the change package, and you set control parameters that determine how the package will behave during the rest of the package life cycle.

Change Man ZMF automatically copies additional control parameters from Global and Application Administration records on the package master to make your package follow the rules set up by your administrators at the ChangeMan ZMF instance level and at the application level.

When you have entered valid information in all required fields on all required Create Package panels, ChangeMan ZMF assigns a 10-character change package ID and writes package records to the package master file. The package ID consists of a four character application mnemonic (three characters and a blank if the application mnemonic has only three characters) and a six character package number, which is incremented by one for each new package in an application.

When your package has been successfully created, no application components are associated with your change package. Although your Application Administrator can set parameters that allocate staging libraries for some library types when your package is created, staging libraries for your package are usually not allocated until you check out components into your change package. However, even if staging libraries are allocated at create package, they are empty.

Rules for Creating Packages

These rules and restrictions apply to the create package function.

- To create a new change package in an application, you must have update access to the application, which is defined under the ChangeMan ZMF resource class in your security system.
- A new change package is not created and assigned a package ID until you have entered valid information in all required fields in all required Create Package panels,

Administration Settings for the Create Package Function

Your Global and Application Administrators make settings in ChangeMan ZMF administration that control how the create package function works in your application. Ask your administrator if any of the following apply to your application.

- Restrict entries in the package Scheduler field on the Create: Installation Instructions panel. (Global Administration Parameters: Install Job Scheduler and Default Job Scheduler)
- Restrict the number of packages that can be scheduled for installation on a particular Install Date through the use of the Installation Calendar. (Global Administration Parameters: Disable Installation Calendar)
- Prohibit creation of temporary change packages. (Application Administration Parameters: Allow Temporary Packages)
- Require data in the Work Request field on the Create: Create a New Package panel. (Application Administration Parameters: Require Work Request Number)
- Require data in the Department field on the Create: Create a New Package panel. (Application Administration Parameters: Require Department Number)
- Enable the Package User Information facility to display one or two panels that you use to set package-level variables defined by your administrator. (Global Administration Parameters: Display Package User Option Panel)
- Set Normal Business Hours in Global Administration that determine whether an unplanned package is assigned the Planned Approval List or the Unplanned Approval list when the package is created. (Application Administration Parameters: Normal Business Hours)

> **Note**
>
> If you create an unplanned change package outside of Normal Business Hours, the Unplanned Approval List is assigned. Normal Business Hours are compared to the time you create your package, not the time that your package is frozen or the time that it is scheduled for install. Your administrator may set Normal Business Hours to times that are not "normal" to force all unplanned packages to use the Unplanned Approval List.

Set the difference in system clock time between the development site where your change package is created and a remote site where it is scheduled for install. The time difference can allow you to enter an install time for a remote site that has already passed at your site. (Global Administration Sites: Time Difference)

Exit Programs for the Create Package Function

Your ChangeMan ZMF installer or administrators may implement exit programs to alter the behavior ChangeMan ZMF. Ask your administrator if any of the following apply to your application.

- Restrict Install date by one of more of the following: Application, Global or Application Administration authority, specified date, specified install time, today's date, day of week, package type, values in Package User Information variables. (Exit program CMNEX002)
- Impose a lead time between today's date and the package install date. Other criteria available includes Application, Global or Application Administrator authority, specified date, day of week, package type, values in Package User Information variables. (Exit program CMNEX003)
- Restrict creation of specified package types by one or more of the following: Application, Global or Application Administrator authority. (Exit program CMNEX006)
- Restrict install date by one of more of the following: Application, administrator authority, specified date, day of week, package type, today's date. (Exit program CMNEX007)
- Restrict install day of week by one of more of the following: Application, global or application administrator authority, day of week, package type. (Exit program CMNEX012)
- Validate information entered in Work Request Number and/or Department against a specified a list of values. Cross-edit Work Request Number, Department, and Package User Information when package user information is entered. (Exit program CMNEX014)
- Validate group site names used on the Site Information panel, and specify the sites associated with each group site name. (Exit program CMNEX039)
- Add custom processes, executed outside of ChangeMan ZMF, at the end of the create package process. (Exit program CMNEX043)

Accessing Create Package Panels

Use one of these methods to display the **Create: Create a New Package** panel, which is the first panel in the Create Package function.

- Using the Menu Hierarchy:

  a On the **Primary Option Menu**, select **1 Build**.

  b On the **Build Option** panel, select **1 Create**.

- Using Direct Panel Access:

  On any ChangeMan ZMF panel, enter **=1.1** in the **Command** or **Option** line and press **Enter.**

- **Using the Change Package List:**

  On the **Change Package List** panel, type **CREATE** or **CR** on the Command line and press **Enter**.

Package Description Methods

When you execute the Create Package function, many of the ISPF panel fields in the process are displayed with values stored in your ISPF profile from the last time you created a package. You can use the displayed information for your new package, or you can overtype that information with new values.

You have three choices for entering the rest of the descriptive and control information required to create your new package: the long method, the short method, and copy package forward option. You choose which method to use on the first Create Package panel.

Long Method

When you choose the Long Method, all panels in the Create Package process are displayed. You are required to type information in free form text description fields.

Short Method

When you choose the Short Method, two panels (CMNCRT02, CMNCRT03) in the Create Package process are skipped, and the fields on those panels are set to default values. This table shows the default field values for the two skipped panels:

Skipped Panel	Panel Field	Short Method Default Value
CMNCRT02 Create: Package Description	Free form description text	“NONE”

Skipped Panel	Panel Field	Short Method Default Value
CMNCRT03 Create: Implementation Instructions	Contingency	The default will be option 1 if you have never created a package using the long method, otherwise it will use the last option selected by the user.
		1 -Hold production and contact analyst
		2 -Backout change, continue production
		3 -Other:
	Job scheduler	Value in Default job scheduler fields from Application Parameters
	Free form instruction text	“NONE”

You can use the Update Package Information function to go back later and replace the default information with your own values and text.

No Package Description

When you choose option D (No package description), panel CMNCRT02 in the Create Package process is not displayed. Package description will be ’None’. All other panels are displayed.

No Implementation Instructions

When you choose option I (No implementation instructions), panel CMNCRT03 in the Create Package process is not displayed. Package implementation instructions will be ’None’. All other panels are displayed.

Copy Package Forward

You can use an existing package as a model to create a new package. When you type a package ID in the Package to Copy Forward field on the first Create Package panel, the fields displayed on all *subsequent* panels are copied from the package you specified. You can overtype this information with new information.

The list of package components from the original package is copied into your new package. The status of these components is INACTIVE because the components on this list have not been copied to staging libraries for the new package. Later, you can delete

component names from this list or use the list of component names to check components out from baseline.

Note

Utility requests (scratch and rename components) are not copied forward into the new package.

Dynamic Panels in Create Package

The panels in the Create Package process are displayed dynamically, and what you see depends on Global Administration settings, Application Administration settings, exit programs enabled by your Administrators, and on information you enter as you proceed through the Create Package process.

This table shows when panels in the Create Package process are displayed.

Panel Title	Panel ID	When Displayed
Create: Create A New Package	CMNCRT0R	All new change packages.
Create: Package Description	CMNCRT02	All packages when the Long create package method is used.
Create: Installation Instructions	CMNCRT03	All simple and participating packages when the Long create package method is used. Enter up to a maximum of 46 lines of free- form instructions for the installation of the change package. Caution: Text beyond 46 lines will not be saved.
Create: Scheduling Dependencies	CMNCRT04	Simple and participating package when you specify Other in the Scheduler field on the Create: Installation Instructions panel
Create: Affected Applications	CMNCRT05	All Participating packages except short.
Create: Complex/ Super Information	CMNCRT08	Super or Complex packages.
Create: Package User Information	CMNDPUP1	Simple and participating package if your administrator enabled the Package User Information Facility. This custom panel, and a second panel CMNDPUP2, may have a different panel title coded by your administrator.
Create: On Site Information	CMNCRT06	Simple and participating packages in an All environment.
Create: Site Information	CMNCRT07	Simple and participating packages in a Development or Development/ Production environment.

Defining Package Control Information

The first panel displayed in the Create Package process is the **Create: Create A New Package** panel (CMNCRT0R).

```
CMNCRT0R        Create: Create a New Package
Option ===>____________________________________
        L Long method                S Short method
        D No package description     I No implementation instructions

Package title
  Demo package-----------------------------------------------
Application . . . . . . . . . DEMO_                (Blank or pattern for list)
Requester's name . . . . . .  John Doe_______________________
Requester's phone . . . . . . (555)_555-5555_
Work request . . . . . . . . .____________
Department . . . . . . . . .  IDD___
Package level . . . . . . . . 1_                   (1. Simple 2. Complex
                                                    3. Super 4. Participating)
Package type . . . . . . . .  PLANNED_             (Planned or Unplanned)
Package time span . . . . . . PERM_                (Permanent or Temporary)
Package to copy forward . . . __________           (Optional package name)
Unplanned reason code . . . . _________            (* for list)
Temporary change duration . . ____                 (In days)
Notify user . . . . . . . . . JDOE_

Enter "/" to select option
__Attach package to release
```

This table describes the fields and options available on the **Create: Create A New Package** panel.

Field	Description
Option	Choose a method for creating your package. See Package Description Methods
Long method	L: Show all panels in the Create Package process, require input for text descriptions. All package creation panels will be displayed for required user input, including Package Description and Implementation Instructions.
Short method	S: Create a package using the SHORT method: only the first and last panels will be displayed.
No package description	D: Use Default Package Description Omit the display of panel CMNCRT02 (Package Description) but still display the other panels. Package description will be 'NONE'.
No implementation instructions	I: Use Default Implementation Instructions. Omit the display of panel CMNCRT03 (Implementation Instructions) but still display the other panels. Implementation Instructions will be 'NONE'
Package title	Enter free-form text to identify this package.
Application	Enter a valid 3 or 4 character application mnemonic, as established by your ChangeMan ZMF Administrator. ChangeMan ZMF will assign a sequential number as part of the Package ID. You must have UPDATE access to this mnemonic in your security system.

Field	Description
Requester’s name	Enter your name or the name of the person requesting this change package. This field is not case sensitive and has a maximum length of 25 characters.
Requester’s phone	Enter a telephone number (extension) for the person whose name you entered in Requestor Name. This field has a maximum length of 15 characters.
Work request	This field may be required by your Administrator. Also, depending upon the INFO MANAGEMENT RULE currently in effect, this number may be tied to an INFO Change Record number. This field has a maximum length of 12 characters and is not case sensitive.
Department	Enter the department for this change package if required. This field is used for reporting only. This field is not case sensitive and has a maximum length of 4 characters.
Package level	Enter a one-character code (1-4) representing the package level. Note: If you 'copy forward' an existing change package to create this one, its 'level' will be copied also.
	1. Simple - The change package contains a change that is unrelated to any other change package. This level of change package does not affect any other application, nor does it require changes to software or operational procedures in other applications.
	2. Complex - The parent for two or more participating change packages that have interdependent changes to software or operational procedures. To create a super or complex package, you provide only this first panel of control and general information, a package description, and a list of participating packages. Remote sites and installation dates are set in each participating package. No staging libraries are allocated to super or complex packages. There are no processing difference between super and complex change packages.
	3. Super - The "parent" for change packages that contain major changes to several applications. To create a super or complex package, you provide only this first panel of control and general information, a package description, and a list of participating packages. Remote sites and installation dates are set in each participating package. No staging libraries are allocated to super or complex packages. There are no processing difference between super and complex change packages.
	4. Participating - The change package is related to one or more other participating change packages which are listed under the same super or complex package, otherwise it is much like a Simple package. SYSLIB statements in build process jobs include staging libraries from other participating packages under the same super or complex package.
Package type	Choose a package type. P: **PLANNED -** The package contains PLANNED change. These are change packages that have been previously scheduled. (Abbreviation: P) U: **UNPLANNED -** The package contains UNPLANNED changes. These packages are commonly referred to as “emergencies” or “emergency fixes” and are unscheduled. Depending on Global and Application Administration settings, some package lifecycle steps and requirements may be skipped. (Abbreviation: U)

Field	Description
Package time span	Choose a package time span. P: **PERMANENT** - Package components are installed in baseline and production libraries. A permanent package may be either planned or unplanned. Super, complex, and participating packages must be permanent. (Abbreviation: P) T: **TEMPORARY** - Changes are installed in special override libraries concatenated on top of your production environment library concatenations. These changes may be special one-time or short-term processing and are removed from these libraries after the Temporary Change Duration has passed. Baseline and production libraries are not changed. A temporary package may be either planned or unplanned. You must provide a Temporary Change Duration on this same panel. (Abbreviation: T)
Package to copy forward	Enter Package ID that you want to use as a model for the package you are creating. Package control and descriptive information from the model is displayed on create package panels for your new package, and a list of the components in the model are copied into your package. See Package Description Methods
Unplanned reason code	Enter the 3-digit change reason code that best represents why you are installing an unplanned change package. These codes were established by your ChangeMan Administrator. Enter '*' to display a reason code selection list.
Temporary change duration	The number of days that changes in temporary packages are to remain in override libraries. Valid days are 1-999. The count of calendar days for Temporary Change Duration is incremented at 23:59 system time each night. Temporary package components are automatically removed at the time-of-day specified in the Install Date/Time. Therefore, the number entered in Temporary Change Duration does not necessarily equate to 24- hour days. For example, a temporary package has an Install Date/Time of 20041015 / 1800, a Temporary Change Duration of 1, and the package is actually installed at 2100 on the install date 10/15/ 2004. Package components are automatically removed from temporary libraries at 1800 on 10/16/2001, which is less than 24 hours after they were installed.
Notify user	Enter the TSO userid of the user to be notified of major package changes, such as when the package is frozen. If not specified, this defaults to the package creator.
Attach package to release	Enter (/) to attach this change package to a release. When selected another panel will be displayed where you will select the release that this package will be attached to.

When you finish typing information on the **Create: Create A New Package** panel, press

Enter. If no error message are displayed, the next panel in the create package process is

displayed. Your new change package is not created until all new package information has been entered and validated.

Providing a Package Description

The **Create: Package Description** panel (CMNCRT02) is displayed for all new packages when you use the Long create package method.

```
CMNCRT02    Create: Package Description     Row 1 to 12 of 12
Command ===>________________________________ Scroll ===> PAGE

Description:
___ _________________________________________________________
___ _________________________________________________________
___ _________________________________________________________
___ _________________________________________________________
___ _________________________________________________________
___ _________________________________________________________
___ _________________________________________________________
___ _________________________________________________________
___ _________________________________________________________
___ _________________________________________________________
___ _________________________________________________________
___ _________________________________________________________

******************************* Bottom of data **************
```

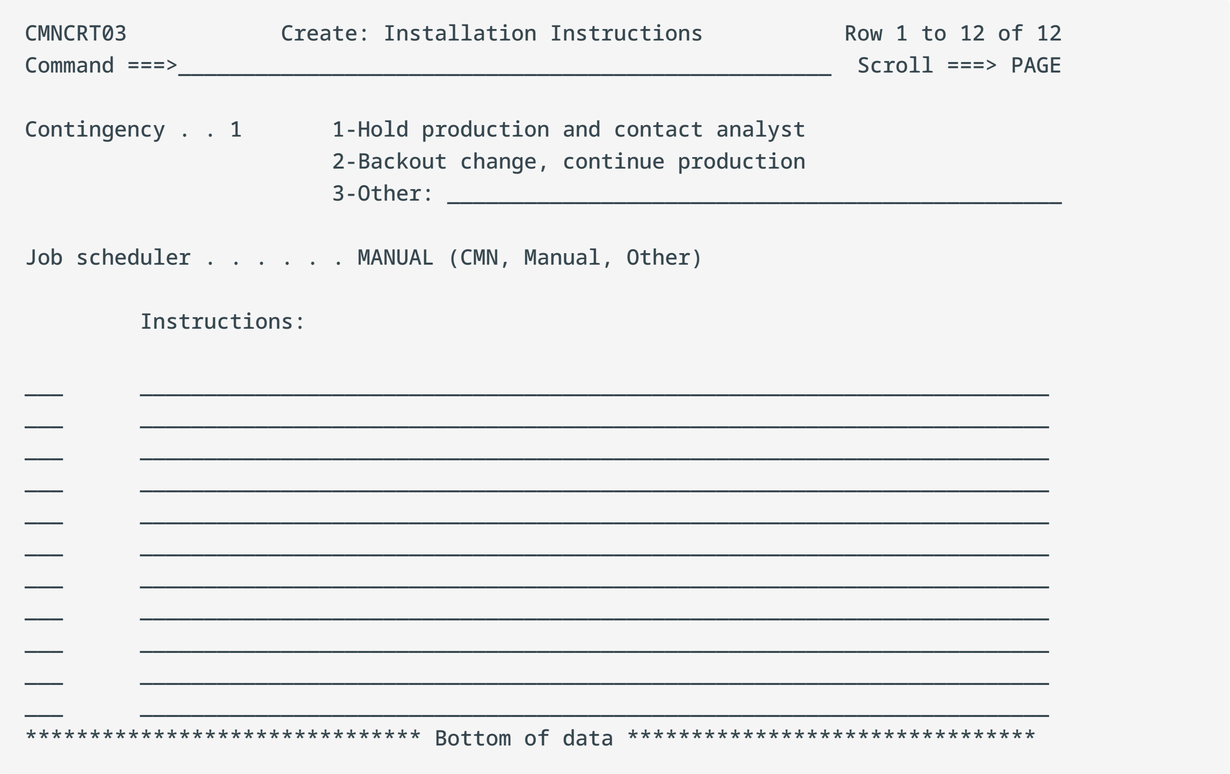
Use this panel to provide details about your package in addition to the information in the Package Title. Your change control procedures and standards may specify what kind of package description you are required to provide.

This table describes the line commands you can use on the **Create: Package Description | Field | Description | |:----|:---| | Line Command | I: Insert a new line.
R: Repeat an existing line.
D: Delete an existing line. | | Description|Enter up to a maximum of 46 lines of free-form text to describe the package.
CAUTION: Text beyond 46 lines will not be saved. |

When you finish typing information on the **Create: Package Description** panel, press **Enter**, and the next panel in the create package process is displayed. Your new change package is not created until all new package information has been entered and validated.

Providing Installation Instructions

The **Create: Installation Instructions** panel (CMNCRT03) is displayed for new simple and participating packages (Package Type 1 and 4) when you use the Long create package method.



```
CMNCRT03            Create: Installation Instructions          Row 1 to 12 of 12
Command ===>___________________________________________________  Scroll ===> PAGE

Contingency . . 1       1-Hold production and contact analyst
                        2-Backout change, continue production
                        3-Other: ________________________________________________

Job scheduler . . . . . . MANUAL (CMN, Manual, Other)

         Instructions:

___      _______________________________________________________________________
___      _______________________________________________________________________
___      _______________________________________________________________________
___      _______________________________________________________________________
___      _______________________________________________________________________
___      _______________________________________________________________________
___      _______________________________________________________________________
___      _______________________________________________________________________
___      _______________________________________________________________________
___      _______________________________________________________________________
___      _______________________________________________________________________
******************************* Bottom of data ********************************
```

Use this panel to provide instructions for installing your change package. Your data center standards and change control procedures may specify what procedures to follow for ChangeMan ZMF change package installs and what information you should provide on this panel.

This table describes the fields and options available on the **Create: Installation Instructions** panel.

Field	Description
Contingency	Select the option that describes what should be done if the installation of your change package fails. This field is required, but it has no effect on the lifecycle of your package or its installation process. 1. Select option 1 to discontinue running your application's production jobs until a supporting analyst can be reached for instructions. 2. Select option 2 if the package should be backed out and the production jobs for your application should continue to be run as scheduled. 3. Select option 3 to describe alternate action to be taken if there are problems installing the package. In the space provided, you will be able to enter a description (up to 44 characters) following your selection.

Field	Description
Scheduler	This field determines how the package installation process is initiated. Enter the globally-allowed scheduling system for ChangeMan to use for implementation into production. **CMN:** Select CMN to have ChangeMan ZMF schedule the submission of the package installation job. Package installation is initiated by the internal ChangeMan ZMF scheduler when it determines that the package Install Date and From Time have arrived at the specified Site. **Manual:** Select Manual to have package installation initiated when the package is fully approved. **Other:** Select Other to have ChangeMan ZMF perform a batch interface to add the package install job information to the external scheduler's database. Package installation will be initiated by an external scheduler like CA-7®, CA-Scheduler®, or CA-ADC2™. Scheduling records are inserted into the scheduler database by job CMN17, which runs when the package is distributed.
Instructions (installation)	Enter up to 46 lines of instructions for the installation and backout of your change package. This is a required field. CAUTION: Text beyond 46 lines will not be saved. Line commands: I- Insert new lines R- Repeat an existing line. D- Delete an existing line.

When you finish typing information on the **Create: Installation Instructions** panel, press **Enter**, and the next panel in the create package process is displayed. Your new change package is not created until all new package information has been entered and validated.

Setting Job Scheduling Dependencies

The **Create: Scheduling Dependencies** panel (CMNCRT04) is displayed if you specify **Other** in the **Scheduler** field on the **Create: Installation Instructions** panel.

```
CMNCRT04        Create: Scheduling Dependencies             Row 1 to 12 of 12
Command ===>                                                 Scroll ===> PAGE

        Successor           Predecessor
___     ACPD106             ACPD105
___     ACPW345             ACPW335
___     _______             _______
___     _______             _______
___     _______             _______
___     _______             _______

******************************* Bottom of data ********************************
```

The information on this panel is inserted into your job scheduler database by job CMN17, which runs when your package is distributed. The jobs you listed in the **Successor** and **Predecessor** fields should be related to the first installation job for your package, such as CMN20, CMN21, CMN30, or CMN32. All installation jobs after the first job are submitted by the ChangeMan ZMF internal scheduler.

This table describes the fields and options available on the **Create: Scheduling Dependencies** panel.

Field	Description
Line Command	I*n*: Insert *n* lines. The *n* is optional.
	R*n*: Repeat a line *n* times. The *n* is optional.
	D*n*: Delete *n* lines. The *n* is optional.
Successor	Jobs in your automated job scheduler that run after the first installation job for your package.
Predecessor	Jobs in your automated job scheduler that run before the first installation job for your package.

You can leave the **Create: Scheduling Dependencies** panel blank when you create your package and use the Update Package function to provide the information later.

When you finish typing information on the **Create: Scheduling Dependencies** panel, press **Enter**, and the next panel in the create package process is displayed. Your new change package is not created until all new package information has been entered and validated.

Adding Affected Applications

The **Create: Affected Applications** panel (CMNCRT05) is displayed if you are creating a participating package (Package Level 4).

```
CMNCRT05       Create: Affected Applications             Row 1 to 12 of 12
Command ===>                                              Scroll ===> PAGE

Complex/super package .......ACTP4

        Appl
___     GENL
___     ACTP
___     ____
___     ____
___     ____
___     ____
___     ____
******************************* Bottom of data ********************************
```

This table describes the fields and options available on the **Create: Affected Applications** panel.

Field	Description
Line Command	I*n*: Insert *n* lines. The *n* is optional.
	R*n*: Repeat a line *n* times. The *n* is optional.
	D*n*: Delete *n* lines. The *n* is optional.
Complex/ super package	Enter the package ID of the Complex or Super package associated with this participating change package.
Appl	Enter the three or four (3-4) character mnemonic for the application affected by this change package. When you freeze your participating package, planned approvals that are defined as Interfacing Approvals in the affected application are assigned to your package.

You can leave the **Create: Affected Applications** panel blank when you create your package and use the Update Package function to provide the information later.

When you finish typing information on the **Create: Affected Applications** panel, press **Enter**, and the next panel in the create package process is displayed. Your new change package is not created until all new package information has been entered and validated.

Defining Complex/Super Package Information

The **Create: Complex/Super Information** panel (CMNCRT08) is displayed if you are creating a super or complex package (Package Level 2 or 3).

```
CMNCRT08 Create:        Complex/Super Information       Row 1 to 12 of 12
Command ===> ___________________________________________ Scroll ===> PAGE

        Package
____    ACTP000009
____    ACTP000008
____    __________
____    __________
____    __________
____    __________
____    __________
____    __________
____    __________
____    __________
____    __________
____    __________
******************************* Bottom of data ********************************
```

This table describes the fields and options available on the **Create: Complex/Super Information** panel.

Field	Description
Line Command	I: Insert new line.
	R*n*: Repeat existing line.
	D*n*: Delete existing line.
Package	Enter the change package ID's of the Level 4 (Participating) change packages which are part of this Complex or Super package. Note: They must be in DEV status.

You can leave the **Create: Complex/Super Information** panel blank when you create your package, and you can use the Update Package function to provide the information later.

When you finish typing information on the **Create: Complex/Super Information** panel, press **Enter**, and the next panel in the create package process is displayed. Your new change package is not created until all new package information has been entered and validated.

If you are reconfiguring a pre-existing complex/super group of participating packages, the audit RC for all relevant packages will be reset. This is done because any audit performed prior to this reconfiguration is potentially invalidated by this change in environment. A warning panel will advise you that this is about to happen and you can cancel the change if you are not sure.

Note

Only those packages in DEV status will have their audit RC's reset. There may be frozen packages belonging to the same complex package that will be affected by this reconfiguration. You may choose to revert these and re-audit.

Entering Package User Information

The **Create: Package User Information** panel (CMNDPUP1) is displayed if your administrator enabled the Package User Information Facility. This panel and an optional second panel (CMNDPUP2) are used to populate package level custom variables defined by your administrator and used elsewhere in the change package lifecycle. Note: CMNDPUP2 is only displayed if **Next panel** is selected. Your administrator will probably modify these panels, so the panel title and format may not look like this sample panel that is delivered from Serena.

```
CMNDPUP1        CREATE - Sample Package User Panel 1
Command ===> __________________________________________

Enter "yes" or "no" to indicate value of variable

Field 1 . . . . . . . NO
Field 2 . . . . . . . NO
Field 3 . . . . . . . NO
Field 4 . . . . . . . NO
Field 5 . . . . . . . NO
Field 6 . . . . . . . NO

Enter "/" to select option
Next panel
```

The descriptions for the fields on these panels are in the help panels that you access by pressing **PF1**.

You can leave the **Create: Package User Information** panel blank when you create your package, and you can use the Update Package function to provide the information later.

Note

You will have the option to go to **Create: Package User Information** (CMNDPUP2) panel 2 by selecting **Next panel**, press **Enter**. The next panel in the create package process is displayed. Your new change package is not created until all information has been entered and validated.

```
CMNDPUP2        CREATE - Sample Package User Panel 2
Command ===>____________________________________________

Enter "yes" or "no" to indicate value of variable:

Test Value1 . . . . . NO
Test Value2 . . . . . NO
Test Value3 . . . . . NO
Test Value4 . . . . . NO
```

Defining Install Date and Site Information

The **Create: On Site Information** panel (CMNCRT06) is displayed for all simple and participating packages (Package Type 1 or 4) in an All environment. A change package in an All environment is only installed at one site, so you specify only one Install Date/Time on this panel.

```
CMNCRT06        Create: On Site Information         Row 1 to 1 of 1
Command ===> ______________________________________ Scroll ===> PAGE

Install     Time
Date        From To     Primary/backup contacts     Phone numbers
20141231    2100 2200   John Doe                    9174445555
                        Jane Doe                    9177777777
******************************* Bottom of data ***********************************
```

```
CMNCRT07         Create: Site Information           Row 1 to 1 of 1
Command ===>______________________________________ Scroll ===> PAGE
            Install     Time
    Site    Date        From To   Primary/backup contacts Phone numbers
    SERT6 20141231      2100 2200 John Doe                9177777777
                                  Jane Doe                9177777777
******************************* Bottom of data *******************************
```

The **Create: Site Information** panel (CMNCTR07) is displayed for all simple and participating packages (Package Type 1 or 4) in a Development (D) or Development/ Production (DP) environment. A package in a D or DP environment can be installed at multiple sites, so you must specify which site you want your package installed and enter an Install Date/Time for each site.

```
CMNCRT07        Create: Site Information        Row 1 to 1 of 1
Command ===>__________________________________ Scroll ===> PAGE

            Install     Time
Site        Date        From To     Primary/backup contacts Phone numbers
SERT6       20141231    2100 2200   John Doe                9177777777
                                    Jane Doe                9177777777
******************************* Bottom of data *******************************
```

This table describes the fields and options available on the **Create: On Site Information** panel and the **Create: Site Information** panel.

Field	Description
Line Command	I: Insert new line.
	R: Repeat existing line.
	D: Delete existing line.
	*: Display the Site Selection List panel where you can select a site to add to the Create: Site Information panel.
Site	Enter the site name where the change package is to be installed. When the **Create: Site Information** panel is first displayed, it lists the ZMF instance where you are working plus any site for which production libraries are defined in application administration. You can delete the sites where you do not want your package installed. This field only displays for Development or Development/ Production environments.
Install/Date	Enter the system date and time that you want the package to install at each site. If you specified **Manual** in the **Scheduler** field for this package, the install is initiated immediately after the last package approval is entered, regardless of what you entered in the **Install Date/ Time**. Otherwise, the package installation process begins on the specified install date after the "From" time. The exact time that an automated installation is initiated depends on the interval between scheduler events.
	You cannot enter a date and a "To" time that has passed at the site. Your global administrator can define a time difference for each remote site so that an install date/ time that has passed at the development site is still valid at a remote site where the system clock is ahead of the development site.
	If you want a package to install at the same system time at all sites, then type the same date and time in the **Install Date/Time** for all sites. If you want the package to install simultaneously at all sites, then adjust the **Install Date/Time** that you enter for each site by the difference in system clock times between each site and the development site.
	Install Date Enter the calendar date (yyyymmdd) that the package will be installed at the site selected.
	Time From Enter the time of day to begin installation of your package into production. Enter four digits in the form HHMM (24 hour clock).
	Time To Enter the time of day to end installation of your package into production. Enter four digits in the form HHMM. If the selected scheduling system is 'CMN', this field will be ignored.
Primary/ backup contacts	Enter the names of the primary and back-up contacts to be notified at each site if there is a problem with the package installation. These fields are 25 characters long.

Field	Description
Phone numbers	Enter the telephone numbers of the primary and backup contacts. These fields are 15 characters long.

When you finish entering information on the **Create: On Site Information** panel or the **Create: Site Information** panel, press **Enter**. If there are no errors, all of the information you specified on all Create Package panels is saved, you are returned to the **Primary Option Menu**, and a short ISPF message in the upper right corner of the menu displays the Package ID that ChangeMan ZMF generated for the new package you created.

5. Updating Change Package Information

Most of the information that you enter in the Create Package process can be changed later. You use the Update Package Information function to change package-level information in packages that are in Development (DEV) or Open (OPN) status.

Rules for Update Package Information Functions

These rules and restrictions apply to the update package information function.

- To update change package information in an application, you must have update access to the application, which is defined under the ChangeMan ZMF resource class in your security system.
- Package Type and Package Time Span can never be changed after a package is created.
- Package information can be updated when a package is frozen by selectively unfreezing the General package category or the Onsite / Remote Sites category.
- If you are not allowed to update package information, the fields on package update panels display in browse mode.

Administration Settings for Update Package Information

Your global and application administrators make settings in ChangeMan ZMF administration that control how the update package information function works in your application. Ask your administrator if any of the following apply to your application.

- Restrict entries in the package Scheduler field on the Update: Installation Instructions panel. (Global Administration Parameter: Installation Job Scheduler)
- Restrict the number of packages that can be scheduled for installation on a particular Install Date through the use of the Installation Calendar. (Global Administration Parameters: Disable Installation Calendar)
- Require data in the Work Request field on the Update: Control Information panel. (Application Administration Parameters: Require Work Request Number)

- Require data in the Department field on the Update: Control Information panel. (Application Administration Parameters: Require Department Number)
- Enable the Package User Information facility to display one or two panels that you use to set package-level variables defined by your administrator. (Global Administration Parameters: Display Package User Option Panel)
- Set the difference in system clock time between the development site where your change package is created and a remote site where it is scheduled for install. The time difference may allow you to enter an install time for a remote site that has already passed at your site. (Global Administration Sites: Time Difference)

Exit Programs for Update Package Information

With the release of 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/1. Please refer to the **ChangeMan ZMF 8.1 Readme** and **ChangeMan ZMF High Level Language Functional Exits** *Getting Started Guide* for details.

Your ChangeMan ZMF installer or administrators may implement exit programs to alter the behavior ChangeMan ZMF. Ask your administrator if any of the following apply to your application.

- Determine special criteria for who can update package information by package status, package type, administration authority, and other criteria. (Exit program CMNEX001)
- Restrict Install date by one of more of the following: Application, Global or Application Administration authority, specified date, today’s date, day of week, package type, values in Package User Information variables. (Exit program CMNEX002)
- Impose a lead time between today’s date and the package install date. Other criteria available includes Application, Global or Application Administrator authority, specified date, day of week, package type, values in Package User Information variables. (Exit program CMNEX003)
- Restrict install date by one of more of the following: Application, administrator authority, specified date, day of week, package type, today's date. (Exit program CMNEX007)
- Validate information entered in Work Request Number and/or Department against a specified a list of values. Cross-edit Work Request Number, Department, and Package User Information. (Exit program CMNEX014)
- Override package status validation when adding or removing participating packages in complex or super packages. Allow automatic close of super or complex packages after update if all participating packages have completed the package lifecycle. (Exit program CMNEX033)
- Validate group site names used on the Site Information panel, and specify the sites associated with each group site name. (Exit program CMNEX039)

- Define rules for package IMS information update. (Exit program CMNEX041) Your administrator may also have applied other exit programs to your application.

Accessing Package Information Update Functions

Go to Primary Option Menu (CMN@PRIM) select Option 1. The Build Option panel (CMNBUILD) is displayed. Select Option 2 to access the **Update: Package Information** panel (CMNPGNL0).

```
CMNPGNL0        UPDATE: Package Information
Option ===>_______________________________________

Package . . . . ACTP000013

1 Control           Package control information
2 General           General description
3 Instruction       Installation instructions
4 Dependencies      Job Scheduling dependencies
5 Affected Apps     Affected applications
6 Participation     Participation packages
7 Install Dates     Install date and site information
8 Close Package     Close complex/super package
9 Open Package      Open complex/super package
D Db2               Db2 package information
E User              Package user information
I IMS               IMS package information
R Release           Package release information
```

The **Update: Package Information** panel dynamically displays update options depending on the ChangeMan ZMF licensable options that you license. If you license one or more licensable options, one or more of these panel options are displayed:

```
D Db2 - Update Db2 package information
E User - Package user information
I IMS - Update IMS package information
R Release - Change package Release information (ERO)
```

Use one of these methods to display the **Update: Package Information** panel:

- Using the Menu Hierarchy:

  a On the **Primary Option Menu**, select **1 Build**.

  b On the **Build Option** panel, select **2 Update**.

- Using Direct Panel Access:

  You can directly access the panel from any ChangeMan ZMF panel, by entering **=1.2** in the **Command** or **Option** line and pressing **Enter**.

Using the Change Package List to Update Package Information

You can access the options listed on the **Update: Package Information** menu directly from the **Change Package List**.

On the **Change Package List** panel, type one of the following in the line command for a change package and press **Enter**.

- **U1** Update control information
- **U2** Update general information
- **U3** Update installation instructions
- **U4** Update job scheduling dependencies
- **U5** Update affected application
- **U6** Update participating package information
- **U7** Update install date and site (remote) information
- **U8** Close complex/super package (Change to CLO status)
- **U9** Open complex/super package (Change to OPN status)
- **UE** Change package user information

The package update line commands available on the **Change Package List** panel depend on the ChangeMan ZMF licensable options that you license. If you license one or more licensable options, one or more of these line commands is available:

- **UD** Update package Db2 information
- **UI** Update package IMS system information
- **UR** Update package Release information

Updating a ChangeMan ZMF Package

The ISPF panels used to update package information are nearly identical to the panels you use to create a package. Differences include:

- The panels titles say “Update” instead of “Create”. For example, you first enter a package description on the **Create: Package Description** panel, and you change that description after the package is created by using the **Update: Package Description** panel.
- Some fields on update panels are read-only. For example, the **Package Type** field on the **Update: Control Information** panel is protected and cannot be changed after the package is created.

- Some panels display package information that was not available when the package was being created. For example, the **Update: Control Information** panel displays the Package ID, Status, and Install Date in display-only fields.

Use the following table to find the topic in "Creating a Change Package" that tells you about the Create panel that corresponds to the Update function you want to perform. The comments in this table tell you about differences between the Create and Update versions of the panels.

Update Option	Corresponding "Create" Topic	Comments
1 Control	See Defining Package Control Information	You can switch the **Package Level** between **1-Simple** and **4-Participating** only.
2 General	See Providing a Package Description	Press **Enter** twice to save your updates on the **Update: Package Description** panel.
3 Instruction	See Providing Installation Instruction	Press **Enter** twice to save your updates on the **Update: Installation Instructions** panel.
4 Dependencies	See Setting Job Scheduling Dependencies	You can add or update this information only if the package **Scheduler** is **OTHER**. See option **3 Instructions**.
5 Affected Apps	See Adding Affected Applications	The **Update: Affected Applications** panel is displayed for participating packages only.
6 Participating	See Defining Complex/ Super Package Information	Use the **Update: Complex/ Super Information** panel to add or remove participating packages in a Super or Complex package.
7 Install Dates	See Defining Install Date and Site Information	
8 Close Package		This option changes the status of a Complex or Super package to CLO. No other panel is displayed.
9 Open Packages		This option changes the status of a Complex or Super package to OPN. No other panel is displayed.
E - User	See Entering Package User Information	
D - Db2		See the *ChangeMan ZMF Db2 Option Getting Started Guide
I - IMS		See the *ChangeMan ZMF IMS Option Getting Started Guide
R - Release		See the *ChangeMan ZMF ERO Getting Started Guide*.

6. Checking Out a Component

The checkout component function copies a component from a baseline or promotion library into a staging library allocated exclusively to your package. If a package staging library has not already been created for the type of component you are checking out, a staging library for that type is allocated.

The checkout function can show you the baseline and promotion libraries that contain a component before you check it out. The display shows prior versions in baseline libraries as relative minus versions (-1, -2, -3). Future versions in promotion libraries are shown as plus (+) promotion level numbers (e.g. +10, +20, +30).

- Rules for Checking Out Components
- Accessing Checkout Component Functions
- Checking Out from Baseline or Promotion Libraries
- Checking Out From a Package
- Checking Out Package Components
- Checkout Warnings and Messages

Rules for Checking Out Components

These rules and restrictions apply to the checkout component function:

- You can check out a component from a baseline library or a promotion library. When you check out from a promotion library, you are assuming that the component in the promotion library will be installed by the time your package is ready to be installed.
- You may be able to check out a component into a personal library. Administration Settings for Checkout Component Functions. If you check out to a personal library, the component is also copied to a package staging library.
- After you check out a component, the status of the component in your package is CHECKOUT.
- You can check out a component into your package if the component is already in your package in CHECKOUT status. You do not have to delete the component to do so.
- Checkout can be executed online or in a batch job. Batch checkout releases your terminal while components are copied into staging or personal libraries and component history is built.

- When you check out to a personal library or data set, ChangeMan ZMF verifies that the data set is cataloged and that you have at least UPDATE authority for the data set in your security system.
- If checkout to a CA Panvalet or CA Librarian library is allowed, you must use batch checkout to target those proprietary library technologies.
- When you check out a component that is already checked out to one or more active packages, every user who checked out the same component to an active package is notified with an MVS SEND message.
- When you check out a component from either baseline or promotion, the vv.mm ISPF statistics in the staging library are set according to this formula, relative to the statistics in the origin library: vv=vv+1, mm=00.
- When you check out a component that has an associated Component Activity File library type, the Component Activity File member with the same name is checked out into a package staging library, regardless of the whether you choose a staging or personal library as the target for the original component.
- When you check out a component from a promotion library, if that component has an associated Component Activity File library type, the Component Activity File member is always copied from the 0-level baseline library.

Administration Settings for Checkout Component Functions

Your global and application administrators make settings in ChangeMan ZMF administration that control how the checkout component function works in your application. Ask your administrator if any of the following apply to your application.

- Require use of the checkout function to add a component to your package if the component is in a baseline library. Prohibit stage from development, or require special security authorization to use stage from development, if the component is in baseline. (Application Administration Parameters: Checkout Enforcement Rule)
- Restrict checkout of a component to one package at a time. (Application Administration Parameters: Allow Concurrent Checkout)
- Allow checkout of components into personal libraries. Copies of components checked out to personal libraries are kept in package staging libraries. (Application Administration Parameters: Allow Checkout to Personal Lib).

Exit Programs for Checkout Component Functions

Your ChangeMan ZMF installer or administrator may implement exit programs to alter the behavior ChangeMan ZMF. Ask your administrator if any of the following apply to your application.

- Add custom processes before and/or after checkout. (Exit program CMNEX019)
- Bypass the Checkout Enforcement Rule for specified library types to permit stage from development when a component is in baseline. See Checkout Enforcement Rule in Administration Settings for Checkout Component Functions. (Exit program CMNEX030)
- Restrict the library types displayed on the valid library selection list for checkout, stage, browse baseline, browse compressed listing, compare, scan, scratch/rename, and relink functions. This exit effectively disables these functions for the specified library types. (Exit program CMNEX035)

Accessing Checkout Component Functions

Checkout functions are accessed from the **Checkout Options** menu (CMNMCKOR).

```
CMNMCKOR                        Checkout Options
Option ===>_______________________________________________________

Package . . . . ACTP000014

1 Baseline/Promo    From baseline/promotion libraries
2 Package           Package components
3 Release           Components from a release
4 Package           From package
```

Use one of these methods to display the **Checkout Options** menu:

- Using the Menu Hierarchy:

  a On the **Primary Option Menu**, select **1 Build**.

  b On the **Build Option** panel, select **5 Checkout**.

- Using Direct Panel Access:

  You can also access the **Checkout Options** panel by entering **=1.5** on the **Command** line from any ChangeMan ZMF panel.

Using the Change Package List to Checkout Components

You can access the options listed on the **Checkout Options** menu directly from the Change Package List.

On the **Change Package List** panel, type one of the following in the line command for a change package and press **Enter**.

- **C1** Check out components from Baseline/promotion
- **C2** Check out components from package list
- **C3** Check out components from package release
- **C4** Check out (from package)

Checking Out from Baseline or Promotion Libraries

Follow these steps to check out one or more components from a baseline or promotion library into your change package.

1. On the **Checkout Options** menu, select Option **1 Baseline/Promo** to display the **Checkout** panel (CMNCKOT1).

```
CMNCKOT1                    Checkout
Command ===>____________________________________________________________________

Package . . . . . . .  TEST001852__

Component name . . ._______________________________________________________________ +
                                        (Blank or pattern for list; * for all members)
Library type . . . . . ____             (Blank for list)
Source library . . . . 0___             (Baseline 0 to -n; Promotion +1 to +n)

Checkout to . . . . . .  S___             (S-Staging library, P-Personal library)
Personal library . . .  ___________________________________________________________ +
Library dsorg . . . .  ____             (Personal lib: PDS, PDSE, SEQ, PAN, LIB, ZFS)
Checkout mode . . . .  1__              (1-Online, 2-Batch)
Data Encoding          ___              (1-ASCII, 2-UTF-8)

Enter "/" to select option
/ Confirm overlay                                / Lock component
/ Display component user options
_ Build member list from scan results            _ Case sensitive

Records to select .    0__ .            (0 = all)
From column .          0__ .            (0 = all) To column          0      (Max = 80)
Data string . ______________________________________________________
```

This table describes the fields on the **Checkout** panel.

Field	Description
Command	Enter **L** to display the **component.type Library List** where component name (entered below) is found. Panel shows baseline and promotion libraries that contain the component. Note: For this option, you must enter a fully-qualified component name - no masking is permitted.

Field	Description
Package	This field displays the package ID associated with the component being checked out. The package must be in DEV status.
Component name	Enter the name of the component you want to check out. Other options:
	Pattern: Build and display a filtered list of components from the origin library by entering a component pattern (aaa*). You can select components from the filtered list for check out. **Note:** Do not type * by itself unless you want to check out all of the components in the library.
	Blank: Leave this field blank or enter a pattern to select from a list of components; Fill in SOURCE LIBRARY. To list all libraries where a particular component is found, enter the component name here, enter the LIBRARY TYPE, leave the SOURCE LIBRARY blank and enter 'L' on the command line.
	: Enter '' to check out all members of the SOURCE LIBRARY.
Library type	Enter the library type of the component to be checked out. Other options:
	Pattern: Display a filtered list of library types.
	Blank: Display a list of all library types so you can select the type you want.
Language	Note:** Language is no longer specified on this panel. It can be specified on the "STAGE: COMPILE AND LINK EDIT" panel. ChangeMan will 'assume' the language based on:
	1 Designated procedure at force level
	2 Component history
	3 Designated procedure at force level 1
	? Enter ? to specify the language of the designated procedure.
Source library	Enter an integer to indicate the baseline version or promotion level that you want to check out from. Plus or minus signs are required.
	0: Check out from the current or 0-level baseline library.
	-1 to -999: Check out from a prior version baseline library. The 999 integer following the minus sign represents a relative prior version (e.g. -1 means the version immediately prior to the current version). Librarian Archie and Stacked Reverse Delta baseline libraries can range as low as -255 and -999 respectively.
	+1 to 99: Level 1 refers to the lowest promotion level, up to +99 n as the closest to production. Promotion libraries can range as high as 99.
	Blank: You can leave this field blank if you type **L** in the **Command** field.

Field	Description
	Note: **If you type** L **in the** Command **field, List libraries where 'Component Name' entered is found. For this option, you must enter a fully-qualified component name - no masking is permitted. You can select a library on this list to check out from regardless of the version or level originally specified in the** Source Library** field.
Checkout to	Enter S or P to determine whether components are checked out to a staging library or a personal library or data set. When you check out to a personal library or data set, the component is also copied to a package staging library.
	S: Check out components to a ChangeMan ZMF staging library.
	P: Check out component to a personal development library or data set.
Personal library	If '**P**' is selected in the **Checkout to** field (above), you must enter the name of the target personal library. If a dataset name is entered and it does not exist, ChangeMan ZMF will allocate it based on the Library Type attributes. **Note:** Batch checkout is required for any personal library including PAN or LIB.
Library dsorg	Enter a data set organization for the library or data set named in the **Personal Library** field.
	PDS: Partitioned dataset
	SEQ: Sequential data set.
	PAN: Panvalet file.
	LIB: Librarian file.
	ZFS: z/OS File System
Checkout mode	Enter 1 or 2 to determine whether checkout is performed online or in a batch job. See Online and Batch Execution Modes.
	1. Check out components online - The components to be checked out reside in a Baseline library whose Storage Means is defined as (D)elta deck or (C)ompressed PDS.
	2. Check out components in a batch job - The PERSONAL LIBRARY NAME to which the component is to be checked out is in PANVALET or LIBRARIAN format, or does not yet exist. If the components being checked out is a unique library type, not defined as Like S, C, L, or P (See Admin - Library definitions) it must be checked out in batch.
Data Encoding	This option can be used to view components stored using a CCSID representing ASCII or UTF-8 characters. This option has no effect when using Browse.
	1. Enables ASCII conversion for ISPF view.
	2. Enables UTF-8 conversion for ISPF view.
Confirm overlay	Select to overlay component in target library before checkout is executed. Omit to bypass confirmation panels and perform checkout.

Field	Description
	Note:** If your application is configured to maintain baseline by site, a site selection list will be displayed from which a site must be selected prior to a checkout.
Lock component	Select to immediately lock components in your package after checkout. Omit to bypass component locking.
	Notes:** This only applies to the current package, it does not cross packages. See Locking and Unlocking Package Components
	Batch checkout will not update the member's ISPF statistics. Batch checkout will also bypass the OVERLAY PRIOR STAGED MODULE Admin rule. If a component is locked by another user you must go to the stage package driven list (1.6.2) to UNLOCK the component prior to checkout. UC Unlock Component to allow for Checkout/Stage.
Display component user options	Select to allow display of component user option panel(s) for non-source components during stage process. It has no effect during checkout. Setting the value here allows you to bypass the stage selection panel to set the value prior to edit or stage. This allows user data to be associated with a component for later use in exits or XML services. If the '**Enable component user variables**' global option is not enabled this setting is ignored.
	Note:** Component user variables for source components are accessible during the build process.
Build memberlist from scan results	Select to scan the source library to limit the member list to members containing a specified data string. If not selected, all fields listed below (Case Sensitive, Records to Select, From Column, To Column, and Data String) are ignored. Member list scan does not support zFS files or data sets with RECFM=U.
Case Sensitive	Select if the data string entered should be used exactly as entered (with regard to upper and lower case characters). Omit if the scan is to be conducted without regard to case sensitivity.
Records to Select	Enter a numeric value used to determine when the scan is to be discontinued. When this value is '0', the scan displays a list of all members in the source library that meet the data string search criteria.
From Column	Specify upon which column, for each record, that the scan is to begin. When this value is '0', each record is scanned, beginning in column one.
To Column	Specify upon which column, for each record, that the scan is to end. When this value is '0', records are scanned to the 80th column.
Data String	Enter a data string to be scanned to build the member selection list.

2. If you enter **L** on the **Command** line the **Checkout** (CMNCKOT1) panel, the **component.type Library List** (CMNCMLSL) panel is displayed. Select the library from which to check out the component.

```
CMNCMLSL            ACPSR*.SRC Library List             Row 1 to 3 of 3
Command ===>___________________________________________Scroll ===> CSR

    Lvl     Dataset/pathname                +        Prom.name   Site
__ +010    CMNTP.S6.V810.PROM.S6P1UT.SRC            S6P1UT      SERT6
__ +020    CMNTP.S6.V810.PROM.S6P1IT.SRC            S6P1IT      SERT6
__ 0000    CMNTP.S6.V810.BASE.ACTP.SRC              BASELINE
******************************* Bottom of data ********************************
```

3. On the **Checkout** panel, if you leave the **Component Name** field blank or use a pattern that selects multiple components, the **Promotion Library Selection List** (CMNCPLSL) is displayed. Enter **S** in the line command field to select from the list of promotion libraries you wish to use.

```
CMNCPLSL            Promotion Library Selection List            Row 1 to 2 of 2
Command ===>__________________________________________________ Scroll ===> CSR

        Library type: SRC           Promotion level: +10

    Dataset                 Name        Site
__  CMNTP.S6.V810.PROM.S6P1UT.SRC            + S6P1UT      SERT6
__  CMNTP.S6.V810.PROM.S6P1UT1.SRC           + S6P1UT1     SERT6P1
******************************* Bottom of data ********************************
```

```
CMNCCMSL            Checkout from SRC/LVL(0)                Row 1 to 7 of 7
Command ===>_______________________________________________ Scroll ===> PAGE

    Input library:
    CMNTP.S6.V810.BASE.ACTP.SRC
Name     Status  vv.mm   Created     Last modified       Size    Init    User
__ ACPSRCCA         01.16   2002/05/07 2014/11/19 03:13     34      23      USER016
__ ACPSRCCC         01.05   2002/05/07 2014/11/19 03:10     29      23      USER016
s  ACPSRCCE         01.03   2002/05/07 2014/11/19 03:10     29      23      USER016
__ ACPSRCD1         03.01   2002/05/07 2011/03/11 20:09     52      15      USER016
__ ACPSRC80         01.03   2002/05/07 2014/11/19 03:04     24       1      USER016
__ PLIPGM01         01.07   2007/04/16 2014/11/19 03:06      6       6      USER016
__ PLIPGM1          01.04   2011/05/27 2014/11/19 03:06      6       6      USER016
******************************* Bottom of data ********************************
```

Valid primary commands are:

Command	Description	Abbreviation
CANCEL	Cancel the checkout.	C
DCD	Display component general descriptions.	D
LOCATE name	Locate an entry.	L
LONG	Display zFS components expanded.	LON
REFRESH	Refresh the panel display.	R
SHORT	Display an abbreviated component list.	SH

Command	Description	Abbreviation
SORT name	Sort the list by the column name entered in the command.	SO

Enter **S** in the line command for one or more components and press **Enter** to check the components out to your package.

Valid line commands are:

Command	Description
B	Browse a component.
H	Display component history.
S	Select a component for checkout.
V	View a component.

4. If a component you select for checkout is in another active change package, or concurrent checkout has been disallowed, a concurrent development warning message is displayed on the History *component.type* (CMNCMPSW) panel. The results of the line commands are displayed in the STATUS column.

```
CMNCMPSW            History - ACPSRCCE.SRC                  Row 1 to 5 of 5
Command ===>______________________________________________ Scroll ===> CSR

This component is included in the following packages:

Package     Sta Promoted vv.mm Last action      Size  Procname  User    Release
ACTP000065  DEV          04.00 2015/03/30 13:56 00030 CMNCOB2   USER016
ACTP000060  DEV          03.00 2015/03/16 14:56 00029 CMNCOB2   USER017
ACTP000038  DEV S6P1UT   03.05 2015/02/12 10:21 00030 CMNCOB2   USER016
ACTP000028  DIS          02.01 2015/01/20 22:55 00029 CMNCOB2   USER016
ACTP000027  DEV          03.00 2015/03/06 11:59 00029 CMNCOB2   USER017
******************************* Bottom of data ********************************
```

Make sure that you are coordinating your changes with the other developers who have checked out or staged the component into their packages.

If you decide to abandon the checkout for this component, type **CANCEL** in the **Command** line and press **Enter**. Otherwise, press **PF3** or **Enter** to check out the component.

Checking Out From Promotion Sites

The **Checkout** panel gives you the option of checking out from a promotion library by specifying a promotion level number in the **Source Library** field. If the application has multiple promotion sites, the same promotion level number may point to different libraries in different sites, so multiple libraries may be displayed on the **Promotion Library Selection List** (CMNCPLSL) panel.

```
CMNCPLSL        Promotion Library Selection List            Row 1 to 2 of 2
Command ===>______________________________________________ Scroll ===> CSR

Library type: SRC       Promotion level: +10

Dataset             Name                    Site
CMNTP.S6.V810.PROM.S6P1UT.SRC       + S6P1UT                  SERT6
CMNTP.S6.V810.PROM.S6P1UT1.SRC      + S6P1UT1                 SERT6P1
******************************* Bottom of data ********************************
```

Enter **S** in the line command to select the library from which to check out the component.

```
CMNCCMSL            Checkout from SRC/LVL(+10)              Row 1 to 5 of 5
Command ===>       Scroll ===> CSR

    Input library:
    CMNTP.S6.V810.PROM.S6P1UT.SRC

    Name    Status          vv.mm Created       Last modified     Size  Init User
S   ACPSRCCE                03.04 2002/05/07    2015/02/01 22:07  30    23   USER016
__  ACPSRCEE                01.01 2015/02/01    2015/02/01 22:08  30    30   USER016
__  ACPSRC1A                02.03 2014/12/18    2015/02/01 22:09  33    34   USER016
__  ACPSRC6A                03.02 2012/09/26    2015/02/01 19:45  44    27   USER016
__  ACPSRC92                03.01 2002/05/07 2 015/01/15 00:06   30    15   USER016
******************************* Bottom of data ********************************
```

Enter **S** in the line command to select the component for checkout.

Replying to the Save Previous Version Panel

If the component you want to check out is in a library type that is enabled for staging versions with the PROMPT parameter, ChangeMan ZMF displays the **Save Previous Version** (CMNCMP03) panel before executing the checkout, regardless of whether the component is actually in the package staging library.

```
dataset name . . . 'CMNTP.A00A2.#CE9B4B9.#81CAF70.OUTLIST'
        |------------------------------------------------------------|
Report  | CMNCMP03 Save Previous Version         |
        | Command ===>_____________________________________________ |
        |                   | rinting
        | Package: ACTP000027                    | nting
        | Component: ACPSRC94                  + |
Job st  | Type: SRC          |
//US    |                   |
//      | Enter "/" to select option             |
//  |  / Save previous staging version        |
//  |  _ Set Save Previous Version prompt off                     |
        |                   |
        |                   |
        |------------------------------------------------------------|
```

This table describes the fields on the **Save Previous Version** panel.

Field	Description
Package	Displays the ID of the package you are checking out to.
Component	Displays the component name
Type	Displays component type.
Save previous staging version	Select to save previous component version that is in staging library.
	NOTE If the component you are checking out is not already in the package staging library, your response in this field is ignored.
Set Save Previous Version prompt off	Select field 'Set Save Previous Version prompt off' to skip the display of this panel for subsequent requests within the current function.

See Using Staging Versions.

Checking Out Package Components

You can use the list of components in your package to check those components out into your package.

- If a component was previously checked out or staged from development into your package, the current staging library member is overlaid by the new component you check out from baseline or promotion.
- If you copied a package forward when you created your package, the list of components from the package you specified is copied into your new package. However, the components in that list are not copied into staging libraries for your package, and the status of the components

listed in your package is **INACTIVE**. When you use this list to check out components, there is nothing to overlay.

The general rule still applies that if a component in your package is in **CHECKOUT** status, you cannot check the component out again into your package until you stage or delete the component.

On the **Checkout Option** panel, select Option **2 Package** to display the **Checkout: package Components** panel (CMNCKOTS).

```
CMNCKOTS        CHECKOUT: ACTP000013 Components          Row 16 to 22 of 22
Command ===>____________________________________________  Scroll ===> PAGE

    Name +  Type Status     Changed         Language Procname User    Request
__ ACPSRC95 SRC CHECKOUT   20141210 045636 COBOL2   CMNCOB2  USER016 LOCKED
__ ACPSRC96 SRC CHECKOUT   20141210 045637 COBOL2   CMNCOB2  USER016 LOCKED
__ ACPSRC97 SRC CHECKOUT   20141210 045638 COBOL2   CMNCOB2  USER016 LOCKED
__ ACPSRC98 SRC CHECKOUT   20141210 045639 COBOL2   CMNCOB2  USER016 LOCKED
__ ACPSRC99 SRC CHECKOUT   20141210 045641 COBOL2   CMNCOB2  USER016 LOCKED
__ PLIPGM01 SRC CHECKOUT   20141210 045642 PLIE     CMNPLIE  USER016 LOCKED
__ PLIPGM1  SRC CHECKOUT   20141210 045643 PLIE     CMNPLIE  USER016 LOCKED
******************************* Bottom of data ********************************
```

Enter **S** in the **line command** for one or more components that you want to check out. To find out all available line commands, press PF1.

Note that **RECOMP** is displayed in the **Procname** field if the component has been recompiled in the source package.

Checking Out From a Package

You can check out from an existing package by selecting option 4, and this will result in the display of panel CMNCKOT3.

```
CMNCKOT3                            Checkout
Command ===>________________________________________________________

Package . . . . . . . ACTP000032

Source package . . . __________________________
Component name . . . _____________________________________________________+
Library type . . . . _____

Checkout to . . . . . ____        (S-Staging library, P-Personal library)
Personal library . . _____________________________________________________+
Library dsorg . . . . ____        (Personal lib: PDS, PDSE, SEQ, PAN, LIB, ZFS)
Checkout mode . . . . ____________(1-Online, 2-Batch)

Enter "/" to select option
__ Confirm overlay
/  Lock component
__ Display component user options
__ Mixed case
```

This function is used to copy components from another package into a personal library or staging library.

Field	Description
Package	This is the change package ID with which the component being checked out will be associated. The package must be in DEV status.
Source package	Enter the package from which to check out components.
Component name	Enter a component name, pattern or blank.
Library Type	Enter a library type, pattern or blank.
Checkout to	Specify the component's destination. Enter 'S' to check out the component to the package's Staging library. Enter 'P' to check out the component to a personal library.
Personal library	If 'P' is entered above, enter the name of the target personal library (use apostrophes). If PAN or LIB is selected, the check- out must be done in batch. If a dataset name is entered and it does not exist, ChangeMan will allocate it based on the Library Type attributes - and batch checkout is required.
Library dsorg	If checking out to a personal library, enter that library's dataset organization based upon the following:
	PDS: Partitioned dataset
	SEQ: Sequential dataset
	PAN: Panvalet file
	LIB: Librarian file

Field	Description
	ZFS: z/OS File System
Checkout mode	Enter 1 to check out the component online or 2 for batch. A batch checkout will be required if any of the following conditions apply:
	1. The components to be checked out reside in a Baseline library whose Storage Means is defined as (D)elta deck or (C)ompressed whose Storage Means is defined as (D)elta deck or (C)ompressed PDS.
	2. The PERSONAL LIBRARY NAME to which the component is to be checked out is in PANVALET or LIBRARIAN format, or does not yet exist.
	3. The component being checked out is defined as like-other.
Confirm overlay	Enter the option desired for confirmation panels: Select to view and respond to any confirmation required where the target file will be overlaid by performing this checkout. Omit to bypass confirmation panels and perform the checkout.
Lock component	Enter the option desired for component locking: Select to place an immediate lock on the component. This only applies to the current package, it does not cross packages. Omit to bypass component locking.
	NOTE: Batch checkout will not update the member's ISPF statistics. Batch checkout will also bypass the OVERLAY PRIOR STAGED MODULE Admin rule. If a component is locked by another user you must go to the stage package driven list (1.6.2) to UNLOCK the component prior to checkout. **UC** Unlock Component to allow for Checkout/Stage.
Display component user options	Allow display of component user option panel(s) for non-source components during stage process. It has no effect during checkout. Setting the value here allows you to bypass the stage selection panel to set the value prior to edit or stage. This allows user data to be associated with a component for later use in exits or XML services. If the 'Enable component user variables' global option is not enabled this setting is ignored. Note: Component user variables for source components are accessible during the build process.
Mixed Case	By default, all fields are folded to upper case. However, if the option 'Mixed Case' is selected, then the component name is left as is. Whichever case you enter will then be processed.

Checkout Warnings and Messages

The ChangeMan ZMF checkout function displays warning panels and issues MVS SEND messages to help you manage concurrent development and reduce the risk of overlaying an existing package component, losing the work already done.

> **Note**
>
> MVS SEND messages are displayed on a TSO terminal when a user presses Enter or a PF key. If a user is not logged on when the message is sent, the message is displayed when they logon to TSO.

Concurrent Development Warning

If you initiate a checkout (from the **Checkout** panel (CMNCKOT1)) for a component that is already checked out or staged from development into another active package, and select the **Confirm Overlay** field, the **History - component.type** (CMNCMPSW) panel is displayed before your checkout is executed with the following warning message: 'This component is included in the following packages:'.

```
CMNCMPSW        History - ACPSRCD1.SRC                     Row 1 to 2 of 2
Command ===>_____________________________________________ Scroll ===> PAGE

This component is included in the following packages:

Package     Sta Promoted vv.mm Last action      Size Procname User    Release
ACTP000021  DEV          04.00 2014/12/15 18:34 00052         USER016
ACTP000013  DEV          04.00 2014/12/10 04:56 00052         USER017
******************************* Bottom of data ********************************
```

You can use the information on the History - *concurrent.type panel* to contact the other developers working on the component in other packages, to coordinate your work.

> **Caution**
>
> The concurrent development warning is suppressed if the Confirm Overlay field on the Checkout (CMNCKOT1) panel is selected.

If you still want to check out the component into your change package, press **PF3** or **Enter**. If you want to cancel your checkout, type **CANCEL** in the Command line and press **Enter**.

Checkout Conflict Message

If you check out a component that is already checked out or staged in another active package by someone else, the user who performed the last checkout or stage of the component in the other package receives a checkout conflict SEND message:

```
Checkout Conflict: ACPSRC1A.SRC between ACTP000027 and ACTP000038 2015/03/09 16
:55 USER016 CN(INTERNAL)
 Checkout Conflict: ACPSRC6A.SRC between ACTP000027 and ACTP000038 2015/03/09 16
:56 USER016 CN(INTERNAL)
 Checkout Conflict: ACPSRC91.SRC between ACTP000027 and ACTP000028 2015/03/09 16
:56 USER016 CN(INTERNAL)
 Checkout Conflict: ACPSRC92.SRC between ACTP000027 and ACTP000038 2015/03/09 16
:56 USER016 CN(INTERNAL)
***
```

Users who performed the last checkout or stage on your component in multiple other packages receive multiple checkout conflict messages.

Checkout Overlay Warning

If you initiate a checkout for a component that is already in your package, and if you select the **Confirm Overlay** field on the **Checkout** panel (CMNCKOT1), the **Checkout Warning** (CMNCKWTP) panel is displayed.

```
CMNCMPSW        History - ACPSRCD1.SRC                  Row 1 to 2 of 2
Command ===>                                           Scroll ===> PAGE

This component is included in the following packages:
        |--------------------- Checkout Warning ----------------------|
Packag  | CMNCKWTP                                                    | Release
ACTP00  | Command ===>                                                |
ACTP00  |                                                             |
  |  Checking out component                                       |   |
ACPSRCD1                                                             |
| Will overlay JDOE version.                                          |
|                                                                     |
|                                                                     |
|                                                                     |
|                                                                     |
|---------------------------------------------------------------------|
```

You can use the information on the Checkout Warning panel to avoid overlaying an existing package component and losing the work already done on that component. You can also use the Checkout Warning information to contact another developer who is working on the component in your package to coordinate your work.

Caution

The checkout overlay warning is suppressed if the Confirm Overlay field is not selected on Checkout panel (CMNCKOT1).

If you still want to check out the component into your change package and overlay the existing package component, press **Enter**. If you want to cancel your checkout, press **PF3** or type **CANCEL** in the Command line and press **Enter**.

Note

If the package component you want to overlay is in CHECKOUT status, message CMN2330I is displayed, and your checkout is blocked.

Checkout Overlay Message

If you check out and overlay a component that was staged in your package by someone else, that user receives a checkout overlay SEND message.

```
Checkout Overlay: ACPJCL53.JCL in ACTP000090 by USER240 CN(INTERNAL)
```

Confirming version regression override

When you check out a component into a package, ZMF records information about the baseline component at that time. If, at some later time, a different package is used to update the baseline version of the component, that baseline information no longer matches that recorded in your package. When audit sees this, it flags a synch10 version regression error.

Audit knows nothing about the contents of your component. You may well have included the recent baseline changes into your package component, but audit will continue to flag synch10 until this information is updated.

The longhand way to update this information is to save your package version to an external library, checkout the newly updated baseline version over the top of what is in your package, and then re-introduce your "saved-off" changes.

If you have already accounted for the baseline updates, this is not necessary.

The facility discussed in this section allows you to dynamically update the information used by audit to flag Synch10s and confirm that there is, in fact, no version regression. When this process is used, the synch10 is no longer reported by audit.

You can access this feature in two ways:

You can select option 5 of the build checkout menu:

```
Checkout Options

Option ===>

Package . . . . . STEV001576

1 Baseline/Promo    From baseline/promotion libraries
2 Package           Package components
3 Release           Components from a release
4 Package           From package
5 Confirm           Confirm no regression
```

Or you can access it directly from the package list using action C5:

```
                Change Package List                 Row 1 to 8 of 8
Command ===>                                        Scroll ===> CSR

    Package     Sta     Install     Lvl     Type    Work request    Dept    Promote     Aud
Creator
    STEV001570  DEV     20221021    SMPL    PL/PE   REQ001          R&D     00
WSER58
    STEV001571  DEV     20221021    PART    PL/PE   REQ001          R&D     00           12
WSER58
    STEV001572  DEV     20221021    PART    PL/PE   REQ001          R&D     00           00
WSER58
    STEV001573  OPN                 CMPX    PL/PE   REQ001          R&D     00           12
WSER58
    STEV001574  DEV     20221021    SMPL    PL/PE   REQ001          R&D     00
WSER58
    STEV001575  DEV     20221021    SMPL    PL/PE   REQ001          R&D     00           12
WSER58
 C5 STEV001576  DEV     20221021    SMPL    PL/PE   REQ001          R&D     00
WSER58
    STEV001577  DEV     20211230    SMPL    PL/PE   ERO             DEV     00
WSER42
```

Both approaches lead to the version regression component selection panel:

```
                Baseline Version Regression Query Criteria
Command ===>

         Package: STEV001576

Component name . . . . ASMPGM04
Component type . . . . SRC

Enter "/" to select option
    / Mixed case
```

In this panel, enter the exact specification of the component you want to test for a potential version regression flag (note: wildcards/patterns are not supported). If no version regression is detected, the following message is displayed:

```
                  Baseline Version Regression Query C No version regression
Command ===>

        Package: STEV001576

Component name . . . . ASMPGM04
Component type . . . . SRC

Enter "/" to select option
/ Mixed case


+-----------------------------------------------------------------------+
| CMN7814I - No baseline version regression issues were found for this |
| component. |
+-----------------------------------------------------------------------+
```

Otherwise, a component action display is shown:

```
                Baseline Version Regression
Command ===>                                        Scroll ===> CSR

    Package: STEV001576 Libtype: SRC

    Component               -------------- Baseline ---------------
                            Package     Changed             User
    ASMPGM04              +             2020/11/16 08:25:00 WSER58
```

The three actions available to you are:

- S - Show meta-data
- C - Compare staging with baseline
- P - Process version regression override

You should make use of S and C (especially C) to convince yourself that there is no version regression before proceeding to use P.

Using the line command S displays something like this:

```
                Baseline Version Regression Component Details
Command ===>

    Component:  ASMPGM04 +
    Libtype:    SRC

                Baseline                Staging
    Package:                            STEV001576
    User:       WSER58                  WSER58
    vv.mm:      12.01                   13.05
    Changed:    2020/11/16              2021/12/15
                08:25:00                03:13:14
    Hash token: 9DD2EC00000001EC        9083466B000002E2

                                        Check out time
    Changed:    2020/11/16              2021/12/12
                08:25:00                01:01:00
    Hash token: 9DD2EC00000001EC        BBBBBBBB0000000B
```

This panel displays attributes for the selected component to allow you to compare the version in the current release with that in the baseline.

The top half of the panel shows a comparison, for the component, between what is in baseline now and what is in the package.

The bottom half compares the fields that audit uses to decide whether a version regression (synch10) error should be flagged. We can see a repeat of the timestamp and hash token of the baseline component as it exists right now. This is compared with the values of these attributes as they were when the component was checked out into the package. Any differences in these attributes indicate the baseline was updated after the component was checked out and cause audit to flag a synch10.

If you choose to confirm that all potentially regressed code has actually been included in the current package (option P from the component list), this mismatch will be reconciled.

Line command C presents a comparison panel showing what is about to be compared:

```
Baseline Version Regression Comparison
Command ===>

    Baseline version (SYSUT1):

    WSER58.BASE.SRC(ASMPGM04)                                               +

    Current package version (SYSUT2):

    CMNDEV.CMNL.STEV.#001576.SRC(ASMPGM04)                                  +
```

Press Enter.

SERCOMPR(Comparex) is used to compare the two members:

```
BROWSE     WSERWRK.WSER58.#DAC4223.#90B3CAC.OUTLIST
Line 0000000000 Col 001 132
Command
===>
Scroll ===> CSR
******************************************************* Top of Data
************************************************
    S E R C M P A R (MVS - 880 - 20201205) 2   TEXTONLY   WEDNESDAY DECEMBER 15, 2021
(2021/349)  08:32:42 PAGE 1
SYSUT1=WSERWRK.WSER58.#DAC4220.#3E5FF02.STG,SYSUT2=WSERWRK.WSER58.#DAC4220.#4425151.STG
    ASMPGM04 CSECT
00010000            O N E 1
    ********
00010100            O N E 2
++++++++<+++.++++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>++.++++8++++
+++++++++++++++++
D               COPY    AACPY02                                                 00010402
DIF     O N E 3 +
--------|---.----1----.----2----.----3----.----4----.----5----.----6----.----7----.----8------
I               COPY    CPYV8                                                   00010402
DIF     T W O 3 +
I               COPY    AACPY01                                                 00010403
DIF     T W O 4 +
I               COPY    CPY22104                                                00010404
DIF     T W O 5 +
I               COPY    BAS22104                                                00010405
DIF     T W O 6 +
++++++++<+++.++++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>++.++++8++++
+++++++++++++++++
                XR      15,15
00020000            O N E 4
                BR      14
00030000            O N E 5
                END
00040000            O N E 6
SER71I - END OF TEXT ON FILE SYSUT1
SER72I - END OF TEXT ON FILE SYSUT2
SER75I - RECORDS PROCESSED: SYSUT1(6)/SYSUT2(9),DIFFERENCES(1,0,3)
                            EXPLANATION - 1 RECORD DIFFER THAT SYNCHRONIZED TOGETHER
                                          0 RECORDS WERE CONSIDERED INSERTED ON SYSUT1
                                          3 RECORDS WERE CONSIDERED INSERTED ON SYSUT2
SER80I - TIME OF DAY AT END OF JOB: 08:32:42 - CONDITION CODE ON EXIT: 4
```

Finally, line action P starts the process to alter the meta-data so that the synch10 is no longer flagged. First, a confirmation panel is displayed:

```
                Baseline Version Regression Confirmation
Command ===>

    Component: ASMPGM04 + Libtype: SRC

    When this component was last checked out, the version identifying values
    for the component in baseline were

    Timestamp:  2021/12/12  01:01:00    Hash token: BBBBBBBB0000000B

    Since it was checked out the baseline version of this component has
    been updated and now has the following values

    Timestamp:  2020/11/16  08:25:00    Hash token: 9DD2EC00000001E

    To confirm that the current package component is 'up to date' with all
    changes made to the prior release simply press enter, else hit PF3 to
    cancel.
```

As shown on the panel, you can abandon this process using cancel or pf3/end. Pressing Enter applies the meta-data change.

7. Using Staging Versions

ChangeMan ZMF can save multiple versions of a package component that a developer might create in a staging library between the time the component is first added to a change package and the time the package is baselined.

About Staging Versions

You can use saved staging versions to:

- Examine changes you made to a component earlier in a project.
- Compare versions of a component to isolate coding changes made by a particular developer on a project.
- Compare versions of a component to isolate coding changes made at a particular time in a project.
- Browse the version of the component that is currently in the baseline library.
- Compare the version in the staging library, or any saved version to the component in the baseline library.
- Recover a complete version of a component that was saved earlier in the project.
- Merge multiple versions of a component with the M+R Option to create a new version.
- Protect against accidentally overlaying a component with checkout or stage from development.
- Protect against accidentally deleting a component from a package.

How Staging Versions Work

Some features of staging versions are available if the staging versions facility is installed. More features are available for package components in a library type if the facility is enabled for the library type by the application administrator. The merge facility of staging versions is only available if your company licensed the M+R Option.

Staging Version Installed

If the staging versions facility is installed, use the VC Staging Selection Code on any component on the **Stage: Package Components** panel to display the **Version Control** panel. The **Version Control** panel shows an STG version and a BAS version. These versions are pointers to the components in the staging library and baseline library, respectively. From the **Version Control** panel you can browse and view these versions, and you can compare them.

Staging Versions Enabled for Library Type

If your application administrator enabled the staging versions facility for a library type, you can save a new staging version whenever you change a component on the **Stage: Package Components** panel. Staging versions are stored as full copies in a compressed format in a VSAM file. Each version is labeled with an optional 35-character description.

Your administrator can make ChangeMan ZMF create a staging version every time a staging library member is changed, or ChangeMan ZMF can ask you if you want to overlay the component in the staging library without first creating a staging version.

From the **Version Control** panel, browse and view component versions. You can compare any two versions listed on the **Version Control** panel, including the STG and BAS versions. Your administrator can use the **Version Control** panel to delete saved staging versions. You can recover any saved staging version by copying it to a data set external to ChangeMan ZMF.

M+R Option Licensed

If your company licensed the M+R Option and installed the software, you can merge two to eight versions listed on the **Version Control** panel to create a new version of a component.

Rules, Restrictions, and Options

The following apply if the staging version facility is installed.

- The **Version Control** panel is displayed for any component on the **Stage: Package Components** panel.
- The **Version Control** panel always displays a STG version. The STG version points to the component in the staging library.
- The **Version Control** panel displays a BAS version if there is a component of the same name in the baseline library. The BAS version points to the component currently in the baseline library.
- The BAS version changes as the component in the baseline changes. It might not be the same as the component you checked out.

The following apply to library types for which staging versions are enabled.

- If your administrator configured the library type to ALWAYS save staging versions, the component in the staging library is saved as a staging version whenever you:
    - Save changes from an edit-in-stage session
    - Overlay the staged version with stage from development.
    - Overlay the staged version by checking out the component.
    - Delete the component.
- If your Administrator enabled staging versions with the PROMPT parameter for a library type, the **Save Previous Version** panel asks if you want to save the component in the staging library before it is changed, overlaid, or deleted. You are only allowed to skip creating a staging version if:
    - You are overlaying the component, not deleting it.
    - Your UserID matches the ID on the component in the staging library.
    - The component is not promoted.
- If you delete a component from your package, and then put the component back into your package with checkout or stage from development, all saved staging versions are available.
- An unlimited number of versions can be saved in staging versions.
- Staging versions can be deleted manually only by a ChangeMan ZMF Administrator.
- STG and BAS staging versions cannot be deleted manually.
- Staging versions can be browsed, recovered, compared, merged, and deleted throughout the package life cycle until the package stage libraries are deleted by housekeeping.

- All staging versions for a package are deleted automatically when package stage libraries are deleted for aging by ChangeMan ZMF housekeeping. Ask your administrator how long staging libraries are kept after a change package is installed.
- Staging versions can be saved only for components that are displayed on the **Stage: Package Components** panel. Staging versions are never saved for components like load modules that are created by staging build processes.

Creating Staging Versions

If the staging versions facility is enabled for a library type, staging versions panels display when you perform a ChangeMan ZMF function that changes a member in the staging library.

Adding A Component Change Description

When you copy a component to the staging library for a library type that has staging versions enabled, the **Component Change Description** (CMNCMP02) panel is displayed.

```
CMNCMPSW    History - ACPSRCCE.SRC      Row 1 to 2 of 2
Command ===>__________________________ Scroll ===> CSR

This component is included in the following packages:
       +---------------------------------------------------+
Packag | CMNCMP02 Component Change Description | User Release
ACTP00 | Command ===>                          | USER016
ACTP00 |                                       | USER016
  |  Package: ACTP000038                 |   |  Component:
ACPSRCCE                         + |
| Type: SRC                                    |
|                                              |
| Component change description (optional):     |
| Update program Id                            |
|                                              |
|                                              |
|                                              |
+----------------------------------------------------+
```

This table describes the fields on the **Component Change Description** panel.

Field	Description
Package	Displays the ID of the package containing the component being processed.
Component	Displays the component name.
Type	Displays the library type of the component.

Field	Description
Component change description	Enter description of the component you are saving to the staging library. The maximum length of the description is 35 characters. This field is optional.

> **Note**
>
> The description you enter on the Component Change Description panel applies to the new member in the staging library, not to the old member that is being saved as a staging version.

ChangeMan ZMF automatically generates a Component Change Description for BAS versions. It also creates descriptions for STG versions that are created by checkout. Examples of these generated Component Change Descriptions include:

```
Baseline version
Checkout from baseline 0
Checkout from promotion +10
Checkout from promotion +20
```

Replying to the Staging Version Prompt

If your Administrator specified PROMPT when staging versions were enabled for a library type, the **Save Previous Version** (CMNCMP03) panel is displayed to ask you if you want to save the staging library member before overlaying or deleting it.

```
CMNCMPSW        History - ACPSRCCE.SRC              Row 1 to 2 of 2
Command ===>______________________________________ Scroll ===> CSR

This component is included in the following packages:
       +---------------------------------------------------+
Packag | CMNCMP03 Save Previous Version                    | User Release
ACTP00 | Command ===>_____________________________________ | USER016
ACTP00 |                                                   | USER016
  |  Package: ACTP000038                               |   |  Component:
ACPSRCCE                             + |
       | Type: SRC                                         |
       |                                                   |
       | Enter "/" to select option                        |
       | / Save previous staging version                   |
       | _ Set Save Previous Version prompt off            |
       |                                                   |
       |                                                   |
       |                                                   |
       |                                                   |
       +---------------------------------------------------+
```

This table describes the fields on the **Save Previous Version** panel.

Field	Description
Package	Displays the ID of the package containing the component being processed.
Component	Displays the component name.
Type	Displays the library type of the component.
Save version	Select to save previous staging version. Omit if you do not want to save previous version.
Set save previous version prompt off	Enter any character to suppress the **Save Previous Version** panel until you exit the checkout or stage function. The last response entered in the Save Version field is given for each component as long as the panel is suppressed.

Managing Staging Versions

Staging versions are managed with the **Version Control** panel. Using this panel, you can browse, compare, recover, delete, and view staging versions.

Version Control Panel

On the **Stage: Package Component** panel, type **VC** on the line command for a component.

```
CMNSTG01        STAGE: GENL000006 Components            Row 1 to 10 of 10
Command ===>____________________________________________ Scroll ===> CSR

    Name    + Type Status   Changed          Procname    User    Request
___ GNLCPY00  CPY  ACTIVE   20150106 180258             USER016
___ GNLSRC1A  SRC  ACTIVE   20150106 195420 *RECOMP*    USER016
___ GNLSRC50  LOS  ACTIVE   20150106 211617 *RELINK*    USER016
___ GNLSRC99  SRC  INCOMP   20150116 102220             USER016 LOCKED
___ GNLSRS00  SRS  ACTIVE   20150106 194852 CMNCOB2     USER016
___ GNLSRS1B  SRS  ACTIVE   20150106 192057 *RECOMP*    USER016
___ GNLSRS1C  SRS  ACTIVE   20150106 192124 *RECOMP*    USER016
___ GNLSRS5A  SRS  ACTIVE   20150106 192150 *RECOMP*    USER016
___ GNLSRS5B  SRS  ACTIVE   20150106 192214 *RECOMP*    USER016
___ GNLSRS5C  SRS  ACTIVE   20150106 192228 *RECOMP*    USER016
******************************* Bottom of data ********************************
```

The **Version Control** panel is displayed.

```
CMNSIDV1                Version Control                 Row 1 to 3 of 3
Command ===>__________________________________________ Scroll ===> CSR

Type: CPY Package: GENL000006                       Staging view
Component: GNLCPY00                                                  +

    Level Prompt Changed          User    Description
___ STG   _____  2015/01/06 18:02 USER016 Test
___-001   _____  2015/01/06 17:19 USER016 Checkout from baseline0
___ BAS   _____  2015/01/01 21:43 USER016 Baseline version
******************************* Bottom of data ********************************
```

You can also access this panel in display mode from the **Source** and **Non-source** categories of the Query Package function. This table describes the fields on the **Version Control** panel.

Field	Description
Command	CANCEL: Cancel processing and exit the version control panel. Abbreviation: C
	REFRESH: Display updated information on this panel.
	LOCATE *level*: Scroll to row specified by *level*. For example: `LOCATE - 025`
	RESET: Clear all Line Command and Prompt fields.
	VIEW *parm*: Display a different view of the **Version Control** panel. Valid *parm* values include:
	STG: Display the component changed date, the UserID of the person who changed the component, and component change description. STG is the default view.
	SAVE: Display the staging version save date, the UserID of the person who changed the component, and the component change description.
	PROMO: Display the version promotion site, level, nickname and description.
	DIR: Display staging library directory size, date created, date changed and UserID.
	DIR2: Display staging library directory size, initial size, modified size, version number, modification level and UserID.

Field	Description
	The following *parm* values let you cycle through the **Version Control** panel views, which are presented in the sequence listed above.
	NEXT: Display the next view.
	PREV: Display the previous view.
Type	Displays the library type of the component.
Package ID	Displays the ID of the package containing the component being processed.
Component	Displays the component name.
View	Indicates what view of the **Version Control** panel is displayed. See the View command above. Displayed values:
	Staging view
	Save view
	Promotion view
	Directory view (1)
	Directory view (2)
Line Command	Enter the line command for the action you want to perform against each version.
	B: Browse a temporary copy of the staging version using ISPF Browse.
	C: Compare two staging versions and display a compare report. C must be entered on two versions listed on the **Version Control** panel.
	M: Merge 2 to 8 staging versions with the M+R Option. M must be entered on two to eight versions listed on the **Version Control** panel.
	D: Delete a staging version. (May only be used by administrators)
	R: Recover a staging version by copying the version to a user specified data set that is external to ChangeMan ZMF or to a ZMF package. Only saved staging versions (level -nnn) may be copied to a package.
	V: View a temporary copy of a staging version using ISPF View.

Field	Description
Level	Displays the level for each version managed by staging versions.
	STG Refers to the component in the package staging library.
	- Staging versions that were previously in the staging library,
	nnn listed in reverse chronological order by the changed or staged date. Level -001 is the most recently saved version.
	BAS refers to the component in the baseline library.
Prompt	Displays the status of the last request against the component version.
VIEW STG: The fields below are displayed in the Staging view of the **Version Control** panel.	
Version saved	Displays the date and time that the component version was last modified or staged.
ID	Displays the TSO UserID or JOB name that last modified or staged the component version.
Description	Displays the component change description that was automatically created by ChangeMan ZMF or that was entered by the developer when the component was staged.
VIEW SAVE: The fields below are displayed in the Save view of the **Version Control** panel.	
Version saved	Displays the date and time that the staging version was saved.
Stamp	Displays the date/time stamp on the saved version in the staging versions VSAM file.
Description	Displays the component change description that was automatically created by ChangeMan ZMF or that was entered by the developer when the component was staged.
VIEW PROMO: The fields below are displayed in the Promotion view of the **Version Control** panel.	
Site	Displays the site to which the version was last promoted when it was the STG version.

Field	Description
Lvl	Displays the promotion level to which the version was last promoted when it was the STG version.
Nickname	Displays nickname of the promotion level to which the version was last promoted when it was the STG version.
Description	Displays the component change description that was automatically created by ChangeMan ZMF or that was entered by the developer when the component was staged.
VIEW DIR: The fields below are displayed in the Directory view of the **Version Control** panel. These ISPF statistics were recorded in the Staging Versions VSAM file when the member in the staging library was saved as a staging version.	
Size	Displays the number of lines of the saved version.
Created	Displays the date and time that the component version was created.
Changed	Displays the date and time that the component version was last modified or staged.
ID	Displays the TSO UserID or JOB name that last modified or staged the component version.
VIEW DIR2: The fields below are displayed in the Directory2 view of the **Version Control** panel. These ISPF statistics were recorded in the Staging Versions VSAM file when the member in the staging library was saved as a staging version.	
Init	Displays the initial size (number of lines) of the component version.
Mod	Displays the number of lines changed of the component version.
VV.MM	Displays the ISPF version number and modification level of the component version.
Per	The file permissions in octal format.
F	File format 0-7
	0 Not specified
	1 Binary data
	2 New line
	3 Carriage return

Field	Description
	4 Line feed
	5 CR & LF
	6 LF & CR
	7 CR & NL

You can scroll through the list of versions and type line commands on one or more versions. After entering line commands, press Enter. Line commands are processed in the following order:

1. **B** (Browse), **D** (Delete), **R** (Recover) and **V** (View) commands are processed in the order that they appear in the version control list, top to bottom.
2. **C** (Compare) commands are processed next. The first version with a C line command (searching top to bottom) is processed as the “new” version, and the second version with a C line command is processed as the “old” version.
3. **M** (Merge) commands are processed last.

Recovering A Staging Version

Staging versions are recovered by copying the version to a data set that is external to ChangeMan ZMF or a package.

Copying a version to a data set

Note

You must have UPDATE security access to the target **TO DATASET**, and the UserID of the SERNET instance running ChangeMan ZMF must also have UPDATE authority.

⚠ **Caution**

Do not specify a package staging library as the target **TO DATASET**. If yourecover a staging version directly to a staging library, the overlaid member is not saved in staging versions, and package master records for the component are not updated.

1. On the **Version Control** panel, type **R** (Recover) on a line command for the version you want to recover.

```
CMNSIDV4                     Copy Component Version
Option ===>

Package . . . . . GENL000006
Component . . . . GNLCPY00                                             +
Type . . . . . .  CPY
Version . . . . . -001

1  Dataset        To dataset
2  Package        To package
```

2. Select option **1** to copy the version to a data set.

   The **Copy To Dataset** (CMNSIDV2) panel is displayed.

```
CMNSIDV2            Copy to Dataset Information
Command ===>_________________________________________________________

Package . . . . . GENL000006
Component . . . . GNLCPY00                                              +
Type . . . . . . CPY
Version . . . . . STG
To dataset . . . _______________________________________________________+
```

   This table describes the fields on the **Copy To Dataset Information** panel.

Field	Description
Package	Displays the ID of the package containing the component being processed.
Component	Displays the component name.
Type	Displays the library type of the component.
Version	Displays the staging versions level selected for recovery.
	STG Refers to the component in the package staging library.
	-nnn Staging versions that were previously in the staging library, listed in reverse chronological order by the changed or staged date. Level -001 is the most recently saved version.
	BAS Refers to the component in the baseline library.
To dataset	The fully qualified data set name to which the recovered version will be copied. This is a data set outside of ChangeMan ZMF.

3. Enter the name of the target data set and press Enter to execute the copy.

Copying a version to a package

> **Notes**
>
> Only saved staging versions (level -nnn) may be copied to a package.
>
> A component version can be copied to the same or a different package and component.

1. On the **Version Control** panel, type **R** (Recover) on a line command for the version you want to recover.

```
CMNSIDV4                   Copy Component Version
Option ===>

Package . . . . . GENL000006
Component . . . . GNLCPY00
   +
Type . . . . . . CPY
Version . . . . . -001

1 Dataset         To dataset
2 Package         To package
```

2. Select option **2** to copy the version to a package.

   The Copy to Package (CMNSIDV5) panel is displayed.

```
CMNSIDV5                           Copy to Package
Command ===>

Package . . . . . . . PETE000963
Component . . . . . . PJWA0000
     +
Type . . . . . . .    JCL
Version . . . . . . . -003

Target package . . .  PETE000963
Target component . .  PJWA0000
    +
Target type . . . . .JCL
Change description .

Enter "/" to select option
    / Confirm overlay
    Lock component
    / Save previous staging version
    Mixed case
```

This table describes the fields on the **Copy To Package** panel.

Field	Description
Package	The current change package.

Field	Description
Component	The name of the component to be copied.
Type	The library type of the component to be copied.
Version	The level number of the component version.
Target package	The package to copy the component to.
Target component	The name of the component to be copied to. If blank, thesource component name will be used.
Target type	The library type of the component to be copied to.
Change description	Describe the change being made to the component. If save staging versions is enabled, this description is displayed for each component version in the version control function.
Lock component	Select to lock the component in the target package. Omit to bypass locking the component. If the component is already locked in the target package, it will remain locked.
Confirm overlay	Select to view and respond to any confirmation required where the target file will be overlaid. Omit to bypass the confirmation panel.
Save previous staging version	Select to save the previous component version in the target package. This option applies only when the target library type 'Save staging version' option is set to 'PROMPT' and the component is already in the target package.
Mixed case	By default, all fields are folded to upper case. However, if the "Mixed Case" option is selected, the target component name is left as is.

3. Select which options you want to use and press Enter to execute the copy.

Merging Staging Versions

When you type **M** in the Line Command for two to eight versions on the **Version Control** panel and press **Enter**, each selected version is copied to a separate temporary sequential data set, and your ChangeMan ZMF session is transferred to the M+R Option.

With the M+R Option, you can merge and reconcile multiple versions of a component, or multiple files, to create a single new version.

What you see first in the M+R Option depends on whether you used the M+R Option previously with this ChangeMan ZMF instance, either through the staging versions merge function or through option **C (M+R Option)** on the **Primary Option Menu**.

- If the first M+R Option panel you see is the **Member Reconciliation Menu** panel, go to Member Reconciliation.

- If the first M+R Option panel you see is the **M+R Projects Repository** panel, go to New M+R Option Users.

New M+R Option Users

If you have never used the M+R Option before on this instance of ChangeMan ZMF, the following **M+R Projects Repository** (SEROJECT) panel is displayed the first time you merge staging versions.

```
SEROJECT            M+R Projects Repository
Command ===>

This panel displays only because the Merge+Reconcile installation has not been
completed successfully.
Press ENTER and ask your System Programmer to edit this SEROJECT panel.
You won't be able to use the project management facility - Option 5 PROJECT
and Option W Walk through M+R of the Primary Menu.

A SEQuential data set needs to be allocated with Record format FB, Record
length 80, Block size 6000, 1st extent 20 blocks, Secondary blocks 400.

All M+R users should have read/write access to this data set that will
contain information about the profiles of all team-oriented M+R Projects.
The fully qualified name of this data set needs to be set in the )INIT section
of this panel, as well as the fully qualified name of the M+R Samples dataset.

    M+R Projects DSNAME . . .
    M+R Samples DSNAME . . .
Each M+R project needs two profile data sets: one as an intermediate storage
("Work In Suspense" or WIS) and one for writing different M+R reports. The
preferred Unit Name and High Level Qualifiers for their allocation need to be
set in the )INIT section. Defaults are SYSDA and User logon ID, respectively.
    WIS / Print Unit Name . . SYSDA
    High Level Qualifiers . . CDFUSER
```

1. Read the Reconcile walkthrough for an overview of the M+R Option, and practice the line commands as instructed. To navigate the walkthrough panels, press **PF8** to page down and **PF7** to page up.
2. When you finish the tutorial, press **PF3**, or type **End** on the Command line and press **Enter**. Do this as many times as it takes to get back to the **Version Control** panel.
3. Maneuver to the following panel (from the primary option panel it’s C, M, 0 (that’s a zero) then 5 to display **Profile Data Sets** (SERDSNML) panel.

```
SERDSNML                Profile Data Sets
Command ===>_______________________________________________
WIS dsname . . . . . . . CMNTP.S6.MMR.WIS
Print dsname . . . . . . CMNTP.S6.MMR.PRINT
Export dsname . . . . . CMNTP.S6.MMR.EXPORT
Export DSORG . . . . . . PDS    (PDS, SEQ, PAN, LIB, or blank)
Export member mask . . . ______ (Mask to rename Base members on export)
LIB module name . . . .  ______ (LIBrarian update module name: LIBR, LIBRARY)
```

4. Fill in the appropriate data set names. If these datasets do not exist, ChangeMan ZMF will allocate them.
5. On the **Version Control** (CMNSIDV1) panel, select staging versions to merge by typing **M** on the Line Command for three to eight versions, then press **Enter**. The **Merge+Reconcile** (SER@PRIM) panel is displayed.

```
CMNSIDV1                Version Control                 Row 1 to 3 of 3
Command ===>__________________________________________ Scroll ===> CSR

Type: CPY Package: GENL000006                       Staging view
Component: GNLCPY00
                                                  +
     Level Prompt Changed          User    Description
M    STG   ______ 2015/01/06 18:02 USER016 Test
___  -001  ______ 2015/01/06 17:19 USER016 Checkout from baseline0
M    BAS   ______ 2015/01/01 21:43 USER016 Baseline version
******************************* Bottom of data ********************************
```

You are ready to proceed with Member Reconciliation.

Member Reconciliation

Follow this procedure to merge and reconcile staging versions.

If you have used the M+R Option previously with this ChangeMan ZMF instance, the **Member Reconciliation Menu** (SER@MAIN) panel is displayed after you use the M+R option.

~~~
```
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
```
~~~

1. On the **Member Reconciliation Menu**, choose Option **1 Versions** and press **Enter**. The **Versions** (SERCDFCM) panel is displayed.

```
SERCDFCM                            Versions
Command ===>____________________________________________________________________

    CMN Export package: GENL000006       Libtype: CPY    Export member: GNLCPY00
                                                      -Member---   Type Nickname Id
Base: USER001.#DACAC13.#2395231.STG _________________ _____________ SEQ STG_____ -_
                                          Last Member:
       or CMN appl/pkg: ___________       Libtype:___ _____________ ___ ________ __
Drv1: USER001.#DACAC13.#2395157.BAS _________________ _____________ SEQ BAS_____ A_
       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:___ _____________ ___ ________ __
```

Notice on the **Versions** panel:

- The temporary data set created for each selected staging version is assigned to a base or derivative version in the M+R Option. The oldest staging version is assigned to the **Base** version on this panel, and the newest staging version is the derivative (**Drvn**) with the highest number.
- The staging version **Level** is used as the version **Nickname** in the M+R Option.
- The version **ID** in the M+R Option is an alpha character assigned in ascending order from the oldest derivative to the newest. (There is no **ID** assigned to the M+R Option Base version.)

On the **Versions** panel, you can change the files assigned to the Base and derivatives, and you can add derivatives that reside in other sequential or PDS files. You can change the **Nicknames** and the **ID**s assigned to versions on this panel.

2. When you are satisfied with the entries on the **Versions** panel, press **PF3** or type **End** on the Command line and press **Enter**, to return to the **Member Reconciliation Menu**.
3. On the **Member Reconciliation Menu**, choose Option **2 Execute**. The **Reconcile** (CDFPANEL) panel is displayed.

```
CDFPANEL ------------------------ RECONCILE --------------- ROW 1 TO 20 O
Command ===>                                                  Scroll ===>
A:SOURCE#A
******************************** TOP OF DATA *****************************
000000 ********/* 2015/03/13 09:02:08 CMNTP.S6.V810.BASE.ACTP.SRC
000001          IDENTIFICATION DIVISION.
000002          PROGRAM-ID. ACPSRCCE.
000003
> A___          *PACKAGE ACTP000038 S6.V810
000004          *PACKAGE ACTP000076 S4.V712      ERO S4712COM RELATED COMM
000005          *PACKAGE ACTP000001 S6.V810
000006
000007            ENVIRONMENT DIVISION.
000008            CONFIGURATION SECTION.
< 0009            SOURCE-COMPUTER. IBM-370.
<>A___            SOURCE-COMPUTER. TRS-80.
< 0010            OBJECT-COMPUTER. IBM-370.
<>A___            OBJECT-COMPUTER. HP-3000.
000011
000012            INPUT-OUTPUT SECTION.
000013            FILE-CONTROL.
000014
000015            DATA DIVISION.
000016            WORKING-STORAGE SECTION
```

4. Merge and reconcile the Base and derivative versions. Press **PF1** for Help.
5. To end your reconciliation session and return to the **Member Reconciliation Menu,** press **PF3** or type **End** on the Command line and press **Enter**. When you exit your session, the results of your reconciliation are saved in your **WIS** file
6. If you exit from the M+R Option, the temporary data sets created by staging versions are deleted.
7. To resume the reconciliation effort from your WIS file:

    a. On the ChangeMan ZMF **Primary Option Menu**, select Option **C M+R**.

    b. On the **Merge+Reconcile** panel, choose Option **M Reconcile**.

    c. On the **Member Reconciliation Menu**, choose Option **4 Load**.

    d. On the **Member Reconciliation Menu**, choose Option **5 Edit**.

8. To export the results of your reconciliation into your M+R Option Export file, choose Option **6 Export** on the **Member Reconciliation Menu**.
9. To see the data set name of your Export file:

    a. Choose Option **0 Profile** on the **Member Reconciliation Menu** to display the **Profile** panel

    b. Choose Option **5 Datasets** to display the **Profile Data Sets** panel.

10. After you export the reconciled component into your Export file, use one of these methods to get the new version into your change package:

    a. Stage the component from development to overlay the staged version.

b. From inside an edit-in-stage session, delete all of the lines, then use the COPY command to copy all lines from the new version.

8. Recompiling Components

Recompile lets you perform build processing without checking out like-source components into your package. This process performs the minimum processing necessary to build like- load components from source, reducing the risk that you will inadvertently change source that does not need to be changed.

About Recompiling Components

Recompile executes an entire stage procedure, but the like-source input is obtained from a promotion or baseline library instead of from your package staging library. You use recompile when you want to include new versions of like-copy components in a build process where the like-source component is not changing.

An important secondary benefit of recompile is that component history and prior baseline versions reflect actual changes, not just multiple copies of the same component. Both component history and prior versions in baseline libraries are limited, and if you checkout, stage, and install the same source with no changes, real changes are eventually lost.

You can initiate recompile from your change package to add like-load components to your package. Recompile can be automatically invoked by the auto resolve feature of package audit to fix SYNCH errors. For example, if audit detects a like-source component in baseline that uses a copybook in your package, auto resolve initiates a recompile for the source to create a new load to add to your package.

Rules for Recompiling Source

These rules and restrictions apply to the recompile source function.

- You can recompile a like-source component that resides in a baseline library or a promotion library to create a new like-load in your package.
- When you recompile from a promotion library, you are assuming that the component in the promotion library will be installed by the time your package is ready to be installed.
- You can recompile a component that was recompiled previously in your package.
- You cannot recompile a component that was checked out to your package or staged from development. If a like-source component is in your package staging libraries, delete the component from the package before you recompile the component.
- Recompile can be executed in the online or in a batch job. Batch recompile releases your terminal while JCL for one or more recompile jobs is being created.
- If you recompile in batch mode, you can perform mass recompile to initiate multiple recompile jobs from one online operation.
- When you recompile a component that is already in other active packages, a SEND message is broadcast to the userid associated with the component in those packages. Multiple messages may be sent to the same user.
- To recompile a component, ChangeMan ZMF needs the component language, the name of the compile procedure skeleton, the compile and binder options, and the User Options. "Build Processing Controls" describes how this information is obtained.

Exit Programs for Recompile Source

Your ChangeMan ZMF installer or administrators may implement exit programs to alter the behavior ChangeMan ZMF. Ask your administrator if any of the following business rules apply to your application.

Dynamically allocate additional staging libraries based on component type, user options, and other data passed to the exit. Define relationships between library types so that when a component is deleted, components with the same name in related library types are also deleted. Prevent deletion of a component if it is promoted or if it generates components that are promoted. (Exit program CMNEX026)

> **Note**
>
> Components with the same name in related library types are deleted only if the TSO user ID of the user who staged the components matches the TSO user ID of the user who is deleting the like-named component.

Restrict the library types displayed on the valid library selection list for checkout, stage, browse baseline, browse compressed listing, compare, scan, scratch/rename, and relink functions. This exit effectively disables these functions for the specified library types. (Exit program CMNEX035)

Accessing Recompile Source Panels

Recompile functions are executed from the **Recompile Source** panel (CMNRCMP0).

```
CMNRCMP0                    Recompile Source
Option ===>__________________________________________________________
        blank Member selection list                 L List libraries containing component

Package . . . . . . . . . GENL000006
Work request . . . . . . 10010678
Department . . . . . . . IDD
Component name . . . . . ________________________________________________________________+
Source library type . . . SRC                       (Blank for list)
Library level . . . . . . 0                         (Baseline 0, Promotion 1 to nn)
Recompile mode . . . . . 1                          (1-Online, 2-Batch)
```

Use one of these methods to display the **Recompile Source** panel:

- Using the Menu Hierarchy:

  a. On the **Primary Option Menu**, select **1 Build**.

  b. On the **Build Option** panel, select **8 Recompile**.

- Using Direct Panel Access:

  a. On a ChangeMan ZMF panel, type **=1.8** in the **Command** or **Option** line and press **Enter**.

Using the Change Package List to Recompile Components

You can access the **Recompile Source** panel directly from the **Change Package List**.

On the **Change Package List** panel, type **RC** in the line command for a change package and press **Enter**.

Recompiling Source Components Online

Follow these steps to recompile one or more components from a baseline or promotion library into your change package using online processing.

1. Access the **Recompile Source** panel (CMNRCMP0).

```
CMNRCMP0                     Recompile Source
Option ===>_________________________________________________
    blank Member selection list          L List libraries containing component
Package . . . . . . . . . ACTP000028
Work request . . . . . . 10010678
Department . . . . . . . IDD
Component name . . . . . ACPSRC9* +
Source library type . . . SRC            (Blank for list)
Library level . . . . . . 0              (Baseline 0, Promotion 1 to nn)
Recompile mode . . . . . 1              (1-Online, 2-Batch)
```

This table describes the fields on the **Recompile Source** panel.

Field	Description
Option	Type **L** to display the **component.type Library List** panel that shows baseline and promotion libraries that contain the component specified in the **Component Name** and **Library Type** fields. You cannot use a pattern in the **Component Name** field if you use this option.
Package	Displays the package ID of the change package you are recompiling into.
Work request	Displays the Work request of the change package you are recompiling into.
Department	Displays the Department of the change package you are recompiling into.
Component name	Type the name of the component you want to recompile. Other options:
	Pattern: Build and display a filtered list of components from the origin library. You can select components to recompile from the list. See Building Lists Using Patterns.
	Note: Do not type * by itself unless you want to recompile all components in the library.
	Blank: Display a list of all components in the origin library. You can select components to recompile from the list.
	*: Recompile all components in the library.
Source library type	Type the library type of the components you want to recompile. Leave this field blank to display a list of library types to select from. You can select a like-copy library type to subsequently select a like -copy component to then select a like-source component that uses that like-copy component

Field	Description
Library level	Type an integer to indicate the baseline version or promotion level where you want to get the source to recompile. your Administrator may have defined Levels such as 10 20 30, so in that case a value of 1 would fail with a short message UNDEFINED PROMO LIBRARY.
	0: Recompile from the current or 0 level baseline library.
	1 to 99: Integers from 1 to 99 mean recompile a future version in a promotion library. The integers are level numbers in a promotion site/level. If there are multiple promotion sites with the level number you specify, the **Promotion Library Selection List** panel is displayed, where you select one of the listed promotion libraries.
	Blank: You can leave this field blank if you type **L** in the **Option** field.
	NOTE If you type **L** in the **Option** field, the **component.type Library List** panel is displayed that shows you the baseline and promotion libraries that contain the component specified in the **Component Name** and **Source Library Type** fields. You can select a library on this list to check out from regardless of the version or level that you specified in the **Library Level** field.
Recompile mode	Type 1 or 2 to determine whether recompile is performed in online mode or batch mode. See Online and Batch Execution Modes.
	1: Recompile components in online mode.
	2: Recompile components in batch mode.

Type your entries on the **Recompile Source** panel, selecting option **1-Online** for **Recompile Mode**, and press **Enter**.

2. On the **Recompile Source** panel, if you leave the **Component Name** field blank or use a pattern that selects multiple components, as shown on the last panel above, the **Recompile From: library** panel (CMNRCMP2) is displayed.

```
CMNRCMP2        Recompile Member List           Row 1 to 10 of 10
Command ===> ___________________________________ Scroll ===> CSR
                                                             ID
    Input library:
    CMNTP.S6.V810.BASE.ACTP.SRC

Name    Function   vv.mm Created    Changed           Size Init User
__ ACPSRC90        02.00 2002/05/07 2015/01/05 14:57  28      1 USER015
__ ACPSRC91        02.00 2002/05/07 2015/01/05 14:57  29     15 USER015
__ ACPSRC92        02.00 2002/05/07 2015/01/05 14:58  29     15 USER015
__ ACPSRC93        02.01 2002/05/07 2015/01/04 22:22  29     15 USER015
__ ACPSRC94        02.00 2002/05/07 2015/01/05 14:58  24      1 USER015
__ ACPSRC95        02.03 2002/05/07 2014/12/16 19:46  25     15 USER015
__ ACPSRC96        02.00 2002/05/07 2015/01/05 14:58  24     15 USER015
__ ACPSRC97        02.01 2002/05/07 2015/01/05 15:02  25     15 USER015
__ ACPSRC98        02.00 2002/05/07 2015/01/05 14:58  24     15 USER015
__ ACPSRC99        02.00 2002/05/07 2015/01/05 14:58  25      1 USER015
******************************* Bottom of data ********************************
```

This table describes the commands and line commands you can use on the **Recompile From: library** panel.

Field	Description
Command	Type one of the following commands, or leave the command line blank and type a line command next to a component name.
	REFRESH: Display updated information on this panel. Abbreviation: R
	SORT *heading*: Sort listed components by information under the specified column heading. Abbreviation: SO
	LONG: Display additional component information on a second line.
	SHORT: Suppress the additional component information displayed with the LONG command.
	CANCEL: Cancel the function and return to the previous panel. Abbreviation: C
	BROWSE *member*: Browse the specified component in the selected library. Abbreviation: B
	SELECT *member*: Select the specified component for recompile. Abbreviation: S
	HISTORY *member*: Display component history for the specified component. Abbreviation: H
Line Command	Type a line command to the left of the Name row:
	S: Select the component for recompile processing.
	B: Browse the component in the baseline library.
	H: Display component history.
	V: View a component in edit like mode.

Make your selection(s) on the **Recompile From: library** panel and press **Enter**.

> **Note**
>
> The next several panels are displayed in a series for the one component you specified in the Component Name field of the Recompile Source panel or for each component you selected on the Recompile From: library panel.

3. If you choose a component for recompile that is in another active change package, **Component Warning** panel is displayed.

```
CMNCMPSW            Component History                   Row 1 to 2 of 2
Command ===>__________________________________________ Scroll ===> CSR

Component: ACPSRC91.SRC

This component is included in the following packages:

Package    Sta Promoted vv.mm Last action       Size Procname User Release
ACTP000032 DEV                2015/01/12 21:29      *RENAME   USER016
ACTP000028 DEV          03.03 2015/01/14 11:13 00030 CMNCOB2 USER017
******************************* Bottom of data ********************************
```

This panel lists the other packages that contain the component that you want to recompile. Make sure that you are coordinating your work with the other developers who have checked out or staged the component into their packages.

If you want to abandon the recompile for this component, type **CANCEL** in the **Command** line and press **Enter**. Otherwise, press **PF3** or **Enter** to recompile the component.

```
CMNRCMPC            Confirm RECOMPILE Request
Command ===> __________________________________________
Package:        ACTP000028
Component name: ACPSRC91                                             +
        type: SRC
Language:       COBOL2
Changed date:   20150105
        time:   14:57
Procedure:      CMNCOB2
User:           USER015
Site:           BASELINE
Level:
```

4. If you continue the recompile process, the **Online Recompile Job Information** panel (CMNRCMP1) is displayed.

```
CMNRCMP1        ONLINE Recompile Job Information            HISTORY ASSUMED
Command ===>_________________________________________________

    Package:  ACTP000028     Status: DEV            Install date: 20150228
Component:    ACPSRC91                                                  +
Library type: SRC - Source for Programs to be Linked Executable
Dataset name: CMNTP.S6.V810.BASE.ACTP.SRC                               +
Language . . . . . . . COBOL2   (Blank for list)
Compile procedure . . CMNCOB2   (Blank for list; ? for designated procedure)
Compile parms . . . .  ____________________________
Pgm binder parms . . . ____________________________
Enter "/" to select option
__Db2 processing           _ Other Db2 options
__Display Other options    _ User Variables
__Suppress batch messages

Job statement information:
//USER015A JOB (SM-1IKF-SM),'ACTP28',
// CLASS=A,MSGCLASS=X,NOTIFY=USER015
//*
//* JOB TO RUN ONLINE RECOMPILE
```

This table describes the fields on the Online Recompile Job Information panel.

Field	Description
Component	Displays the component name in the baseline library.
Library type	Displays the component library type and library type description.
Dataset name	Displays the name of the library you are recompiling from.
	If there is a **designated procedure** for the component being recompiled, and the **Force Level** in that procedure is **2**, then the fields on the **Online Recompile Job Information** panel are presented in display mode. If the **Force Level** of the **designated procedure** is 1, you can restore the fields on this panel to the values required for freeze by typing ? in the Compile Procedure field.
Language	Type the source language. If you leave the **Language** field blank, the **Language Selection List** is displayed when you press **Enter**. If the **Language** field contains information when the panel is first displayed, the short ISPF message in the upper right corner of the panel tells you where the language was obtained:
	HISTORY ASSUMED Obtained from component history.
	DESIGNATED ENFORCED Obtained from a designated procedure.
	PRIOR ASSUMED Obtained from a prior use of this panel for another component during your current ChangeMan ZMF session.

Field	Description
Compile procedure	Type the compile procedure to be used to recompile the component. Compile procedures are members in the ChangeMan ZMF skeleton library that are processed by ISPF file tailoring to create stage JCL. If you leave the **Compile Procedure** blank, but there is only one compile procedure defined for the specified language, the **Compile Procedure** field is filled automatically when you press **Enter**. If you leave the **Compile Procedure** field blank, and if there are multiple compile procedures defined in administration for the specified language, the **Compile Procedure Selection List** panel is displayed when you press **Enter**. Type ? to restore **Language**, **Compile Procedure**, **Compile Parms**, **Binder Parms**, **Db2 Precompile**, and **User Options** from a **designated procedure** with **Force Level 1**.
Compile parms	Type compile options that are not already in effect through: System defaults. Compile Procedure skeletons. User Options. See **Other Options** below.
Pgm binder parms	Type binder options that are not already in effect through: System defaults for the compiler. Compile Procedure skeletons. User Options. See **Other Options** below.
Db2 processing	Indicates whether or not a Db2 pre-process or co-process is to be invoked during the compile.
Other Db2 options	Change or refer to values affecting the Db2 processing.
Display other options	Select to display the **Stage: User Options** panel before submitting the stage job. Otherwise do not display the **Stage: User Options** panel before submitting the stage job. Obtain User Options from component history or from a **designated procedure** with **Force Level 2**.
User Variables	Enter '/' to specify user variables used to pass information to skeleton file tailoring.
Suppress batch messages	Select to suppress the component activation messages which are normally issued by the batch job as it stages components into the requested package.
Job statement information	Type valid job card information.

5. Type your entries on the **Online Recompile Job Information** panel and press **Enter**. If you set the **Other Options** field to **Y**, the **Stage: User Options** panel is displayed. Type your entries on this panel and press **Enter**. A recompile job is submitted.
6. If you use the field **source library type** as a like-copy component then the process is similar, except you will be presented with a list of like-source components that use the like-copy component, and you can then select those components to be recompiled via **recompile member list** panel CMNCMPRC8

```
CMNRCMP8        Recompile Member List       Row 1 to 2 of 2
Command ===>______________________________ Scroll ===> CSR

Name       Library     Request
__ACPSRCCA SRC
__DUSR56DE SRC
******************************* Bottom of data *******************************
```

Mass Recompile Source Components in Batch

You can initiate multiple recompile jobs using mass recompile in batch if the following are true:

- The components have the same library type.
- The components are in the same baseline or promotion library.
- You want to use stage parameters from component history for all components, or you want to override component history for all components and use language, compile procedure, compile parms, binder parms, and other stage parameters from a recompile panel.

Mass recompile submits a batch job that file tailors recompile JCL and submits a recompile job for every like-source component selected for mass recompile.

Read the instructions in Recompiling Source Components Online. Then, follow these instructions to mass recompile multiple components from a baseline or promotion library into your change package.

1. On the **Recompile Source** panel (CMNRCMP0):

   Leave the **Component Name** blank or use a pattern to filter like-source component names.
2. Select option **2-Batch** for **Recompile Mode**. (Must be set to 2 to Mass recompile)
3. Set other fields according to instructions for recompiling online.
4. Press **Enter**.
5. The **Recompile From: library** panel (CMNRCMP2) is displayed. On this panel, use one of these procedures:

   Type S in the line command for every source component you want to recompile, then type **MASS** in the Command Line. Press **Enter** to mass recompile selected components.

```
CMNRCMP2             Recompile Member List      Row 1 to 10 of 10
Command ===>mass                                 Scroll ===> CSR
                                                            ID
Input library:
CMNTP.S6.V810.BASE.ACTP.SRC

Name    Function    vv.mm Created    Changed          Size    Init User
__ACPSRC90          02.00 2002/05/07 2015/01/05 14:57 28         1 USER015
__ACPSRC91          02.00 2002/05/07 2015/01/05 14:57 29        15 USER015
__ACPSRC92          02.00 2002/05/07 2015/01/05 14:58 29        15 USER015
s ACPSRC93          02.01 2002/05/07 2015/01/04 22:22 29        15 USER015
s ACPSRC94          02.00 2002/05/07 2015/01/05 14:58 24         1 USER015
s ACPSRC95          02.03 2002/05/07 2014/12/16 19:46 25        15 USER015
s ACPSRC96          02.00 2002/05/07 2015/01/05 14:58 24        15 USER015
s ACPSRC97          02.01 2002/05/07 2015/01/05 15:02 25        15 USER015
s ACPSRC98          02.00 2002/05/07 2015/01/05 14:58 24        15 USER015
s ACPSRC99          02.00 2002/05/07 2015/01/05 14:58 25         1 USER015
******************************* Bottom of data ********************************
```

6. Alternatively type **MASSALL** in the Command line and press **Enter** to mass recompile **every** component listed on the panel.

Caution

If a component you select for mass recompile is in another active change package, the Component Warning panel is not displayed.

The **Batch Mass Recompile Job Information** panel (CMNRCMP3) is displayed.

```
CMNRCMP3    BATCH MASS Recompile Job Information
Command ===>___________________________________________
        Package: ACTP000028     Status: DEV     Install date: 20150228

Library type:   SRC - Source for Programs to be Linked Executable
Dataset name:   CMNTP.S6.V810.BASE.ACTP.SRC                          +

Language . . . . . . .  ________________(Blank for list)
Compile procedure . .   ________________(Blank for list)
Compile parms . . . .   ________________
Pgm binder parms . . .  ________________

Enter "/" to select options
__Db2 precompile
__Precompile variables
__Display other options
__Continue after error
__Suppress history
__Suppress batch messages

Job statement information:
//  USER015A JOB (SM-1IKF-SM),'ACTP28',
//  CLASS=A,MSGCLASS=X,NOTIFY=USER015 ,TYPRUN=HOLD
//*
//* JOB TO RUN MASS Recompile
```

All fields on this panel must be filled, but the fields may not be used. See Build Information Search Order to see how the **Suppress History** field determines whether information you type on this panel is used when components are recompiled.

This table describes the fields on the **Batch Mass Recompile Job Information** panel.

Field	Description
Library type	Displays the library type and library type description.
Dataset name	Displays the name of the library you are recompiling from.
Language	Type the source language or press **Enter** to display the **Language Selection List**.
Compile procedure	Type the compile procedure to be used to recompile the components, or press **Enter** to display the **Compile Procedure Selection List** panel. (If there is only one compile procedure defined for the specified language, the **Compile Procedure** field is filled automatically.
Compile parms	If you set the **Suppress History** field to **Y**, type compile options that will be used for all mass recompiled components except those with Force Level 2 Designated Compile Procedures. Enter options that are not already in effect through: System defaults. Compile Procedure skeletons. User Options. See **Other Options** below.
Pgm binder parms	If you select the **Suppress History** field, type binder options that will be used for all mass recompiled components except those with Force Level 2 Designated Compile Procedures. Enter options that are not already in effect through: System defaults for the compiler. Compile Procedure skeletons. User Options. See **Other Options** below.
Db2 precompile	If you select the **Suppress History** field, select this field to determine whether all mass recompile jobs include a Db2 precompile step, unless otherwise specified for a component in a Force Level 2 Designated Compile Procedure.
Precompile variables	If you select the **Suppress History** field, select this field to determine whether the **Db2 Physical Subsystems** panel is displayed to set Db2 options for all mass recompile jobs, unless otherwise specified for a component in a Force Level 2 Designated Compile Procedure.
Display other options	If you select the **Suppress History** field, set this field to determine whether the **Stage: User Option** panel is displayed to set user options for all mass recompile jobs, unless otherwise specified for a component in a Force Level 2 Designated Compile Procedure.
Continue after error	If you select this option, the mass recompile job continues to submit more recompiles after an error is found. If this option isunselected, the job stops immediately when an error is found.
Suppress history	Select this to suppress component history and use the language, compile procedure, and other options entered on the Batch Mass Recompile Job Information panel (and subsequent optional panels), except for components with Force Level 2 Designated Compile Procedures. See Build Information Search Order for information on where language, compile procedure, and options are obtained when history is not suppressed.

Field	Description
Suppress Batch messages	Select this to suppress the SEND message normally broadcast at the completion of the stage job that tells you whether the component was staged successfully.
Job statement information	Type valid job card information.

7. Type your entries on the **Batch Mass Recompile Job Information** panel and press **Enter**. If you set the **Other Options** field to **Y**, the **Stage: User Options** panel is displayed. Type your entries on this panel and press **Enter**. A mass recompile job is submitted.

9. Staging a Component

Stage Component is where you edit and build package components to meet project requirements.

About Staging Components

When a like-source component is staged, a compile procedure skeleton is file tailored and a batch job is submitted to transform the source into the like-load component type specified in application library types for the source type.

When a like-other component of type *ttt* is staged, a compile procedure skeleton named CMN$$*ttt* is file tailored and a batch job is submitted. Your ChangeMan ZMF installer or administrator defines the build job steps and processes in the like-other skeleton.

When a like-PDS component is staged, there is no transform processing.

Depending on the restrictions established by your administrator, you may also use the stage function to:

- Copy a component into your change package instead of using checkout.
- Add a new component directly to a staging library from an edit session.
- Mass stage multiple components into your change package.

Rules for Staging Components

These rules and restrictions always apply to the stage component function:

- You can stage a component that is in CHECKOUT, INCOMP, UNFROZEN, or ACTIVE status.
- You cannot stage a component that is in FROZEN status.
- When a like-source or like-other component is staged, the status of the component is set to INCOMP (incomplete). If the stage job completes successfully, the component status is changed to ACTIVE (active). If the stage job fails for any reason, such as a compile error or JCL error, the component status remains INCOMP.
- To stage a component, ChangeMan ZMF needs the component language, the name of the compile procedure skeleton, the compile and link edit options, and the User Options. Build Processing Controls describes how this information is obtained.

Administration Settings for Stage Component Functions

Your global and application administrators make settings in ChangeMan ZMF administration that control how the stage component function works in your application. Ask your administrator if any of the following business rules apply to your application.

- If a component is in baseline, prohibit stage from development or require special security authorization. Require checkout to add a component into your package if the component is in a baseline library. (Application Administration Parameters: Checkout Enforcement Rule)
- Prohibit stage from development or require special security authorization to use this function. (Application Administration Parameters: Staging Restriction Level)
- If a member is staged from development, then edited in the package, save the changes into the member in the development library and into the member in the package staging library. (Application Administration Parameters: Eliminate Save To Personal Lib)
- If a member is checked out to a personal library, then edited in the package, save the changes into the member in the personal library and into the member in the package staging library. (Application Administration Parameters: Eliminate Save To Personal Lib)
- Prohibit stage from development when that would overlay a package component that is identified by another person’s user ID. (Application Administration: Overlay Prior Staged Module)
- Prohibit stage or discard edit-in-stage changes for package components checked out from baseline or promotion if the baseline component has changed since the checkout. (Application Administration: Validate Version During Staging)

Exit Programs for Stage Component Functions

Your ChangeMan ZMF installer or administrators may implement exit programs to alter the behavior ChangeMan ZMF. Ask your administrator if any of the following business rules apply to your application.

- Add custom processes before checkout and/or after stage. (Exit program CMNEX019)
- Dynamically allocate additional staging libraries based on component type, user options, and other data passed to the exit. Define relationships between library types so that when a component is deleted, components with the same name in related library types are also deleted. Prevent deletion of a component if it is promoted or if it generates components that are promoted. (Exit program CMNEX026)

> **Note**
>
> Components with the same name in related library types are deleted only if the TSO user ID of the user who staged the components matches the TSO user ID of the user who is deleting the like-named component.

- Bypass checkout enforcement rule depending on the component library type. (Exit program CMNEX030)
- Restrict the library types displayed on the valid library selection list for checkout, stage, browse baseline, browse compressed listing, compare, scan, scratch/rename, and relink functions. This exit effectively disables these functions for the specified library types. (Exit program CMNEX035)
- Call a preprocessor or a different editor. For example, SMART EDIT instead of ISREDIT. (Exit program CMNEX036)
- Call an edit macro such as one of the commercially available JCL validation tools. (Exit program CMNEX037)
- Display panel CMNEX042 and store information entered on the panel in the package master. (Exit program CMNEX042)

Accessing Stage Component Functions

You access stage functions from the **Stage Options** menu (CMNSTG00).

```
CMNSTG00            Stage Options
Option ===>___________________________________________

Package . . . . ACTP000003

1   Dev         Stage components from development libraries
2   Package     Process package components
3   Parms       Specify component list selection criteria
```

Use one of these methods to display the **Stage Options** menu:

- Using the Menu Hierarchy:

  a. On the **Primary Option Menu**, select **1 Build**.

  b. On the **Build Option** panel, select **6 Stage**.

- Using Direct Panel Access:

On a ChangeMan ZMF panel, type **=1.6** in the **Command** or **Option** line and press

Enter.

Using the Change Package List to Access Stage Functions

You can also access the stage options listed on the **Checkout Options** menu directly from the **Package List Parameters** panel (CMNLIST0).

On the **Change Package List** panel, type one of the following on the line command for a change package and press **Enter**.

- **S1** Stage components (Dev Lib list)
- **S2** Stage components (Package list)
- **S3** Specify package driven selection parameters

Choosing a Stage Option

There are two types of stage processes:

- Stage a component that is already in your change package.
- Stage a component into your package from a file.

  Use the **Stage Options** (CMNSTG00) menu to choose which stage process you want to execute. This panel also offers a filter function if you want to stage particular components in your change package.

- Select option **2 Package** to stage components already in your change package. See Staging from Package.
- Select option **3 Parms** to filter components in your change package before you process them as if you had chosen option 2 Package. See Filtering with Component List Parameters
- Select option **1 Dev** to stage components from a file into your package, and to perform mass stage. See Staging from Development

Staging from Package

Use Option **2 Package** on the **Stage Options** menu to stage and manage components that are already in your change package.

A typical use of the stage from package function is to edit a program in your package and then compile and link-edit the updated source:

1. On the **Stage Options menu**, choose Option **2** and press **Enter**. The **Stage: package Components** (CMNSTG01) panel is displayed.

```
CMNSTG01         STAGE: ACTP000027 Components       Row 1 to 5 of 5
Command ===>______________________________________ Scroll ===> CSR

Name        + Type  Status   Changed          Procname User     Request
__ACPSRCCA  SRC     CHECKOUT 20150306 115925  CMNCOB2  USER016
__ACPSRCCC  SRC     CHECKOUT 20150306 115928  CMNCOB2  USER016  LOCKED
__ACPSRCCE  SRC     CHECKOUT 20150306 115940  CMNCOB2  USER016  LOCKED
__ACPSRCD1  SRC     CHECKOUT 20150306 115943  CMNCOB2  USER016  LOCKED
__ACPSRC94  SRC     ACTIVE   20150115 164426  CMNCOB2  USER016
******************************* Bottom of data ********************************
```

This table describes the fields on the **Stage: package Components** panel:

Field	Description
COMMAND	Enter one of the following commands, or leave the command line blank and type a line command next to a component name.

Field	Description
	DCD: Display the first line of the component general description on an additional line.
	EDIT *member.typ*: Edit and stage a component in the package. If the component is not in the package, an empty edit session is provided, which is saved under the component name and library type specified in the EDIT command. Abbreviation: E
	REFRESH: Display updated information on this panel. Abbreviation: R
	SORT *heading*: Sort listed components under one of two field names or specified column headings. The sort direction can be specified by following the field name with A for ascending or D for descending. If not specified, the default sort direction for the field will be used. SORT DEFAULT restores the default sort, which is by NAME.
	LOCATE *member*: Locate a listed component by information in the last sorted column or by Name if the components were not sorted. Abbreviation: L
	LONG: Display additional component information on a second line.
	XLONG: Display long component name on the first line, and display additional component information on a third line.
	SHORT: Suppress the additional component information displayed with the LONG command and the long name displayed with the XLONG command.
	CANCEL: Cancel the function and return to the previous panel. Abbreviation: C
Line Command	Enter a line command to the left of the Name row:
	B: Browse component from personal or staging library.
	BA: Browse component activity file. (Set up by Administrator).
	BD: Browse component general description. (Set up by Administrator).
	BL: Browse the component listing.
	CB: Compare package component with its baseline version
	D: Delete component from staging library.
	E: Edit and stage component from personal or staging library.
	E: A Edit and stage component activity file.
	ED: Edit component general description.

Field	Description
	H: Display component history.
	LC: Lock component to disallow a checkout/stage by another user. See Locking and Unlocking Package Components
	R: Restage component from staging library. This is **not** the same as the Recompile function.
	SB: Stage component in batch - Free the terminal and submit a batch job to perform ISPF file tailoring to build JCL for the stage job; then submit the stage job. See Batch Stage Job Card
	SL: Display source-to-load relationship (source components only).
	ST: Stage one or multiple components in a single batch job submission. (Abbreviated S is also acceptable)
	UC: Unlock component (if locked) to allow for checkout/ stage. See Locking and Unlocking Package Components
	UL: Display Component UserID Work List (if enabled by administrator).
	UV: Update component user variables
	V: View component from personal or staging library.
	VC: Display Version Control panel (if enabled by administrator). See Staging Version Installed
	VL: View the component listing.
	To display the Valid Staging Selection Codes panel, which shows valid line commands and allows you to enter the command from that panel, type an invalid command or type nonsense characters such as / or ? or 99 in the line command for a component.
Name	Displays the name of the component.
Type	Displays the library type under which the component is checked out or staged in this package.
Status	Displays the status of the component in the package:
	ACTIVE: Component has been successfully staged
	CHECKOUT: Component has been checked out from baseline but not yet staged
	FROZEN: Component is frozen and may not be updated
	INACTIVE: Component name has been copied forward from another package, but the component has not been checked out to the staging library
	INCOMP: Compile job has not completed or has failed

Field	Description
	UNFROZEN: Previously frozen component is available for update
Changed	Displays the date and time that the component was last modified (and or staged).
Language	The language under which the source (SRC) or 'like-SRC' component was staged or checked out.
Procname	Displays one of the following for SRC or like-source components: The name of the last ISPF skeleton used to build JCL to stage the component. *RECOMP* if the component was recompiled. Displays *RELINK* for LOD or like-load components that were last built with relink.
User	Displays the User ID of the person who performed the last action on the component or the jobname if Batch Stage was used.
Request	Displays the last action performed on the component since the Stage: Package Components panel was first displayed.
The following information is displayed on a second line for each listed package component if LONG or XLONG is entered on the Command Line:	
Org	Displays the data set organization of the data set from which the member was staged. Valid options include:
	PDS: Partitioned dataset or PDSE
	SEQ: Sequential dataset
	PAN: PANVALET library
	LIB: LIBRARIAN Master
	ZFS: z/OS File System
	OTH: OTHER proprietary structures.
Input dataset name	Displays the data set from which the component was staged. If the component was relinked, the LCT library name is displayed.
Target lib	Displays target load library type for SRC and like-source components.
	If DCD is entered on the command line, the first line of the component general description is displayed on an additional line.

2. On the **Stage:** ***package*** **Components** panel, choose an action you want to perform against a listed component, enter the appropriate code in the **line command** for the component, and press **Enter**.

   For this example, enter **E** in the **line command** for an SRC component and press **Enter**.

If this dataset has been migrated, the following message is displayed that allows you to cancel the edit.

```
CMNSTG01              STAGE: FLO002605 Components              Row 1 to 9 of 12
Command ===> ______________________________________________ Scroll ===> CSR__

   Name         + Type  Status   Changed        Procname    User    Request
_E ASMS _______________________Dataset Migrated____________________
       |                                                          |   LOD
__ ASMS| Command ===> _________________________________________   |   LOD
__ ASMS|                                                          |   LOD
__ ASMS|     Dataset: CMNDEV.CMNL.FLOO.#002605.SP1                |
__ ASMS|                                                          |
__ ASMS|     Press Enter to recall, END to bypass.                |   LOD
__ ASMS|__________________________________________________________|
__ ASMSRC01       SRC   CHECKOUT 20150306 115943 CMNCOB2   USER016    LOD
__ ASMSRC02       SRC   ACTIVE   20150115 164426 CMNCOB2   USER016
******************************* Bottom of data ********************************
```

If this dataset has not been migrated, an edit session opens.

```
ISREDDE2    CMNTP.A00A2.#CE9B479.#0638571.ACPSRC94      Columns 00001 00072
Command ===>______________________________________________ Scroll ===> PAGE

****** ***************************** Top of Data ******************************
000001      IDENTIFICATION DIVISION.
000002      PROGRAM-ID. ACPSRC94.
000003
000004      *PACKAGE ACTP000027 S6.V810
000005      *PACKAGE ACTP000001 S6.V810
000006      *PACKAGE ACTP000016 S4.V710T19
000007      *PACKAGE ACTP000007 S4.V711
000008
000009      ENVIRONMENT DIVISION.
000010      CONFIGURATION SECTION.
000011      SOURCE-COMPUTER. TRS-80.
000012      OBJECT-COMPUTER. IBM-370.
000013
000014      INPUT-OUTPUT SECTION.
000015      FILE-CONTROL.
```

3. The edit session in the ChangeMan ZMF stage function is an ISPF edit session.

   The file being edited is a ChangeMan ZMF temporary data set. The stage function copies the component from the package staging library into a dynamically allocated temporary edit data set so that you are not editing directly in the staging library.

   Because you are working in a temporary data set, the ISPF SAVE function has no meaning and is disabled. If your administrator enabled the **EDIT STAGING RECOVERY MODE ON** parameter for your application, and your ChangeMan ZMF session ends abnormally, the temporary data set is saved, and is presented to you when ChangeMan ZMF is restarted and you connect again.

   See Editing Components in ChangeMan ZMF for details about recovering your work from an interrupted edit session in ChangeMan ZMF.

4. When you finish your edit changes, press **PF3** or type **End** in the **Command** line and press **Enter**.

   If the **COMPARISON REPORT** option is selected, the updated component in the temporary edit data set is compared to the original component in the staging library. A comparison report is displayed online.

```
ISRBROBA    CMNTP.A00A2.#CE9B479.#0638571.OUTLIST    Line 00000000 Col 001 080
Command __________________________________________________===> Scroll ===> PAGE

********************************* Top of Data **********************************
    S E R C M P A R (MVS - 871 - 20140828) 2 TEXTONLY FRIDAY MARCH
SYSUT1=CMNTP.S6.ACTP.STG6.#000027.SRC(ACPSRC94),SYSUT2=CMNTP.A00A2.#CE9B479.#063
            IDENTIFICATION DIVISION.
            PROGRAM-ID. ACPSRC94.
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>
D           *PACKAGE ACTP000027 S6.V810
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>
            *PACKAGE ACTP000001 S6.V810
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>
I           *PACKAGE ACTP000001 S6.V810
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>
            *PACKAGE ACTP000016 S4.V710T19
            *PACKAGE ACTP000007 S4.V711
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>
I           *PACKAGE ACTP000008 S4.V711
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>
```

5. After reviewing your edit changes in the comparison report, press **PF3** or type **End** in the **Command** line and press **Enter**. The **Comparison Report Disposition** (CMNSTG16) panel is displayed.

```
CMNSTG16            Comparison Report Disposition
Command ===>_______________________________________________
Report dataset name . . . 'CMNTP.A00A2.#CE9B479.#0638571.OUTLIST'
Report disposition . . . 3          1 - Print dataset and delete
                                    2 - Print dataset and keep
                                    3 - Delete dataset without printing
                                    4 - Keep dataset without printing

Job statement information if printing:
//USER016 JOB (0000),'CHANGE MAN',
//          CLASS=A,NOTIFY=USER016,MSGCLASS=X
//*
//*
```

   The most important function of this panel is to let you **cancel** the changes you made in your edit session if you do not like the results that displayed in the comparison report. You can also print the comparison report from this panel.

   If you want to keep your edit changes, use **PF3** or type **End** in the **Command** line and press **Enter**.

6. If the **CONFIRM OTHER REQUESTS** option is selected, and the component you edited is in another change package that has not been baselined, the **History- component.type** (CMNCMPSW) warning panel is displayed.

```
CMNCMPSW            History - ACTCPY00.CPY          Row 1 to 2 of 2
Command ===>______________________________________ Scroll ===> CSR

This component is included in the following packages:

Package     Sta Promoted vv.mm Last action  Size    Procname    User    Release
ACTP000008  DEV                2015/01/12 20:44     *RENAME     USER016
ACTP000007  DEV 04.01          2015/03/05 11:16 00006           USER016
******************************* Bottom of data ********************************
```

This panel lists the other packages that contain the component that you edited. You should make sure that you are coordinating your changes with the other developers who have checked out or staged the component into their packages.

If you decide to abandon the changes you made in your edit session, type **CANCEL** in the **Command** line and press **Enter**.

7. If staging versions are enabled for the library type being staged, the **Component Change Description** (CMNCMP02) panel is displayed, which allows you to enter an optional description of the component you are about to stage. You will also be able to provide change description for each component in the version control function within the package staging and query functions.

```
CMNSTG16        Comparison Report Disposition
Command ===>_________________________________________________________

Report dataset name . . . 'CMNTP.A00A2.#CE9B4B9.#81CAF70.OUTLIST'
____________________________________________________
Report | CMNCMP02 Component Change Description      |
       | Command ===>                               |
       |                                            |rinting
       | Package: ACTP000027                        |nting
       | Component: ACPSRCCA                      + |
Job st | Type: SRC                                  |
//US   |                                            |
//     | Component change description (optional):   |
//  |  Deleted lines of code                      |
//  |                                             |
       |                                            |
       |                                            |
       |____________________________________________|
```

This table describes the fields on the **Component Change** Description panel:

Field	Description
Package	Displays the ID of the package containing the component being processed.
Component	Displays the component name followed by the library type formatted as a file extension.

Field	Description
Type	Displays the component type
Component change description	Enter a description of the component you are saving to the staging library. The maximum length of the description is 35 characters. You may leave the description blank.

Note

The description you enter on the Component Change Description panel applies to the new member in the staging library, not to the old member that is being saved as a staging version.

For a full description of the staging versions function, see Using Staging Versions

8. If your Administrator specified PROMPT when staging versions were enabled for the library type, the **Save Previous Version** (CMNCMP03) panel is displayed to ask you if you want to save a member that might already be in the staging library before it is overlaid.

```
CMNSTG16                Comparison Report Disposition
Command ===>__________________________________________________

Report dataset name . . . 'CMNTP.A00A2.#CE9B4B9.#81CAF70.OUTLIST'
    |---------------------------------------------------|
Report | CMNCMP03 Save Previous Version                    |
    | Command ===>_____________________________________ |
    |                                                   | rinting
    | Package: ACTP000027                               | nting
    | Component: ACPSRC94                             + |
Job st | Type: SRC                                         |
//US   |                                                   |
//     | Enter "/" to select option                        |
//  |  / Save previous staging version                   |
//  |  __Set Save Previous Version prompt off            |
    |                                                   |
    |                                                   |
    |---------------------------------------------------|
```

This table describes the fields on the **Save Previous Version** panel

Field	Description
Package	Displays the ID of the package containing the component being processed.
Component	Displays the name of the component that will be saved as previous version.
Type	Displays the component type of the component being processed.
Save previous staging version	Select to create a staging version from the component
	Omit to replace the component in staging without saving it first as a staging version.

Field	Description
Set Save Previous Version prompt off	Select to suppress the Save Previous Version panel until you exit the checkout or stage function. The last response entered in the Save version field is given for each component as long as the panel is suppressed.

9. If you are staging a component that is not SRC or like-source, the system returns to the **Stage: package Components** panel, and the stage process is complete.
10. For SRC and like-source components, the stage process continues. The **Stage: Build** (CMNSTG04) panel is displayed.

```
CMNSTG04            Stage: Build                HISTORY ASSUMED
Command ===>___________________________________

    Package: ACTP000003     Status: DEV         Install Date: 20141231
Staged name . . . .  ACPSRCCA                                                 +
Library type . . . . SRC - Source for Programs to be Linked Executable
Dataset name . . . . CMNTP.S4.V711.BASE.ACTP.SRC                              +
Language . . . . . . . . . COBOL2 (Blank for list)
Compile procedure . . . . CMNCOB2 (Blank for list; ? for designated proc.)
Compile parms . . . . . . _________________________________
Pgm binder parms . . . . ._________________________________
Enter "/" to select option
__Db2 processing
__Other Db2 options
__Other options             / User variables
__Suppress messages
Job statement information:
//USER001 JOB (0000),'S6 V810',
//           CLASS=A,MSGCLASS=X,NOTIFY=USER001
//*
//*
```

Note

The component status is changed to INCOMP as soon as this panel is displayed. Even if you press PF3 or type Cancel in the Command line, the component status changes

This table describes the fields on the **Stage: Build** panel:

Field	Description
Package ID	The package ID of the current change package.
Status	The current status of this change package.
Install date	The date this change package is scheduled for installation.
Staged name	Displays the component name in the staging library.
Library type	Displays the component library type.

Field	Description
Dataset name	Displays the name of the staging dataset from which the component is to be staged; a personal or development library, or the temporary dataset created by ChangeMan if the module is to be checked out to staging.
	If there is a **Designated Procedure** for the component being staged, and the **Force Level** in that procedure is **2**, then the fields on the **Stage: Build** panel are presented in display mode. If the **Force Level** of the **Designated Procedure** is 1, you can restore the fields on this panel to the values required for freeze by typing ? in the Compile Procedure field.
Language	Enter the source language. If the **Language** field is blank, the **Language Selection List** is displayed when you press **Enter**. If the **Language** field contains information when the panel is first displayed, the short ISPF message in the upper right corner of the panel tells you where the language was obtained:
	HISTORY ASSUMED: Obtained from component history.
	DESIGNATED ENFORCED: Obtained from a Designated Procedure.
	PRIOR ASSUMED: Obtained from a prior use of this panel for another component during your current ChangeMan ZMF session.
Compile Procedure	Enter the compile procedure to be used to stage the component. Compile procedures are members in the ChangeMan ZMF skeleton library that are processed by ISPF File Tailoring to create stage JCL. Type ? to restore **Language**, **Compile Procedure**, **Compile Parms**, **Link Edit Parms**, **Db2 Precompile**, and **User Options** from a **Designated Procedure** with **Force Level 1**. If the **Compile Procedure** field is blank, and if there are multiple Compile Procedures defined in administration for the specified Language, the **Compile Procedure Selection List** panel is displayed when you press **Enter**. If the **Compile Procedure** is blank, but there is only one Compile Procedure defined for the specified Language, the **Compile Procedure** field is filled automatically when you press **Enter**. If the **Compile Procedure** field contains information when the panel is first displayed, the short ISPF message tells you where the procedure was obtained. See **Language** above.
Compile parms	Enter additional compile options that are not already in effect through:
	System defaults.
	Compile Procedure skeletons.
	User Options. See **Other Options** below.
Pgm binder parms	Any additional program binder (link edit) parms as appropriate.
Db2 Processing	Indicates whether or not a Db2 pre-process or co-process is to be invoked during the compile.
Other Db2 options	Select to display the **Db2 Physical Subsystems** panel to set additional options for Db2 components. Field **Db2 Processing** must be selected. Leave blank to omit display of Db2 Physical Subsystems panel.

Field	Description
Other options	Select to display the **Stage: User Options** panel before submitting the stage job. Leave blank to omit display of **Stage: User Options** panel before submitting the stage job. Obtain User Options from component history or from a **Designated Procedure** with **Force Level 2**. **Note:** If the component has an enforced designated procedure associated with it, the values defined in the designated procedure are used whether or not you choose to display the options panel(s).
User Variable	Enter '/' to specify user variables V01 through V10 on panel CMNUSV1 to pass information to skeleton file tailoring.
Suppress messages	Select to suppress the SEND message normally broadcast at the completion of the stage job that tells you whether the component was staged successfully. (The job submitted message for the stage job is also suppressed.) Broadcast a SEND message at the completion of the stage job to tell you whether the component was staged successfully.
Job statement information	Enter valid job card information for your site.

11. If you do not select **Other Options** field, stage job JCL is built by ISPF file tailoring from the **Compile Procedure** skeleton, and a stage job is submitted.

    If you select **Other Options** field, as many as four **Stage: User Options** (CMNUSR01, CMNUSR02, CMNUSR03, CMNUSR04) panels are displayed.

    Your ChangeMan ZMF installer and administrator may have modified these panels to fit your company needs. The example below shows only the first **User Options Part 1** (CMNUSR01) panel that is delivered with ChangeMan ZMF.

```
CMNUSR01            User Options Part 1
Command ===>_____________________________________________________
Name: GNLSRC99                                                      +
Type: SRC       Language: COBOL2
Compile only . . . . .             IMS DLITxxx entry . . .
CICS precompile . . .               Drop include stmts . .
Easytrieve object . .               User option 06 . . . .
User option 07 . . . .              User option 08 . . . .
User option 09 . . . .              User option 10 . . . .
User option 11 . . . .              User option 12 . . . .
User option 13 . . . .              User option 14 . . . .
User option 15 . . . .              User option 16 . . . .
User option 17 . . . .              User option 18 . . . .
User option 19 . . . .              User option 20 . . . .

Enter "/" to select option
_ Mixed Case
```

12. After setting use options, press **Enter**. ISPF file tailoring builds stage job JCL from the **Compile Procedure** skeleton. The stage job is submitted, and you are returned to the **Stage: package Components** (CMNSTG01) panel.

```
CMNSTG01        STAGE: ACTP000027 Components           Row 1 to 5 of 5
Command ===>__________________________________________ Scroll ===> CSR

Name        + Type Status   Changed         Procname User    Request
__ACPSRCCA    SRC  CHECKOUT 20150306 115925 CMNCOB2  USER016
__ACPSRCCC    SRC  CHECKOUT 20150306 115928 CMNCOB2  USER016 LOCKED
__ACPSRCCE    SRC  CHECKOUT 20150306 115940 CMNCOB2  USER016 LOCKED
__ACPSRCD1    SRC  CHECKOUT 20150306 115943 CMNCOB2  USER016 LOCKED
__ACPSRC94    SRC  INCOMP   20150115 164426 CMNCOB2  USER016 *BUILD
******************************* Bottom of data ********************************
```

The **Request** column for the component now indicates the action taken, and the component **Status** is changed to **INCOMP** until the build process completes successfully.

13. Upon completion, the stage job issues an MVSSEND message to indicate whether the build process was successful. Type **REFRESH** on the Command line of the **Stage: package Components** (CMNSTG01) panel, and the **Status** column is changed to **ACTIVE** if the stage job was successful. If build processing failed, the component status remains **INCOMP**.

```
CMNSTG01                STAGE: ACTP000027 Components                Row 1 to 5 of 5
Command ===>______________________________________________________ Scroll ===> CSR

Name        + Type Status   Changed         Procname User    Request
__ACPSRCCA    SRC  CHECKOUT 20150306 115925 CMNCOB2  USER016 LOCKED
__ACPSRCCC    SRC  CHECKOUT 20150306 115928 CMNCOB2  USER016 LOCKED
__ACPSRCCE    SRC  CHECKOUT 20150306 115940 CMNCOB2  USER016 LOCKED
__ACPSRCD1    SRC  CHECKOUT 20150306 115943 CMNCOB2  USER016 LOCKED
__ACPSRC94    SRC  ACTIVE   20150306 131649 CMNCOB2  USER016
******************************* Bottom of data ********************************
```

Filtering with Component List Parameters

You use option **2 Package** on the **Stage Options menu** to display all package components on the **Stage: package Components** panel.

Use option **3 Parms** and the **Component List Parameters** (CMNSTG12) panel to select certain package components for display on the **Stage: package Components** list.

You can also set indicators on the **Component List Parameters** panel to control features of checkout and stage processing.

1. On the **Stage Options menu**, choose option **3 Parms** and press **Enter**. The **Component List Parameters** (CMNSTG12) panel is displayed.

```
CMNSTG12                 Component List Parameters
Command ===>________________________________________________
Package: ACTP000050         Status: DEV         Install date: 20180405
Work request: Work Req WR2                          Department: Dept
Component name . . . . . . acpdoc60                                    +
Component type . . . . . . SRC
Language . . . . . . . . . _________
Component status . . . .   __ Active __ Checkout __ Frozen __ Inactive
                           __ Incomp __ Unfrozen
Changed from date . . . . ._________    (yyyymmdd)
            time . . . . . ____________(hhmmss)
Changed to date . . . . .  ____________(yyyymmdd)
            time . . . . . __________  (hhmmss)
Compile procedure . . . . ._________
User . . . . . . . . . . . _________
Display mode . . . . . . . S           (S-short, L-long, X-extra long)
Data Encoding . . . . . . .            (1-ASCII, 2-UTF-8)
Enter "/" to select option
/ Confirm component delete              / Confirm other requests
/ Display component user options        __Mixed case
/ Comparison report for edit           Text type . . . . $.
__Ignore recompiled components
```

The following table describes the fields on the panel.

Field	Description
Package	Displays the current package
Status	Displays the current status of the change package.
Install date	Displays the date when the package is proposed for installation in *yyyymmdd* format.
Work request	Work Request ID.
Department	Department.
Component name	Enter a component name or a pattern containing one or more characters followed by an * wildcard.
Component type	Enter a component library type or a pattern containing one or more characters followed by an * wildcard.
Language	Enter the source language associated with the component.
Component status	Enter a 1-character code to select a component status:
	1: Display components in ACTIVE status (successfully staged).
	2: Display components in INCOMP status (not staged successfully).
	3: Display components in CHECKOUT 3 status (checked out but not yet staged).
	4: Display components in INACTIVE status (copied forward from another package but not yet staged).

Field	Description
Change from - to dates and times	Specify a range of dates, and, optionally, times representing when the component was last modified (and/or staged).
Compile procedure	Enter the name of the compile procedure used to stage the component.
User	Enter a User ID to select components last changed by a particular user or a pattern containing one or more characters followed by an * wildcard.
Display mode	Enter one of the desired display options listed below. This value can be overridden at the command line of the package-driven list.
	Short: Display of the package-driven list, but exclude data set organization, source data set, and the target load library for SRC or Like-SRC components. Abbreviated S.
	Long: Display of the package-driven list with all data set information. Abbreviated L.
	Extra-long: Display of the package-driven list with all data set information and the component name in a longer field. Abbreviated X.
Confirm component delete	Select option to display **Confirm Delete Request** panel before deleting a component from the package. Leave blank to delete a component without displaying the **Confirm Delete Request** panel.
Confirm other requests	Select to display and respond to any ERROR or WARNING panel. This setting will also display concurrent development warnings if members staged into the package also exist in other ChangeMan ZMF packages. Do not select to bypass any ERROR or WARNING panel; do not display concurrent development warnings. The settings made for this parameter will be active for any checkout or stage function.
Display component user options	Allow display of component user option panel(s) for non-source components. This allows user data to be associated with a component for later use in exits or XML services. If the 'Enable component user variables' global option is not enabled, this setting is ignored. Note: Component user variables for source components are accessible during the build process.
Mixed case	By default, all fields are folded to upper case. However, if this option is selected, the component name is left as is. Whichever case you enter will then be processed.
Comparison report for edit	Select to display a summary of the changes made when you exit from an edit-in-stage session. ChangeMan ZMF will compare the original member contents with the member as it is when the 'SAVE' was issued. The comparison report may be printed for hard copy storage and reference. Do not select to bypass displaying the comparison report when a user exits from an edit-in-stage session. The setting used for this parameter will be active for any edit-in- stage session.
Ignore recompiled components	Recompiled components have the same compile procedure as a source component, so you cannot filter them out using the procedure name. This option allows them to be ignored.

Field	Description
Text type	Enter the text type to be compared if Comparison Report is selected.
	.(period): The first four records are analyzed to identify the target language to determine what kind of text compare should be done. If you specify a period, changes to spacing and commas are ignored during the comparison as described for the values below. Enter $. to force a position-by-position comparison and to flag lines as changes even if the only differences are the positions of spaces and commas.
	COBOL: Positions 7 through 72 are compared. Spaces and commas are ignored during the comparison so that any lines where only spaces or commas are changed are not flagged as modified. Thus, if you insert one line of code in the program, for example, and subsequent lines of code are thereby renumbered, only the inserted line of code is identified as a change. Renumbered lines are not identified as modified in the report.
	PANEL REPORT SCRIPT: Positions 1 through 80 are compared. No characters are ignored during the comparison.
	ALC,BAL,JCL, PASCAL,C,CLIST,FORTRAN,PL/1,PL/I,PLI,NATURAL,REXX,RPG: Positions 1 through 72 are compared. Spaces are ignored during the comparison so that any lines where only spaces are changed are not flagged as modified.
	You can precede any of the above values with the $ character. If you do, spaces and commas are included in the comparison and lines where only spaces or commas are in different character positions are flagged as changed in the report. For example, specify $COBOL if you want to identify lines as changed even if the positions of only spaces or commas differ. The setting made for this parameter will be active for any comparison report run online in ChangeMan ZMF.
Display component user options	Select to allow display of component user option panel(s) for non- source components. This allows user data to be associated with a component for later use in exits or XML services. If the 'Enable component user variables' global option is not enabled, this setting is ignored. Note: Component user variables for source components are accessible during the build process.
Mixed case	Select to process input exactly as you type it, upper and lower case.

2. Type information on the **Component List Parameters** panel to select components for the **Stage: package Components** (CMNSTG01) panel.

    - Selection fields on this panel are combined with logical AND operators to select components from the package.
    - All fields on this panel are optional except for the **Short or Long Display** field.
    - Blank selection fields are treated as "select all."
    - If you set no selection criteria, all package components display.

3. When you finish entering information on the **Component List Parameters** panel, press **Enter**. The **Stage: package Components** (CMNSTG01) panel is displayed.

```
CMNSTG01            STAGE: GENL000008 Components        Row 1 to 3 of 3
Command ===>___________________________________________ Scroll ===> CSR

Name            + Type Status Changed           Procname User   Request
__ CPY001         CPY  ACTIVE 20150115 165613            USER016
__ GNLCPY1X       CPY  ACTIVE 20150112 172032            USER016
__ GNLSRC1A       SRC  ACTIVE 20150113 224848 CMNCOB2    USER016
******************************* Bottom of data ********************************
```

4. Proceed with your stage processing as described in Staging from Package.

Staging from Development

On the **Stage Options menu**, choose Option **1 Dev** to stage a component that resides in a library other than your package staging libraries. The component you stage from development is first copied into a package staging library. Then, if the component is like- source, build processing is initiated.

Three administration entries can restrict your use of stage from development:

- Checkout Enforcement can require you to use checkout to get a component into your package if it already exists in an application baseline library.
- Staging Restriction can prevent you from staging a component that does not already reside in your application baseline.
- Overlay Prior Staged Module can prevent you from overlaying an existing package component that was last staged by someone else.

Ask your administrator if checkout is enforced and if you are permitted to create new components in ChangeMan ZMF.

You can use stage from development to process multiple components in a mass stage, but all components in a mass stage must be staged in the same library type.

1. On the **Stage Options menu**, choose Option **1 Dev** and press **Enter**. The **Stage: From Development** (CMNSTG02) panel is displayed.

```
CMNSTG02                    Stage from Development
Command ===>_____________________________________________________
        Package: GENL000008         Status: DEV             Install date: 20180228

ISPF Library:
    Project . . . . __________
    Group . . . . . ______________________________
    Type . . . . .  __________
    Member . . . .  __________      (Blank/pattern for list; * for all members)

Other partitioned, sequential or zFS dataset:
    DSN . . . . . . ______________________________________________ +
Org . . . . . .     _______         (PDS, Seq, PAN, LIB, Oth, zFS)

Library type . . . . . ______       (Blank for list)
Stage name . . . . . . ____________________________________________+
Stage mode . . . . . . 1            (1-Online, 2-Batch)

Enter "/" to select option
    / Confirm request           ___ Expand zFS subdirectories
    / Lock component            ___ Display component user options
__ Extract Stored Procedure from Db2 catalog
```

This table describes the fields on the **Stage: From Development** panel.

Field	Description
ISPF Library	Use this group of fields to type the name of an ISPF library containing the component to be staged. An ISPF library is a cataloged partitioned data set with a three-level data set name in this format: project.group.type The library name can also be entered in the fields under the heading **Other Partitioned, Sequential, or zFS Dataset**.
Project Group Type	Enter the three nodes of an ISPF Library name in these fields. **Member** - Type one of the following:
	Blank: Display a library member selection list after the **Stage: From Development** panel.
	Member name: Stage one library member.
	Pattern: Display a partial member selection list after the **Stage: From Development** panel. A member name pattern contains one or more characters followed by an * wildcard
	*: Mass stage all members in the library.
	Caution: If you enter * in the **Member** field, every member in the library will be staged into your change package.
Other, partitioned, sequential or zFS dataset	Use this group of fields to enter the name of a partitioned or sequential data set or an zFS path that contains the component to be staged. An ISPF library name can be entered here or in the fields under the heading **ISPF Library**.

Field	Description
DSN	Enter the name of a partitioned data set, PDSE, sequential data set, or zFS directory or path. Quotes work the same as in your standard ISPF session. If the file is in zFS, the **DSN** must specify (end with) a directory, not a file name. If the file is a sequential data set, you must type a component name in the STAGE NAME below. If the file is a PDS, PDSE, a CA Panvalet library, or a CA Librarian master, type the member name in parentheses as follows:
	No member: Display a library member selection list after the Stage: From Development panel.
	Member name: Stage one library member.
	Pattern: Display a partial member selection list after the Stage: From Development panel. A member name pattern contains one or more characters followed by an * wildcard
	*: Mass stage all members in the library.
	Caution: If you type (*) for a member name, every member in the library will be staged into your change package.
Org	Enter the type of file from which you are staging:
	PDS: Partitioned dataset or PDSE
	SEQ: Sequential dataset
	PAN: CA Panvalet library
	LIB: CA Librarian master
	ZFS: z/OS File System
	Oth: Other, such as ROSCOE
	Blank: ChangeMan ZMF determines whether the file organization is PDS or SEQ by the presence of a member name and/or a STAGE NAME
Library type	Enter the library type under which you will stage the component. Leave this field blank to display the **Library Type Selection List** panel.
Stage name	Enter the name under which you want to stage the component, if: You are staging a component from a sequential file. You want to change the name of a member staged from a PDS, CA Panvalet, or CA Librarian library. This field is often cleared when an error message is displayed on the **Stage: From Development** panel. If you type information in **Stage Name**, and panel processing stops with an error message, you may need to reenter Stage Name.
Stage mode	See Online and Batch Execution Modes
	1. Stage in online mode.
	2. Stage in batch mode.
	Enter '1' to submit multiple members online, however, online execution may tie up your terminal with JES messages.

Field	Description
	Enter '2' to allow staging of several components in background from a single job submission.
Expand	If you specify **zFS** in the **ORG** field, a component selection panel is displayed after the **Stage: From Development** panel. The **Expand** field determines what is displayed on that selection list.
	Select to list files in the last directory path you entered in the **DSN** field. Also, list subdirectories and paths in the last directory in the **DSN** path. If not selected it will not list subdirectories or paths in the last directory in the **DSN** path.
Confirm request	Activate confirmation panels for checkout, stage, and component delete that permit you to cancel a function after you have initiated the process but before it is executed. There are two confirmation panels for Stage From Development: Display a list of other packages that contain the component being staged. Display a warning if the stage process will overlay a component already resident in the staging library. N Suppress confirmation panels in checkout, stage, and component delete.
Lock component	Select to lock component being staged from development. See Locking and Unlocking Package Components Omit to stage the component without locking it. This only applies to the current package. Omit to bypass locking the component. If the component is already locked in the package, it will remain locked.
Display component user options	Allow display of component user option panel(s) for non-source components. This allows user data to be associated with a component for later use in exits or XML services. Only applies to the current package. Omit to bypass locking the component. If the component is already locked in the package, it remains locked.
Extract Stored Procedure from Db2 catalog	Connect to Db2 (v11 and later) and extract a native SQL Stored Procedure definition directly into a ZMF package component.

2. Type the information you want on the **Stage: From Development** panel and press **Enter**.

3. When you stage from development, you are allowed to copy a component into your package that already resides in your staging library. However, before the copy is executed, an **Overlay Warning** (CMNSTGWP) panel is displayed.

```
+----------------------- Stage Warning ----------------------+
| CMNSTGWP                                                    |
| Command ===> ______________________________________________|
|                                                             |
| Staging Member:                                             |
|   GNLSRC1A                                                  |
| Will overlay USER106 version.                               |
|                                                             |
+-------------------------------------------------------------+
```

If you want to overlay the component in your staging library, press **Enter**. Otherwise, press **PF3** or type **End in** the **Command** line. Press **Enter** to stop the stage from development process.

4. When a component is copied to a staging library by stage from development, ISPF statistics in the staging library directory are set as follows.

ISPF Stat	Staged From Sequential File	Staged From PDS(E)
VV.MM	Set to 01.01	Set to 01.01
CREATED	Set to the stage date	Copied from the directory of the originating library
CHANGED	Set to the stage date/time	Set to the stage date/time
SIZE	Set to the number of records in the file	Copied from the directory of the originating library
INIT	Set to the number of records in the file	Copied from the directory of the originating library
ID	Set to the TSO ID of the person who staged the component	Set to the TSO ID of the person who staged the component

5. If you stage from a sequential file, or if you specify a member name on the **Stage: From Development** (CMNSTG03) panel, only one component is staged, and the rest of the stage from development process is like a stage from package. For a description of the rest of the stage process.
6. If you stage from a PDS(E) and you specify * for the member name, mass stage is invoked. Go to Mass Stage.
7. If you stage from a PDS(E) but do not specify a member on the **Stage: From Development** (CMNSTG03) panel, or if you specify a name pattern for a member name, the **Stage: From - library** is displayed to give you a list of components.

```
CMNSTG03            Stage from Development      Row 1 to 2 of 2
Command ===>__________________________________ Scroll ===> CSR

From dataset name
USER016.USER.SRCLIB

Name    Function vv.mm Created          Changed          Size  Init  User
GNLSRC1A         01.01 2002/05/07       2015/01/05 01:22 00028 00022 USER016
GNLSRC99         01.01 2002/05/07       2015/01/05 00:54 00022 00015 USER016
******************************* Bottom of data ********************************
```

This table describes the fields on the **Stage From - library** panel.

Field	Description
COMMAND	Enter one of the following commands, or leave the command line blank and type a line command next to a component name.
	BROWSE *member*: Browse the specified member in the development library. Abbreviation: B
	HISTORY *member*: List component history for the specified member. Abbreviation: H

Field	Description
	STAGE: Mass Stage members selected on this panel. See Mass Stage
	STAGE ALL: Mass Stage all members on the component list. Selection is not required. See Mass Stage
	REFRESH: Display updated information on this panel. Abbreviation: R
	SORT *heading*: Sort listed components by information under the specified column heading.
	LOCATE *member*: Locate a listed component by information in the last sorted column or by Name if the components were not sorted. Abbreviation: L
	CANCEL: Cancel selections and exit this panel. Abbreviation: C.
Line Command	Enter a line command to the left of the Name row. If the **Stage From -** ***library*** panel extends over multiple screens, line commands on the current screen are processed before the next screen is displayed.
	B: Browse the member in the development library.
	H: List component history for the member
	S: Select a member for stage
	M: Select for mass stage. This line command is equivalent to using the S line command plus the Stage command at the top of the panel.
	V: View a component in edit like mode
	If S and M line commands are entered on the same panel, all selections are treated as M and are mass staged.

8. If you select one or more components on the **Stage From - library** panel that are not SRC or like-source:

   - Selected components are copied to the staging library.
   - The component status for staged components is set to **Active**.
   - The **Function** field on the **Stage From - library** panel is set to ***Stage**.

   Stage from development is complete. Press **PF3** or type **End** in the **Command** line. Press **Enter** to return to the **Stage: From Development** panel.

9. If you select SRC and like-source components on the **Stage From - library** panel, the next steps depend on how you select those components:

   - If you type **S** in the **line command** for one or more components, and if you do not type **STAGE** in the **Command** line, each selected member is staged as an individual component. Go to Step 6 in Staging from Package for a description of the steps each component must pass through to complete the stage process.
   - All other selection methods on the **Stage: From library** panel invoke mass stage.

Mass Stage

```
Mass stage:

- Stages multiple components under the same library type.
- Submits a separate stage job for each component.
- Can force all components to be staged using the same procedure and parameters, or it can
use procedures and parameters found in component history and in **Designated Procedures**.
- Uses the same staging versions **Component Change Description** for all staged components.

You invoke mass stage when you do any of the following for components in stage from
development:
- Enter **\*** for the member name on the **Stage: From Development** (CMNSTG02) panel.
- Select members on the **Stage: From library** (CMNSTG03) panel with **M** in the **line
command**.
- Select members on the **Stage: From library** (CMNSTG03) panel with **S** in the **line
command** and enter **STAGE** in the **Command** line.
- Enter **STAGE ALL** in the **Command** line of the **Stage: From library** (CMNSTG03)
panel.
```

1. In this example, the library outside of ChangeMan ZMF where components reside is coded in the **Partitioned, Sequential, Panvalet, Librarian, Other or zFS Dataset** field, and the **Library Type** field is set.

```
CMNSTG02                Stage from Development
Command ===> __________________________________________
    Package: GENL000008     Status: DEV     Install date: 20150228

ISPF Library:
Project . . . . ___________
Group . . . . . __________________________
Type . . . . .  ___________
Member . . . .  ___________     (Blank/pattern for list; * for all members)

Other partitioned, sequential or zFS dataset:
DSN . . . . . . . USER016.USER.SRCLIB                                     +
Org . . . . . . ______          (PDS, Seq, PAN, LIB, Oth, zFS)
Library type . . . . . SRC      (Blank for list)
Stage name . . . . . . __________________________________________________ +
Stage mode . . . . . . 1        (1-Online, 2-Batch)

Enter "/" to select option
/ Confirm request       __Expand zFS subdirectories
/ Lock component        __Display component user options
__Extract Stored Procedure from Db2 catalog
```

2. Press Enter to display the **Stage From - library** (CMNSTG03) panel.

```
CMNSTG03           Stage from Development                 Row 1 to 2 of 2
Command ===> _____________________________________________ Scroll ===> CSR

From dataset name
USER016.USER.SRCLIB

  Name     Function vv.mm Created    Changed          Size  Init  User
M GNLSRC1A CANCEL   01.01 2002/05/07 2015/01/05 01:22 00028 00022 USER016
M GNLSRC99 CANCEL   01.01 2002/05/07 2015/01/05 00:54 00022 00015 USER016
******************************* Bottom of data ******************************
```

3. On the **Stage From - library** (CMNSTG03) panel, select components for mass stage. In this case, M is entered in the Line Command for components to be staged. Then press Enter to display the **Stage: Mass Build** (CMNSTG05) panel where you set the build process procedure and options.

```
CMNSTG05                Stage: Mass Build
Command ===>___________________________________________

    Package: GENL000006     Status: DEV         Install date: 20150228
Library type . . . . SRC - Source for Programs to be Linked Executable
Dataset name . . .   USER016.USER.SRCLIB
Language . . . . . . . . _________ (Blank for list)
Compile procedure . . . .  _________ (Blank for list)
Compile parms . . . . . .  _________
Pgm binder parms . . . . . ____________________________________
Enter "/" to select option ____________________________________
    Db2 precompile
    Precompile variables
/ Other options
__ Suppress history
__ Suppress messages
/  Increment jobname
Job statement information:
    //USER016 JOB (0000),'CHANGE MAN',
    //            CLASS=A, NOTIFY=USER016,MSGCLASS=X
//*
//*
```

This table describes the fields on the **Stage: Mass Build** panel.

Field	Description
Package	Displays the current package ID.
Status	Displays the current status of the change package.
Install date	Displays the date when this package is proposed for installation. (yyyymmdd).
Library type	Displays the SRC or like-source library type specified on the **Stage: From Development** panel.
Dataset name	Displays the name of the file specified on the **Stage: From Development** panel.
NOTE:	

Field	Description
	There is no component history for a component, or
	The **Suppress History** field below is selected and there is no **Designated Procedure** with **Force Level 2** for a component.
Language	Enter the source language. If the Language field is blank, the **Language Selection List** is displayed when you press **Enter**.
Compile procedure	Identify the compile procedure to be used to stage the component. Compile procedures are members in the ChangeMan ZMF skeleton library that are processed by ISPF file tailoring to create stage JCL. If the Compile Procedure field is blank, and if there are multiple Compile Procedures defined in administration for the specified Language, the **Compile Procedure Selection List** panel is displayed when you press Enter. If the Compile Procedure is blank, but there is only one Compile Procedure defined for the specified Language, the Compile Procedure is filled automatically when you press **Enter**.
Compile parms	Enter compile options that are not already in effect through: System defaults. Compile Procedure skeletons. User Options. See **Other Options** below.
Pgm binder parms	Enter pgm binder parms that are not already in effect through: System defaults for the compiler. Compile Procedure skeletons. User Options. See **Other Options** below.
Db2 precompile	Select to include a Db2 precompile step in the stage job.
	Omit to bypass Db2 precompile step in the stage job.
Precompile variables	Select to display the **Db2 Physical Subsystems** panel to set additional options for Db2 components. Field **Db2 Precompile** must be selected.
	Omit to bypass display of **Db2 Physical Subsystems** panel.
Other options	Select to display the **Stage: User Options panel** before submitting the stage job.
	Omit to bypass display of **Stage: User Options** panel before submitting the stage job.
Suppress history	Select to suppress component history and use the procedure and parameters on this panel (and on the **Stage: User Options** panel, if displayed) to stage all components except those with a **Designated Procedure** that has a **Force Level 2**.
	Obtain stage procedure and parameters for each component in this order:
	1. Designated Procedure with Force Level 2.
	2. Component history.
	3. Entries on this panel (and on the **Stage: User Options** panel, if displayed).
**NOTE:	
Suppress message	Suppress the SEND message normally broadcast at the completion of each stage job that tells you whether the component was staged successfully. (The job submitted message for the stage job is also suppressed.)

Field	Description
	Broadcast a SEND message at the completion of each stage job to tell you whether the component was staged successfully.
Increment jobname	Determines whether job name suffixes for staging jobs submitted by mass state are incriminated to allow multiple jobs to run at the same time. The job name in the **Job Statement Information** field must consist of the submitter's TSO userid followed by a one character alpha suffix, A-Z. Job name suffixes 0-9, @, #, and $ are not incriminated. There is no relationship between this option and **JOB NAME INCREMENT OVERRIDE** in ZMF administration.
	Increment job name suffix, allowing multiple jobs to run simultaneously.
	Do not increment the job name suffix. Force jobs to single thread.
Job statement information	Enter valid job card information for your site.

When you finish entries on the **Stage: Mass Build** panel, press **Enter**.

4. Stage job JCL for each mass staged component is created in a separate file tailoring started task and submitted through the internal reader.

Staging Like-Other Components

Your ChangeMan ZMF installer or administrator can create custom build processes from the standard compile and link-edit ISPF skeletons that are delivered from Serena. However, some custom processes cannot meet the restrictions of "like-source" processing, so a "like-other" process is created.

You initiate stage processing for like-other components in the same way that you initiate stage processes for like-source, by using one of the Stage Options. When you stage a like- other component, the **Stage: other Components** panel is displayed.

```
CMNSTG09                Stage: OTH Components
Command ===> ___________________________________________________

Package: ACTP000043         Status: DEV         Install date: 20171130
Dataset name: CMNTP.SERT7.BASE.ACTR.OTH
Library type: OTH

Job statement information:
===> //USER2391 JOB (X170,374),'S6.V814',
===> // CLASS=A,MSGCLASS=X,NOTIFY=USER239
===> //*
===> //*

Enter "/" to select option
__User variables
/ Other options
```

Inputs allowed on this panel are the **Job Statement Information** and the two options, **User variables** and **Other options**. If these are selected, then if you selected **User variables**, then panels CMNUSR01-4 are displayed for inputs. If you selected **Other options**, then panel CMNUSV1 is displayed for inputs. When you press **Enter**, a batch job is created and submitted.

Staging LOD Components

You initiate stage processing for stage from recfm=U development library components in the same way that you initiate stage processes for like-source, by using one of the Stage Options. When you stage a component from a recfm=U development library, the **Stage from Development** panel (CMNSTG11) is displayed.

```
CMNSTG11            Stage from Development              Row 1 to 35 of 35
Command ===>___________________________________________   Scroll ===> CSR

From dataset name
USER015.ISPLLIB

Module       Function Length ttr        Alias-of AC R/M A/M Setssi --Attributes--
__CLS       *STAGE    0000B8 000016              00 24 24
__COLOURS             0013F0 000012              00 24 24 610A8BE2 RN RU
__CURPOS    CANCEL    000498 000011              00 24 24
__DELINKI             00FE38 000005              00 ANY 31         RN RU RF
__DWNSPDSR            000478 000006              00 24 24          RN RU RF
...
```

The line commands available are

S To stage a member into ChangeMan staging libraries

B To browse a member

H Display component history for a member

M To select members for selective mass-stage

V To view a member

ISPF converts a View request to a browse for recfm=u libraries.

Batch Stage Job Card

When you set field **Stage Mode** to **2-Batch** on the **Stage: From Development** panel, or when you use line command **SB** on the **Stage: package Components** panel to stage a component that is not like-source, the series of stage panels concludes with the **Batch Stage Job Statements** (CMNSTG19) panel.

```
CMNSTG19                            Batch Stage Job Statements
Command ===>______________________________________________________________

Package: GENL000009

Job statement information:
//USER016      JOB (0000),'CHANGE MAN',
//                CLASS=A,NOTIFY=USER016,MSGCLASS=X
//*
//*

Enter "/" to select option
__  Suppress batch messages
```

This table describes the fields on the **Batch Stage Job Card** panel.

Field	Description
Package	The package ID associated with the stage.
Job statement information	Enter the appropriate Job card information to submit a background job to stage component(s) into this package. For details, see Batch Job Statement Information
Suppress batch messages	Select this option to suppress the component activation messages which are normally issued by the batch job as it stages components into the requested package.

When you have typed the information on the panel, press **Enter** to process the batch stage.

Adding New Components

Depending on the staging restriction rule that your administrator defined, you can add new components to your application baseline libraries by adding components to a change package, then installing the package.

You can add a component to your change package with either stage from package or stage from development:

- You can copy a component that you have developed in a library or sequential data set outside of ChangeMan ZMF by using stage from development.
- You can clone a component in a ChangeMan ZMF staging or baseline library by typing the library and member name in the **Data Set Name** field on the **Stage: From Development** panel, and then type a new component name in the **Stage Name** field.
- You can create an empty edit-in-stage session in a package that already contains components by typing the **Edit** command in the **Command** line of the **Stage: package Components** panel. The format of the **Edit** command is:

EDIT *member.typ*

where *typ* is a valid library type in your application.

You can fill the empty edit-in-stage session by manually coding the component, or you can use the ISPF edit COPY command to copy in any file or member to which you have read access.

Locking and Unlocking Package Components

Component lock grants dynamic ownership of package components to users. If you lock a package component, other users are prevented from performing these functions against that component in your package:

- Edit
- Stage
- Delete from package
- Overlay with stage from development
- Overlay with another checkout

  Select the **Lock Component** field on the **Checkout, Stage: From Development** or

  Stage: package Components panel.

  Note: Components can only be unlocked by the user who locked it, or by an administrator. When an administrator unlocks a component that was locked by someone else, the ID in component history is not changed, and the person who locked the component remains as the owner.

You can unlock components from the stage-from-package function by using the **UC** line command on the **Stage package Components** panel.

10. Rebinding Components

Rebind lets you perform build processing without checking out like-source components into your package. This process performs the minimum processing necessary to build like- load components, reducing the risk that you will inadvertently change source or load that does not need to be changed.

- About Rebinding Components
- Rules for Rebinding Load
- Accessing Rebind Load Panels
- Rebinding Load Components

About Rebinding Components

Rebind executes only the program bind portion of a stage procedure to rebuild a composite load module composed of a statically linked main program and subroutines. Rebind obtains object and load modules from a SYSLIB concatenation of staging, promotion, and baseline libraries. If one exists, a link edit control member (with the same name as the component being rebound) is obtained from either a staging or baseline library.

An important secondary benefit of rebind is that component history and prior baseline versions reflect actual changes, not just multiple copies of the same component. Prior versions in baseline libraries are limited, and if you checkout, stage, and install the same component with no changes, real changes are lost from the oldest versions in baseline libraries.

You can initiate rebind from your change package to add like-load components to the package. Rebind may also be invoked by the auto resolve feature of package audit to fix some SYNCH errors.

In order to ensure that rebinds rebuild composite load modules correctly it is important to separate constituent NCAL subroutines from the final composite executables by using different library types for each (object code subroutines are forced to have a different library type owing to their differing DCB requirements).

Then, when rebuilding a composite executable, you can rebind from the top level NCAL (or object code) component library type to the composite executable library type. This will not only ensure that manual rebinds pick up the latest versions of subroutines (either via automatic library call or via link edit control card direction or a combination of both) but it will also allow audit autoresolve to correctly rebuild your executables should they be found to be out of synch.

Note that rebinding a fully resolved composite executable achieves nothing. Unless directed by REPLACE link edit control cards the program binder will not disturb a fully resolved load module.

Rules for Rebinding Load

These rules and restrictions apply to the rebind load function.

- You can rebind a like-load component that resides in a baseline library or a staging library.
- You can rebind a component that was rebinded previously in your package.
- You cannot rebind a component was checked out to your package, staged from development, or recompiled into your package.
- When you rebind a component that is already in other active packages, a SEND message is broadcast to the userid associated with the component in those packages.

Exit Programs for Rebind Load

Your ChangeMan ZMF installer or administrators may implement exit programs to alter the behavior of ChangeMan ZMF. Ask your administrator if any of the following business rules apply to your application.

> Restrict the library types displayed on the valid library selection list for checkout, stage, browse baseline, browse compressed listing, compare, scan, scratch/rename, and relink functions. This exit effectively disables these functions for the specified library types. (Exit program CMNEX035)

You can also use the HLL exit capability via the BULD fnctional area of the *HLLX Getting Started Guide*.

Obtaining Build Information

ChangeMan ZMF needs the language, procedure, link edit options, and user options to create the rebind job. Because the options used to rebind a component are independent from those used during the compile and link process, rebind uses the latest component history from rebind, not stage or recompile. The latest component history is defined as the first found in the this search order.

1. Current package.
2. Baselined package with the latest package baselined date/time.
3. Deleted/archived package with the latest component history changed date/time.
4. Package in motion with the latest package component history changed date/time.

5. Initial history record in component history.

Accessing Rebind Load Panels

Rebind functions are executed from the **Rebind Load Modules** panel (CMNRLNKR if licensed for ERO, otherwise CMNRLNK0).

```
CMNRLNKR                    Rebind Load Modules
Command ===> _____________________________________________________

Package . . . . . . . . . ACTP000016
Work request . . . . . . 1907D90
Department . . . . . . . IDD

Rebind from . . . . . . . S          (S(tage),B(aseline),R(elease))
Input library type . . . LOD        (Blank for list)
Target library type . . . LOD        (Blank for list)
Member name . . . . . . . ________+ (Blank/Pattern for list)

Enter "/" to select option
__ Specify release area
__ LCT member list
```

Use one of these methods to display the **Rebind Load Modules** panel:

- Using the Menu Hierarchy:

  a On the **Primary Option Menu**, select **1 Build**.

  b On the **Build Option** panel, select **9 Rebind**.

- Using Direct Panel Access:

  On a ChangeMan ZMF panel, type **=1.9** in the **Command** or **Option** line and press **Enter**.

Using the Change Package List to Rebind Load Modules

You can access the **Rebind Load Modules** panel directly from the **Change Package List**.

On the **Change Package List** panel, type **RL** in the line command for a change package and press **Enter**.

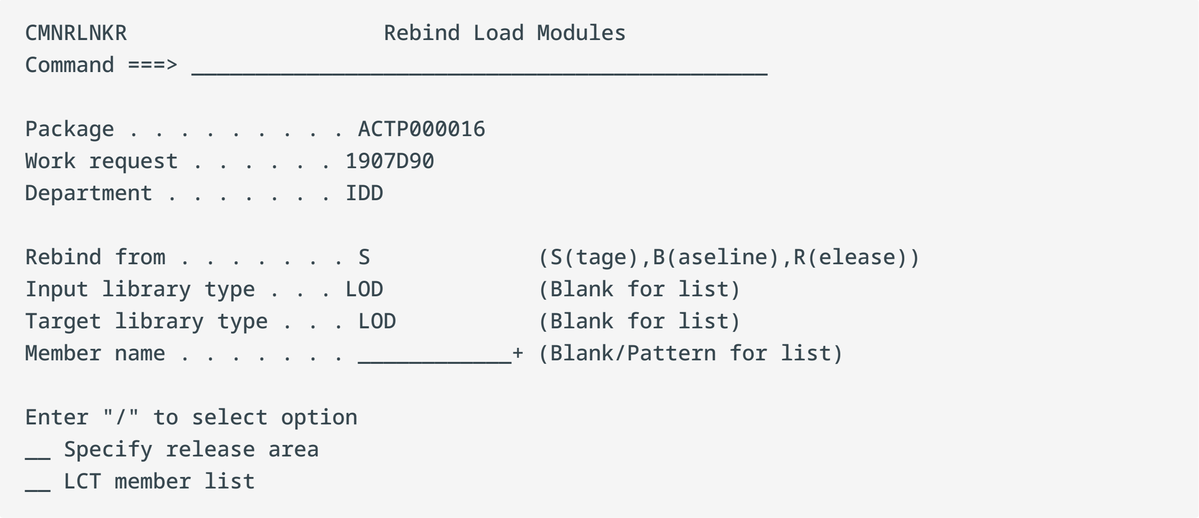
Rebinding Load Components

A rebind that you initiate through the Rebind Load Modules panel is driven by either:

- A link-edit control member in a package staging library or baseline library.
- A load module in a package staging library or a baseline library.

Follow these steps to rebind a load component into your change package.

1. Bring up the **Rebind Load Modules** panel (CMNRLNKR).



```
CMNRLNKR                    Rebind Load Modules
Command ===> _____________________________________________

Package . . . . . . . . . ACTP000016
Work request . . . . . . 1907D90
Department . . . . . . . IDD

Rebind from . . . . . . . S             (S(tage),B(aseline),R(elease))
Input library type . . . LOD            (Blank for list)
Target library type . . . LOD           (Blank for list)
Member name . . . . . . . ____________+ (Blank/Pattern for list)

Enter "/" to select option
__ Specify release area
__ LCT member list
```

This table describes the information on the panel.

Field	Description
Package	Type the ID of the package you want to rebind a load module into.
Rebind from	Type the location of the link-edit control member or the like- load member to be rebound.
	S: Package staging library
	B: Baseline library
Input library type	Type the library type of the module to be rebound from. Leave blank to display a selection list. The input library type can be like-object, like-NCAL, or like- load.
Target library type	Type the library type of the module to be rebound. Leave blank to display a selection list. The output library type must be like-load.
Member name	Type the name of the link-edit control member or the like-load member to be rebound. Type a name pattern or blank to display a selection list. See Building Lists Using Patterns.
Specify release area	Select if you want to select a specific release area to process. This will only process against the one selected release area library. Omit if you want to use a consolidated list of release area libraries.

Field	Description
LCT member list	Select this to specify whether this rebind is driven by a link- edit control member or a load module, otherwise the bind is driven by a load module.

Make your entries on the **Rebind Load Modules** panel and press **Enter**. If you specified a full Member Name, to go step 3.

2. If you left the **Member Name** field blank, or if you typed a pattern, a selection list is displayed showing members in a staging or baseline library for link-edit control members or load modules

```
CMNRMLST             Library Member List                    Row 1 to 1 of 1
Command ===>______________________________________________ Scroll ===> CSR

Input library:
CMNTP.S6.V810.BASE.GENL.LCT

Name         +  Function Created            Changed             Size       User
__ GNLSRC25                2015/01/01         2015/01/01 20:32    00000003   USER015
s  GNLSRC50                2015/01/01         2015/01/01 22:32    00000005   USER015
******************************* Bottom of data *******************************
```

Type **S** or **R** to rebind, or type **B** to browse the member, **H** to display component history, or **V** to view the member in edit like mode.

3. The **Rebind Job Information** panel (CMNRLNK1) is displayed.

```
CMNRLNK1                Rebind Job Information
Command ===> ______________________________________________________

    Package: GENL000006        Status: DEV            Install date: 20180228

Member name:        GNLSRC50                                            +
Dataset name:       CMNTP.S6.V810.BASE.GENL.LCT                         +
LCT member list:    YES Input library type: LOS Target library type: LOD

Language . . . . . . . . COBOL2     (Blank for list)
Compile procedure . . . CMNCOB2     (Blank for list)
Pgm binder parms . . . .____________________________________

Enter "/" to select option
__ Db2 processing       _ Other Db2 options
__ Other options        _ User variables
__ Suppress batch messages

Job statement information:
//USER015A JOB (SM-1IKF-SM),'GENL06',
//          CLASS=A,MSGCLASS=X,NOTIFY=USER015
//*
//*     JOB TO RUN REBIND
```

4. Proceed from here as normal to submit the rebind job and complete the process.

11. Utility Requests

A ChangeMan ZMF utility request is used to rename or delete an existing component in baseline and production libraries.

About Utility Requests

The utility request facility is designed to rename or delete components in production and baseline libraries. These requests can be the only items in a change package, or they can be part of a change package with staged components. ChangeMan ZMF processes these requests in the package installation process, *after* installing and rippling staged components.

When you create a utility request to scratch a like-source component, all components with a source-to-load relationship to that component are also deleted. For example if you create a utility request to scratch like-source component A, then like-load or like NCAL component A is also be scratched, as well as the compressed listing from the build process for A.

When you create a utility request to rename a component, only that component is renamed. A request to rename a like-source component has no effect on a like-load that was created in build processing from the like-source component.

Use the **Utility: Rename/Scratch Options** menu to access these capabilities.

Accessing Utility Request Functions

Rename and scratch utility functions are accessed through the **Utility: Rename/Scratch Options** menu.

```
CMNUTL00                Utility: Rename/Scratch Options
Option ===> _____________________________________________________

Package . . . . . ACTP0000028

1 Baseline        Select baseline components to scratch or rename
2 Package         Process scratch or rename requests in package
```

Use one of these methods to display the **Utility: Rename/Scratch Options** menu:

- Using the Menu Hierarchy:

  a. On the **Primary Option Menu**, select **1 Build**.

  b. On the **Build Option** panel, select **4 Utility**.

- Using Direct Panel Access:

  On a ChangeMan ZMF panel, type =1.4 in the Command or Option line and press Enter.

Using the Change Package List to Access Utility Requests

You can access the options listed on the **Utility: Rename/Scratch Options** menu directly from the **Change Package List**.

On the **Change Package List** panel, type one of the following in the line command for a change package and press **Enter**.

- **UA** Rename and Scratch components (package driven)
- **UB** Rename and Scratch components (baseline driven)

Creating Utility Requests from Baseline

Execute the following steps to create utility requests from baseline.

1. Select Option **1 Baseline** on the **Utility: Rename/Scratch Options** panel (CMNUTL00). The **Utility: Baseline Selections** panel (CMNUTL01) is displayed.

```
CMNUTL01        Utility: Baseline Selections
Option ===>____________________________________________

    blank Display component selection list R Rename component
        S Scratch a component

Library type . . . . . CPY      (Blank for list)
Component . . . . . .  ACPCPYC*                              +
New name . . . . . . . _____________________________________+

Enter "/" to select option
/ Confirm request
```

This table describes the fields on the **Utility: Baseline Selections** panel.

Column	Description
Option	Type an **Option** to create a scratch or rename request for one component from this panel.
	R: Create a Rename request for one component in a baseline library.
	S: Create a Scratch (delete) request for one component in a baseline library.
	Leave the **Option** line blank to display the **Utility: Baseline Member List** panel where you can create utility requests for one or more baseline library members.
Library type	Type the library type of the component or components you want to scratch or rename. Leave this field blank to display a library type selection list.
Component	If you typed **R** or **S** in the **Option** line, type a component name for the utility request. If you left the **Option** line blank, do one of the following:
	Leave the Component field blank to display all members in the baseline library on the **Utility: Baseline Member List** panel.
	Type a pattern to filter the members displayed on the **Utility: Baseline Member List** panel.
	Type a component name to position the **Utility: Baseline Member List** panel at that component.
New name	If you typed **R** in the **Option** line, type the new name for the component.

Column	Description
Confirm request	Select this field to display the Component Warning panel that shows a list of other packages that contain the component being scratched or renamed (otherwise this is suppressed).

2. If you left the **Option** line blank on the **Utility: Baseline Selections** panel, the **Utility: Baseline Member List** panel (CMNUTL02) is displayed.

```
CMNUTL02        Utility:  Baseline Member List                  Row 1 to 6 of 6
Command ===>    ________________________________________________Scroll ===> CSR

Baseline library:
CMNTP.S6.V810.BASE.ACTP.CPY

Name       Rename   vv.mm Created    Changed            Size    User    Function
__ACPCPYCA ________ 02.00 2002/05/07 2014/12/12 01:02     3    JPRESTO
__ACPCPYCB ________ 02.00 2002/05/07 2014/12/12 01:02     3    JPRESTO
__ACPCPYCC ________ 02.00 2002/05/07 2014/12/12 01:02     3    JPRESTO
__ACPCPYCD ________ 02.00 2002/05/07 2014/12/12 01:02     3    JPRESTO
__ACPCPYCE ________ 02.00 2002/05/07 2014/12/12 01:02     3    JPRESTO
__ACPCPYCF ________ 02.00 2002/05/07 2014/12/12 01:02     3    JPRESTO
******************************* Bottom of data ********************************
```

This table describes the fields on the **Utility: Baseline Member List** panel.

Column	Description	
Command	Type a *panel* command	
	CANCEL	Cancel and return to the previous panel.
	SORT *heading	Sort panel rows by the named heading.
	Or, type a *component* command followed by the name of a component on the **Utility: Baseline Member List** panel.	
	LOCATE	Scroll to the component.
	SCRATCH	Create a scratch request for the component.
	RENAME	Create a rename request for the component.
	BROWSE	Browse the component.
	VIEW	View the component.
		Display the long history for the component.
	I	Display the short history for the component.

Column	Description	
Line Command	S	Create a scratch request for the component
	R	Create a rename request for the component
	B	Browse the component
	V	View the component
	H	Display the long history for the component
	I	Display the short history for the component
	L	Display a list of libraries where the component resides
Name	Components in the baseline library	
Rename	New name for rename utility requests	
Function	The line command function that was requested, after it has been executed.	

3. On the **Utility: Baseline Member List** panel, type a command on the Command line, or type a line command on one or more components. For a rename request, provide a new name for the component. Press **Enter** to process your requests.
4. If the **Confirm Request** option on the **Utility: Baseline Selections** panel was selected, and if you request a rename or scratch for a component that is in another package that has not been baselined, the **Component Warning** panel (CMNCMPSW) is displayed.

```
CMNCMPSW     History - GNLCPY1A.CPY            Row 1 to 1 of 1
Command ===> _________________________________ Scroll ===> CSR

This component is included in the following packages:

Package     Sta Promoted vv.mm Last action      Size Procname User    Release
GENL000007  DEV          02.02 2015/01/08 01:06 00006         USER015
******************************* Bottom of data ******************************
```

Also, a conflict message is sent to the user specified in the Notify user field of the other packages:

```
    Scratch/rename Conflict: ACPJCL50.JCL between ACTP000043 and     ACTP000055
2010/04/30 23:43 CN(INTERNAL)
```

Make sure that you are coordinating your utility requests with the other developers who are working on the same component in their packages.

If you decide to abandon your rename or scratch request, type CANCEL in the **Command** line of the panel and press **Enter**.

> **Note**
>
> When there is a concurrent development situation, a conflict message is sent to the user specified in the Notify user field of the other packages: Scratch/rename Conflict: ACPJCL10.JCL between ACTP000043 and ACTP000047 2010/04/30 23:43 CN(INTERNAL)

5. If you decide to proceed with your utility requests, press **Enter** on the **Component Warning** panel, and your requests are processed. If you ask to rename a component that is already in the package it will fail and you will receive a message similar to:

```
CMN2250A - ACPSRC91.SRC exists in the package. Cannot scr/ren existing component.
```

The **Utility: Baseline Member List** panel shows the action you requested in the **Function** column.

```
CMNUTL02               Utility: Baseline Member List                   RENAME INFO.
SAVED
Command ===> ____________________________________________________________   Scroll ===>
CSR
CMN000 CMN2220I - COBOL002.SRC rename information saved.
    Baseline library:
    CMNTP.S6.V810.BASE.GENL.SRC

    Name        Rename      vv.mm   Created     Changed            Size  user
Function
___ COBOL001                01.03   2013/08/28  2015/01/28 14:54   12    DJACOBS
___ COBOL002    COBOL003    01.03   2013/08/28  2015/01/28 15:12   13    DJACOBS  *Rename
******************************* Bottom of data*******************************************
```

6. When you are finished with rename and scratch requests, press **PF3**, or type **END** on the **Command** line and press **Enter**, to exit the **Utility: Baseline Member List** panel and return to the **Utility: Baseline Selections**.

Reviewing and Deleting Utility Requests from Packages

Access and complete the **Utility: package List** to review or delete utility requests from a package list.

1. Select Option **2 Package** from the **Utility: Rename/Scratch Options** menu. The **Utility: package List** panel (CMNUTL03) is displayed.

```
CMNUTL03     Utility: GENL000010 List           Row 1 to 2 of 2
Command ===>_________________________________  Scroll ===> CSR

Req Name   + Rename + Type Changed          User    Status Function
__ SCR COBOL001          SRC  2015/02/04 04:25 USER015 ACTIVE
__ REN COBOL002 COBOL003 SRC 2015/02/04 04:25  USER015 ACTIVE
******************************* Bottom of data *****************************
```

This table describes the fields on the **Utility: package List** panel.

Column	Description	
Command	LOCATE *name	Scroll to the named component
	REFRESH	Reload panel data from master files
	CANCEL	Cancel action and return to the previous panel, note that any deleted entries are removed from the list after pressing ENTER, and remain removed.
Line Command	D	Delete the utility request from the package
	H	Display the long history for the component
	I	Display the short history for the component
	L	Display a list of libraries where the component resides
Req	Displays the type of utility request:	
	SCR	Scratch request
	REN	Rename request
Name	Component name	
Rename	Displays the new name for rename utility requests.	
Type	Displays the library type of the component.	
Changed	Displays the date and time that the utility request was created.	
User	Displays the userid of the person who created or activated the utility request.	
Status	Displays the status of the component.	

Column	Description
Function	Displays the line command function that was requested, after it has been executed.

2. Type a line command to initiate an action and press Enter to process the line commands.

   This panel shows the list after issuing a delete for the scratch request for the SRC component COBOL001 which was the first entry.

```
CMNUTL03         Utility: GENL000010 List           Row 1 to 1 of 1
Command ===>______________________________________ Scroll ===> CSR

Req  Name    + Rename +  Type  Changed          User    Status Function
__ REN  COBOL002 COBOL003  SRC   2015/02/04 04:25 USER015 ACTIVE
****************************** Bottom of data ******************************
```

> **Note**
>
> Deleting a utility request for a like-source component does not automatically delete the utility requests for source-load related components (even though those utility requests were generated automatically). In that case, the scratch requests for SRC, LOD, and LST members must be deleted individually. SYNCH22! or SYNCH23! will occur if a component conflict is found, see Auditing a Package.

Scratch/Rename Updates to Impact Analysis

Scratch and rename utility requests can alter component relationships that are tracked by ChangeMan ZMF impact analysis.

- COPYBOOK - Copybooks referenced in source code or in other copybooks
- SUBROUTINE - Subprograms statically linked in composite load modules
- JCL-PROCEDURE - Cataloged procedures that are invoked by the EXEC statement in JCL components
- PGM NAME/SYMBOL - Programs that are executed by the EXEC PGM= statement in JCL components
- DSN NAME/SYMBOL - Datasets that are referenced in DSN= statements in JCL components

  Impact analysis data is used by audit and by the online query impact and query bill of materials functions.

  Impact analysis relationships are stored in a VSAM LDS, which is dynamically updated in baseline ripple jobs at package install and by reverse baseline ripple jobs at package backout.

When a scratch or rename utility request is processed for the *superior* component in a relationship, table rows are deleted from impact analysis data. When the utility request is a rename, the table rows are added back to the impact analysis data under the new name.

When a scratch or rename request is processed for a *subordinate* component in a relationship, impact analysis data is not changed. The **Baseline ripple - SCRATCH/ RENAME - Potential Component Restage Report** displays a warning that you may need to restage the superior component.

```
ChangeMan(R) ZMF        CMNDSPTM - 8.1.0 2015/01/14 01:09:27
Baseline ripple - SCRATCH/RENAME - Potential Component Restage Report
Copybook    GNLCPY00 has been renamed to GNLCOP00 but may still be referenced by ...
GNLSRS00
GNLSRS1B
GNLSRS1C
GNLSRS5A
GNLSRS5B
GNLSRS5C
Copybook    GNLCPY99 has been scratched but may still be referenced by ...
GNLSRC1A
GNLSRC99
Session terminated with ChangeMan ZMF started task
I/A dataspace update processing complete; rc=0000
```

This report is written by program CMNDSPTM at the RESTAGE DD statement in job step DSPTM in these install jobs:

- CMN30
- CMN30I
- CMN37

If you did not fully analyze component relationships (audit not being required, or modified by exits) before installing a package containing scratch or rename utility requests, you must examine the **Potential Component Restage Report** for relationships that may require additional action.

Important

When a package containing scratch or rename requests is backed out, impact analysis data is restored only for renamed components. Although components deleted by scratch utility requests are restored to baseline and production libraries when the package is backed out, the impact analysis relationships are not restored.

12. Auditing a Package

The ChangeMan ZMF package audit function ensures the synchronization of components in an active change package with:

- The staging, baseline, and promotion libraries of those components.
- The staging, baseline, and promotion libraries of components in other participating packages.
- The staging, baseline, and promotion libraries of components in other applications.

About Package Audit

Package audit uses information in the package master, component master, and the impact analysis data space to validate the integrity of package components. A package must pass audit before it can be installed. Your ChangeMan ZMF administrator determines the level of level of audit flexibility that is allowed for packages to pass. Audit results are displayed in the Audit Report.

Package audit verifies the synchronization of each package component with other versions of the component in baseline libraries, promotion libraries, and participating packages.

Audit checks the synchronization of components contained within components, such as subprograms in statically linked load modules and copybooks included in compiled source. Audit enforces other rules such as designated compile procedures, and it ensures that package components have not been changed by tools outside of ChangeMan ZMF.

These are examples of errors that audit is intended to catch before package installation:

- You modify a copybook in your package after you compile a source component that references it.
- Someone else installs a change to a subprogram after you link edit a composite load that includes the baseline version of the subprogram NCAL load.
- You compile a program that includes a copybook in your package, then you delete the copybook from your package.
- You check a component out but do not stage it.
- You check out and stage a component, but you make no changes.

  The auto resolve facility of package audit automatically submits restage, recompile, and relink jobs to resolve synchronization problems detected by audit.

Note

The audit return code is also written to the Activity Log. Select Activity Log Code 50 to display this information.

Administrative Parameters that Affect an Audit

Your ChangeMan ZMF administrator can set specific global and application parameters that affect an audit. These parameters include the:

- Audit level.
- Installation date for processing participating packages.
- Department number.
- The order in which the audit auto-resolve function submits stage, recompile, and relink jobs. This order is determined by the sequence number that is assigned to Like- Source and Like-Load libraries on the application Library Types (CMNCLLT0) panel (option A.A.2 from the ChangeMan ZMF Primary Option Menu).
- The value of the SYSLIB EXCLUDE flag that is specified for Like-Copy promotion libraries.

Although you may not have access to the administrator options that allow you to set these parameters, the following sections summarize their effects on audit. Consult your application administrator for information about the audit rules that are in force for your application. Refer to the *ChangeMan ZMF Administrator's Guide* for a complete explanation of these parameters and how to set them.

Audit Level

Your package must pass an audit before you can freeze it for promotion or installation. For your package to pass an audit, the audit return code cannot exceed the value that your application administrator specifies for the audit level:

Audit Level	**Meaning
0	Audit is recommended but optional.
1	Audit is required. Any return code, except abend, is acceptable.
2	Audit is required and the return code cannot exceed 12. This return code indicates that there are out-of-sync conditions in the staging libraries of the package. Any out-of-sync conditions in baseline libraries are reported as well. The package passes the audit only if the return code is not higher than 12.
3	Audit is required and the return code cannot exceed 8. This return code indicates that there are no out-of-sync conditions in the staging libraries but that there are out-of-sync conditions in the baseline libraries. The package passes the audit only if the return code is not higher than 8.
4	Audit is required and the return code cannot exceed 4. This return code indicates that there are no out-of-sync conditions in the staging or baseline libraries, but at least one component in a staging library is a duplicate (unchanged) of the corresponding baseline component. The package passes the audit only if the return code is not higher than 4.

Audit Level	**Meaning
5	Audit is required and the return code cannot exceed 0. This return code indicates that there are no out-of-sync conditions in the staging or baseline libraries. Also, there are no duplicate (unmodified) components in the staging libraries. The package passes the audit only if the return code is not higher than zero. Some components that are unknown to ChangeMan ZMF, do not have ISPF statistics, or are unparsable load modules are flagged as out of sync but are allowed to pass the audit. Examples of components that are unknown to ChangeMan ZMF include language environment subroutines which the binder automatically links into a load module.

Processing Participating Packages by Installation Date

If your site plans software changes months ahead and installs packages regularly, your global administrator should set the *Process Participating Packages by Install Date* parameter on the Global Parameters panel (CMNGGP03) on. If this parameter is set to on, participating packages that are scheduled for installation earlier or simultaneously with the participating package that you want to audit are included in the audit report.

In this case, staging libraries are placed in descending order by installation date in concatenation lists. The compile and link-edit job procedures use these lists. Having staging libraries in descending order ensures that the latest copybooks and called load modules are included during staging and audit.

Presumably, participating packages with installation dates later than the package you want to audit are in early development stages. If these participating packages are included in the audit, they will generate invalid out-of-sync conditions. Setting the *Process Participating Packages by Install Date* parameter on will exclude packages with later installation dates from the audit.

Department Number

If you want to audit participating packages with the same department as a group, your application administrator should set the *Require Department* parameter on the Global Parameters panel (CMNGGP02) on. This setting requires the package creator to specify a department at the time a package is created.

Complex or super packages can have participating packages with different department values. If the application administrator sets the *Require Department* parameter on, you can request that an audit be restricted to participating packages that have the same department at the time you submit the audit request.

Sequence Number in Library Type Definitions

The following items determine the order in which the audit auto-resolve function submits jobs:

- The sequence number that is assigned to the Like-Source libraries that are defined for your application determines the order in which staging and recompile jobs are submitted.
- The sequence number that is assigned to the Like-Load libraries that are defined for your application determines the order in which relink jobs are submitted.

The sequence numbers that are assigned to Like-Copy library types have no effect on the audit auto-resolve function.

Syslib exclude Field

The application administrator can set the **Syslib exclude** field on the **application Promotion Libraries** (CMNLRPM3) panel in application administration (option A.A.7 from the ChangeMan ZMF Primary Option Menu). If the value of this field is set to Y for a Like- Copy promotion library, that library is excluded from audit processing. See the *ChangeMan ZMF Administrator's Guide* for more information about this field.

User Exits that Affect a Package Audit

The following user exits affect the results of a package audit:

- CMNEX020 - Resets return codes for package audit SYNCHnn! conditions.
- CMNEX021 - Uses library type, application, package creator, and other fields passed from package audit to bypass SYNCHnn! processing, component relationship processing, and promotion libraries in SYNCH15! processing.
- CMNEX022 - Excludes specified load and non-load components from processing by package audit and by the impact analysis LDS build; includes CSECT with the same name as the composite load module in the Impact Analysis Table to show LOD relationships and in package audit processing to detect SYNCH8!.
- CMNEX028 - Adds JOB statements 5 and 6 to package audit batch JCL.
- CMNEX031 - Excludes library types from audit processing.
- CMNEX040 - Specifies the name of a data set that contains audit auto resolve runtime parameters.
- CMNEX044 - Specifies like-copy library types to be excluded from package audit SYNCH15! processing.

See the *ChangeMan ZMF Customization Guide* for more information about these exits. See comments in the exit program source code in the CMNZMF.ASMSRC library for instructions on enabling each exit and examples of use.

Auditing a Package Attached to a Release

If you have licensed the Enterprise Release Option (ERO), a package must pass a package audit and an area audit before a release can be processed.

If you perform a package-level audit against a package that is attached to a release, the following sync conditions are not flagged:

- DUPLIC!
- SYNCH0!
- SYNCH5!
- SYNCH8!
- SYNCH10!
- SYNCH13!
- SYNCH15! for copybooks (the LCT version of SYNCH15! is still flagged)
- SYNCH16!
- SYNCH24!

These sync conditions are related either to baseline components or copybooks.

A baselined component without ISPF statistics returns a SYNCH1! in package audit, but not in release audit.

A staged component without ISPF statistics returns a SYNCH1! in package audit and release audit.

Two messages appear in the SYSPRINT data set for audit job step AD000 to indicate that the package being audited is attached to a release:

```
CMN2614I - Package is attached to a Release, package audit processing will be restricted to
staging contents.

CMN2617I - Use ERO Area Audit to ensure integrity across the release areas and baseline.
```

A slightly different format heading appears in the audit report (AUDITRPT data set of the audit job) to indicate that the package is attached to a release. For example:

```
ChangeMan(R) ZMF (8.1.0 - 20141010) Audit FRIDAY DECEMBER 12, 2014 (2014/346)
01:52:26                       Page 1
            ******************************(NOTE: Package attached to
Release)*******************************
            * ERO Change Package       ===> ACTP00023 Created 2014/12/12 at 01:26:56 by
USER015            *
            *Package Installation Date ===> 2014/12/31       Package Status:
DEV                           *
            *Component Analysis Type   ===> Description of member from library directory
entry             *
            *Library Appl:Libtype      ===>
ACTP:CPY                                                       *
```

Note

Each time the package is either attached to or detached from a release, the package audit return code is reset. This requires the package to be audited again according to the audit-level rules.

See the *ChangeMan ZMF ERO Option Getting Started Guide* for information about auditing release areas.

Requesting An Audit

You request an audit by filling in the information on the Audit Change Package (CMNAUDIT) panel and submitting the audit job. Your response to prompts on this panel determine the focus and scope of the audit.

Take the following steps to request an audit and generate an audit report:

1. Select option 1 Build from the Primary Option Menu.
2. Select option 7 Audit from the Build Options menu (CMNBUILD). The Audit Change Package panel (CMNAUDIT) is displayed.

```
CMNAUDIT                Audit Change Package
Command ===> __________________________________________

Package . . . . . . . . ACTP000081
Audit part. pkg . . . . (1-As simple,2-As primary,3-By Department)
Autoresolve parms . . . __________________________________________

Enter "/" to select option
__ Audit staging libraries only
/  Include history records
__ Format report for printing
__ Specify application               __ User variables
__ Suppress batch messages
/  Include cross appl headings       __ Print top line only
__ Lock package                      __ Reset lock
__ Auto resolve
__ Update only this pkg rc
Job statement information:
//USER015A JOB (SM-1IKF-SM),'CMN6 AUDIT 8',
//          CLASS=A,
//          NOTIFY=USER015,
//      MSGCLASS=X
```

Fields on this panel are:

Field	Description
Command	TRACE. Write diagnostic information to SYSPRINT DD at job step AD000.
Package	The package that you want to audit (a simple package, a complex or super package, or a participating package).
Audit part. pkg	The value of this field can be set only if the Package identifies a participating package. This field value affects the scope of the audit as follows:
	**1- As Simple
	Audit the specified participating package as a simple package. Only the staging and baseline libraries associated with this package are considered in the audit. If you select this, you must **not** select the **Update Only this Pkg rc** field.
	2 - As Primary**
	Audit the libraries of all participating packages in the audit, but reports out-of-sync conditions only in the target participating package and updates the audit return code for only the target participating package. The text *Primary Package Processing* appears in the header of each section of the audit report to indicate that you have requested that the target participating package be audited as a primary package.
	3 - By Department**
	Exclude the participating packages that have a department number that differs from the department number of the target participating package.

Field	Description
Autoresolve parms	Fully qualified data set name for an optional file that contains audit auto resolve runtime parameters that are intended to provide the most efficient process for resolving out-of-synch conditions. The administrator may force the use of a standard data set with a ZMF exit program. See Auto Resolve Runtime Parameters.
Audit staging libraries only	This field indicates whether you want to check for out-of-sync conditions in the package's staging libraries only or in the staging and baseline libraries. If audit is required for your package before freeze, then it must be a full audit of staging and baseline libraries.
	If not selected, then audit will check for out-of-sync conditions in both the staging and baseline libraries.
	Select this with a / for audit to check for out-of-sync conditions only in the staging libraries. This option will automatically be enforced if the package being audited is attached to an ERO release.
Include history records	This field allows you to include or exclude component history information in the audit report. Component history shows other packages that contain the staged component and may be helpful in resolving out-of-sync conditions.
	If not selected then the audit will not display component history information when generating the audit report.
	elect this with a / to extract component history for each staged component and display it in the audit report.
Format report for printing	This field enables you to customize the appearance of the audit report:
	If not selected, then the audit report will be formatted for browsing. The text starts in column one and blank lines separate the header and sections of the audit report.
	Select this to format the audit report for printing. The first character is an ASA printer spacing control character that controls the vertical printer motion.
Specify application	If not selected, then do not change the mix of applications that will be included in the audit by default. The applications included by default depend on the type of package being audited, and are as follows:
	Only the application associated with a simple package or with a participating package that is being audited as a simple or primary package (1 - As simple or 2 - As primary).
	The applications associated with all participating packages if the package being audited is a:
	Complex or super package.
	Participating package that is not being audited as a simple or primary package.

Field	Description
	If selected then ChangeMan ZMF will bring up the Application Scope Selection List panel (CMNAUDAP) to select additional applications whose libraries you want to include in the audit. See Performing a Cross-Application Audit for a description of this panel.
User Variables	Enter '/' to specify user variables V01 through V10 on panel CMNUSV1 to pass information to skeleton file tailoring.
Suppress batch message	Select to suppress the SEND message normally broadcast at the completion of the batch audit job that tells you whether your package passed or failed the audit.
Include cross appl headings	If numerous application:libtype combinations share the same baseline library, this field enables you to specify the number of application:libtype combinations that are printed in the heading of an audit report section.
	Select to list all appl:libtype combinations. Up to six appl:libtype combinations are printed on each line.
	If not selected, then suppress all appl:libtype combinations.
	Also select "**Print top line only**" to print only the first line of appl:libtype combinations.
	For example, if the same baseline library is shared among 300 applications, then selecting this cross application headings field prints 50 lines of appl:libtype combinations in the heading of a report section. This field enables you to control the number of heading lines that are printed if a large number of appl:libtype lines are undesirable.
Lock package	Select this to lock the package during audit to prevent users from acting on the package until the audit completes. For complex, super and participating packages, other participating packages being audited are also locked.
Reset lock	Select this option to unlock a package that remains locked for audit when a previous audit job did not complete. For complex or super packages, all participating packages are unlocked. For other packages, only the requested package is unlocked. **Note:** When you select the **Reset Lock** field and press Enter, audit is not submitted. The package(s) will be unlocked and you will remain on the panel, and the field is cleared so that you may continue.
Auto resolve	Normally audit will only report out-of-sync conditions but does not attempt to resolve them. Select this field to not only report out-of- sync conditions, but also to automatically submit restage, recompile and relink jobs to resolve SYNCH2, SYNCH4, SYNCH5, SYNCH7, SYNCH8, SYNCH9, SYNCH16, and SYNCH24 conditions.. See Using Audit Auto Resolve.
Update only this pkg rc	Normally audit will update the return code for all participating packages that are included in the audit. If the Package is a complex or super package, the return code for the target package is also updated and is set to the highest return code issued for the participating packages. If you select this field then the audit will only update the return code for this package.

Field	Description
Job statement information	These JOB statements are used when you press Enter to submit the audit job. Verify this information before you submit the job.

Audit Report Contents

An audit report contains several sections. The number of sections and their contents depend on the types of components represented in the package or packages that are audited and the parameters you specify on the Audit Change Package (CMNAUDIT) panel when you request the audit.

An audit report can contain the following sections:

Report Section	Description
Description of Member from Library Directory Entry	Lists package components for each application:library type. Each application:library type that is represented in the package or packages being audited has its own section.
History of Changes for Audited Components	Lists the history of each component of an application:library type. Each application:library type that is represented in the package or packages being audited has its own section. These report sections are generated only if you respond YES to the Include History Records prompt on the Audit Change Package panel (CMNAUDIT).
Copybooks Found within Source Code	Lists copybook members that source programs and other copybooks reference. Each like-copy and like-source application:library type that is represented in the package or packages being audited has its own section.
Static Subroutines Called within Load Members	Lists the statically link-edited subprograms that each composite load module calls. Each like-load application:library type that is represented in the package has its own section.
Legend and Summary Report	Shows how many occurrences of each out-of-sync condition are flagged in the audit report.
Recommendation Summary	Contains recommendations for fixing each out-of-sync condition.

An example of each audit report section follows.

Note

The package audit report shown in this topic is run with these settings both selected:

- Include history records
- Include cross appl headings

Section Header

A header introduces each section of the audit report. The header identifies the contents of each report section and includes information such as the following:

- Banner that identifies the ChangeMan ZMF version, the date and time that the audit report was run, and the page number.
- Type of package being audited (simple, participating, complex/super), package name, package create date and time, and the TSO userid of the package creator.
- The number of participating packages (for a complex or super package) and the package status.
- The scheduled installation date (for a simple or participating package).
- The type of component analysis information that appears in the body of this report section.
- The application:library type of the components represented in this report section. You can control the number of application:library type lines that are printed with the value that you specify for the Include Cross Appl Headings field on the Audit Change Package (CMNAUDIT) panel. See Requesting An Audit for a description of this field.

Here's a sample section header from an audit report:

```
ChangeMan(R) ZMF (8.1.4 - 20170707) Audit WEDNESDAY DECEMBER 13, 2017 (2017/347) 15:30:05
Page 1
**********************************************************************************************
*  Simple Change Package     ===> ACTP000081 Created 2017/11/22 at 18:43:44 by
USER015         *
*  Package Installation Date ===> 2018/06/06 Package Status:
DEV                                *
*  Component Analysis Type   ===> Description of member from library directory
entry           *
*  Library Appl:Libtype      ===> ACTP:CPY
ACTR:CPY                                             *
```

Description of Member from Library Directory Entry

This section lists the components for each application:library type represented in the package or packages being audited. The ISPF directory entry for the library provides the information. The report section has two parts:

- The left side of the report with the heading *Baseline* contains ISPF statistics about the baselined version of the components of the target application:library type. If a component was not checked out from a baseline library, the baseline information for the component is blank.
- The right side of the report with the heading *Staging* contains ISPF statistics about the version of the components in the staging library associated with the component type.

The following excerpt shows the start of a sample report section that lists the components of the CPY library type for the ACTP and ACTR applications.

```
ChangeMan(R) ZMF (8.1.4 - 20170707) Audit WEDNESDAY DECEMBER 13, 2017 (2017/347) 15:30:05
Page 6
**********************************************************************************************
*       Simple Change Package            ===> ACTP000081 Created 2017/11/22 at 18:43:44 by
USER015 *
*       Package Installation Date        ===> 2018/06/06 Package Status:
DEV                       *
*       Component Analysis Type          ===> Copybooks found within Source
code                   *
*       Source Appl:Libtype              ===> ACTP:SRC
ACTR:SRC                                    *
**********************************************************************************************
*--------------------- B a s e l i n e ---------------------*
*---------------------- S t a g i n g ----------------------*
*************************************************************
*************************************************************
Copybook                                                         Source   Copybook
Name     VV.MM   Created    Changed        Size  Init   Tso-id   Name     Name     VV.MM
Created       Changed       Size Init Tso-id

________ _____ __________ ________________ ____  ____   ________ ________ ________ _____

__________ ________________ ____ ____ ________
         01.03 2002/05/07 2015/01/05 00:44 34     23    USER015  ACPSRCCA          02.00
2017/12/13 2017/12/13 15:17   34 34   USER015
ACPCPYCA 02.00 2002/05/07 2014/12/12 01:02 3       1    USER015
ACPCPYCB 02.00 2002/05/07 2014/12/12 01:02 3       1    USER015
ACPCPY00 02.00 2002/05/07 2014/12/12 01:02 4       1    USER015
COMCPY00 --------- Not found or unknown -------                  SYNCH0!
```

As another example, the following excerpt shows a report section that lists the staged components of the LOD library type for the ACTP and ACTR applications.

```
ChangeMan(R) ZMF (8.1.4 - 20170707) Audit WEDNESDAY DECEMBER 13, 2017 (2017/347) 15:30:05
Page 2
**********************************************************************************************
*Simple Change Package              ===> ACTP000081 Created 2017/11/22 at 18:43:44 by
USER015  *
*Package Installation Date          ===> 2018/06/06   Package Status:
DEV                      *
*Component Analysis Type            ===> Description of member from library directory
entry    *
*Library Appl:Libtype               ===> ACTP:LOD
ACTR:LOD                                     *
**********************************************************************************************
*--------------------- B a s e l i n e --------------------*
*---------------------- S t a g i n g ---------------------*
*************************************************************
*************************************************************

Name     Size     Linkdate   Alias-of AC Attr Setssi                  Name     Size
Linkdate   Alias-of AC Attr Setssi

________ ________ __________ ________ __ ____ ________                ________ ________
__________ ________ __ ____ ________
ACPSRCCA 00001510 2015/01/05          00 C2E2 6779D60D                 ACPSRCCA 00001538
2017/12/13          00 C2E2 6D013C27
                                                                       CTST     00001088
2017/11/26          00 C2E2 6CEB2ED4
```

Refer to Audit Report Field Descriptions for a description of the fields in the Baseline and Staging sections of the report.

History of Changes for Audited Components

These report sections provide historical information about the components of each application:library type that is represented in the package or packages being audited. Here’s the start of a sample history report section for the LOD library type. It shows the packages in which the target component appears:

```
ChangeMan(R) ZMF (8.1.4 - 20170707) Audit WEDNESDAY DECEMBER 13, 2017 (2017/347) 16:39:56
Page 4
***************************************************************************************
*Simple Change Package     ===> ACTP000081 Created 2017/11/22 at 18:43:44 by USER015 *
*Package Installation Date ===> 2018/06/06      Package Status: DEV                   *
*Component Analysis Type   ===> History of changes for audited components            *
***************************************************************************************
*----------------- C o m p o n e n t H i s t o r y - LOD -----------------*
********************************************************************************
Name     VV.MM Changed             Tso-id   Package    Status

________ _____ ___________________ ________ __________ ______
ACPSRCCA 01.03 2014/11/19 03:14    USER015  ACTP000001 BAS
ACPSRCCA 01.00 2015/01/05 00:45    USER015  ACTP000003 BAS
ACPSRCCA 01.00 2015/01/04 22:52    USER015  ACTP000020 BAS
ACPSRCCA 01.01 2016/01/20 20:37    USER015  ACTP000060 DEV
ACPSRCCA 01.00 2017/12/13 15:18    USER015  ACTP000081 DEV
CTST     01.00 2017/11/26 21:51    USER015  ACTP000081 DEV
```

Refer to Audit Report Field Descriptions for a description of the fields in the component history sections of the report.

> **Note**
>
> To display this report section, you must select field Include history records on the Audit Change Package panel (CMNAUDIT).

Copybooks Found within Source Code

This report section lists copybook members that source programs and other copybooks reference. Each like-copy and like-source application:library type that is represented in the package or packages being audited has its own section.

The report provides baseline information (on the left side) and staging information (on the right side) about each copybook member. The component name (from a like-copy or like- source library type) that references the associated copybook member appears in Source Name column in the middle of the report between the Baseline and Staging sections. Any out-of-sync conditions that are flagged for the listed components also appear in the Source Name column.

The component is new to ChangeMan ZMF if no baseline information is shown for it. If no Staging information is shown, the component resides only in the baseline library and has not been staged in the package or packages being audited.

Here's a sample *Copybooks Found within Source Code* report section. The excerpt shows an audit of a simple package (ACTP000081).

Typically, you stage only those copybooks that you want to change. (That is, there is no reason to stage a copybook unless you intend to change it.) Three out-of-sync conditions are flagged in this example (SYNCH5! - cpy/hdr baseline problem).

```
Changeman R) ZMF (8.1.4 - 20170707) Audit WEDNESDAY DECEMBER 13, 2017 (2017/347)
17:24:57               Page 11
**********************************************************************************************
*Simple Change Package                      ===> ACTP000081 Created 2017/11/22 at 18:43:44
by USER015 *
*Package Installation Date                  ===> 2018/06/06         Package Status:
DEV               *
*Component Analysis Type                    ===> Copybooks found within Source
code                   *
*Source Appl:Libtype                        ===> ACTP:SRS
ACTR:SRS                                    *
**********************************************************************************************
*--------------------- B a s e l i n e --------------------*
*---------------------- S t a g i n g ---------------------*
*************************************************************
*************************************************************
Copybook                                                        Source Copybook
Name     VV.MM Created    Changed          Size Init Tso-id     Name    Name     VV.MM
Created    Changed            Size Init Tso-id
________ _____ __________ ________________ ____ ____ ________ ________ ________ _____
__________ ________________  ____ ____ ________
         01.02 2014/12/18 2014/12/18 02:10 23   19   USER015  ACPSRS00
ACPCPY00 02.00 2002/05/07 2014/12/12 01:02  4    1   USER015  SYNCH5!  ACPCPY00 03.01
2017/12/13   2017/12/13 17:24  4    5   USER015
         01.02 2014/12/18 2014/12/18 02:11 26   22   USER015  ACPSRS1B
ACPCPY00 02.00 2002/05/07 2014/12/12 01:02  4    1   USER015  SYNCH5!  ACPCPY00 03.01
2017/12/13   2017/12/13 17:24  4    5   USER015
ACPCPY1B 02.00 2002/05/07 2014/12/12 01:03  4    1   USER015
ACPCPY1C 02.00 2002/05/07 2014/12/12 01:03  4    1   USER015
         01.01 2014/12/18 2014/12/18 01:28 21   21   USER015  ACPSRS1C
ACPCPY00 02.00 2002/05/07 2014/12/12 01:02  4    1   USER015  SYNCH5!  ACPCPY00 03.01
2017/12/13    2017/12/13 17:24 4    5   USER015
ACPCPY1C 02.00 2002/05/07 2014/12/12 01:03  4    1   USER015
```

Refer to Audit Report Field Descriptions for a description of the fields in the Baseline and Staging sections of the report.

Static Subroutines Called within Load Members

These report sections list all composite load modules and the statically linked subroutines that they call for each application included in the audit report. Each application:like-load library type has its own report section.

The report shows information from the *Baseline* library (on the left side) and information from the package *Staging* library (on the right side). The calling composite load module name and flagged out-of-sync conditions appear in the middle of the report under the *Calling Module* column.

A load module is new to ChangeMan ZMF if *Staging* information but no *Baseline* information is shown for it. If no *Staging* information is associated with a module, the module resides in the baseline library but has not been staged into the package.

Here’s a sample report section for *Static Subroutines Called within Load Members*. The following information is shown in the excerpt:

- One composite load module named ACPSRCCA of the LOD library type for the ACTP application is shown in the report.
- ACPSRCCA calls several statically linked subroutines. None of the subroutines have been staged from baseline into the staging library associated with package ACTP000025.

Note

The modules whose names begin with the letters CEE and IGZ identify Language Environment routines that the binder automatically links into a module. They are flagged with the SYNCH0! condition code, which means that they have not been defined to ChangeMan ZMF. You can ignore such modules for application purposes. (You can suppress SYNCH0! condition codes from audit reports with a user exit. User Exits that Affect a Package Audit

```
ChangeMan(R) ZMF (8.1.4 - 20170707) Audit WEDNESDAY DECEMBER 13, 2017 (2017/347) 17:24:57
Page 12
**************************************************************************************
*Simple Change Package ===> ACTP000081 Created 2017/11/22 at 18:43:44 by USER015 *
*Package Installation Date ===> 2018/06/06 Package Status: DEV *
*Component Analysis Type ===> Static subroutines called within Load members *
*Load member Appl:Libtype ===> ACTP:LOD ACTR:LOD *
**************************************************************************************
*--------- B a s e l i n e ---------*      *---------- S t a g i n g ---------- *
*************************************     *************************************
Called    Module                     Calling    Called   Module
Module    Size    Linkdate   Setssi  Module     Module   Size     Linkdate   Setssi

________ ________ __________ ________ ________ ________ ________ __________ ________
         00001510 2015/01/05 6779D60D ACPSRCCA           00001538 2017/12/13 6D013C27
ACPSRS00 00000458 2014/12/18 67622F30
CEEARLU  -- Not found or unknown SYNCH0!
CEEBETBL -- Not found or unknown SYNCH0!
CEEBINT  -- Not found or unknown SYNCH0!
CEEBLLST -- Not found or unknown SYNCH0!
CEEBPIRA -- Not found or unknown SYNCH0!
CEEBPUBT -- Not found or unknown SYNCH0!
CEEBTRM  -- Not found or unknown SYNCH0!
CEECPYRT -- Not found or unknown SYNCH0!
CEESG005 -- Not found or unknown SYNCH0!
CEESTART -- Not found or unknown SYNCH0!
IGZCBSO  -- Not found or unknown SYNCH0
```

Refer to Audit Report Field Descriptions for a description of the fields in the Baseline and Staging sections of the report.

Legend and Summary Report

The Legend and Summary Report section of the audit report provides the following information:

- The audit level that the application administrator has defined for the application (0, 1, 2, 3, 4, or 5). If you are auditing a complex package that contains components from different applications, the more restrictive audit level is used if the administrator has defined different audit levels for the applications.
- The text description associated with the audit level that is defined.
- A message that indicates if the package passed or failed the audit and the highest audit return code that was encountered for each package included in the audit.
- The number of occurrences of each out-of-sync condition.
- A message indicating whether each change package included in the audit passed or failed the audit and the highest return code issued for each package. A complex package is assigned the highest return code that is issued to its participating packages.

If you are auditing a participating package and have selected any of the following prompts on the Audit Change Package (CMNAUDIT) panel, only the audit return code for the target participating package is updated:

- Audit Part. Pkg : 1 - as simple
- Audit Part. Pkg : 2 - as primary
- Update only this pkg rc

The following illustration shows a sample Legend and Summary Report section of the audit. Audit level 4 is in effect for the audit. The audit encountered the following conditions: 1 DUPLIC!, 24 SYNCH0!, 1 SYNCH4!, 7 SYNCH5!, 3 SYNCH14! and 2 SYNCH22.

```
ChangeMan(R) ZMF (8.1.4 - 20170707) Audit WEDNESDAY DECEMBER 13, 2017 (2017/347)
17:24:57               Page 13

                                                  Legend and Summary Report

The local level of audit chosen at this point;      4
4 - Audit is required and the return code must not exceed 4 which implies
that there are no "out-of-synch" situations within the staging libraries
nor the baseline libraries but at least one module of a staging library
is a "duplicate" of its baseline counterpart
Out-of-synch messages (hint - search for "!" marks)
DUPLIC!  (Staging duplicates baseline)      ===>  1
SYNCH0!  (Not in scope of audit or unknown) ===> 24
SYNCH1!  (Cmpnt statistics not available)   ===>  0
SYNCH2!  (Compile/designated proc differ)   ===>  0
SYNCH3!  (Unparsable load module)           ===>  0
SYNCH4!  (cpy/hdr staging problem)          ===>  1
SYNCH5!  (cpy/hdr baseline problem)         ===>  7
SYNCH6!  (Activity file not checked out)    ===>  0
SYNCH7!  (Static subcomponent stage problem) ==>  0
SYNCH8!  (Static subcomponent base problem) ===>  0
SYNCH9!  (Source and load discrepancy)      ===>  0
SYNCH10! (Version regression problem)       ===>  0
SYNCH11! (Component hash discrepancy)       ===>  0
SYNCH12! (Orphan module in staging)         ===>  0
SYNCH13! (Baseline/staging discrepancy)     ===>  0
SYNCH14! (Components not in active status)  ===>  3
SYNCH15! (Source to relationship problem)   ===>  0
SYNCH16! (CPY low-date problem in baseline) ===>  0
SYNCH17! (CPY deleted problem in staging)   ===>  0
SYNCH18! (LOD deleted problem in staging)   ===>  0
SYNCH19! (Missing module in staging)        ===>  0
SYNCH20! (Inconsistent subroutine)          ===>  0
SYNCH21! (Scr/rename pkg component)         ===>  0
SYNCH22! (Scratch subcmpnt is in use)       ===>  2
SYNCH23! (Rename subcmpnt is in use)        ===>  0
SYNCH24! (LOD low-date problem in baseline) ===>  0
SYNCH25! (LCT high-date problem in package) ===>  4
SYNCH26! (LCT -> missing executable)        ===>  1
SYNCH27! (Missing LCT for executable)       ===>  1
SYNCH28! (Same component in multiple ppkgs) ===>  8

Highest return code encountered             ===> 12
CMN3060A - This package has failed the audit.
CMN2696I - PACKAGE ACTP000081 FAILED THE AUDIT WITH A RETURN CODE OF 12.
```

The highest return code encountered for any package that was included in the audit is 12.

Recommendation Summary Report

The Recommendation Summary Report section of the audit report:

- Identifies the out-of-sync conditions that have been flagged in the audit report.
- Provides a brief explanation of how to correct them.

Identifies the highest return code issued for this audit report (RC=00, in this example).

```
Recommendation Summary Report
Listed below are some solutions to resolving out of synch situations
    that can be flagged within this audit report.
        DUPLIC! (Staging duplicates baseline)
            Delete component from staging or change contents of staging component.
            For package in BAS status, indicates no changes in baseline since pkg installed
- no action required.
        SYNCH0! (Not found within scope of this Audit)
            The component is either unknown to ChangeMan ZMF or not found within the
            criteria set for this audit. Check if component resides in an application
            or package excluded from the scope of this audit.
            To register an unknown component to ChangeMan ZMF stage it into a package.
        SYNCH4! (cpy/hdr staging problem)
            Stage source component to include copybooks/header files from the staging
libraries.
        SYNCH5! (cpy/hdr baseline problem)
            Recompile source component from baseline to include copybooks/header files
residing in the staging libraries.
        SYNCH14! (Components not in active status)
            Re-stage component to activate
        SYNCH22! (Scratch subcmpnt is in use)
            The sub-component to be scratched is in use by the indicated top-level
component.
            Remove the reference to the sub-component or remove the scratch request.
CMN7540I - End of job; RC = 12
```

If the package does not pass the audit, you cannot freeze it for approval, promotion, or installation. For the package to pass the audit, the audit return code must be consistent with the audit level set by your application administrator See Audit Level.

Audit Report Field Descriptions

The following table describes the fields (columns) you see in the various audit report sections. The entries are ordered in ascending sequence by field name to help you locate the description of a report field quickly. Not all fields appear in all sections of the report.

Field (Column) Name	Description
AC	For a like-load component, the authorization code (usually 00).

Field (Column) Name	Description
Alias-of	For a like-load component, the name of the member of which the target component is an alias.
Attr	For a like-load component, the hexadecimal representation of the PDS2ATR1 and PDS2ATR2 attributes of the PDS directory entry. Refer to the IBM *z/OS DFSMS Program Management* manual for a description of these values.
Called Module	The name of the called baselined or staged load module.
Calling Module	The name of the baselined or staged composite load module.
Changed	On the Baseline side of the report, the date (*yyyy/mm/dd*) of the most recently baselined version of the component. On the Staging side of the report, the date that the target component was last staged in this package.
Copybook Name	The name of the baselined or staged copybook.
Created	The date (*yyyy/mm/dd*) that the component was first included in the ChangeMan ZMF system.
Dept	The department number field, which you specify on the Create: Create a New *Package* panel (CMNCRT0R) when you create the package.
Init	The number of text lines of an editable component when first it was first created (version 01.01).
Linkdate	For like-load components, the date (yyyy/mm/dd) that the component in the baseline library was link-edited (on the Baseline side of the report). On the Staging side of the report, the date (yyyy/mm/dd) that the component in the package's staging library was last link-edited.
Module Size	The size (in bytes) of the module represented as a hexadecimal value. If the module is of libtype like-object, the text "Object" appears in this field.
Name	The name of the component.
Package	The name of the participating package where the component resides (if applicable).
Procedure	For a like-source component, this field describes the procedure that was used to stage (compile and link-edit) the component.
Setssi	For a like-load component, a timestamp that ChangeMan ZMF sets or that is extracted from a vendor load module. (The Setssi value is an 8-byte alphanumeric representation of a 4-byte binary number which is the number of seconds since January 1, 1960.)

Field (Column) Name	Description
Size	For editable components (such as like-source), the number of text lines (decimal) in the most recently baselined version of the target component (on the Baseline side of the report). On the Staging side of the report, the number of text lines of the target component in the package's staging library. For like-load components, the size (hexadecimal) in bytes of the component in the target baseline or staging library.
Source Name	The name of the source member that references the component members that are listed below it.
Status	The status of the associated change package.
Tso-id	The TSO identification of the person responsible for the associated component version and modification (*vv.mm*). For CA Panvalet library components, this field displays the USER and LANG TYPE because CA Panvalet does not have a true directory to read for the TSO-ID.
VV.MM	The version and modification number for the component following IBM standards for ISPF statistics. (*mm* starts at 01, increments to 99; *vv* starts at 01, increments to 99, and recycles to 01). The first time the component is processed by ChangeMan ZMF the *vv.mm* is 01.01. Each successive stage request (in the same change package) increments the *mm* portion (for example, 01.02, 01.03). The next package using the component causes the *vv* portion to be incremented (02.01). If an existing library is added to ChangeMan ZMF the *vv.mm* is not reset.

Audit Return Codes

Audit detects and reports out-of-sync conditions among components in package staging and baseline libraries in the target applications. Each out-of-sync condition sets a return code based on its impact on the integrity of the audited change package.

Return Code	Description
0	No out-of-sync conditions were found in your package. No user action is required.
4	This return code indicates that there are no out-of-sync conditions in the staging or baseline libraries, but at least one component in a staging library is a duplicate (unchanged) of the corresponding baseline component.
8	This return code indicates that there are no out-of-sync conditions in the staging libraries but that there are out-of-sync conditions in the baseline libraries.
12	This return code indicates that there are out-of-sync conditions in the staging libraries of the package and in baseline libraries.

Out-of-Sync Conditions

The following table identifies:

- Each out-of-sync condition.
- A description of the condition and recommendations for resolving it.
- The return code (RC) associated with the out-of-sync condition.

Out-of-Sync Condition	Description	RC
!DUPLIC	Component in the staging library is identical to the component in the baseline library. This condition identifies a component that shows no difference in the staging and baseline library after a line-by-line comparison. This condition, if not resolved, causes older versions of the component to drop off prematurely during baseline ripple. Suggested Resolution: Delete the unchanged component from the change package.	4
SYNCH0!	The component is not defined to the application. The package master contains no record of this component. For a simple package audit, the component cannot be found in the package that is being audited or in any of the application's baseline libraries. For a complex package audit, the component cannot be found in any of the participating packages that are being audited or in any of the baseline libraries that are defined for the application associated with the target participating package. The *target participating package* is one of the following: The participating package that you have explicitly specified to be audited. The participating package with the latest install date if you have specified that a complex or super package be audited. Language Environment subroutines that are automatically linked-edited into a load module are typically flagged with a SYNCH0! condition. Suggested Resolution: No action required. You can modify a user exit to suppress the issuing of SYNCH0! conditions. See the *ChangeMan ZMF Customization Guide*.	0
SYNCH1!	No ISPF statistics are available for the component in the baseline or staging library. Some of the audit checks cannot be performed for this component because it has no ISPF statistics (such as Create Date or Date Last Modified). Suggested Resolution: Create or reset the ISPF statistics for the component.	0
SYNCH2!	You staged a like-source component without using the designated compile procedure and options. Suggested Resolution: Recompile the component using the designated compile procedure or set Auto Resolve of Out-Of- Synchs to YES on the Audit Change Package panel (CMNAUDIT) and resubmit the audit.	12
SYNCH3!	Audit is unable to parse a load module in a staging library. (IDRDATA in a load component cannot be read.) ChangeMan ZMF cannot perform some of the audit checks for this component. Suggested Resolution: Restage the component.	0

Out-of-Sync Condition	Description	RC
SYNCH4!	You changed a like-copy component in a staging library more recently than you changed the *staging* library component that references it in the same package or in any relevant participating package. Suggested Resolution: Recompile the calling source component in the *staging* (not baseline) library. (Use option =1.6.2, line command R on the Stage: *package* Components panel (CMNSTG01)).	12
SYNCH5!	You changed a like-copy component in a *staging* library more recently than you changed the *baselined* component that references it. (That is, the referencing component resides in a baseline library and does not appear in the same package or in any relevant participating package as the component that it references.) Suggested Resolution: Recompile the source component from the *baseline* library. (Use option =1.8 to bring up the Recompile Source panel (CMNRCMP0)). Alternatively, you can set Auto Resolve of Out-Of-Synchs to YES on the Audit Change Package panel (CMNAUDIT) and resubmit the audit.	8
SYNCH6!	The component in the staging library has no corresponding Component Activity File member in the package. If the global or application administrator has defined an activity file for the target library type, the activity file is checked out automatically with the target component if the administrator has specified YES for the Check Out Component Activity File on library administration panels. If the corresponding activity file does not exist in baseline and these administrative parameters have been specified, audit flags the SYNCH6! condition. Suggested Resolution: Stage a corresponding Activity File for the component into the target package from a development library. **Note:** Audit does not perform this check if an activity file library type is not defined for the library type of the target component or if the global or application administrator has specified NO for the Check Out Component Activity File on library administration panels.	12
SYNCH7!	You changed a statically linked called load module, a like-object libtype component, or a like-NCAL libtype component in a *staging* library more recently than the calling load component in a *staging* library in the same package or relevant participating package. (That is, the SSI of the called module is more recent than the SSI of the calling module.) Thus, an obsolete version of the called module is link-edited with the calling module. Suggested Resolution: A relink is required. Set Auto Resolve of Out-Of-Synchs to YES on the Audit Change Package panel (CMNAUDIT) and resubmit the audit.	12
SYNCH8!	You changed a statically linked called load module, a like-object libtype component, or a like-NCAL libtype component in a *staging* library more recently than the calling load component in the *baseline* library. (That is, the SSI of the called module in the package or relevant participating package is more recent than the SSI of the calling module in the baseline library. The calling module has not been staged into the package.) Thus, an obsolete version of the called module is link-edited with the calling module in the baseline library. Suggested Resolution: A relink is required. Set Auto Resolve of Out-Of-Synchs to YES on the Audit Change Package panel (CMNAUDIT) and resubmit the audit.,	8

Out-of-Sync Condition	Description	RC
SYNCH9!	A load component in a staging library does not match its corresponding source component in the package. This condition can occur if you link-edit a load module outside of ChangeMan ZMF and replace the load module that was created when you originally staged the source component from a baselined library. Suggested Resolution: Stage the source component from within ChangeMan ZMF to create a new load module. Alternatively, you can set Auto Resolve of Out-Of-Synchs to YES on the Audit Change Package panel (CMNAUDIT) and resubmit the audit.	12
SYNCH10!	The component in the baseline library has changed since you checked the component out to a package staging library. This condition can occur if the same component is in a package that has been installed since you checked the component out to your package. To avoid version regression due to concurrent development, you must merge the changes in the newly baselined version of the component with the changes in the staged component in your package. Suggested Resolution: Check the component out from baseline to obtain the latest baselined version and stage the component with your changes. You can use the ChangeMan ZMF Merge and Reconcile facility to merge the changes. Refer to the *ChangeMan ZMF Merge and Reconcile Getting Started Guide*.	12
SYNCH11!	You used a tool other than ChangeMan ZMF to change and replace a component in a staging library. Suggested Resolution: Use ChangeMan ZMF to restage the component from baseline.	12
SYNCH12!	Either you used a tool other than ChangeMan ZMF to copy a component into a staging library or the component was left over from a staging job that abended. (The staged component does not have a corresponding record in the package master.) Suggested Resolution: Delete the component from the staging library or restage it from baseline. **Note:** This condition is in contrast to SYNCH19! where there is a record on the package master but no corresponding component in a staging library.	12
SYNCH13!	A component in the baseline library was changed more recently than the component in the staging library. The staged component is thus out of sync with the baseline library. For load modules, the SETSSI of the staged and baselined components are compared and the SETSSI of the baselined component is more recent than the SETSSI of the staged component. Suggested Resolution: Check the component out from baseline again and restage it with your changes. If the target component is a load module in a staging library that you created by compiling and linking a source component from baseline, recompile and relink the source component from baseline.	12
SYNCH14!	The package component is not in Active status. (You have checked the component out but not yet staged it.) Suggested Resolution: Stage the component or delete it from the package.	12

Out-of-Sync Condition	Description	RC
SYNCH15!	A more recent version of a like-copy or LCT component resides in a staging, promotion, or baseline library than the version that was used when you staged the referencing like-source component in a package. This condition can occur if:	12
	- A copybook in a staging, promotion, or baseline library was changed after the source component that references it in your package was compiled.	
	- A new version of the copybook was added to the SYSLIB search order for the referencing source component.	
	- The staged LCT component has a more current activation date that the corresponding source component.	
	- The search process for finding out-of-sync copybooks occurs in the following sequence for those copybooks that participate in the audit:	
	1. All copybooks in the simple package (or in relevant participating packages) staging libraries are searched first.	
	2. Then, all included copybook libraries defined to the target application promotion levels are searched in promotion-level order.	
	3. Finally, for the source components that exist in the package, audit picks up the application and library type where the copybooks came from that were used in the compile of the component (CMNWRITE). Audit then picks up the copybook from the baseline indicated by the application/library type. When detecting the SYNCH15! condition, audit compares the hash token of the copybook actually used in the compile against the token of the first matching copybook from the above list. If the tokens are not the same, a SYNCH15! condition occurs.	
	Suggested Resolution: Restage the like-source component to update the source-to-copybook or source-to-LCT relationship. Alternatively, you can set Auto Resolve of Out-Of-Synchs to YES on the Audit Change Package panel (CMNAUDIT) and resubmit the audit.	
SYNCH16!	A source-to-copy relationship is out of sync for a component in your package. This can happen if: you check a copybook out to two different packages (packages 1 and 2), and you change and activate the copybook in packages 1 and 2, and you stage a source program that references the copybook from a development library into package 2 and then baseline package 2.	8
	Suggested Resolution: Recompile the referencing like-source component from baseline. This action will store an updated like- load component in your package. Alternatively, you can set Auto Resolve of Out-Of-Synchs to YES on the Audit Change Package panel (CMNAUDIT) and resubmit the audit.	

Out-of-Sync Condition	Description	RC
SYNCH17!	A like-copy component in a staging library was deleted after it was referenced in a like-source component that was staged or recompiled. If the component was checkout out from Baseline then a SYNCH15! occurs instead, a SYNCH17! is generated only if a new like-copy component is deleted from the package. Suggested Resolution: If you inadvertently deleted the like-copy component from the package, restage the component from baseline, make the desired changes to it, and recompile the like-source modules from staging and baseline that reference it. If you no longer need the deleted copybook, remove the references to it from the like-source modules that reference it, and recompile the like-source modules.	12
	Note:** No SYNCH17! is issued if a like-copy component is checked out or staged from development from the baseline library, referenced in a like-source stage or recompile, but is never changed in the package before it is deleted.	
SYNCH18!	A called load module in a staging library was deleted after it was statically linked into a composite load module in the package. Suggested Resolution: If you inadvertently deleted the called load module from the package, restage it from baseline, and relink the composite load modules that reference it. If you no longer need the load module, remove references to it from link-edit control cards and from the modules that call it, and recompile and relink the callers.	12
SYNCH19!	A component is missing from a staging library. A record on the package master points to a component that is not there, as if the component was deleted from the staging library using a tool outside of ZMF.	12
	Recommended Resolution:	
	If the component is like-source, check the component out, reapply your changes, and stage the component. (You may be able to recover some of your changes from prior versions in staging versions.)	
	If the component is the product of build processing (like- load, etc.), then restage, recompile, or relink the component.	
	Note:** This condition is in contrast to SYNCH12! where there is a component in a staging library but no corresponding record on the package master.	
SYNCH20!	A subroutine that is statically linked in a composite load module in your package has a different timestamp (SETSSI) than the version of the subroutine load module that audit expects would be obtained by link edit from your package staging libraries (or eligible participating package) or from baselines (first found location). Recommended Resolution: Rebuild the composite load module in your package to link the correct version of all subcomponents.	
	Note:** SYNCH7! and SYNCH8! flag a package subroutine load that is newer than a composite load that contains it in the package or in a baseline. SYNCH20! looks to match the SETSSI in the subroutine loads exactly.	12

Out-of-Sync Condition	Description	RC
SYNCH21!	A scratch or rename request has been recorded for a component which is physically within the current set of packages. If the package being audited is a simple package then SYNCH21! should never normally occur, but it will be flagged if the situation arises. More likely to occur in a group of participating packages.	12
	Recommended Resolution: Remove the physical component or the scratch rename request.	
SYNCH22!	A sub-component (e.g. copybook, subroutine, archive constituent) identified in a scratch request is still in use by a top level component. This top level component may either be in a different package belonging to the same complex or in baseline. Note that if a component, which previously included this sub- component, is in the package with all reference to the sub-component removed then the synch error will not be flagged.	12
	Recommended Resolution: Remove the references to the component or the scratch request.	
SYNCH23!	Similar to SYNCH22! except, in this case, the sub-component is being renamed (rather than scratched). This error may not be resolvable within the current package(s). It may be that the rename request has to be baselined first before the affected components can be rebuilt using the renamed component. Recommended Resolution: Remove the references to the component or the rename request or baseline the rename request and rebuild.	4
SYNCH24!	A composite-to-static subcomponent relationship is out of sync for a subcomponent in your package. This is different to a synch8 as the baseline composite has a later link date than your package subcomponent (else, it would be a synch8). However, the baseline composite is not using the package subcomponent.	8
	This can happen if you check a subcomponent out to two different packages (packages 1 and 2), you change and activate the subcomponent in both packages, you build a composite that references the subcomponent into package 2 and then baseline package 2. This leaves the subcomponent in package 1 not used by the composite component now in baseline.	
	Suggested Resolution: Either delete the subcomponent in package 1 (if it is no longer needed) or build the composite in package 1 to include the subcomponent. Alternatively, you can set Auto Resolve of Out-Of-Synchs to YES on the Audit Change Package panel (CMNAUDIT) and resubmit the audit.	
SYNCH25!	A ‘linkcard’ (LCT) component in the package has a later change timestamp than one or more of the target executables to which it is related.	12
	Suggested Resolution: re-build the target executables so that the updated version of the LCT component is used in the build. Alternatively, you can set Auto Resolve of Out-Of-Synchs to YES on the audit.	

Out-of-Sync Condition	Description	RC
SYNCH26!	A ‘linkcard’ (LCT) component in the package is related to one or more target executables which are missing.	12
	Suggested Resolution: re-build the target executables so that the updated version of the LCT component is used in the build. Alternatively, you can set Auto Resolve of Out-Of-Synchs to YES on the audit.	
SYNCH27!	An executable in the package was built using a 'linkcard' (LCT) component which can no longer be found (in the package, participating packages, baseline or release areas).	12
	Suggested Resolution: Recover the missing LCT component and re-build the target executable (or, if possible, just re-build the target executable without the LCT component). This error is not Auto Resolvable.	
SYNCH28!	This component exists in more than one of the eligible participating packages for this audit. Consolidate changes into a single version of the component and delete all others from the participating package set. This error is only flagged when the complex/super package is audited.	8

Accessing the Audit Report

Use your site’s output utility, such as the Spool Display and Search Facility (SDSF), to view the contents of the audit report. Select the entry for AUDITRPT in the DDNAME list.

This is an example of multiple SYNCH23! and a SYNCH22! errors in an Audit Report.

```
ChangeMan(R) ZMF (8.1.0 - 20141010) Audit WEDNESDAY JANUARY 14, 2015 (2015/014) 00:11:27
Page 5

**********************************************************************************************
        *           Simple Change Package     ===> GENL000008 Created 2015/01/08 at
20:33:10 by USER015   *
        *           Package Installation Date ===> 2015/02/28 Package Status:
DEV                         *
        *           Component Analysis Type   ===> Scratch/Rename problems with non-zFS
components        *
        *           Library Appl:Libtype      ===>
GENL:CPY                                               *

**********************************************************************************************
    *--- C o m p o n e n t  C o n f l i c t ----*        *------ U t i l i t y  R e q u e
s t --------------*
    **********************************************
***********************************************************
         Name     Package    Appl Ltp                  Name     Package           Request
time     Tso-id
         ________ __________ ____ ___                  ________ __________
________________ ________
         GNLSRS00 BASELINE   GENL SRS          SYNCH23! GNLCPY00 GENL000008
2015/01/12 20:44 USER015
         GNLSRS1B BASELINE   GENL SRS          SYNCH23! GNLCPY00 GENL000008
2015/01/12 20:44 USER015
         GNLSRS1C BASELINE   GENL SRS          SYNCH23! GNLCPY00 GENL000008
2015/01/12 20:44 USER015
         GNLSRS5A BASELINE   GENL SRS          SYNCH23! GNLCPY00 GENL000008
2015/01/12 20:44 USER015
         GNLSRS5B BASELINE   GENL SRS          SYNCH23! GNLCPY00 GENL000008
2015/01/12 20:44 USER015
         GNLSRS5C BASELINE   GENL SRS          SYNCH23! GNLCPY00 GENL000008
2015/01/12 20:44 USER015
         GNLSRC99 BASELINE   GENL SRC          SYNCH22! GNLCPY99 GENL000008
2015/01/14 00:10 USER015
    ...
SYNCH22! (Scratch subcmpnt is in use)
  The sub-component to be scratched is in use by the indicated top-level component.
  Remove the reference to the sub-component or remove the scratch request.

SYNCH23! (Rename subcmpnt is in use)
  The sub-component to be renamed is in use by the indicated top-level component.
  It may not be possible to resolve this until after the rename has been processed in
baseline.
  Change the reference to the sub-component or remove the rename request.
```

Single Line Audit Summary Report

Program CMNADSUM is included as a step in skeleton CMN$$AUD, just after the execution of CMNAD500 (which generates the full audit report). CMNAD500 can produce a CSV file (see the Administrator’s guide for further details) that is used by CMNADSUM to produce single line summaries for each audit synch error flagged in the main report.

The single line summary of audit errors is output to ddname AUDITSUM. The sysin parameters used by CMNADSUM are:

Parameter	Description
PACKAGE	The requested package for the audit.
USER	The userid who requested the audit.
ZMFID	The subsystem character for the target ZMF instance.
REPORTLEVEL	Only errors carrying this RC and above will be reported here (default=0).
IGNORE	Errors with this synch number (DP=duplicate) will be ignored by this report. You can specify as many of these as you like.
	The default is for all errors (which pass the REPORTLEVEL criterion) to be reported.

This is an example output:

```
SYSPRINT:

ChangeMan(R) ZMF    CMNADSUM - 8.3 Change Man Package Audit Error Summary

CMNADSUM        Processing begins at 03:48:50 on 03/08/2021

CMNADSUM SYSIN: PACKAGE=STEV000456
CMNADSUM SYSIN: USER=WSER58
CMNADSUM SYSIN: ZMFID=M
CMNADSUM SYSIN: REPORTLEVEL=4
CMNADSUM SYSIN: IGNORE=DP
CMNADSUM SYSIN: IGNORE=00

----------------------------------------------------------------------------------------------
CMNADSUM        Processing completed at 03:48:50 on 03/08/2021 MAX RC = 00

AUDITSUM:

**********************************************************************************************
*               Summary of audit errors for audited package: STEV000456 *
**********************************************************************************************
```

```
*******************************************
* Text based component stand-alone errors *
*******************************************

        Lib Sync Component              Baseline            Baseline    Staging
Staging              Staging
Appl    Typ Err     Name                Timestamp           Userid      Package
Timestamp            Userid
____ ___ ____ ________________________________ ___________________ ________ __________
___________________ ________
STEV    JCL *06*    CCC00100            2016-05-28-11.44.00 SDOWNES     STEV000455
2021-03-05-03.14.00   WSER58
STEV    JCL *06*    CCC00101            2016-05-28-11.44.00 SDOWNES     STEV000455
2021-03-05-03.14.00   WSER58
STEV    JCL *06*    CCC00102            2016-05-28-11.44.00 SDOWNES     STEV000455
2021-03-05-03.14.00   WSER58
STEV    JCL *06*    CCC00104            2016-05-28-11.44.00 SDOWNES     STEV000455
2021-03-05-03.14.00   WSER58
STEV    JCL *06*    CCC00105            2016-05-28-11.44.00 SDOWNES     STEV000455
2021-03-05-03.14.00   WSER58

******************************************
* Source-to-copybook relationship errors *
******************************************

Main component                                      Included component
Appl Ltp Err  Name      Package     Timestamp         Userid Name           Package
Timestamp         Userid
____ ___ ____ ____________________ __________ _________________ ________
____________________ __________ ________
STEV SRC *05* RENAME01  Baseline    16-05-24-02.03.00 WSER58 ACHILL01       STEV000455
21-03-05-03.15.00 WSER58
STEV SRC *05* RENAME02  Baseline    16-05-24-02.03.00 WSER58 ACHILL01       STEV000455
21-03-05-03.15.00 WSER58
STEV SRC *05* RENAME93  Baseline    16-05-24-02.03.00 WSER58 ACHILL01       STEV000455
21-03-05-03.15.00 WSER58
STEV SRC *05* SCRTCH01  Baseline    16-05-24-02.04.00 WSER58 ACHILL01       STEV000455
21-03-05-03.15.00 WSER58
STEV SRC *05* SCRTCH02  Baseline    16-05-24-02.04.00 WSER58 ACHILL01       STEV000455
21-03-05-03.15.00 WSER58
STEV SRC *05* SOLSTICE  Baseline    16-05-11-08.58.00 WSER58 ACHILL01       STEV000455
21-03-05-03.15.00 WSER58
STEV SRC *05* SOLSTICE  Baseline    16-05-11-08.58.00 WSER58 ACHILL02       STEV000455
21-03-05-03.16.00 WSER58
STEV SUS *05* NEWPORT   Baseline    09-02-04-23.05.00 WSER58 ACHILL01       STEV000455
21-03-05-03.15.00 WSER58
STEV SUS *05* NEWPORT   Baseline    09-02-04-23.05.00 WSER58 ACHILL02       STEV000455
21-03-05-03.16.00 WSER58
STEV SUS *05* WESTPORT  Baseline    08-06-10-06.12.00 WSER58 ACHILL01       STEV000455
21-03-05-03.15.00 WSER58
STEV SUS *05* WESTPORT  Baseline    08-06-10-06.12.00 WSER58 ACHILL02       STEV000455
21-03-05-03.16.00 WSER58
STV2 SR5 *15* XAPSRCA   STV2000024  21-03-05-06.44.00 WSER58 XAPCPY01
STV2 SR5 *15* XAPSRCA   STV2000024  21-03-05-06.44.00 WSER58 XAPCPY02       STEV000454
21-03-05-06.45.00 WSER58
STV2 SR5 *15* XAPSRCA   STV2000024  21-03-05-06.44.00 WSER58 XAPCPY03       STEV000454
21-03-05-06.41.00 WSER58
STV2 SR5 *15* XAPSRCB   STV2000024  21-03-05-06.45.00 Recomp XAPCPY02       STEV000454
21-03-05-06.45.00 WSER58
STV2 SR5 *05* XAPSRCC   Baseline    17-02-27-02.17.00 WSER58 XAPCPY03       STEV000454
```

```
21-03-05-06.41.00 WSER58
STV2 SR5 *05* XAPSRCD   Baseline    17-02-27-02.17.00 WSER58 XAPCPY03       STEV000454
21-03-05-06.41.00 WSER58
STV2 SR5 *05* XAPSRCE   Baseline    17-02-27-02.18.00 WSER58 XAPCPY03       STEV000454
21-03-05-06.41.00 WSER58
STV2 SR5 *05* XAPSRCXX  Baseline    18-06-29-08.42.00 WSER58 ACHILL01       STEV000455
21-03-05-03.15.00 WSER58

- End of summary report
-----------------------------------------------------------------------------------------

If component names are too long to fit in this summary then please refer to the main report.

************************************************************
* A brief explanation of each reported error code follows: *
************************************************************

RC(08) : SYNCH5! (cpy/hdr baseline problem)
RC(12) : SYNCH6! (Activity file not checked out)
RC(12) : SYNCH15! (Source to relationship problem)
```

There are 4 different sections which may be reported. (Two of these are shown in the above example.) These sections are:

1. Text based component stand-alone errors. Each error involves just a single component which is not like-LOD.
2. Binary component stand-alone errors (not shown). This section is similar to the text based component stand-alone errors section described above, except it involves like- LOD components.
3. Source-to-copybook relationship errors.
4. Composite executable-to-static subroutine errors (not shown).

Each error is reported in a single summary line:

- The error code is highlighted with an asterisk on either side.
- A short-description (like that produced by CMNAD500) is output at the end of the report for each error present in the report.
- The rc associated with each error can be changed by CMNEX020, in the same way as for the main package audit programs.
- The stand-alone sections contain information about the component in baseline followed by the component as present in a staging library.
- The relationship errors contain information about the main component and the included component on the same line.
- The userid in the source-to-copybook section contains Recomp if the main component is present in the staging library as a recompile.

Performing a Cross-Application Audit

The following items determine the application libraries that are included by default in the scope of an audit:

- Level of the package that you are auditing (simple, participating, complex/super).
- Your responses to prompts on the Audit Change Package (CMNAUDIT) panel.

The following table summarizes the application libraries that are included in an audit by default:

Package Level	Audit part. pkg value on CMNAUDIT Panel	Applications Included in the Audit by Default
Simple Package	Not applicable	The libraries of the application associated with the package.
Participating Package Being Audited as a Simple Package	1	The libraries of the application associated with the package.
Participating Package Being Audited as a Primary Package	2	The libraries of all the applications associated with all packages that participate in the complex or super package.
Participating Package	blank	The libraries of all the applications associated with all packages that participate in the complex or super package.
Complex or super package	blank	The libraries of all the applications associated with all packages that participate in the complex or super package.

Whether baseline and staging libraries or only staging libraries are included in the audit depends on whether you select the field **Audit staging libraries only** on the Audit Change Package (CMNAUDIT) panel.

You can override the default and include other application libraries in the audit as follows:

1. Bring up the Audit Change Package (CMNAUDIT) panel by selecting option 1 Build from the Primary Option Menu and then option 7 Audit from the Build Options (CMNBUILD) panel.
2. Select the field **Specify application scope**.

```
CMNAUDIT            Audit Change Package
Command ===> ______________________________________

Package . . . . . . . . ACTP000025
Audit part. pkg . . . . (1-As simple,2-As primary,3-By Department)
Autoresolve parms . . . _______________________________________________

Enter "/" to select option
__Audit staging libraries only
__ Include history records
__Format report for printing
/  Specify application scope
__ Suppress batch messages
__Include cross appl headings      Print top line only
__ Lock package                     Reset lock
__Auto resolve
__ Update only this pkg rc
Job statement information:
//USER015L JOB (SM-1IKF-SM),'CMN6 AUDIT     ',
//          CLASS=A,
//          MSGCLASS=X,
//          NOTIFY=USER015
```

3. The Application Scope Selection List (CMNAUDAP) panel is displayed. The default applications are already selected on this panel.

```
CMNAUDAP    Application Scope Selection List    Row 1 to 3 of 3
Command ===>__________________________________ Scroll ===> CSR

Appl Description
S    ACTP ACTP Accounts Payable (Base ZMF)
_    COMM COMM Common Components (Base ZMF)
S    GENL GENL General Ledger (Db2 Option)
******************************* Bottom of data ********************************
```

4. Type the letter **S** in the selection field preceding the application mnemonic to select other applications that you want to include in the audit. (In the example above, the ACTP application is selected by default because the package being audited, ACTP000025, is a simple package.) Type the letter S in the selection field associated with the GENL application to include the libraries of that application in the audit.

To select all applications listed on the panel, you can type SELECT ALL on the Command line. Note that your administrator can customize the CMN$$AUD skeleton to always perform as if a SELECT ALL has been selected. Refer to the ZMF Customization Guide.

Press **PF3/End** to submit the audit job or CANCEL to return to the CMNAUDIT panel.

Use Case Scenario: Fixing Out-of-Sync Conditions

This use case scenario:

- Shows you how to request an audit.
- Explains how to locate the out-of-sync conditions that are flagged in the audit report.
- Explains why the out-of-sync conditions occurred.
- Illustrates a step-by-step process for resolving the out-of-sync conditions.
- Explains why the sequence of steps you take to resolve out-of-sync conditions is important.
- Gives useful tips.

The Sample Application

This scenario uses a complex package (ACTP000029) that has two participating packages:

- ACTP000028, which contains components from the sample ACTP application.
- GENL000006, which contains components from the sample GENL application.

  In this example you have staged the following components in package GENL000006:

- A copybook named GNLCPY00, which you have modified.
- A subprogram named GNLSRS00, which you have modified and which also references the copybook GNLCPY00.

Requesting an Audit

Take the following actions:

1. Bring up the Audit Change Package (CMNAUDIT) panel (=1.7). On the Audit Change Package (CMNAUDIT) panel, you:

   Specify the complex package ACTP000029 as the Package.

   Only accept the one option, **Include cross appl headings**.

```
CMNAUDIT            Audit Change Package
Command ===> ___________________________________________________
Package . . . . . . . . ACTP000029
Audit part. pkg . . . . . _ (1-As simple,2-As primary,3-By Department)
Autoresolve parms . . ._______________________________________________

Enter "/" to select option
_Audit staging libraries only
_ Include history records
_Format report for printing
_ Specify application scope
_Suppress batch messages
/ Include cross appl headings       Print top line only
_ Lock package                      Reset lock
_Auto resolve
_ Update only this pkg rc
Job statement information:
//USER015L JOB (SM-1IKF-SM),'CMN6 AUDIT     ',
//       CLASS=A,
//       MSGCLASS=X,
//       NOTIFY=USER015
```

2. Press Enter to submit the audit job.

   The system issues MVSSEND messages and also logs the messages in the AUDITRPT DD, which indicate that there are audit errors:

```
CMN2669I - COMPLEX PACKAGE ACTP000029 FAILED THE AUDIT WITH A RETURN CODE OF 12. CMN2666I -
PARTICIPATING PACKAGE ACTP000028 PASSED THE AUDIT WITH A RETURN CODE OF 00.

CMN3060A - This package has failed the audit.

CMN2667I - PARTICIPATING PACKAGE GENL000006 FAILED THE AUDIT WITH A RETURN CODE OF 12.
```

Examining the Audit Report

Here’s an excerpt from the Legend and Summary Report section of the audit report.

```
ChangeMan(R) ZMF (8.1.0 - 20141010) Audit TUESDAY JANUARY 6, 2015 (2015/006)
18:18:13           Page 13
                                                 Legend and Summary Report
The local level of audit chosen at this point;    4
    4 - Audit is required and the return code must not exceed 4 which implies
    that there are no "out-of-synch" situations within the staging libraries
    nor the baseline libraries but at least one module of a staging library
    is a "duplicate" of its baseline counterpart
Out-of-synch messages (hint - search for "!" marks)
    DUPLIC!  (Staging duplicates baseline)        ===>  0
    SYNCH0!  (Not in scope of audit or unknown)   ===> 22
    SYNCH1!  (Cmpnt statistics not available)      ==>  0
    SYNCH2!  (Compile/designated proc differ)     ===>  0
    SYNCH3!  (Unparsable load module)             ===>  0
    SYNCH4!  (cpy/hdr staging problem)            ===>  1
    SYNCH5!  (cpy/hdr baseline problem)           ===>  6
    SYNCH6!  (Activity file not checked out)      ===>  0
    SYNCH7!  (Static subcomponent stage problem)   ==>  0
    SYNCH8!  (Static subcomponent base problem)   ===>  2
    SYNCH9!  (Source and load discrepancy)        ===>  0
    SYNCH10! (Version regression problem)         ===>  0
    SYNCH11! (Component hash discrepancy)         ===>  0
    SYNCH12! (Orphan module in staging)           ===>  0
    SYNCH13! (Baseline/staging discrepancy)       ===>  0
    SYNCH14! (Components not in active status)    ===>  0
    SYNCH15! (Source to relationship problem)     ===>  0
    SYNCH16! (CPY low-date problem in baseline)   ===>  0
    SYNCH17! (CPY deleted problem in staging)     ===>  0
    SYNCH18! (LOD deleted problem in staging)     ===>  0
    SYNCH19! (Missing module in staging)          ===>  0
    SYNCH20! (Inconsistent subroutine)            ===>  0
    SYNCH21! (Scr/rename pkg component)           ===>  0
    SYNCH22! (Scratch subcmpnt is in use)         ===>  0
    SYNCH23! (Rename subcmpnt is in use)          ===>  0
    SYNCH24! (LOD low-date problem in baseline)   ===>  0
    SYNCH25! (LCT high-date problem in package)   ===>  0
    SYNCH26! (LCT -> missing executable)          ===>  0
    SYNCH27! (Missing LCT for executable)         ===>  0
    SYNCH28! (Same component in multiple pkgs)    ===>  0

Highest return code encountered                   ===> 12
```

It shows the following conditions:

- Twenty two SYNCH0! conditions.

  A SYNCH0! condition identifies a component that cannot be found in the scope of the audit (that is, in the baseline or staging libraries that are examined). These components are typically runtime language environment modules to which the compiler automatically generates references and which the binder includes in the load modules that are generated for the application. SYNCH0! conditions generate a return code of 0. You can typically ignore them; they do not prevent a package from passing the audit.

  Resolving the SYNCH0! Conditions in the Example provides more details about these SYNCH0! conditions.

- One SYNCH4! condition.

  Here we have a component GNLSRS00 in the package that has used the copybook GNLCPY00 (also in the package), but the copybook GNLCPY00 has been changed since. Recompile the GNLSRS00 component.

- Six SYNCH5! conditions.

  When you stage a copybook into a package, the source components that reference the copybook but that are not also staged into the package receive a SYNCH5! condition. This makes sense, because those non-staged source components now reference an obsolete version of the copybook. You typically have to recompile those source components from their baseline libraries to remove the SYNCH5! flags.

- Two SYNCH8! conditions.

When you stage the source code for a statically linked subprogram into a package and change it, the composite load modules that call this subprogram but that are not also staged into the package receive a SYNCH8! condition. This also makes sense, because those non-staged composite load modules now call an obsolete version of the subprogram. You typically have to relink these composite load modules from their baseline libraries to remove the SYNCH8! flags.

We examine the SYNCH8! conditions that are generated in this sample audit, and show you how to resolve them in Resolving the SYNCH8! Conditions in the Example

Sequencing Considerations

You should plan a sync-resolution strategy before you start to resolve the out-of-sync conditions. The sequence in which you correct the conditions is important. In general, you should recompile and relink subprograms that reference obsolete versions of a copybook before you relink the composite modules that call those subprograms.

This makes sense if you think about it. If you relink a composite module first, and then recompile the subprograms that it calls, you will have to relink the composite module all over again because the subprograms will have a more recent link date than their caller.

Resolving the SYNCH0! Conditions in the Example

We have twenty two SYNCH0! conditions flagged in the sample audit report. If you search the sample report for the text SYNCH0!, you reach the following report section:

```
ChangeMan(R) ZMF (8.1.0 - 20141010) Audit TUESDAY JANUARY 6, 2015 (2015/006)
18:18:13           Page 11
***************************************************************************************
*Complex/Super Package      ===> ACTP000029 Created 2015/01/06 at 17:56:43 by USER015 *
*Participating Packages     ===> 2              Package Status: OPN                   *
*Component Analysis Type    ===> Static subroutines called within Load members        *
**********************************************************************************************
*--------- B a s e l i n e ---------*       *------------------ S t a g i n g
------------------*
*************************************
*****************************************************
Called Module                         Calling     Called                  Module
Module   Size      Linkdate  Setssi   Module      Module  Package  Dept   Size
Linkdate    Setssi
________ ________ __________ ________ ________ ________ __________ ____ ________
__________  ________
         00001080 2015/01/05 677A9E06 ACPSRC91          ACTP000028 IDD  00001080
2015/01/06  677C168B
CEEARLU  --  Not found or unknown      SYNCH0!
CEEBETBL --  Not found or unknown      SYNCH0!
CEEBINT  --  Not found or unknown      SYNCH0!
CEEBLLST --  Not found or unknown      SYNCH0!
CEEBPIRA --  Not found or unknown      SYNCH0!
CEEBPUBT --  Not found or unknown      SYNCH0!
CEEBTRM  --  Not found or unknown      SYNCH0!
CEECPYRT --  Not found or unknown      SYNCH0!
CEESG005 --  Not found or unknown      SYNCH0!
CEESTART --  Not found or unknown      SYNCH0!
IGZCBSO  --  Not found or unknown      SYNCH0!
        00001080  2015/01/05 677A9E3C ACPSRC99          ACTP000028 IDD   00001080
2015/01/06  677C1D97
CEEARLU  --  Not found or unknown      SYNCH0!
CEEBETBL --  Not found or unknown      SYNCH0!
CEEBINT  --  Not found or unknown      SYNCH0!
CEEBLLST --  Not found or unknown      SYNCH0!
CEEBPIRA --  Not found or unknown      SYNCH0!
CEEBPUBT --  Not found or unknown      SYNCH0!
CEEBTRM  --  Not found or unknown      SYNCH0!
CEECPYRT --  Not found or unknown      SYNCH0!
CEESG005 --  Not found or unknown      SYNCH0!
CEESTART --  Not found or unknown      SYNCH0!
IGZCBSO  --  Not found or unknown      SYNCH0!
```

In this section of the report:

- The section heading indicates that this section of the report analyzes statically linked subprograms that are called by main program modules in the LOD library type of the ACTP application.
- The twenty two subroutines that are flagged with the SYNCH0! condition are all called by main programs ACPSRC91 and ACPSRC99 (which you staged into package ACTP000028).
- You did not stage these twenty two subroutines into the package (the *Staging* side of the report has no entries for these routines).

- ChangeMan ZMF cannot locate these subroutines in the baseline LOD library of the ACTP application (that's what SYNCH0! means).
- Upon closer inspection of the subroutine names, you realize that the routines are all IBM language environment runtime modules. The compiler automatically generates a reference to them at the time you compile the caller (ACPSRC91 and ACPSRC99).
- The return code of a SYNCH0! condition is zero, which means that you can ignore the condition: it does not prevent your package from passing the audit.

> **Note**
>
> You can suppress the appearance of SYNCH0! conditions in an audit report with user exit program CMNEX022. Refer to the ChangeMan ZMF Customization Guide for details.

Resolving the SYNCH5! Conditions in the Example

There are six SYNCH5! conditions in the sample audit report:

- Five SYNCH5! conditions are flagged for statically linked subroutines. These SYNCH5! conditions are described and resolved in Resolving a SYNCH5! Condition for a Statically Linked Subroutine.
- The sixth SYNCH5! condition is flagged for the main program module. This SYNCH5! condition is discussed in Resolving the SYNCH5! Condition for a Main Program.

Resolving a SYNCH5! Condition for a Statically Linked Subroutine

A search on the text SYNCH5! brings up the section of the audit report that describes Copybooks Found within Source Code for the SRS library type of the GENL application.

This section lists the baselined copybooks that each subroutine that is listed in the Source Name column references (see the Baseline side of the report section).

```
    ChangeMan(R) ZMF (8.1.0 - 20141010) Audit TUESDAY JANUARY 6, 2015 (2015/006)
18:18:13                   Page 9
***************************************************************************************
*Complex/Super Package      ===> ACTP000029 Created 2015/01/06 at 17:56:43 by USER015 *
*Participating Packages     ===> 2 Package Status: OPN                                *
*Component Analysis Type    ===> Copybooks found within Source code                   *
**********************************************************************************************
*--------------------- B a s e l i n e --------------------*       *----------------------
S t a g i n g ----------------------*
*************************************************************
*************************************************************
Copybook                                    Source Copybook
Name     VV.MM Created    Changed          Size Init Tso-id   Name     Name     VV.MM
Package     Changed       Size  Dept Tso-id
________ _____ __________ ________________ ____ ____ ________ ________ ________ _____
__________ ________________ ____ ____ ________
         01.02 2014/12/19 2015/01/04 19:34 23   21   USER015 GNLSRS00          02.01
GENL000006  2015/01/06 17:21 24   IDD  USER015
GNLCPY00 03.01 2002/05/07 2015/01/01 21:43 5     1   USER015 SYNCH4! GNLCPY00   04.01
GENL000006  2015/01/06 18:03  6   IDD  USER015
         01.01 2015/01/01 2015/01/04 19:48 29   29   USER015 GNLSRS1B
GNLCPY00 03.01 2002/05/07 2015/01/01 21:43 5     1   USER015 SYNCH5! GNLCPY00   04.01
GENL000006  2015/01/06 18:03  6   IDD  USER015
GNLCPY1B 01.01 2014/12/23 2014/12/23 17:41 4     4   USER015
GNLCPY1C 01.01 2014/12/23 2014/12/23 17:38 4     4   USER015
         01.02 2014/12/23 2014/12/23 18:09 26   26   USER015 GNLSRS1C
GNLCPY00 03.01 2002/05/07 2015/01/01 21:43 5     1   USER015 SYNCH5! GNLCPY00   04.01
GENL000006  2015/01/06 18:03  6   IDD  USER015
GNLCPY1C 01.01 2014/12/23 2014/12/23 17:38 4     4   USER015
         01.04 2002/05/07 2014/12/23 18:09 29   23   USER015 GNLSRS5A
GNLCPY00 03.01 2002/05/07 2015/01/01 21:43 5     1   USER015 SYNCH5! GNLCPY00   04.01
GENL000006  2015/01/06 18:03  6   IDD  USER015
GNLCPY1X 01.02 2002/05/07 2014/12/23 17:56 4     1   USER015
GNLCPY5A 01.02 2002/05/07 2014/12/23 17:56 5     1   USER015
GNLCPY5B 01.02 2002/05/07 2014/12/23 17:57 4     1   USER015
         01.02 2002/05/07 2014/12/23 18:05 27   15   USER015 GNLSRS5B
GNLCPY00 03.01 2002/05/07 2015/01/01 21:43 5     1   USER015 SYNCH5! GNLCPY00   04.01
GENL000006  2015/01/06 18:03  6   IDD  USER015
GNLCPY5B 01.02 2002/05/07 2014/12/23 17:57 4     1   USER015
GNLCPY5C 01.02 2002/05/07 2014/12/23 18:00 4     1   USER015
         01.02 2002/05/07 2014/12/23 18:06 25   15   USER015 GNLSRS5C
GNLCPY00 03.01 2002/05/07 2015/01/01 21:43 5     1   USER015 SYNCH5! GNLCPY00   04.01
GENL000006  2015/01/06 18:03  6   IDD  USER015
GNLCPY5C 01.02 2002/05/07 2014/12/23 18:00 4     1   USER015
```

The GNLCPY00 copybook is flagged with a SYNCH5! condition for the following statically linked subroutines:

- GNLSRS1B
- GNLSRS1C
- GNLSRS5A
- GNLSRS5B
- GNLSRS5C

The line item for each of these subroutines on the Staging side of the report is blank, which means that you did not stage the subroutine into a package. However, because you changed copybook GNLCPY00 and staged it into the package, these five baselined subroutines now reference an obsolete copy of the GNLCPY00 copybook.

To resolve the SYNCH5! conditions, you need to recompile and relink each subroutine from baseline (option =1.8, Recompile Source Code from Baseline or Production). This action will place a new load module of each subroutine into your package. This action does not stage the corresponding source module into the package. You don’t need the source modules in the package because you are not changing them. That is, you only need a new load module to ensure that the modules reference the latest version of the copybook when you baseline the package.

Compare the entries for these subroutines with the entry for the GNLSRS00 subroutine (the first entry on the report), which shows staging information because you checked that subroutine out to the package and changed, recompiled, and relinked it. Therefore, it

references the latest version of the GNLCPY00 copybook and the reference is not flagged with a SYNCH5! condition.

To resolve these SYNCH5! conditions:

1. Select the Recompile option (=1.8) from the Build Options menu. The Recompile Source panel (CMNRCMPR) panel appears.

```
CMNRCMPR Recompile Source
Option ===>
blank Member selection list L List libraries containing component
Package . . . . . . . . . GENL000006
Component name . . . . . +
Source library type . . . SRS (Blank for list)
Library level . . . . . . 0 (Baseline 0, Promotion 1 to nn)
Recompile mode . . . . . 1 (1-Online, 2-Batch)
Enter "/" to select option
Search release areas
Specify release areas
```

2. Fill in the panel as follows:

   a. You get the Package Id (GENL000006, in this example) from the Copybooks Found within Source Code section of the audit report.

   b. The Source Library Type (SRS, in this example) is also identified in the Copybooks Found with Source Code section of the audit report.

   c. The Library Level is 0 (you want to recompile the baseline image of the subroutines in this example).

   d. You can take the action in online or batch mode. (We choose online in this example.)

3. If you have multiple subroutines to recompile, you can leave the Component Name blank to display the Recompile Member List (CMNRCMP2) panel, from which you can select the

subroutines that you want to recompile. In this example, the five subroutines that were associated with the SYNCH5! condition are selected (with the S line command in the line command field for each line item):

```
CMNRCMP2 Recompile Member List          Row 1 to 7 of 7
Command ===>__________________________ Scroll ===> CSR
                                                    ID
Input library:
CMNTP.S6.V810.BASE.GENL.SRS

   Name     Function vv.mm Created    Changed          Size Init User
__ GNLSRS00          01.02 2014/12/19 2015/01/04 19:34 23   21   USER015
__ GNLSRS01          01.03 2008/09/08 2014/12/23 18:08 23   22   USER015
s  GNLSRS1B          01.01 2015/01/01 2015/01/04 19:48 29   29   USER015
s  GNLSRS1C          01.02 2014/12/23 2014/12/23 18:09 26   26   USER015
s  GNLSRS5A          01.04 2002/05/07 2014/12/23 18:09 29   23   USER015
s  GNLSRS5B          01.02 2002/05/07 2014/12/23 18:05 27   15   USER015
s  GNLSRS5C          01.02 2002/05/07 2014/12/23 18:06 25   15   USER015
******************************* Bottom of data  ********************************
```

4. Press Enter to submit the recompile requests. The Confirm Recompile Request (CMNRCMPC) panel is displayed for each subroutine that you have selected. Press Enter to confirm each request and continue.

```
CMNRCMPC                Confirm RECOMPILE Request
Command ===>______________________________________________
Package:        GENL000006
Component name: GNLSRS1B +
          type: SRS
Language:       COBOL2
Changed date:   20150104
        time:   19:48
Procedure:      CMNCOB2
User:           USER015
Site:           BASELINE
Level:
```

5. The Online Recompile Job Information (CMNRCMP1) panel appears. The fields on this panel will be filled in with the values used the last time you recompiled and relinked the target subroutine. You should not need to change these values.

```
CMNRCMP1    ONLINE Recompile Job Information        HISTORY ASSUMED
Command ===>_________________________________________

Package: GENL000006     Status: DEV     Install date: 20150228

Component:      GNLSRS1B                                          +
Library type: SRS - Source for subprograms to be Linked NCAL
Dataset name: CMNTP.S6.V810.BASE.GENL.SRS                         +

Language . . . . . . . COBOL2 (Blank for list)
Compile procedure . . CMNCOB2 (Blank for list; ? for designated procedure)
Compile parms . . . . _________________________________________
Pgm binder parms . . . NCAL

Enter "/" to select option
__ Db2 precompile
__ Precompile variables
__ Display Other options
__ Suppress batch messages

Job statement information:
//USER0151 JOB (SM-1IKF-SM),'GENL06',
//       CLASS=A,
//       MSGCLASS=X,
//       NOTIFY=USER015
```

6. Press Enter to submit the job.
7. You will repeat the steps described here for each subroutine that you have selected to recompile and relink on the Recompile Member List panel shown in Step 3.

You can often introduce more out-of-sync conditions in your packages by submitting jobs to resolve the existing ones in the incorrect sequence. To avoid this problem, wait until you have received notification that the like-load library members for each of the subroutines are active in your package before you continue resolving the out-of-sync conditions that are flagged in the audit.

> **Tip**
>
> If you are unsure of the status of the components in a package, you can display all of the component statuses by selecting option 5 (List) from the Primary Options menu then select the package with an S2, and observe the Status column.

Resolving the SYNCH5! Condition for a Main Program

The reference to copybook GNLCPY00 in module GNLSRC1A is flagged with the remaining SYNCH5! condition. In our sample application, GNLSRC1A is a main program. In the audit report, this SYNCH5! appears in the Copybooks Found within Source Code section for the SRC library type of the GENL application.

```
ChangeMan(R) ZMF (8.1.0 - 20141010) Audit TUESDAY JANUARY 6, 2015 (2015/006) 18:18:13       Page 10
**************************************************************************************
*Complex/Super Package   ===> ACTP000029 Created 2015/01/06 at 17:56:43 by USER015   *
*Participating Packages  ===> 2 Package Status: OPN                                  *
*Component Analysis Type ===> Copybooks found within Source code                     *
************************************************************************************************************************************
*-------------------- B a s e l i n e ---------------------*   *---------------------- S t a g i n g ----------------------*
*************************************************************   *************************************************************
Copybook                                                       Source Copybook
Name      VV.MM Created  Changed           Size Init Tso-id    Name    Name    VV.MM  Package    Changed          Size Dept Tso-id
________ _____ __________ ________________ ____ ____ ________ ________ ________ _____ __________ ________________ ____ ____ ________
         01.01 2002/05/07 2015/01/05 01:22 28   23    USER015 GNLSRC1A
GNLCPY00 03.01 2002/05/07 2015/01/01 21:43 5     1    USER015 SYNCH5! GNLCPY00 04.01 GENL000006   2015/01/06 18:03 6   IDD   USER015
GNLCPY1A 01.02 2002/05/07 2014/12/23 17:55 5     1    USER015
GNLCPY1B 01.01 2014/12/23 2014/12/23 17:41 4     4    USER015
GNLCPY1X 01.02 2002/05/07 2014/12/23 17:56 4     1    USER015
```

The line item for main program GNLSRC1A on the Staging side of the report is blank, which indicates that you did not stage the main program into a package. Because you did change copybook GNLCPY00 and stage it into the package, this main program now references an obsolete copy of the GNLCPY00 copybook and is therefore flagged with a SYNCH5! condition.

To resolve this condition, you need to recompile and relink the main program from baseline (option =1.8, Recompile). This action will place a new load image of the main program into your package. However, the corresponding source program is not staged into the package. You do not need the source program in your package because you are not changing it. You only need a new load image to ensure that the module references the latest version of the copybook when you baseline the package.

You followed the same procedure described in Resolving a SYNCH5! Condition for a Statically Linked Subroutine, so we will not repeat the procedure here.

Running Another Audit

All the SYNCH5! conditions that were flagged in the preceding audit report should now be resolved. You can rerun the audit at any time to check your progress. Here’s the Legend and Summary Report section from a new audit report:

```
    ChangeMan(R) ZMF (8.1.0 - 20141010) Audit TUESDAY JANUARY 6, 2015   (2015/006)
19:49:46         Page 14

                                              Legend and Summary Report
The local level of audit chosen at this point; 4
     4 - Audit is required and the return code must not exceed 4 which implies
         that there are no "out-of-synch" situations within the staging libraries
         nor the baseline libraries but at least one module of a staging library
         is a "duplicate" of its baseline counterpart
Out-of-synch messages (hint - search for "!" marks)
    DUPLIC!  (Staging duplicates baseline)       ===> 0
    SYNCH0!  (Not in scope of audit or unknown)  ===> 33
    SYNCH1!  (Cmpnt statistics not available)    ===> 0
    SYNCH2!  (Compile/designated proc differ)    ===> 0
    SYNCH3!  (Unparsable load module)            ===> 0
    SYNCH4!  (cpy/hdr staging problem)           ===> 0
    SYNCH5!  (cpy/hdr baseline problem)          ===> 0
    SYNCH6!  (Activity file not checked out)     ===> 0
    SYNCH7!  (Static subcomponent stage problem) ==>  1
    SYNCH8!  (Static subcomponent base problem)  ===> 4
    SYNCH9!  (Source and load discrepancy)       ===> 0
    SYNCH10! (Version regression problem)        ===> 0
    SYNCH11! (Component hash discrepancy)        ===> 0
    SYNCH12! (Orphan module in staging)          ===> 0
    SYNCH13! (Baseline/staging discrepancy)      ===> 0
    SYNCH14! (Components not in active status)   ===> 0
    SYNCH15! (Source to relationship problem)    ===> 0
    SYNCH16! (CPY low-date problem in baseline)  ===> 0
    SYNCH17! (CPY deleted problem in staging)    ===> 0
    SYNCH18! (LOD deleted problem in staging)    ===> 0
    SYNCH19! (Missing module in staging)         ===> 0
    SYNCH20! (Inconsistent subroutine)           ===> 0
    SYNCH21! (Scr/rename pkg component)          ===> 0
    SYNCH22! (Scratch subcmpnt is in use)        ===> 0
    SYNCH23! (Rename subcmpnt is in use)         ===> 0
    SYNCH24! (LOD low-date problem in baseline)  ===> 0
    SYNCH25! (LCT high-date problem in package)  ===> 0
    SYNCH26! (LCT -> missing executable)         ===> 0
    SYNCH27! (Missing LCT for executable)        ===> 0
    SYNCH28! (Same component in multiple pkgs)   ===> 0

Highest return code encountered           ===>12
```

Note that:

- All the SYNCH5! conditions have been fixed (which you expect).
- There are eleven new SYNCH0! conditions. Thirty-three SYNCH0! conditions are now flagged instead of the original twenty-two. This is to be expected. Here’s why: During the recompile of main program GNLSRC1A, the compiler automatically generated references to more IBM Language Environment routines that ChangeMan ZMF knows nothing about.
- You can ignore the SYNCH0! messages. They result in a zero return code and do not prevent the package from passing the audit.

Note

Your ChangeMan ZMF administrator can suppress the appearance of SYNCH0!conditions in an audit report with a user exit modification. Refer to the ChangeMan ZMF Customization Guide for details.

- One SYNCH7! condition. This was a result of doing the recompile for one of the subroutines without later recompiling the GNLSRC1A component which statically links it. A quick recompile resolved that.
- Four SYNCH8! conditions are flagged instead of the original two. Here’s why: Recompiling main program GNLSRC1A in a preceding step to resolve a SYNCH5! condition also resolved one of the original SYNCH8! conditions. The process of

recompiling the GNLSRC5A, GNLSRC5B, and GNLSRC5C subroutines, however, introduced three new SYNCH8! conditions.

Tip

The act of resolving one type of out-of-sync condition frequently introduces other out-of-sync conditions. You can expect this to happen. If you have carefully planned the sequence of actions to resolve the out-of-sync conditions, you will eventually get a clean audit. See Sequencing Considerations for guidance.

Resolving the SYNCH7! condition in the example

This excerpt shows the SYNCH7! condition in the audit:

```
ChangeMan(R) ZMF (8.1.0 - 20141010) Audit TUESDAY JANUARY 6, 2015 (2015/006) 19:49:46 Page
13
*************************************************************************************
*Participating Package      ===> GENL000006 Created 2015/01/06 at 17:18:12 by USER015 *
*Package Installation Date  ===> 2015/02/28 Package Status: DEV Department IDD        *
*Component Analysis Type    ===> Static subroutines called within Load members       *
*Load member Appl:Libtype   ===> GENL:LOD                                             *
**********************************************************************************************
*--------- B a s e l i n e ---------*       *------------------ S t a g i n g
------------------*
************************************* *****************************************************
Called   Module                       Calling  Called    Module
Module   Size      Linkdate   Setssi   Module   Module   Package     Dept Size
Linkdate   Setssi
________ ________ __________  ________ ________ ________ __________ ____ ________
__________ ________

          00001EB0 2015/01/05 6779DECC GNLSRC1A           GENL000006 IDD 00001EB0
2015/01/06 677C2F59
CEEARLU  -- Not found or unknown SYNCH0!
CEEBETBL -- Not found or unknown SYNCH0!
CEEBINT  -- Not found or unknown SYNCH0!
CEEBLLST -- Not found or unknown SYNCH0!
CEEBPIRA -- Not found or unknown SYNCH0!
CEEBPUBT -- Not found or unknown SYNCH0!
CEEBTRM  -- Not found or unknown SYNCH0!
CEECPYRT -- Not found or unknown SYNCH0!
CEESG005 -- Not found or unknown SYNCH0!
CEESTART -- Not found or unknown SYNCH0!
GNLSRS00 00000458 2015/01/05 6779D8A9  SYNCH7!  GNLSRS00  GENL000006 IDD 00000458
2015/01/06 677C1124
GNLSRS1B 000004C8 2015/01/05 6779D8A9           GNLSRS1B  GENL000006 IDD 000004C8
2015/01/06 677C2D02
GNLSRS1C 00000498 2015/01/05 6779D8A9           GNLSRS1C  GENL000006 IDD 00000498
2015/01/06 677C2D11
IGZCBSO  -- Not found or unknown SYNCH0!

         00001EB0 2015/01/05 6779D9B2  GNLSRC50
GNLSRS00 00000458 2015/01/05 6779D8A9  SYNCH8!  GNLSRS00  GENL000006 IDD 00000458
2015/01/06 677C338E
GNLSRS5A 000004C8 2015/01/05 6779D8A9  SYNCH8!  GNLSRS5A  GENL000006 IDD 000004C8
2015/01/06 677C2D11
GNLSRS5B 000004C8 2015/01/05 6779D8B6  SYNCH8!  GNLSRS5B  GENL000006 IDD 000004C8
2015/01/06 677C2D1F
GNLSRS5C 00000498 2015/01/05 6779D8B6  SYNCH8!  GNLSRS5C  GENL000006 IDD 00000498
2015/01/06 677C2D2C
```

Note

The setssi values for GNLSRS00 both in the GNLSRC1A module and in the GNLSRC50 module - they are different and the one in the GNLSRC1A module is earlier than the setssi value for GNLSRS00 in GNLSRC50. To resolve that, all that is needed is to recompile GNLSRC1A.

Resolving the SYNCH8! Conditions in the Example

The following excerpt from the next audit report shows the four SYNCH8! conditions.

```
ChangeMan(R) ZMF (8.1.0 - 20141010) Audit TUESDAY JANUARY 6, 2015 (2015/006)
19:55:16             Page 13
**************************************************************************************
*Participating Package      ===> GENL000006 Created 2015/01/06 at 17:18:12 by USER015 *
*Package Installation Date  ===> 2015/02/28 Package Status: DEV Department IDD        *
*Component Analysis Type    ===> Static subroutines called within Load members        *
*Load member Appl:Libtype   ===> GENL:LOD                                             *
**********************************************************************************************
*--------- B a s e l i n e ---------*          *------------------ S t a g i n g
------------------*
************************************* *****************************************************
Called   Module                       Calling  Called   Module
Module   Size     Linkdate   Setssi   Module   Module   Package   Dept Size     Linkdate
Setssi
________ ________ __________ ________ ________ ________ __________ ____ ________ __________
________
         00001EB0 2015/01/05 6779DECC GNLSRC1A          GENL000006 IDD  00001EB0 2015/01/06
677C34D6
CEEARLU  -- Not found or unknown SYNCH0!
CEEBETBL -- Not found or unknown SYNCH0!
CEEBINT  -- Not found or unknown SYNCH0!
CEEBLLST -- Not found or unknown SYNCH0!
CEEBPIRA -- Not found or unknown SYNCH0!
CEEBPUBT -- Not found or unknown SYNCH0!
CEEBTRM  -- Not found or unknown SYNCH0!
CEECPYRT -- Not found or unknown SYNCH0!
CEESG005 -- Not found or unknown SYNCH0!
CEESTART -- Not found or unknown SYNCH0!
GNLSRS00 00000458 2015/01/05 6779D8A9 GNLSRS00          GENL000006 IDD 00000458  2015/01/06
677C338E
GNLSRS1B 000004C8 2015/01/05 6779D8A9 GNLSRS1B          GENL000006 IDD 000004C8  2015/01/06
677C2D02
GNLSRS1C 00000498 2015/01/05 6779D8A9 GNLSRS1C          GENL000006 IDD 00000498  2015/01/06
677C2D11
IGZCBSO  -- Not found or unknown SYNCH0!
         00001EB0 2015/01/05 6779D9B2 GNLSRC50
GNLSRS00 00000458 2015/01/05 6779D8A9 SYNCH8!  GNLSRS00 GENL000006 IDD 00000458  2015/01/06
677C338E
GNLSRS5A 000004C8 2015/01/05 6779D8A9 SYNCH8!  GNLSRS5A GENL000006 IDD 000004C8  2015/01/06
677C2D11
GNLSRS5B 000004C8 2015/01/05 6779D8B6 SYNCH8!  GNLSRS5B GENL000006 IDD 000004C8  2015/01/06
677C2D1F
GNLSRS5C 00000498 2015/01/05 6779D8B6 SYNCH8!  GNLSRS5C GENL000006 IDD 00000498  2015/01/06
677C2D2C
```

A SYNCH8! condition occurs when statically linked subroutines in a staging library are changed more recently than a composite load module in baseline that calls them. That is what happened here:

- You originally staged GNLSRS00 into the sample GENL000006 change package.

- You staged load components for GNLSRS5A, GNLSRS5B, and GNLSRS5C when you recompiled and relinked these subroutines to fix the SYNCH5! conditions described in Resolving the SYNCH5! Conditions in the Example.
- The composite load module GNLSRC50, which resides in a baseline library and which you did not stage into the sample packages used in this example, refers to these four subroutines. Thus, GNLSRC50 now calls obsolete versions of these statically linked subroutines.

You should be familiar with the components of the applications you work with so you know what to look for when you resolve an out-of-sync condition. In our sample application, GNLSRC50 consists of link-edit control statements that gather these subroutines together into a composite load module named GNLSRC50. GNLSRC50 is a member of the LCT library associated with the sample GENL application. Here's a listing of the contents of this component:

```
BROWSE CMNTP.S6.V810.BASE.GENL.LCT(GNLSRC50) - 01.01 Line 00000000 Col 001 080
Command ===> ________________________________________________ Scroll ===> CSR
********************************* Top of Data **********************************
INCLUDE SYSLIB(GNLSRS5A)
INCLUDE SYSLIB(GNLSRS5B)
INCLUDE SYSLIB(GNLSRS5C)
INCLUDE SYSLIB(GNLSRS00)
    NAME GNLSRC50(R)
******************************** Bottom of Data ********************************
```

You need to relink this composite load module to resolve the SYNCH8! conditions. Take the following steps:

1. Select Relink (=1.9) and the Relink Load Modules (CMNRLNK0) panel is displayed.

```
CMNRLNKR                Rebind Load Modules
Command ===>____________________________________________
Package . . . . . . . . . GENL000006
Rebind from . . . . . . . B                   (S(tage),B(aseline),R(elease))
Input library type . . .  LOS                 (Blank for list)
Target library type . . . LOD                 (Blank for list)
Member name . . . . . . . _________________+ (Blank/Pattern for list)

Enter "/" to select option
__ Specify release area
/  LCT member list
```

2. Fill in the panel as shown above:

   a. Package GENL000006 contains load images of the statically linked subroutines that are to be relinked into the GNLSRC50 composite.

   b. You want to relink from the Baseline library because the target LCT member has not been staged into the package.

   c. Select LCT Member List because the LCT member (GNLSRC50) exists.

d. Specify the library type of the subroutines to be relinked in response to the Input Library Type prompt. The library type associated with the load images of these subroutines is LOS in this example.

e. Specify the library type of the composite load module that will be created during the relink operation in response to the Output Library Type prompt. The library type associated with composite load modules is LOD in this example.

f. You can leave the member name blank and press Enter to get a list of LCT library members to choose from. If you do, the CMNRMLST panel is displayed. Select LCT member GNLSRC50.

```
CMNRMLST        Library Member List         Row 1 to 1 of 1
Command ===>______________________________ Scroll ===> CSR

Input library:
CMNTP.S6.V810.BASE.GENL.LCT

Name        +   Function Created        Changed          Size     User
s GNLSRC50               2015/01/01     2015/01/01 22:32 00000005 USER015
******************************* Bottom of data ********************************
```

3. Type the letter S in the line command field to select the desired member and press Enter.

   An informational panel appears that lists all currently active packages that include the target LCT member as a component (if any). If this panel appears, you need to check with the user specified in the `Notify user` field of the packages to be sure you understand how the component is being treated in those packages before you continue with the relink operation.

4. The Relink Job Information (CMNRLNK1) panel appears.

```
CMNRLNK1                Rebind Job Information
Command ===> ____________________________________________________

Package:         GENL000006        Status: BAS         Install date:

Member name:     GNLSRC50                                      +
Dataset name:    CMNTP.S6.V810.BASE.GENL.LCT                   +
LCT member list: YES Input library type: LOS Target library type: LOD

Language . . . . . . . . COBOL2 (Blank for list)
Compile procedure . . .  CMNCOB2 (Blank for list)
Pgm binder parms . . . . _________________________________

Enter "/" to select option
_ Db2 processing    _ Other Db2 options
_ Other options     _ User variables
_ Suppress batch messages

Job statement information:
//USER015R JOB (SM-1IKF-SM),'GENL06',
//       CLASS=A,
//       MSGCLASS=X,
//       NOTIFY=USER015
```

The fields are primed with the values that you specified on the Relink Load Modules (CMNRLNK0) panel above and with values that were specified the last time you relinked the module. You should not have to change the field values.

5. Press Enter to submit the job.
6. Examine the job's output to ensure that the target composite load module was created without error.
7. Run another audit. All the out-of-sync conditions in this scenario should now be resolved. The following messages are displayed after the audit completes, indicating that the complex package and its two participating packages passed the audit with zero return codes:

```
CMN2668I - COMPLEX PACKAGE ACTP000029 PASSED THE AUDIT WITH A RETURN CODE OF 00.
CN(INTERNAL)
CMN2666I - PARTICIPATING PACKAGE ACTP000028 PASSED THE AUDIT WITH A RETURN CODE OF 00.
CN(INTERNAL)
CMN2666I - PARTICIPATING PACKAGE GENL000006 PASSED THE AUDIT WITH A RETURN CODE OF 00.
CN(INTERNAL)
\*\*\*
```

Using Audit Auto Resolve

Audit auto resolve automatically submits stage, recompile, and relink jobs in an attempt to fix the following out-of-sync conditions detected by audit:

- SYNCH2!
- SYNCH4!
- SYNCH5!
- SYNCH7!
- SYNCH8!
- SYNCH9!
- SYNCH15!
- SYNCH16!
- SYNCH24!
- SYNCH25!
- SYNCH26!

You invoke the auto resolve facility of package audit when you set the **Auto Resolve** field to **YES** on the **Audit Change Package** panel.

After the jobs automatically submitted by audit auto resolve are completed, you must run another audit to prove that all out-of-sync conditions have been eliminated. As you saw in the example in Use Case Scenario: Fixing Out-of-Sync Conditions, resolving one out-of-sync condition may uncover another, so audit auto resolve may not resolve all errors in one execution.

Auto Resolve Actions

Auto resolve submits the following restage, recompile, and relink jobs under the conditions listed with each SYNCH! error.

SYNCH2!

Conditions	Auto Resolve Remedy
All	Restage component using the designated procedure.

SYNCH4!

Conditions	Auto Resolve Remedy
Source staged in package	Restage the source component at its current location.
Recompile in package	Recompile the component in the same package.

SYNCH5!

Conditions	Auto Resolve Remedy
1.Simple package audit 2.Source in same application as the package being audited	Recompile source in the package being audited.
1. Simple package audit 2. Source in a different application than the package being audited (cross application processing);	No recompile submitted because there is no eligible package. This is noted in the audit output.
1. Participating package audit 2. Source in one of the applications represented by the collection of eligible participating packages	Recompile source in the participating package with the latest install date, belonging to the source application.
	Notes: If there are several packages with the same install date, the one with the highest package number will be chosen.

Conditions	Auto Resolve Remedy
	If the baseline containing the source is shared across several applications, auto resolve tries to use the application where the source was last baselined. If this application is not represented in the collection of participating packages, auto resolve works back through component history until it finds the first package that is.
1. Participating package audit 2. Source can not be placed in any application represented by the collection of eligible participating packages;	No recompile submitted because there is no target package. This is noted in the audit output.
Complex package audit	Processing is the same as for participating package audit.

SYNCH7!

Conditions	Auto Resolve Remedy
Composite load in the package created by staging source in the package.	Restage of source in the package.
Composite load in the package created by recompile from baseline.	Recompile from baseline.
Composite load in the package relink from baseline.	Relink from baseline.

SYNCH8!

Conditions	Auto Resolve Remedy
Composite load in baseline is created from source component.	Recompile of source component.
Composite load in baseline is created from: Like-LOD Like-object Like NCAL LCT component	Relink
1. Simple package audit 2. Composite load in the same application as the package;	Action in the package being audited.

Conditions	Auto Resolve Remedy
1. Participating package audit 2. Composite load in one of the applications represented by the collection of eligible participating packages	Action in the participating package with the latest install date belonging to the executable application **Note:** - If there are several packages with the same install date, the one with the highest package number is chosen. - If the baseline containing the composite load is shared across applications, auto resolve tries to use the application where the composite load was last baselined. If this application is not represented in the collection of participating packages, auto resolve works back through component history until it finds the first package that is.
1. Participating package audit 2. Composite load cannot be placed in any application represented by the collection of eligible participating packages	No recompile submitted because there is no target package. This is noted in the audit output.
Complex package audit	Processing is the same as for participating package audit.

SYNCH9!

Conditions	Auto Resolve Remedy
All	Restage the source component in its current location.

SYNCH16!

Conditions	Auto Resolve Remedy
1. Simple package audit 2. Source in same application as the package being audited	Recompile source in the package being audited.
1. Simple package audit 2. Source in a different application than the package being audited (cross application processing);	No recompile submitted because there is no eligible package. This is noted in the audit output.
1. Participating package audit 2. Source in one of the applications represented by the collection of eligible participating packages	Recompile source in the participating package with the latest install date, belonging to the source application.
	Notes: If there are several packages with the same install date, the one with the highest package number will be chosen.

Conditions	Auto Resolve Remedy
	If the baseline containing the source is shared across several applications, auto resolve tries to use the application where the source was last baselined. If this application is not represented in the collection of participating packages, auto resolve works back through component history until it finds the first package that is.
1. Participating package audit 2. Source can not be placed in any application represented by the collection of eligible participating packages;	No recompile submitted because there is no target package. This is noted in the audit output.
Complex package audit	Processing is the same as for participating package audit.

SYNCH24!

Conditions	Auto Resolve Remedy
1 Simple package audit 2 Composite in same application as thepackage being audited	If we can determine the generating source for the composite, a baseline recompile is requested in the audited package. Otherwise, a baseline relink of the composite is requested.
1 Simple package audit 2 Composite in a different application from the package being audited (cross application processing)	No action submitted as there is no eligible package available. This is noted in the audit output.
1 Participating package audit 2 Composite in one of the applications represented by the collection of eligible participating packages	Recompile the composite source (if it can be determined), or relink the composite itself, in the participating package with the latest install date, belonging to the source application.
	Notes If there are several packages with the same install date, the one with the highest package number is chosen. If the baseline containing the source is shared across several applications, auto resolve tries to use the application where the source was last baselined. If this application is not represented in the collection of participating packages, auto resolve works back through component history until it finds the first package that includes this application.

Conditions	Auto Resolve Remedy
1 Participating package audit 2 Source can not be placed in any application represented by the collection of eligible participating packages.	No action submitted as there is no target package. This is noted in the audit output.
Complex package audit	Processing is the same as for participating package audit.

SYNCH25!

Conditions	Auto Resolve Remedy
1 Simple package audit 2 LCT in package 3 Target executable(s) generated by: - Source cmpnt in pkg - Recompile in pkg - Relink in package	Rebuild ‘source’ component in package, that is,one of: - Package build - Baseline recompile - Package relink
1 Participating package audit 2 LCT in package 3 Target executable(s) in same part. pkg	This is effectively the same as the simple package case.
1 Participating package audit 2 LCT in package 3 Target executable(s) in different part. pkg	The relevant rebuild action will take place in the same pkg as the target executables (unless over-ridden elsewhere, e.g., preferred pkg).

SYNCH26!

Conditions	Auto Resolve Remedy
1 Simple package audit 2 LCT in package 3 Target executable(s) have been deleted. They were originally generated by: - Source cmpnt in pkg - Recompile in pkg - Relink in package	If the LCT has never been used by a build action in the package, auto resolve does nothing. You must either delete the unnecessary LCT component from the package or use it in a manual build process.Auto resolve submits a relevant build job that uses an LCT component in cases where a previously built executable has been subsequently deleted. This includes: - A baseline recompile (as there is no source in package) - A baseline recompile - A package relink

Conditions	Auto Resolve Remedy
1 Participating package audit 2 LCT in package 3 Target executable(s) were in the same part. pkg before they were deleted.	This is effectively the same as the simple package case.
1 Participating package audit 2 LCT in package 3 Target executable(s) in different part. pkg prior before they were deleted.	The relevant rebuild action takes place in the same pkg as the target executables (unless over-ridden elsewhere, e.g., preferred pkg).

Duplicate or Overlapping Requests

When package audit detects an error that auto resolve can take action against, an auto resolve event is recorded. When audit processing finished, the list of auto resolve events is sorted and the following are dropped:

- The same action against the same component
- Different actions to create the same output

  For example, if there are two events recorded that require a restage of the same source component, one event is dropped. If a relink and a recompile are required that will build the same load module, the relink event is dropped.

Sequence of Auto Resolve Job Execution

The efficiency of the process to fix audit errors depends on the sequence of jobs that are executed to fix the errors, whether those jobs are submitted manually by you or automatically by audit auto resolve.

For example, statically linked subroutines should be rebuilt before composite loads that contain them are rebuilt.

When auto resolve submits jobs to fix audit errors, the sequence of file tailoring to create the job JCL is controlled in two places:

- The SEQ number in application library type definitions. This is the default.
- Run time parameters in an optional data set named in the **Auto Resolve Parms** field of the **Audit Change Package** panel where you request an audit.

At a minimum, file tailoring for stage, recompile, and relink jobs can be initiated in a sequence that favors the resolution of a maximum number of SYNCH errors.

However, control over the sequence of file tailoring may not control the actual execution sequence of the jobs.

Auto Resolve Runtime Parameters

Optional runtime parameters for audit auto resolve include parameters that give you control over:

- Sequence numbers that override the SEQ value in library type definitions to control the order that file tailoring is initiated for auto resolve jobs
- Auto resolve job name pattern so you can tell the purpose of each auto resolve job from the job name
- Whether auto resolve jobs are submitted with TYPRUN=HOLD so you can manually release the jobs in the sequence that resolves out-of-sync conditions most efficiently
- How job names are incremented so you can control simultaneous execution of auto resolve jobs

Run time parameters are input to audit in an optional data set named in the **Auto Resolve Parms** field of the **Audit Change Package** panel where you request an audit.

This is an example of a file of audit auto resolve runtime parameters.

```
JOBSTEM=USER15
******************************************************
* Seq. ; Appl ; Ltype ; Suffix ; TYPRUN ; Increment *
* No.  ;      ;       ;        ;        ;           *
******************************************************
001    ; COMM ; SRO   ; CMOX   ; RUN    ; RANGE=05
002    ; COMM ; SRN   ; CMNX   ; HOLD   ; RANGE=05
003    ; COMM ; SRC   ; CMCX   ; HOLD   ; RANGE=05
010    ; COMM ; OBJ   ; OBJX   ; HOLD   ; RANGE=05
011    ; COMM ; NCL   ; NCLX   ; HOLD   ; RANGE=05
012    ; COMM ; LOD   ; LODX   ; HOLD   ; RANGE=05
021    ; *    ; SRO   ; ALOX   ; HOLD   ; RANGE=05
022    ; *    ; SRN   ; ALNX   ; HOLD   ; RANGE=05
023    ; *    ; SRC   ; ALCX   ; HOLD   ; RANGE=05
031    ; *    ; OBJ   ; OBJX   ; HOLD   ; RANGE=05
032    ; *    ; NCL   ; NCLX   ; HOLD   ; RANGE=05
033    ; *    ; LOD   ; LODX   ; HOLD   ; RANGE=05
999    ; *    ; *     ; A      ; HOLD   ; NO
```

Parameter Syntax

These are the syntax rules for coding auto resolve runtime parameters except for the JOBSTEM parameter:

- Each line (record) consists of a set of values related to an application/library type combination
- Each value on a line is separated by a comma or a semi-colon
- Spaces are ignored
- Lines starting with an asterisk are treated as comments
- Wild cards are allowed
- Any number of lines can be used to specify the parameters required to drive your auto resolve process.
- Auto resolve uses the parameters associated with the most specific match to the application/library type of the auto resolve action.

This table describes the auto resolve parameters. In this table, application/library type is abbreviated as appl:libtype.

Field	Description
JOBSTEM	1 to 8 characters that define the first part of auto resolve job names
Seq No	Value 001 to 999 that sets the order of file tailoring for the appl:libtype of the component being "autoresolved'
	Within a single appl:libtype, components are processed alphanumerically, but stage jobs are file tailored before recompiles, before relinks.
	If the same sequence number is assigned to multiple appl:libtype combinations, then the combinations are taken in alphanumeric order.
Appl	3 or 4 character application mnemonic. Wild card (*) is permitted.
Ltype	3 character library type. Wild card (*) is permitted.
Suffix	1 to 7 characters that are added to the end of the JOBSTEM to make a job name up to 8 characters. The jobname suffix overlays the end of the JOBSTEM, if necessary.
TYPRUN	Value HOLD forces TYPRUN=HOLD on JOB statements. Value RUN omits any TYPRUN values. Any other value has no meaning.
Increment	**YES:** The final character of the jobname is incremented from 0 through to Z (36 values) for each job submitted under this entry. After Z, start again at 0.
	RANGE=nn (valid values for nn are 01-36). The final character of the jobname is incremented from 0 through to the character that equates to the nnth position in the sequence from 0 to Z. The final jobname character starts again at 0 and it cycles through as many times as necessary.
	NO: No final character increment on jobname.

Tip

To view auto resolve job service requests without executing the jobs, use execution parameter TRACE=AUTO for program CMNAD500. The service requests are written to a dynamically allocated AUTOSRVC DD statement.

Autoresolve Example

Observe the following autoresolve parameters:

```
BROWSE CMNTP.GENL.PARMLIB(AUTOCNTL) - 01.00
Command ===>
********************************* Top of Data ********
JOBSTEM=USER
******************************************************
* Seq. ; Appl ; Ltype ; Suffix ; TYPRUN ; Increment *
* No.  ;      ;       ;        ;        ;           *
******************************************************
001    ; ACTP ; SRO   ; SROX   ; RUN    ; RANGE=05
002    ; ACTP ; SRN   ; SRNX   ; RUN    ; RANGE=05
003    ; ACTP ; SRC   ; SRCX   ; RUN    ; RANGE=05
010    ; ACTP ; OBJ   ; OBJX   ; RUN    ; RANGE=05
011    ; ACTP ; NCL   ; NCLX   ; RUN    ; RANGE=05
012    ; ACTP ; LOD   ; LODX   ; RUN    ; RANGE=05
021    ; GENL ; SR*   ; SRCX   ; RUN    ; RANGE=05
022    ; GENL ; L*    ; LODX   ; RUN    ; RANGE=05
031    ; GENL ; OBJ   ; OBJX   ; RUN    ; RANGE=05
032    ; GENL ; NCL   ; NCLX   ; RUN    ; RANGE=05
999    ; *    ; *     ; OTHR   ; RUN    ; NO
******************************** Bottom of Data ******
```

In this sample all jobnames are built from "USER".

This user has an application (ACTP), which has components that are processed ahead of all other applications. In the ACTP application, and in all other applications, source for object (SRO) is processed ahead of source for NCAL subroutines (SRN), ahead of main module source (SRC).

Similarly, relink requests for object code (OBJ) and NCAL subroutines (NCL) are processed ahead of relinks for composite load modules (LOD). (Note that the OBJ/NCL rules are there just in case there is something wrong with the setup and audit auto resolve requests a relink of an object or NCAL subroutine.)

Finally, if the request does not fit into any of the above categories then it is covered by the "catch all" final entry.

If a series of recompiles are requested for SRn components in application GENL, then the jobnames used to process the recompiles are USERSRC0, then USERSRC1, USERSRC2, USERSRC3, USERSRC4 before returning to USERSRC0 again.

In this example all jobs will not have TYPRUN=HOLD added to the jobcard definitions. Note that if the jobcard parsing code is unable to find somewhere to put TYPRUN=HOLD on the 4 supplied jobcard images then the action will not take place and an error message will be issued. If however the original audit job is built with TYPRUN=HOLD, then all jobs will also be TYPRUN=HOLD (as shown below).

The following is an extract from CMNAD500 SYSPRINT DD for a run in which autoresolve parameters have been used. (The dataset name is enforced here via exit CMNEX040). The lines in black text have not changed from the current CMNAD500 SYSPRINT. The lines in red are produced when CMNEX040 is in use, the lines in blue describe the autoresolve parameters being used by CMNAD500. The lines in magenta describe what actions autoresolve has taken.

```
ChangeMan(R)    ZMF CMNAD500 - 8.1.0 Audit THURSDAY JANUARY 8, 2015 @   00:29:14
CMN2698I - Processed by CMNAD500 version 2014/10/10 10.40
CMN7500I - Attempting to initiate dialog with started task.
CMN7510I - Session established with ChangeMan ZMF started task.
CMN2683I - PRINTING AUDIT REPORT FOR PACKAGE GENL000007.

CMN3060A - This package has failed the audit.
CMN2696I - PACKAGE GENL000007 FAILED THE AUDIT WITH A RETURN CODE OF 08.
AUTOCNTL: installation exit is forcing the use of dataset
          CMNTP.GENL.PARMLIB
AUTOCNTL: and member
AUTOCNTL
The following autoresolve control parameters were used in this execution:
AUTOCNTL: JOBSTEM=USER
AUTOCNTL: *****************************************************
AUTOCNTL: * Seq. ; Appl ; Ltype ; Suffix ; TYPRUN ; Increment  *
AUTOCNTL: * No.  ;      ;       ;        ;        ;            *
AUTOCNTL: *****************************************************
AUTOCNTL:   001  ; ACTP ; SRO   ; SROX   ; RUN    ; RANGE=05
AUTOCNTL:   002  ; ACTP ; SRN   ; SRNX   ; RUN    ; RANGE=05
AUTOCNTL:   003  ; ACTP ; SRC   ; SRCX   ; RUN    ; RANGE=05
AUTOCNTL:   010  ; ACTP ; OBJ   ; OBJX   ; RUN    ; RANGE=05
AUTOCNTL:   011  ; ACTP ; NCL   ; NCLX   ; RUN    ; RANGE=05
AUTOCNTL:   012  ; ACTP ; LOD   ; LODX   ; RUN    ; RANGE=05
AUTOCNTL:   021  ; GENL ; SR*   ; SRCX   ; RUN    ; RANGE=05
AUTOCNTL:   022  ; GENL ; L*    ; LODX   ; RUN    ; RANGE=05
AUTOCNTL:   031  ; GENL ; OBJ   ; OBJX   ; RUN    ; RANGE=05
AUTOCNTL:   032  ; GENL ; NCL   ; NCLX   ; RUN    ; RANGE=05
AUTOCNTL:   999  ; *    ; *     ; OTHR   ; RUN    ; NO
Baseline recompile requested in GENL000007 for SRC/GNLSRC1A
Baseline recompile requested in GENL000007 for SRS/GNLSRS5A
CMN7540I - End of job; RC = 08
CMN1410I - Session terminated with ChangeMan ZMF started task.
```

If any of the service requests should fail for some reason then the message returned by ChangeMan ZMF will be shown immediately following the relevant line (magenta above) for example:

```
Package build requested in GENL000007 for GNL07S01.SRC
Service request failed, message returned by ZMF follows:
CMN8490I - Compile procedure was not specified.
```

If TRACE=AUTO is added to CMNAD500 PARM, then autoresolve trace information is written to the SYSPRINT DD . However, this prevents the action service requests from being issued, but this is useful for testing.

Instead, DD statement AUTOSRVC will be used to write CMNVSRVC batch keywords. These are equivalent to the service requests that would have been issued. This means that a test scenario can be setup and run several times without having to reset the components in the package being autoresolved. The end product of the autoresolve actions can be seen in these batch keywords. Of course, the batch keywords can equally well be used as input to a run of CMNVSRVC if there is a requirement to action them.

However, the main aim of this facility is to allow for ease of testing.

The AUTOSRVC DD statement can be added to the CMNAD500 step pointing to a disk file if the batch keywords need to be kept (e.g. for sending to Serena support). If the AUTOSRVC DD statement is missing then it will be dynamically allocated as a sysout DD.

Sample AUTOSRVC output:

```
OBJ=CMPONENT,MSG=RECOMPILE,
PKN=GENL000007,LTP=SRC,SUP=N,LVL=00,UHS=Y,
JC1=//USERSRCA JOB (SM-1IKF-SM),'GENL07', ,
JC2=//      CLASS=A,MSGCLASS=X,NOTIFY=USER015,TYPRUN=HOLD ,
JC3=//* ,
JC4=//* JOB TO RUN AUDIT ON GENL00007 WITH AUTO RESOLVE AND WITH EXIT ,
CMP=GNLSRC1A,
CNT=0001
OBJ=CMPONENT,MSG=RECOMPILE,
PKN=GENL000007,LTP=SRS,SUP=N,LVL=00,UHS=Y,
JC1=//USERSRCB JOB (SM-1IKF-SM),'GENL07', ,
JC2=//      CLASS=A,MSGCLASS=X,NOTIFY=USER015,TYPRUN=HOLD ,
JC3=//* ,
JC4=//* JOB TO RUN AUDIT ON GENL00007 WITH AUTO RESOLVE AND WITH EXIT ,
CMP=GNLSRS5A,
CNT=0001
```

Resolving Same-Named Members in Multiple (PDS) Libraries

ChangeMan ZMF can distinguish among copybooks and statically link-edited subprograms with the same name in multiple baseline libraries. For example, you may have a COBOL copybook and a PL/I copybook with the same name in different baseline libraries.

To enable ChangeMan ZMF to establish the correct source-to-copy and source-to-load relationships, you need to stage (compile and link) the source programs that reference these copybooks and subroutines in a change package. If ChangeMan ZMF cannot identify the appropriate copybook or subprogram during this process or during impact analysis, it prompts you to identify the correct module.

If you created the change package for the express purpose of resolving the references to copybooks and subprograms that have the same names, you do not have to baseline the staged package; you can leave the package dormant if you wish and it will eventually be aged.

Audit Diagnostic Tracing

You can request diagnostic information for debugging audit problems by specifying the TRACE command on the Command line of the Audit Change Package (CMNAUDIT) panel when you request an audit.

```
CMNAUDIT        Audit Change Package
Command ===> trace
Package . . . . . . . . ACTP000029
Audit part. pkg . . . . __ (1-As simple,2-As primary,3-By Department)
Autoresolve parms . . . ____________________________________

Enter "/" to select option
__ Audit staging libraries only
__ Include history records
__ Format report for printing
__ Specify application scope
__ Suppress batch messages
/ Include cross appl headings       Print top line only
__ Lock package                     Reset lock
__ Auto resolve
__ Update only this pkg rc
Job statement information:
//USER015T JOB (SM-1IKF-SM),'GENL06',
//       CLASS=A,
//       MSGCLASS=X,
//       NOTIFY=USER015
```

Trace output appears in the SYSPRINT DD for audit job step AD000.

```
ChangeMan(R) ZMF 8.1.4 Audit - SUNDAY DECEMBER 17, 2017 @ 21:43:03
CMN2698I - Processed by CMNAD000 version 2017/07/26 01.54
SYSIN: PKG=ACTP000081,TRC=T,TSO=USER015,ARS=NO,STG=NO,URC=YES,XHD=YES
SYSIN: HST=YES,PRT=NO,PSP=NO,PSN=NO,PRF=USER015,PDT=NO,RUT=NO,LCK=NO
SYSIN: S2L=YES
SYSIN: XAP=ACTP
SYSIN: JC1=//USER015C JOB (ACCOUNT),'CHANGEMAN',
SYSIN: JC2=//       CLASS=A,
SYSIN: JC3=//       NOTIFY=USER015,
SYSIN: JC4=//       MSGCLASS=X
CMN7521I - End of data on SYSIN.
CMN7500I - Attempting to initiate dialog with started task.
CMN7510I - Session established with ChangeMan ZMF started task.
2017/12/17 21:43:04 - Dataspace initialisation begins.
2017/12/17 21:43:04 - Dataspace initialised.
2017/12/17 21:43:04 - Staging model dataset name: CMNTP.S6.ACTP.STG6.#000081
2017/12/17 21:43:04 - Package locked: ACTP000081
2017/12/17 21:43:04 - Get local record information from package master.

2017/12/17 21:43:04 - Get global record information from package master.
2017/12/17 21:43:04 - There is no BUN entry in the i/a table for:
2017/12/17 21:43:04 - Application
                      0000 C1C3E3D7                            *ACTP            *
2017/12/17 21:43:04 - Library Type
                      0000 D6E3C8                              *OTH.            *

2017/12/17 21:43:04 - Baseline Unique Number Table
                      0000 00000001 C1C3E3D7 C3D7E8C3 00000000 *... ACTPCPYC....*
                      0010 02C1C3E3 D7C3D7F2 C3000000 0003C1C3 * ACTPCP2C.... AC*
                      0020 E3D7C3E3 C3D70000 00001AC1 C3E3D7C4 *TPCTCP.... ACTPD*
                      0030 C2C2D700 00000015 C1C3E3D7 C4C2D9D7 *BBP.... ACTPDBRP*
                      0040 00000000 04C1C3E3 D7C4D6C3 D7000000 *.... ACTPDOCP...*
                      0050 000EC1C3 E3D7C8E3 C8D74000 00000FC1 *. ACTPHTHP ... A*
                      0060 C3E3D7D1 C1D9D340 00000011 C1C3E3D7 *CTPJARL ... ACTP*
                      0070 D1C3C6D3 40000000 05C1C3E3 D7D1C3D3 *JCFL ... ACTPJCL*
                      0080 D7000000 0010C1C3 E3D7D1C3 E3E24000 *P.... ACTPJCTS .*
                      0090 000017C1 C3E3D7D1 E5D3D340 00000018 *.. ACTPJVLL ... *
                      00A0 C1C3E3D7 D1E5E2E2 40000000 19C1C3E3 *ACTPJVSS ... ACT*
...
2017/12/17 21:43:04 - Target Application
0000 C1C3E3D7                                                  *ACTP            *

2017/12/17 21:43:04 - Target Application BUN Table
                      0000 00000001 00000002 00000003 00000004 *... ... ... ... *
                      0010 00000005 00000006 00000007 00000008 *... ... ... ... *
...
```

Among other messages, audit processing issues a SYSPRINT message as each Like-Copy library is added to the concatenation of copy libraries that is included in the audit, so that you can see the order in which the copy libraries are searched. This can be a useful debugging aid.

13. Freezing a Package

This chapter describes the freeze package function, which locks package information and package components to prevent further changes.

About Freezing a Package

When you are finished making changes to the components in your package, you freeze the package to conduct final testing and prepare for package installation and baseline ripple.

Freeze Package locks package information and package components to prevent further changes. You freeze your change package at some point in your project life cycle before final testing so that you can have confidence in your final test results. Your package must be frozen before the package approval process can begin.

When you freeze a package, ChangeMan ZMF validates the condition of the package and its components, and if no error conditions are found, the package status is changed to FRZ. After a package is frozen, you may be able to selectively unfreeze package information and package components to make changes, then selectively refreeze the information or components.

Your administrator can configure your application to build package installation JCL when you freeze your package or when the final package approval is entered. Installation JCL is created in a file tailoring server that runs in an address space separate from your ChangeMan ZMF session.

When you freeze a package, the status of the package is immediately changed to FRZ, and if you did not create the package, a message is sent to the user specified in the `Notify user` field of the package:

```
`CMN400I - Package ACTP000030 frozen by USER015 on 2015/01/14 at 22:34`
```

Approval notifications are also sent when you freeze a package:

```
`CMN4600I - Change Man package ACTP000010 awaits your approval.`
`CN(INTERNAL)`
```

If installation JCL is built at freeze in your application, file tailoring for installation JCL must complete successfully before approvals can be entered.

Rules for Freezing Packages

These rules and restrictions apply to the freeze package function:

- All components in a package must be in ACTIVE status before a package can be frozen.
- A package cannot be frozen if it contains a utility request to scratch or rename a component that is in the package, which includes a generated component such as a load module created by a build process.
- A package cannot be frozen if a utility request renames a component to the same name as a component in the package, which includes a generated component such as a load module created by a build process.
- If a designated compile procedure is specified for a package component, the package cannot be frozen unless the designated procedure is used to build the component.
- Audit may be required for planned packages before they can be frozen, and if audit is required, the audit return code may not exceed a maximum determined by the audit level set in application administration. See Audit Level.
- Unplanned packages never have to pass audit before freeze, although you may be required to run audit.
- When a package is frozen, the package status is changed from DEV (Development) to FRZ (frozen).
- When a package is frozen, the status of package components is changed from ACTIVE to FROZEN.
- Components cannot be added to a frozen package.
- While a package remains frozen, package information and package components may be selectively unfrozen, changed, and refrozen.
- To selectively unfreeze package information and package components, you must have revert authority or be an approver for a package approval that has not been entered.

- The promotion rule may require that a component be selectively demoted before the component may be selectively unfrozen. See Promotion Rule.
- When a package component is selectively unfrozen, the component status is changed from FROZEN to UNFROZEN. If you then stage the component, the component status is changed from UNFROZEN to either ACTIVE or INCOMP.
- If a package component is unfrozen, but you do not change the component, the component may be refrozen without running audit, regardless of the audit level in force.
- Unlike package revert, selective unfreeze does not reset approvals that are already entered. However, new approvals cannot be entered until all components are in FROZEN status. Approvers are not notified of selective unfreeze or refreeze.
- A frozen package can be returned to development status (DEV) with the Revert function.

Audit Level

The audit level specified in application administration at the time a package is created determines whether audit must be run before the package is frozen. The audit level determines the maximum return code that is acceptable to pass the audit. Each return code indicates a type of out-of-sync condition.

If you unfreeze and change a package component, the audit level determines whether the change package must pass audit before you can refreeze the component.

This table describes the audit levels, the highest acceptable audit return code required to pass audit at each level, and a description of the kind of out-of-sync errors that are indicated by each return code.

Audit Level	Maximum Return Code	**Audit Level and Return Code Description
0	Any	Audit is recommended but optional. If audit is run, any return code, except abend, is acceptable. Any out-of- sync condition is permitted.
1	Any	Audit is required. Any return code, except abend, is acceptable. Any out-of-sync condition is permitted.
2	12	Audit is required. The return code indicates that there are out-of-sync conditions in the staging libraries of the package. Out-of-sync conditions indicated by lower return codes may also exist.
3	8	Audit is required. The return code indicates that there are no out-of-sync conditions in the staging libraries but that there are out-of-sync conditions in the baseline libraries. Out-of-sync conditions indicated by lower return codes may also exist.

Audit Level	Maximum Return Code	**Audit Level and Return Code Description
4	4	Audit is required. The return code indicates that there are no out-of-sync conditions in the staging or baseline libraries, but at least one component in a staging library is a duplicate of the corresponding baseline component.
5	0	Audit is required. The return code indicates that there are no out-of-sync conditions in the staging or baseline libraries. Also, there are no duplicate (unmodified) components in the staging libraries. Some components that are unknown to ChangeMan ZMF, do not have ISPF statistics, or are USER239 load modules are flagged as out of sync but are allowed to pass the audit. Examples of components that are unknown to ChangeMan ZMF include language environment subroutines which the binder automatically links into a load module.

For information about package audit, see Auditing a Package

Administration Settings for Freeze Package Functions

Your global and application administrators make settings in ChangeMan ZMF administration that control how the freeze package function works in your application. Ask your administrator if any of the following apply to your application.

- Require audit for unplanned packages when the application audit level is greater than 0. Unplanned packages can still be frozen, approved, and installed if they fail audit. (Application Administration: Force Audit of Unplanned Packages)
- Create installation JCL at freeze or when the final approval is entered. (Application Administration Parameters: Build Installation JCL At Approve)

Exit Programs for Freeze Package Functions

Your ChangeMan ZMF installer or administrators may implement exit programs to alter the behavior ChangeMan ZMF. Ask your administrator if any of the following apply to your application.

- Prohibit package freeze depending on package conditions like package type, library types in package, installation date, installation from time or to time, installation day of week, Package User Information variable values. (Exit program CMNEX024)
- Prevent package freeze or selective refreeze if certain package or component conditions exist. This exit is commonly used to prevent installation of components where prohibited compile options, link edit options, or User Options were used the last time a component was staged. (Exit program CMNEX025)

Accessing Freeze Package Functions

Freeze functions are accessed from the **Freeze Options** menu (CMNFRZ01).

```
CMNFRZ01                Freeze Options
Option ===>________________________________________
  Package . . . . ACTP000038
1 Online          Freeze package online
2 Batch           Freeze package in batch
3 Selective       Selectively unfreeze/refreeze package components
4 Reset           Reset indicator after unsuccessful batch freeze
5 Re-Build        Re-submit install JCL build request
```

Use one of these methods to display the **Freeze Options** menu:

- Using the Menu Hierarchy:

  On the **Primary Option Menu**, select **2 Freeze**.

- Using Direct Panel Access:

  On a ChangeMan ZMF panel, type **=2** in the **Command** or **Option** line and press **Enter**.

Using the Change Package List to Access Freeze Functions

You can access the options listed on the **Freeze Options** menu directly from the **Change Package List**.

On the **Change Package List** panel, type one of the following in the line command for a change package and press **Enter**.

- **F1** Freeze the change package online
- **F2** Freeze the change package in batch
- **F3** Selectively unfreeze or refreeze package components
- **F4** Reset Freeze Progress indicators
- **F5** Re-submit installation JCL build request

Freezing a Package Online

Follow these steps to freeze a package online.

1. On the **Freeze Options** menu (CMNFRZ01), type your package ID in the **Package** field.
2. Select option **1 Online**.
3. Press **Enter**.
4. While ChangeMan ZMF is processing your freeze request, the **In Progress** panel is displayed, which shows what actions are being performed to freeze your package. For small packages, the panel may display for only a second.
5. When freeze processing is complete, the status of your package is changed to FRZ, and an MVS SEND message is sent to the user specified in the `Notify user` field of the package. (You will not receive this message if you are the package creator and this field is blank or if you are the user specified in this field.)
6. If installation JCL is built at freeze, a file tailoring server is started in a separate address space. When file tailoring for installation JCL completes, you receive this message.

```
CMN8700I - *packageID* Installation JCL Build service completed CN(INTERNAL)
```

Freezing a Package in Batch

When you choose the batch option to freeze your package, ChangeMan ZMF submits a job that starts a batch TSO session, which connects to ChangeMan ZMF to perform freeze validation and change the package status to FRZ. If installation JCL is built at package freeze in your application, a file tailoring server is started in a separate address space.

Batch freeze was important before Version 5 of ChangeMan ZMF when file tailoring was performed in the same address space as your ChangeMan ZMF session. Online freeze for large packages could lock your terminal for a significant period of time while installation JCL was being built, preventing you from doing other work.

Whether batch freeze now provides significant benefit depends on the size of the package you are freezing, the computer resources available for processing, and how much your implementation of ChangeMan ZMF makes use of facilities like designated procedures and freeze exit programs.

Follow these steps to freeze a package a package in batch.

1. On the **Freeze Options** menu (CMNFRZ01), type your package ID in the **Package ID** field.
2. Select option **2 Batch** to display the **Freeze Package in Batch** panel (CMNFRZ02).

```
CMNFRZ02            FREEZE Package in Batch
Command ===>___________________________________
Package: ACTP000038

Enter "/" to select option
/ User variables
_ Job statement information:
    //USER015F JOB (SM-1IKF-SM),'ACTP38',
//      CLASS=A,MSGCLASS=X,NOTIFY=USER015 ,TYPRUN=HOLD
//*
//*   JOB TO RUN FREEZE
```

3. On the **Freeze Package in Batch** panel CMNFRZ02, type **Job Statement Information** and optionally select User Variables and press **Enter**.

   You are returned to the **Freeze Option** panel where short ISPF message **Job Submitted** is displayed.

   When the batch freeze job completes successfully, the status of your package is changed to FRZ, a SEND message is sent to you, and a SEND message is sent to the user specified in the `Notify user` field of the package. (You will not receive this message if you are the package creator and this field is blank or if you are the user specified in this field.)

   If the batch freeze job fails, the status of your package is not changed, and error messages are displayed in the job SYSOUT to tell you what you must fix. SEND messages are also sent.

   If installation JCL is built at freeze, a file tailoring server is started in a separate address space. When file tailoring for installation JCL completes, you receive this message.

   `CMN8700I - *packageID* Installation JCL Build service completed CN(INTERNAL)`

Unfreezing and Refreezing Components

You freeze your change package before final testing to ensure that what you test is what you install. You also freeze a change package to lock in package information and controls.

After you freeze your change package, if you need to add a new component to the package, you must revert the entire package.

You may be able to use selective unfreeze and refreeze instead of revert to change package information and package components in circumstances like these:

- A defect is found in testing that can be fixed by changing a package component.
- An approver requires you to change package information.
- You need to change the installation date or time for your package.
- You must remove a scratch or rename utility request.

Follow these steps to selectively unfreeze package information and components.

1. On the **Freeze Options** menu (CMNFRZ01), type your package ID in the **Package ID** field and select option **3 Selective** to display the **Unfreeze/Refreeze package** panel (CMNUNFRZ).

```
CMNUNFRZ    Unfreeze/Refreeze ACTP000038        Row 1 to 6 of 6
Command ===>__________________________________ Scroll ===> CSR
Category            Status      Function
__ General          FROZEN
__ Non-Source       FROZEN
__ Source and Load  FROZEN
__ Utilities        FROZEN
__ Install Date(s)  FROZEN
__ Online Forms     FROZEN
******************************* Bottom of data ********************************
```

This table describes fields and commands on the **Unfreeze/Refreeze package** panel.

Field	Description
Line Commands	Type a line command for a package category.
	U - Unfreeze all package information or components in the package category.
	F - Refreeze all package information or components in the package category that were previously unfrozen.
	S - Display a selection list to unfreeze or refreeze one or more components in the package category.
Category	General: Package information you can change on these panels:
	- **Update: Control Information** (CMNPGNL1)
	- **Update: Package Description** (CMNPGNL2)
	- **Update: Installation Instructions** (CMNPGNL3)
	Non-Source: Non-source package components.
	Source and Load: Source components and the target component types in source-load relationships.
	Utilities : Scratch and rename utility requests.
	Install Date(s): Package installation site and schedule information you can update on the **Update: Control Information** panel (CMNPRSTI).
	Custom Forms: Type **U** to unfreeze or **F** to refreeze all online Forms forms in your change package. Type **S** to display the **Online Forms Freeze/Unfreeze** panel (CMNZUNF) where you enter **U** to unfreeze one or more online forms or **F** to refreeze one or more online forms.
Status	Displays the freeze status of the package category.
	FROZEN: All category information or components are frozen.

Field	Description
	UNFROZEN: Some or all category information or components are unfrozen.
Function	Displays the result of the last action taken on the category.

2. On the **Unfreeze/Refreeze package** panel, type a line command on one or more package categories and press **Enter**. If you typed line commands **U** or **F**, the information in the **Status** and Function columns is updated.
3. After you have successfully unfrozen package information or package components, use ChangeMan ZMF update or stage functions to make changes.
4. When you have finished making changes to your package, go back to step 1 in this section to refreeze the information or components that you changed.

> ⚠ **Caution**
>
> If package installation JCL is created at freeze, and If you unfreeze and delete a scratch or rename utility request, use freeze option 5 Re-Build to reconstruct the installation JCL after the package is completely frozen again.

Selective Unfreeze and Refreeze Source and Non- Source

Follow these steps to selectively unfreeze or refreeze source and non-source package components.

1. On the **Unfreeze/Refreeze package** panel, type **S** in the **Source** or **Non-Source** category and press **Enter**. The **Unfreeze/Refreeze package Components** panel (CMNUNF01) is displayed.

```
CMNUNF01    Unfreeze/Refreeze ACTP000038 Components Row 1 to 4 of 4
Command ===> _____________________________________ Scroll ===> CSR

Name                     Type  Changed          User    Status Request
__ ACPSRCCE                 + SRC 20150115 000437 USER015 FROZEN
__ ACPSRC1A                 + SRC 20150115 000511 USER015 FROZEN
__ ACPSRC6A                 + SRC 20150115 000619 USER015 FROZEN
__ ACPSRC92                 + SRC 20150115 000657 USER015 FROZEN
******************************* Bottom of data ********************************
```

This panel shows the source or non-source components in your package.

This table describes the fields and commands on the **Unfreeze/Refreeze package Components** panel.

Field	Description
Command	Type one of the following commands, or leave the command line blank and type a line command next to a component name.

Field	Description
	REFRESH: Display updated information on this panel. Abbreviation: R
	SORT *heading*: Sort listed components by information under the specified column heading. The **Changed** column sorts in descending order. Abbreviation: S
	LOCATE *member*: Locate a listed component by information in the last sorted column or by Name if the components were not sorted. Abbreviation: L
	CANCEL: Cancel the function and return to the previous panel. Abbreviation: C
Line Command	Type a line command to the left of the Name row:
	U: Unfreeze the component.
	F: Refreeze the component.
	Note: Line command SL is invalid starting with ZMF 6.1.
Name	Displays the name of the component.
Type	Displays the library type under which the component is checked out or staged in this package.
Changed	Displays the date and time that the component was edited or staged.
User	Displays the UserId of the person who performed the last stage action on the component.
Status	Displays the status of the component in the package:
	ACTIVE: Component has been successfully staged
	FROZEN: Component is frozen and may not be updated
	INCOMP: Compile job has not completed or has failed
	UNFROZEN: Previously frozen component is available for update
Request	Displays the last selective unfreeze/refreeze action performed on the component on this panel.

2. On the **Unfreeze/Refreeze package Components** panel, type a line command on one or more components and press **Enter**.

Type line command **U** to unfreeze or **F** to refreeze a component. Note that the **Status** and **Request** columns are updated.

```
CMNUNF01    Unfreeze/Refreeze ACTP000038 Components Row 1 to 4 of 4
Command ===> _____________________________________ Scroll ===> CSR

Name                   Type Changed         User    Status  Request
__ ACPSRCCE          + SRC  20150115 000437 USER015 FROZEN
__ ACPSRC1A          + SRC  20150115 000511 USER015 UNFROZEN *Ok
__ ACPSRC6A          + SRC  20150115 000619 USER015 FROZEN
__ ACPSRC92          + SRC  20150115 000657 USER015 FROZEN
******************************* Bottom of data *********
```

To see the effects of unfreeze on a like-source component, navigate to the **STAGE: package COMPONENTS** panel (CMNSTG01) using jump command **=1.6**, and then choosing option **2** Package.

```
CMNSTG01        STAGE: ACTP000038 Components        Row 1 to 4 of 4
Command ===> ______________________________________ Scroll ===> CSR

Name            + Type Status   Changed         Procname User    Request
__ ACPSRCCE       SRC  FROZEN   20150115 000437 CMNCOB2  USER015
SL ACPSRC1A       SRC  UNFROZEN 20150115 000511 CMNCOB2  USER015 SRC-LOD
__ ACPSRC6A       SRC  FROZEN   20150115 000619 CMNCOB2  USER015
__ ACPSRC92       SRC  FROZEN   20150115 000657 CMNCOB2  USER015
******************************* Bottom of data ********************************
```

Type line command **SL** on an unfrozen like-source component, and the **Source to Load Relationship** panel (CMNSR2LD) shows you that all generated components are unfrozen.

```
CMNSR2LD        Source to Load Relationship             Row 1 to 2 of 2
Command ===> _________________________________________ Scroll ===> CSR

Package:  ACTP000038    Status: FRZ         Install date: 20150118

Source name . . . . ACPSRC1A +
Lib type . . . . .  SRC
Setssi . . . . . .  6786FB9B

Related Load Modules:

   Name         + Type Promotion Changed         User    Setssi
__ ACPSRC1A       LOD  0 STAGING 20150115 000511 USER015 6786FB9B
__ ACPSRC1A       LST  0 STAGING 20150115 000520 USER015 6786FB9B
******************************* Bottom of data ********************************
```

Selective Unfreeze and Refreeze Utility Requests

Follow these steps to selectively unfreeze or refreeze utility requests (scratch and rename) in your change package.

1. On the **Unfreeze/Refreeze package** panel, type **S** in the **Utilities** category and press **Enter**. The **Unfreeze/Refreeze package Utility List** panel (CMNUNF02) is displayed.

```
CMNUNF02    Unfreeze/Refreeze ACTP000038 Utility List Row 1 to 5 of 5
Command ===>________________________________________ Scroll ===> CSR

Req Name         Rename        Type Status  Request
__ SCR ACPCPY3B     +            + CPY  ACTIVE
__ REN ACPCPY4C     + ACPCOP4C  + CPY  ACTIVE
__ REN ACPCPY5A     + ACPCOP5A  + CPY  ACTIVE
__ REN ACPCPY5B     + ACPCOP5B  + CPY  ACTIVE
__ REN ACPCPY5C     + ACPCOP5C  + CPY  ACTIVE
******************************* Bottom of data ********************************
```

This panel shows the utility requests in your package.

This table describes the fields and commands on the **Unfreeze/Refreeze package Utility List** panel.

Field	Description
Command	Type one of the following commands, or leave the command line blank and type a line command next to a component name.
	REFRESH: Display updated information on this panel. Abbreviation: R
	SORT *heading*: Sort listed components by information under the specified column heading. (The **Changed** column sorts in descending order.) Abbreviation: S
	LOCATE *member*: Locate a listed component by information in the last sorted column or by Name if the components were not sorted. Abbreviation: L
	CANCEL: Cancel the function and return to the previous panel. Abbreviation: C
Line Command	Type a line command to the left of the Name row:
	U: Unfreeze the request.
	F: Refreeze the request.
Req	Displays the kind of utility request.
	REN: Rename request.
	SCR: Scratch request.
Name	Displays the name of the component.
Rename	Displays the new name for a rename utility request.
Type	Displays the library type of the component in baseline.

Field	Description
Status	Displays the status of the utility request.
	FROZEN: Request is frozen and may not be updated
	UNFROZEN: Previously frozen request is available for update
Request	Displays the last selective unfreeze/refreeze action performed on the request on this panel.

2. On the **Unfreeze/Refreeze package Utility List** panel, type line command **U** to unfreeze or **F** to refreeze one or more utility requests and press **Enter**. The information in the **Status** and **Request** columns is updated.

```
CMNUNF02        Unfreeze/Refreeze ACTP000038 Utility List Row 1 to 5 of 5
Command ===> ___________________________________________ Scroll ===> CSR

   Req Name         Rename          Type    Status      Request
__ SCR ACPCPY3B     +               + CPY   FROZEN
__ REN ACPCPY4C     + ACPCOP4C      + CPY   FROZEN
__ REN ACPCPY5A     + ACPCOP5A      + CPY   UNFROZEN    *Ok
__ REN ACPCPY5B     + ACPCOP5B      + CPY   FROZEN
__ REN ACPCPY5C + ACPCOP5C + CPY FROZEN
******************************* Bottom of data ********************************
```

3. Press PF3 to return to the **Unfreeze/Refreeze package** panel.

Selective Unfreeze and Refreeze Online Forms

Follow these steps to selectively unfreeze or refreeze online forms in your change package.

1. On the **Unfreeze/Refreeze package** panel, type **S** in the **Custom Forms** category and press **Enter**. The **Online Forms Freeze/Unfreeze** panel (CMNZUNF) is displayed.

```
CMNZUNF         Online Forms Freeze/Unfreeze        Row 1 to 10 of 10
Command ===> ______________________________________ Scroll ===> CSR

    Package: ACTP000038     Status: DEV     Install date: 20150118

                                                Lead Days           Last
    Form Description                            time left Status    user
___ 010  Build GDG Request                       0    15  FROZEN   USER015
___ 030  DASD Dataset Space Request              7    8
___ 070  Problem Summary                         7    8
___ 100  Report Distribution Information         7    8
___ 110  Standards & Guidelines Revision         7    8
___ 120  Tape Retention Register                 7    8
___ 130  Standards Exemption Request             7    8
___ 140  Security Information                    7    8
___ 160  Scheduler JCL Update                    7    8
___ 170  Scheduler Update                        7    8
******************************* Bottom of data ********************************
```

2. On the **Online Forms Freeze/Unfreeze** panel, type line command **U** to unfreeze a form or **F** to refreeze a form and press **Enter**. The form **Status** is updated.

```
CMNZUNF         Online Forms Freeze/Unfreeze        Row 1 to 10 of 10
Command ===> _______________________________________  Scroll ===> CSR

        Package: ACTP000038     Status: DEV     Install date: 20150118

                                                Lead Days          Last
    Form Description                            time left Status   user
___ 010  Build GDG Request                      0    15   UNFROZEN USER015
___ 030  DASD Dataset Space Request             7    8
___ 070  Problem Summary                        7    8
___ 100  Report Distribution Information        7    8
___ 110  Standards & Guidelines Revision        7    8
___ 120  Tape Retention Register                7    8
___ 130  Standards Exemption Request            7    8
___ 140  Security Information                   7    8
___ 160  Scheduler JCL Update                   7    8
___ 170  Scheduler Update                       7    8
******************************* Bottom of data ********************************
```

3. Press PF3 to return to the **Unfreeze/Refreeze package** panel.

Resetting the Freeze-In-Progress Indicator

When you choose the batch option to freeze your package, ChangeMan ZMF submits a job that starts a batch TSO session, which connects to ChangeMan ZMF to change the package status to FRZ. If installation JCL is built at package freeze for your application, the batch TSO session also initiates file tailoring for installation JCL.

To prevent you from initiating another freeze for a package before the first batch freeze job completes, a freeze-in-progress indicator is turned on when you initiate batch freeze processing. If you attempt to perform batch freeze before the freeze-in-progress indicator is turned off, the short ISPF message PROCESS ERROR is displayed, and when you press PF1, this message is displayed:

```
CMN1525I - Package is in the process of being frozen.
```

The freeze-in-progress indicator is turned off when the batch freeze job completes successfully. If there is a problem with a batch freeze job, such as a JCL error or a system crash, the freeze-in-progress indicator is left on, and you cannot initiate another batch freeze process.

To reset the freeze-in-progress indicator so that you can start over with batch freeze processing, select option **4 Reset** on the **Freeze Options** menu. When processing is complete, the short ISPF message INDICATOR RESET is displayed, and if you press PF1, this message is displayed:

```
CMN3036I - Freeze in Progress indicator reset.
```

Resubmitting Installation JCL Build Request

If your package is in FRZ status, and you want to rebuild the package installation JCL, you can resubmit the request to build installation JCL by using option **5 Re-build** on the **Freeze Options** menu.

Authorization to use this option depends on whether file tailoring for installation JCL was previously executed successfully.

- If package freeze processing successfully changes the status of your package to FRZ, but file tailoring for installation JCL fails, anyone authorized to update the application can resubmit the JCL build request.
- If the package is in FRZ status, and file tailoring for installation JCL has completed successfully, global or application administration authority is required to resubmit a request to build installation JCL.

14. Reverting a Package

This chapter describes the revert package function, which returns a package to development (DEV) status.

About Reverting a Package

The revert package function is used to return packages to development (DEV) status that have been previously frozen (FRZ), partially or fully approved (APR), or backed out (BAK).

Revert removes all previously entered approvals, unlocks package information and components, and opens the package back up to development.

When you revert a package, you must enter text on a Revert Reasons panel. This text explains why the package was returned to development and provides an audit trail for the revert action. Revert reasons can be viewed using the query package function.

A revert cannot be started whilst a promotion is in progress and attempts to do so will fail with the error message CMN3312A - Promote process is under way.

The revert package function changes the status of the entire package. If a package has been frozen and you want to update some package information or some package components, you may be able to selectively unfreeze and refreeze parts of the package instead of executing a revert. See Freezing a Package.

Authorization for Revert Package Functions

In some circumstances you must have revert authority to revert a change package. Revert authority is permitted in your security system.

- READ access to the security entity for the application
- UPDATE access to the security entity for revert

  In other circumstances, a package may be reverted by a package approver. Package approvers must have these privileges in your security system.
- READ access to the security entity for the application.
- UPDATE access to the security entity for the approval specified in the planned or unplanned approval list.

Rules for Reverting a Package

These rules and restrictions apply to the revert package function:

- If a package is in FRZ (frozen) status and no approvals have been entered, the package may be reverted by anyone who has update access to the application.
- If a package is in FRZ (frozen) status and at least one approval has been entered, the package may be reverted by a user with revert authority or by a user with approval authority whose highest approval level has not been entered.
- If a package is in APR (approved) status, it may be reverted by a user with revert authority.
- If a package is in REJ (rejected) status, it may be reverted by a user with revert authority.
- If a package is in DIS (distributed) status, it may be reverted by a user with revert authority. This rule applies to packages in a DP/P configuration.
- If a package is in BAK (backed out) status, it may be reverted by a user with revert authority.
- After a change package has been reverted, the package must be frozen again and the approval process must start over.
- The promotion rule may require that a package be fully demoted before the package may be reverted. See Promotion Rule.

In a DP/P configuration, the rules for DIS and BAK status apply to the package at each installation site. Package backout and revert at P instances can be initiated from the DP instance where the package originated.

Accessing Revert Package Panels

Revert package is executed from the **Revert Change Package** panel (CMNREV00).

```
CMNREV00                    Revert Change Package
Command ===>_______________________________________________

Package . . . . ACTP000027
```

Use one of these methods to display the **Revert Change Package** panel:

- **Using the Menu Hierarchy:**

  On the **Primary Option Menu**, select **R Revert**.

- **Using Direct Panel Access:**

  On a ChangeMan ZMF panel, type **=R** in the **Command** or **Option** line and press **Enter**.

Using the Change Package List to Revert Packages

You can access the revert package function directly from the **Change Package List**.

On the **Change Package List** panel, type **RV** in the line command for a change package and press **Enter**.

Reverting a Change Package

The process of reverting an installed change package differs between an All environment and a DP/P environment.

Reverting in an All Environment

For an A environment, follow these steps.

1. Access the **Revert Change Package** panel.
2. On the **Revert Change Package** panel, type the Package ID of the package you want to revert and press **Enter**. The **site - Revert Reasons** panel (CMNREVRS) is displayed.

```
CMNREVRS                         SERT6 - Revert Reasons
Command ===>

         Package: ACTP000027            Status: BAK          Install date: 20150228
Revert reasons:
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
```

3. On the *site* **- Revert Reasons** panel, type up to nine lines of text explaining the reason why the package is being reverted. You must type at least one character on the panel. Press **Enter** to process the revert request.
4. When revert processing is complete, the package status is changed to DEV as shown here on the **Change Package List** panel.

```
CMNLIST3                Change Package List                PACKAGE REVERTED
Command ===>                                               Scroll ===> CSR

    Package     Sta Install   Lvl Type      Work request Dept Promote  Aud Creator
    ACTP000027  BAK 20150228  SMP PLN/PRM   100001000106 IDD 00        00 USER016
  ******************************* Bottom of data *******************************
           +-------------------------------------------------------+
           | CMN3300I - Package has been reverted to development. |
           +-------------------------------------------------------+
```

Reverting at a Remote Site

Reverting a package in a P instance at a remote site is performed the same as reverting on an A instance.

> **Note**
>
> You can revert a package at all installed sites from the D or DP instance where the package was created. There is no requirement to logon to a P instance to revert a package there.

Reverting in a DP/P Environment

For a DP/P instance, follow these steps.

1. Access the **Revert Change Package** panel.
2. On the **Revert Change Package** panel, type the Package ID of the package you want to revert, and press **Enter**.
3. If the package is scheduled for installation at more than one site, the **Revert: Site Information** panel (CMNRVSTI) is displayed.

```
CMNRVSTI                     Revert: Site Information                 Row 1 to 2 of 2
Command ===>                                                         Scroll ===> CSR

           Package: ACTP000076     Creator: USER239     Status: BAK
           Install Time
   Site     date     From  To     Primary/backup contacts     Phone numbers    Status
 _ SERT7   20100630  0001  2359   KIKA HALEMANU               808-555-1213     BAK
                                  IAN THOMPSON                808-555-1215
 _ SERT9   20100630  0001  2359   HUNG NGUYEN                 808-555-1214     BAK
                                  WENWEI HAN                  808-555-1212
 ******************************* Bottom of data ********************************
```

4. On the **Revert: Site Information** panel, select one or more sites where you want to revert the package, and press **Enter**. The *site* **- Revert Reasons** panel (CMNREVRS) is displayed for the first selected site.

```
CMNREVRS                         SERT7 - Revert Reasons
Command ===>

         Package: ACTP000076             Status: BAK         Install date: 20150228
Revert reasons:
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
______________________________________________________________________________________
```

5. On the *site* **- Revert Reasons** panel, type up to nine lines of text explaining the reason why the package is being reverted. You must type at least one character on the panel. Press **Enter**.
6. If the site from which you are reverting the package is a remote site, the **Submit Remote Revert Request** panel is displayed. The revert request job is submitted by the D or DP instance, and it connects to the P instance using TCP/IP to request the revert.

```
CMNRVJCD                     Submit Remote Revert Request
Command ===>

    Site: SERT9    Package: ACTP000076    Creator: USER239    Status: BAK

Job Statement Information:
===> //USER2391 JOB (X170,374),'S7.V710',____________________________________
===> //         CLASS=A,MSGCLASS=X,NOTIFY=USER239____________________________
===> //*_____________________________________________________________________
===> //*_____________________________________________________________________
```

7. Update the **Job Statement Information**, if necessary, and press **Enter** to submit the remote revert request job.
8. If you selected additional sites for revert, the *site* **- Revert Reasons** panel is displayed for each site. Type revert reasons for each site, and submit the remote revert request if required.
9. When revert processing is completed for a site, the **Site Status** on the **Revert: Site Information** panel is changed to DEV.

```
CMNRVSTI                      Revert: Site Information                    PACKAGE REVERTED
Command ===>_____________________________________________________________ Scroll ===> CSR

            Package: ACTP000076    Creator: USER239    Status: BAK
            Install Time
  Site      date      From  To     Primary/backup contacts      Phone numbers      Status
_ SERT7     20100630  0001  2359   KIKA HALEMANU                808-555-1213       BAK
                                   IAN THOMPSON                 808-555-1215
_ SERT9     20100630  0001  2359   HUNG NGUYEN                  808-555-1214       DEV
                                   WENWEI HAN                   808-555-1212
******************************* Bottom of data ********************************
```

10. When all sites are in DEV status, the package status is changed to DEV.

```
CMNRVSTI                       Revert: Site Information                    PACKAGE REVERTED
Command ===>______________________________________________________________ Scroll ===> CSR

           Package: ACTP000076     Creator: USER239     Status: BAK
           Install Time
  Site      date      From  To       Primary/backup contacts      Phone numbers     Status
_ SERT7     20100630  0001  2359     KIKA HALEMANU                808-555-1213      DEV
                                     IAN THOMPSON                 808-555-1215
_ SERT9     20100630  0001  2359     HUNG NGUYEN                  808-555-1214      DEV
                                     WENWEI HAN                   808-555-1212
******************************* Bottom of data ********************************
```

15. Promoting and Demoting a Package

Package promotion populates test libraries with the contents of package staging libraries.

About Promotion

Promotion is a ChangeMan ZMF facility that applies component changes in a package to libraries used for testing and other purposes. Promotion can populate libraries used for:

- Batch testing, where test libraries are coded in STEPLIB or JOBLIB statements in common application testing JCL.
- Online testing, where application testing libraries are coded in region JCL.
- Unit testing where libraries are loosely controlled and may be updated by any developer who wants to run a test.
- Quality Assurance test libraries that are tightly controlled and can only be updated by authorized test coordinators.
- Training environments, where software changes must be available for training classes before they are installed into production.
- Any purpose that requires package components to be copied into a fixed set of libraries.

Promotion copies components from package staging libraries into libraries that are used for application testing or other purposes. Promotion can also be configured to execute additional processes to prepare promoted components for execution. Such processes might include CICS® PHASEIN, Db2 bind, and IMS™ gens.

Demotion deletes components from libraries that were populated by promotion. Demotion can also execute processes such as CICS PHASEIN, Db2 bind, and IMS gens to adapt an environment to the changes made by demotion.

Each set of libraries that is targeted by promotion is represented by a **promotion level**. The ChangeMan ZMF Administrator defines promotion levels for each application. Each promotion level is defined with the library types that can be promoted and the names of the libraries that are targeted for each type. A promotion level does not have to include all library types in an application. Library types for promotion usually include the executable components in your package and can also include nonexecutable types like copybooks.

Each promotion level is defined under a **site**. Promotion populates libraries and prepares executables on **local** sites, which means environments that are on the same MVS image as the ChangeMan ZMF server. Promotion also populates libraries and prepares executables on **remote** sites, which means environments that are on MVS images separate from the image where the ChangeMan ZMF server runs.

Full promote and demote operate at the package level. All components in a package that are eligible for promotion are promoted or demoted together. The current promotion level is recorded at the package and the component level.

Selective promote and demote operate on individual components in a change package. The package promotion level remains the same, but the component promotion level changes.

Because application test libraries are often shared between developers and projects, promotion looks for potential **overlays** by comparing the names of package components eligible for promotion against the directories of the target libraries. The person promoting the package is given a choice whether or not to proceed and overlay matching components in the promotion libraries or cancel the promotion request.

Promotion should not be confused with the physical movement of components through a series of test libraries and into production libraries. Promotion always copies components from package staging libraries into target promotion libraries. Likewise, at baseline ripple and install, package components are copied from package staging libraries into baseline and production libraries.

Promotion Library Cleanup

If your testing environment allows test libraries to be concatenated over production libraries, your administrator may have configured promotion to delete promoted components from prior promotion libraries when a package is promoted to another level or when a package is baseline rippled or installed. This configuration provides the most control over a testing environment because it guarantees that if no packages are promoted, the environment will function exactly like production.

Your testing environment may not allow concatenation of libraries, especially if it uses a database or data dictionary to build executables. In this case, your administrator may have disabled the cleanup of promotion libraries at promote, demote, baseline ripple, and install.

Ask your administrator which way promotion works for your application.

Promotion Restrictions, Rules, and Options

Promotion Security

Each promotion level in an application is associated with a security entity, which is defined in your mainframe security system (RACF®, CA-ACF2®, or CA-Top Secret®).

Ask your administrator which promotion levels you have access to for promotion. Find out the procedure you must use to have your package promoted to levels to which you have not been granted promotion privileges.

Promotion Rule

The behavior of the promotion function is governed by the Promotion Rule. Your administrator selected a Promotion Rule for your application, and perhaps for individual promotion levels, that provide the level of management for change packages, components, and promotion libraries that is required by your application.

There are five promotion rules. The following table describes how the Promotion Rule determines the requirements for promoting and demoting packages and components.

Rule	Restrictions
0	Full and selective promote and demote are allowed without freezing the package first. Full promote can skip promotion levels. Requires the following sequence to change a promoted package component: Selective unfreeze (only if the package is frozen) Edit Stage (Restage) Selective freeze of the component (only if the package is frozen) Selective promotion to any level up to the package promotion level

Rule	Restrictions
1	Package must be frozen for promote and demote. Full promote cannot skip promotion levels. Requires the following sequence to change a promoted package component: Selective demote of the component Selective unfreeze Edit Stage Audit package Selective freeze of the component Selective promotion back to the package promotion level
2	Package must be frozen for promote and demote. Full promote cannot skip promotion levels. Requires the following sequence to change a promoted package component: Selective demote of the component Selective unfreeze Edit Stage Audit package Selective freeze of the component Selective promotion through all intermediate levels to the package promotion level
3	Package must be frozen for promote and demote. Full promote cannot skip promotion levels. Requires the following sequence to change a promoted package component: Full demote of the package Selective unfreeze of the component Edit Stage Audit package Selective freeze of the component Full promotion through all promotion levels up to the original promotion level

Rule	Restrictions
4	Package must be frozen for promote and demote. Full promote cannot skip promotion levels. Requires the following sequence to change a promoted package component: Full demote of the package Revert the package to development status Edit Stage Audit package Freeze package Full promotion through all intermediate levels to the package promotion level

Note

The Promotion Rule does not change the requirements for audit. If audit is required before freeze, then audit is required before selective freeze.

Ask your administrator which promotion rule applies to each promotion level that you will use.

Promotion Rule 0

In addition to the requirements listed in the Promotion Rule table above, the relaxed controls for Promotion Rule 0 include:

- Components that are not ACTIVE that are in library types eligible for promotion are bypassed and a message is displayed.
- Restaging a component sets the component promotion level to 0. The package promotion level is not changed.
- Package may be promoted to levels that are not the next contiguous level.

First Promote

When a package is at promotion level 0 (not promoted), special procedures apply to the first promotion action. These procedures, especially in combination with Promotion Rule 0, can be useful for testing individual package components before the entire package is ready for testing.

- You can selectively promote a component when the change package is at Level 0 (not promoted). A selective promote in these circumstances is labeled a **first promote**.
- When all components are promoted to the same level as the first promote, the package promotion level is changed to that level. The package may be fully promoted or demoted from this new level.
- You cannot promote a package where components have been promoted with first promote until all components are selectively promoted to that level.

General Restrictions And Options

These are general rules for promoting and demoting packages and components:

- Except for first promote, you cannot selectively promote a component to a promotion level higher than the package promotion level.
- Except for first promote, the package promotion level is set only by a full promotion. The package promotion level is reset after a full demotion, not after all components have been selectively demoted.
- A package can not be promoted or demoted until all of the components eligible for promotion are at the package level. You may have to selectively promote or demote components to align components at the package level.
- A package can not be promoted to its current promotion level. A component can not be selectively promoted to its current promotion level.
- A package can be fully promoted, or a component can be selectively promoted, to a lower level. The prior level is cleaned up, even if it is a higher level.
- Your administrator can configure your application to allow a package to be promoted to levels in more than one site at the same time, or they may require you to demote your package in one site before promoting your package in another site.
- Promotion or demotion for local sites is accomplished in one batch job. Promotion or demotion for remote sites requires three batch jobs, one of which runs at the remote location.
- All promotion jobs that run on the same MVS image as the user obtain JOB card information from the Promote Options Panel. Promotion jobs that run at remote sites obtain JOB card information from the Site Definition in application administration. If you have access to the remote JES queue, ask your administrator for the JOB name used for each remote site.

Scheduled Promotion/Demotion

You can schedule a package promotion/demotion at a future date or time. These rules apply.

- You can schedule package promotion/demotion only; ERO promotion cannot be scheduled, nor can any type of demotion.
- The scheduler holds only one scheduled promotion/demotion request at a time for a package.
- Anyone with monitor privileges can change or delete the schedule for a scheduled promotion/demotion using the monitor function.
- When you request a scheduled promotion/demotion for a package, the package must be in a valid promotion/demotion status (DEV, FRZ, APR).
- If the package contents or status change after you schedule a promotion/demotion, the scheduled promotion/demotion is not changed. You accept the risk that package contents and package status can change between the time that you request the scheduled promotion/demotion and the time that the promotion/demotion is initiated by the scheduler.
- A scheduled promotion/demotion may be full or selective. For full promotion/demotion, the component list is generated at run time. For selective promotion/demotion, you specify the list of components to be promoted or demoted when you request the scheduled promotion/demotion.
- The only scheduler available for scheduled promotion/demotion is CMN. The OTHER scheduler is not available for scheduled promotion/demotion, and MANUAL is equivalent to non-scheduled promotion/demotion.

Promotion Paths

Each promotion level must be associated with a site. Promotion sites can be configured to allow simultaneous promotion to levels in more than one site, or you may be required to demote from a level in one site to promote to a level in another site.

These rules can be manipulated to provide multiple promotion paths for a package in an application.

Ask your administrator which promotion paths are available for packages in your application. Ask about the requirements for demoting from one site to promote to another.

Promotion Libraries In SYSLIB Concatenations

Library concatenations for SYSLIB DD statements in compile and link edit JCL are automatically built by ChangeMan ZMF skeletons. These skeletons put staging libraries at the top of the concatenation and baseline libraries at the bottom. Promotion libraries are placed between staging and baseline libraries.

You can exclude individual promotion libraries from these SYSLIB concatenations by coding the SYSLIB Exclude field in the promotion library definition in application administration. The SYSLIB Exclude field only has meaning for like-copy and like-load library types.

Accessing Promotion Functions

Promote functions are accessed from the **Promote/Demote A Change Package** menu (CMNRPM00).

```
CMNRPM00                        Promote/Demote a Change Package
Option ===>______________________________________________________________

       P Promote D Demote
   Package . . . ACTP000038
```

Use one of these methods to display the **Promote/Demote A Change Package** menu:

- **Using the Menu Hierarchy:**

  On the **Primary Option Menu**, select **3 Promote**.

- **Using Direct Panel Access:**

  On a ChangeMan ZMF panel, type **=3** in the **Command** or **Option** line and press **Enter**.

Using the Change Package List to Promote and Demote

You can access the options listed on the **Promote/Demote A Change Package** menu directly from the **Change Package List**.

On the **Change Package List** panel, type one of the following in the line command for a change package and press **Enter**.

- **PR** Promote the change package
- **DM** Demote the change package

Promoting Packages And Components

This section describes how to promote packages and components.

Selecting A Promotion Site And Options

Follow these steps to select a promotion site and choose promotion options.

1. Bring up the Promote/Demote A Change Package menu.

```
CMNRPM00                               Promote/Demote a Change Package
Option ===>___________________________________________________________

       P Promote                                             D Demote
   Package . . . ACTP000038
```

2. On the Promote/Demote a Change Package menu, type a Package ID, type P in the Option field, and press Enter. The Promote Site List panel is displayed.

```
CMNRPM01                        PROMOTE Site List                      Row 1 to 3 of 3
Command ===>______________________________________________________    Scroll ===> CSR

          Package: DEMO000182        Status: DEV         Install date: 20230120

            Demote reqd  ------------- Last promotion details    --------------
  Site      prior sites  Type          Promotion Lvl Date        Time  User
__D001           N       Full Promote  1  D001LV1    2018/01/11  08:20 WSER27
__LOCAL          Y                     0
******************************* Bottom of data ********************************
```

> **Note**
>
> You can access the Promote Site List panel from the Change Package List by typing option PR beside a package. The Promote Site List is suppressed if there is only one site defined for an application. This table describes the fields on the Promote Site List panel.

Field	Description
*package user variables	Package user variables USR0101-USR0810 and USR1601- USR7205 now appear at the top of the Promote/Demote Site List (CMNRPM01) panel. (Not shown in the example above.)
Package	Displays the Package ID of the current change package
Status	Displays the current status of the change package.
Install date	Displays the date when this package is proposed for installation. (yyyymmdd).
Site	Displays the promotion site names defined for the application.

Field	Description
Demote reqd prior sites	Indicates the restriction on promoting to more than one site at the same time. **Y:** A package cannot be promoted in this site until it is fully demoted in all other sites. **N:** A package can be promoted in this site at the same time it is promoted in other sites if the **Demote reqd prior sites** is **N** for those sites also.
Last promotion details	Displays the last successful promote or demote action for this package in each site.
Type	Displays the type of action taken. **Full Promote:** The package was fully promoted. **Part Promote:** Package components were selectively promoted. **1st Promote:** Package components were selectively promoted while the package was at Level 0 (not promoted). **Full Demote:** The package was fully demoted. **Part Demote:** Package components were selectively demoted.
Promotion lvl	Displays the numeric promotion level and the name of the last promotion or demotion target.
Date	Displays the date of the last action. (yyyymmdd)
Time	Displays the time that the last action was completed. (hh:mm)
User	Displays the UserID of the person who initiated the last action.

3. On the Promote Site List panel, type an **S** next to the Site where you want to promote the package or components, and press Enter. The Promote Options panel is displayed.

```
CMNRPM03                               Promote Options
Option ===>___________________________________________________________________

       F  Full promotion           S   Selective promotion
       H  Display history          O   Check for overlay

          Package: ACTP000001       Status: DEV       Install date: 20171119

Promotion site:               SERT6
Current promotion level:      S6P1UT + 10

Next promotion level . . 20
Schedule: date . . . . . ______     time . . . . . _____

Enter "/" to select option
  _ Short selection list     _ Bypass overlay check
  / User variables           _ Suppress batch messages

Job statement information:
    //USER015B JOB (SM-1IKF-SM),'COMPARE',
    //         CLASS=A,MSGCLASS=X,NOTIFY=USER015
    //*
    //* JOB TO PROMOTE
```

This table describes the fields on the Promote Options panel:

Field	Description
Option	Type a one-character code for the promotion action you want to perform: **F:** Fully promote the change package. **S:** Selectively promote components in the change package. **H:** Display all promote or demote actions that were initiated for this package **O:** Check a target promotion level for potential component overlays.
Schedule	date - The date (yyyymmdd) to promote the package or component.
	time - The time (hh:mm) to promote the package or component.
Short selection list	If not selected, then display all package components on the selective promotion selection list. If selected, then limit the selection list for selective promotion to package components that are not currently promoted to the target level, including components that may have been re- staged, newly activated into the package, or overlaid by promotion of another package.
Bypass overlay check	Type your choice for bypassing the check of the target promotion libraries for potential component overlays before promotion is executed. Select this option to bypass the check for potential component overlays in target promotion libraries.

Field	Description
	NOTE** Components in promotion libraries that have the same name as components in your package are replaced.
	If not selected, then check target promotion libraries for potential component overlays. If potential overlays are detected, display a panel listing the common components, and allow the user the opportunity to discontinue the promotion action.
User Variables	Enter '/' to specify user variables V01 through V10 on panel CMNUSV1 to pass information to skeleton file tailoring.
Suppress batch messages	Type your choice for broadcasting a SEND message at the completion of the promotion job to tell you whether the promotion function was a success. If not selected then broadcast a SEND message at the completion of the promotion job. Select this option to suppress the SEND message normally broadcast at the completion of the promotion job.
Job statement information	Job card information for promotion jobs that will run on the same MVS image as the user.

4. On the Promote Options panel, type the appropriate entries and press Enter.

    - To fully promote your package, type F in the Option field. See Full Promote.

        To selectively promote components in your package, type S in the Option field. See Selective Promote.

    - To display the promotion and demotion history of the package for this site, type H in the Option field. See Displaying Promotion/Demotion History For A Site.

    - To check a target promotion level for potential component overlays, type O in the Option field. See Check for Potential Overlays.

Full Promote

This section assumes that you executed the steps in Selecting A Promotion Site and Options and that you selected Option F on the Promote Options panel.

1. If you typed YES in the Bypass Overlay Check field on the Promotion Options panel, ChangeMan ZMF immediately builds full promote job JCL and submits the first job.

    If the batch promotion process is successful, you will receive an MVS send message telling you that your package was promoted and indicating the site, nickname and promotion level to which it was promoted and the date and time that the process was completed.

2. If you typed NO in the Bypass Overlay Check field on the Promotion Options panel, ChangeMan ZMF examines the target promotion libraries and promotion history to detect potential overlays.

   a. If no potential overlays are detected, ChangeMan ZMF builds full promote job JCL and submits the first job.

      If the batch promotion process is successful, MVS sends a message telling you that your package was promoted and indicating the site, nickname and promotion level to which it was promoted and the date and time that the process was completed.

   b. If potential overlays are detected, the Common Components panel is displayed to show components that may be common between your change package and the target promotion libraries.

```
CMNRPM06                Common Components: Lvl 20 (S6P1IT)                  OVERLAY
WARNING
Command ===>                                                               Scroll
===> CSR

         Package: ACTP000038            Status: DEV                Install date:
20150218

Name + Type   Package      Sta    Promotion     Date         Time     User      Status
ACPSRCEE LOD  ACTP000032   DEV   20 S6P1IT    2015/02/01   22:06    USER015   Common
ACPSRCEE SRC  ACTP000032   DEV   20 S6P1IT    2015/02/01   22:06    USER015   Common
******************************* Bottom of data ********************************
```

      See the full description of the Common Components panel for a description of each field on this panel.

   c. Carefully examine the component and package information displayed on the Common Components panel. Contact the people who promoted packages that are listed on this panel to discuss the impact of overlaying their components in test libraries. They may be able to demote their package so you can promote your package without overlays.

   d. If you want to stop your promotion process, press PF3 or type End on the Command line and press Enter. No test libraries change.

   e. If you choose to ignore the overlay warning, press Enter to continue your promotion. ChangeMan ZMF builds full promote job JCL and submits the first job.

      If the batch promotion process is successful, MVS sends a message telling you that your package was promoted and indicating the site, nickname and promotion level to which it was promoted and the date and time that the process was completed.

⚠ **Caution**

If you ignore the overlay warning and promote your package, you may copy components from your package over existing components in test libraries. This may invalidate test results for other projects.

Selective Promote

This section assumes you executed the steps in Selecting A Promotion Site And Options and that you selected Option S on the Promote Options panel.

1. When you select Option S on the Promotion Option panel and press Enter, the Promote To panel is displayed.

```
CMNRPM05           PROMOTE TO: SERT6 Lvl 20 (S6P1IT)             Row 1 to 10 of 10
Command ===> ______________________________________________________ Scroll ===> CSR

         Package: ACTP000038       Status: DEV       Install date: 20150218
  Name    +   Type Status          Promotion     Date        Time     User     REQUEST
_ ACPSRCCE    LOD                 10  S6P1UT    2015/02/01  22:16    USER015
_ ACPSRCCE    SRC                 10  S6P1UT    2015/02/01  22:16    USER015
_ ACPSRCEE    LOD                 10  S6P1UT    2015/02/01  22:16    USER015
_ ACPSRCEE    SRC                 10  S6P1UT    2015/02/01  22:16    USER015
_ ACPSRC1A    LOD                 10  S6P1UT    2015/02/01  22:16    USER015
_ ACPSRC1A    SRC                 10  S6P1UT    2015/02/01  22:16    USER015
_ ACPSRC6A    LOD                 10  S6P1UT    2015/02/01  22:16    USER015
_ ACPSRC6A    SRC                 10  S6P1UT    2015/02/01  22:16    USER015
_ ACPSRC92    LOD                 10  S6P1UT    2015/02/01  22:16    USER015
_ ACPSRC92    SRC                 10  S6P1UT    2015/02/01  22:16    USER015
******************************* Bottom of data ********************************
```

This table describes the fields on the Promote To:

Field	Description
Package	Displays the Package ID of the current change package.
Status	Displays the current status of the change package.
Install date	Displays the date when this package is proposed for installation. (yyyymmdd).
Name	Displays the names of the package components that are ACTIVE and in a library type set up for promotion for the target promotion level.
Type	Displays the library type of the listed component.

Field	Description
Status	Displays the status of the promoted component. **Blank:** Component has never been promoted in this package or has not changed since it was last promoted. **Restaged:** Component was staged again in this package since it was last promoted in this package. These components are treated as if they are at promotion level 0. **Overlaid:** Component has been overlaid in the promotion library by another package since the last promotion in this package. These components are treated as if they are at promotion level 0. **Deleted:** Component was promoted in this package but deleted from the package. These components will only show on the demotion list and are only eligible for demotion.
Promotion	Displays the level number and nickname of the last promotion level to which the component was promoted in this package.
Date	Displays the date when the component was last promoted in this package. (yyyymmdd).
Time	Displays the time when the component was last promoted in this package. (hh:mm).
User	Displays the TSO ID of the person who last promoted the component in this package.
REQUEST	Displays the action requested for each listed component. Type S beside a component and press Enter to change the Request field to `*SELECT*`. Type D beside a selected component to change the Request back to blank.

2. Type S beside each component that you want to selectively promote.

   a. Press Enter to change the Request field to *SELECT*.

   b. Type D beside any selected component to deselect it and exclude it from selective promotion.

   c. Press PF3 or type End on the Command line and press Enter to execute the selective promote.

3. If you typed Yes in the Bypass Overlay Check field on the Promotion Options panel, ChangeMan ZMF immediately builds selective promote job JCL and submits the first job.

   If the batch promotion process is successful, MVS sends a message telling you that your package was promoted and indicating the site, nickname and promotion level to which it was promoted and the date and time that the process was completed.

4. If you typed No in the Bypass Overlay Check field on the Promotion Options panel, ChangeMan ZMF examines the target promotion library and history to detect potential overlays.

a. If no potential overlays are detected, ChangeMan ZMF builds selective promote job JCL and submits the first job.

   If the batch promotion process is successful, MVS sends a message telling you that your package was promoted and indicating the site, nickname and promotion level to which it was promoted and the date and time that the process was completed.

b. If potential overlays are detected, the Common Components panel is displayed to show components that may be common between your change package and the target promotion libraries. See the full description of the Common Components panel for a description of each field on this panel.

c. Carefully examine the component and package information displayed on the Common Components panel. Contact the people who promoted packages that are listed on this panel to assess the impact of overlaying their components in test libraries. They may be able to demote their package so you can promote your package without overlays.

d. If you want to stop your promotion process, press PF3 or type End on the Command line and press Enter.

e. If you choose to ignore the overlay warning, press Enter to continue your promotion process.

   ChangeMan ZMF builds selective promote job JCL and submits the first job. If the batch promotion process is successful, MVS sends a message telling you that your package was promoted and indicating the site, nickname and promotion level to which it was promoted and the date and time that the process was completed.

> ⚠ **Caution**
>
> If you ignore the overlay warning and promote your package, you may copy components from your package over existing components in test libraries. This can invalidate test results for other projects.

Check for Potential Overlays

This section assumes you executed the steps above and that you selected **Option O** on the Promote Options panel.

Use this promotion option before you promote your package to see if there are other projects that already promoted the components you are changing. If you overlay a component that was promoted from another change package, you might invalidate testing that is being conducted by the person who promoted the component previously.

When you select Option O on the Promotion Option panel and press Enter, the Common Components panel is displayed to show components that may be common between your change package and the target promotion libraries.

```
CMNRPM06              Common Components: Lvl 20 (S6P1IT)           OVERLAY WARNING
Command ===>                                                      Scroll ===> CSR

      Package: ACTP000038        Status: DEV        Install date: 20150218
Name + Type  Package     Sta    Promotion   Date        Time      User         Status
ACPSRCEE LOD ACTP000032  DEV    20  S6P1IT  2015/02/01 22:06     USER015      Common
ACPSRCEE SRC ACTP000032  DEV    20  S6P1IT  2015/02/01 22:06     USER015      Common
******************************* Bottom of data ********************************
```

This table describes the fields on the Common Components panel.

Field	Description
Package	Displays the Package ID of the current change package
Status	Displays the current status of the change package.
Install date	Displays the date when this package is proposed for installation. (yyyymmdd).
Name	Displays the name of the package component that already exists in the target promotion library or in history for this promotion target. This is the member that will potentially be overlaid if the current promotion action is continued.
Type	Displays the library type of the common component.
Package	Displays the Package ID of the package from which the component in the target promotion library was promoted.
Sta	Displays the status of the package from which the component in the target promotion library was promoted.
Promotion	Displays the nickname and level number of the target promotion level.
Date	Displays the date when the component in the target promotion library was promoted. (yyyymmdd)
Time	Displays the time when the component in the target promotion library was promoted. (hh:mm)
User	Displays the TSO ID of the person who promoted the component in the target promotion library.
Status	Displays the type of potential overlay.
	Common: The component is physically present in the target promotion library, and there is history of another package promoting the component to this promotion level in this site.
	Nohist: The component is physically present in the target promotion library, but there is no history of another package promoting the component to this promotion level in this site.
	History: The component is not physically present in the target promotion library, but there is history of another package promoting the component to this promotion level in this site.

Field	Description
	Restaged: The component is physically present in the target promotion library, and there is history of this package promoting the component to this promotion level in this site before the component was restaged.

Carefully examine the component and package information displayed on the Common Components panel. Contact the people who promoted packages that are listed on this panel to assess the impact of overlaying their components in test libraries. They may be able to demote their package so you can promote your package without overlays.

Demoting Packages and Components

This section describes how to demote packages and components.

Selecting A Demotion Site And Options

Follow these steps to select a demotion site and choose demote options.

1. On the Primary Option menu, type 3 in the Option field and press Enter. The Promote/ Demote A Change Package panel is displayed.

```
CMNRPM00                Promote/Demote a Change Package
Option ===>______________________________________________

          P Promote                 D Demote
    Package . . . ACTP000038
```

2. On the Promote/Demote A Change Package panel, type a Package ID, type D in the Option field, and press Enter. The Demote Site List panel is displayed.

```
CMNRPM01                   DEMOTE Site List                     Row 1 to 2 of 2
Command ===> ---------------------------------------------------Scroll ===> CSR

    Package: DEMO000182         Status: DEV     Install date: 20230120

        Demote reqd ------------- Last promotion details ------------------
 Site   prior sites   Type          Promotion Lvl  Date         Time    User
_D001       N         Full Promote  1 D001LV1      2018/01/11   08:20   USER27
_LOCAL      Y                       0
********************************* Bottom of data *******************************
```

Note

You can access the Demote Site List panel from the Change Package List by typing option DM beside a package.

This table describes the fields on the Demote Site List panel.

Field	Description
*package user variables	Package user variables USR0101-USR0810 and USR1601- USR7205 now appear at the top of the Promote/Demote Site List (CMNRPM01) panel. (Not shown in the example above.)
Package	Displays the Package ID of the current change package
Status	Displays the current status of the change package.
Install date	Displays the date when this package is proposed for installation. (yyyymmdd).
Site	Displays the promotion site names defined for the application.
Demote reqd prior sites	Indicates the restriction on promoting to more than one site at the same time. **Y:** A package cannot be promoted in this site until it is fully demoted in all other sites. **N:** A package can be promoted in this site at the same time it is promoted in other sites if the **Demote reqd prior sites** is **N** for those sites also.
Last promotion details	Displays the last successful promote or demote action for this package in each site.
Type	Displays the type of action taken, if any. **Full Promote:** The package was fully promoted. **Part Promote:** Package components were selectively promoted. **1st Promote:** Package components were selectively promoted while the package was at Level 0 (not promoted). **Full Demote:** The package was fully demoted. **Part Demote:** Package components were selectively demoted.
Promotion lvl	Displays the numeric promotion level and the name of the last promotion or demotion target.
Date	Displays the date of the last action. (yyyymmdd)
Time	Displays the time that the last action was completed. (hh:mm)
User	Displays the TSO ID of the person who initiated the last action.

3. On the Demote Site List panel, type an S next to the Site where you want to demote the package or components, and press Enter. The Demote Options panel is displayed.

```
CMNRPM04                    Demote Options
Option ===>______________________________________

    F Full demotion         S Selective demotion
    H Display history

     Package: ACTP000001    Status: DEV    Install date: 20171119

Promotion site:           SERT6
Current promotion level:  S6P1UT + 10

Demote from level . . . . 10       (blank, * or 0 for a list)
Schedule: date . . . . .  _______  time . . . . . . _____

Enter "/" to select option
  / User variables        _ Suppress batch messages

Job statement information:
   //USER015B JOB (SM-1IKF-SM),'DEMOTE',
   // CLASS=A,MSGCLASS=X,NOTIFY=USER015
   //*
   //* JOB TO DEMOTE
```

This table describes the fields on the Demote Options panel.

Field	Description
Option	Type a one-character code for the promotion action you want to perform: **F:** Fully demote the change package. **S:** Selectively demote components in the change package. **H:** Display all promote or demote actions that were initiated for this package.
Promotion site	Displays the promotion site name selected on the previous panel.
Current promotion level	Displays the nickname and level number of the current promotion level. If STAGING + 0 is displayed, then the package is not currently promoted. **NOTE** Component promotion levels are not displayed here.
Demote from level	Type the promotion level from which demotion will take place. This field is preset to the current package promotion level. You can type blank, 0, or `*` to see a list of promotion levels available for this site in this application.
Schedule	date - The date (yyyymmdd) to demote the package or component.
	time - The time (hh:mm) to demote the package or component.
User Variables	Enter '/' to specify user variables V01 through V10 on panel CMNUSV1 to pass information to skeleton file tailoring.
Suppress batch messages	Select to broadcast a SEND message at the completion of the promotion job to tell you whether the promotion function was a success.

Field	Description
Job statement information	Job card information for promotion jobs that will run on the same MVS image as the user.

4. On the Demote Options panel, type the appropriate entries and press Enter.

   a. To fully demote your package, type F in the Option field.

   b. To selectively demote components in your package, type S in the Option field.

   c. To display the promotion and demotion history of the package for this site, type H in the Option field. See Displaying Promotion/Demotion History For A Site

Full Demote

If you executed the steps above, and if you selected **Option F** on the Demote Options panel, ChangeMan ZMF builds full demote job JCL and submits the first job.

If the batch demotion process is successful, MVS sends a message telling you that your package was demoted and indicating the site, nickname and promotion level from which it was demoted and the date and time that the process was completed.

Selective Demote

This section assumes that you executed the steps above and that you selected **Option S** on the Demote Options panel.

1. When you select Option S on the Demotion Option panel and press Enter, the following panel is displayed.

```
CMNRPM05    DEMOTE FROM: SERT6 Lvl 10 (S6P1UT)               Row 1 to 10 of 10
Command ===>                                                  Scroll ===> CSR

    Package: ACTP000038     Status: DEV       Install date: 20150218
  Name     +     Type Status   Promotion     Date        Time    User       REQUEST
_ ACPSRCCE      LOD           10 S6P1UT     2015/02/01 22:16  USER015
_ ACPSRCCE      SRC           10 S6P1UT     2015/02/01 22:16  USER015
_ ACPSRCEE      LOD           10 S6P1UT     2015/02/01 22:16  USER015
_ ACPSRCEE      SRC           10 S6P1UT     2015/02/01 22:16  USER015
_ ACPSRC1A      LOD           10 S6P1UT     2015/02/01 22:16  USER015
_ ACPSRC1A      SRC           10 S6P1UT     2015/02/01 22:16  USER015
_ ACPSRC6A      LOD           10 S6P1UT     2015/02/01 22:16  USER015
_ ACPSRC6A      SRC           10 S6P1UT     2015/02/01 22:16  USER015
_ ACPSRC92      LOD           10 S6P1UT     2015/02/01 22:16  USER015
_ ACPSRC92      SRC           10 S6P1UT     2015/02/01 22:16  USER015
******************************* Bottom of data ********************************
```

   This table describes the fields on the Demote: *package* Components panel.

Field	Description
Package	Displays the Package ID of the current change package.

Field	Description
Status	Displays the current status of the change package.
Install date	Displays the date when this package is proposed for installation. (yyyymmdd).
Name	Displays the names of the package components that are ACTIVE and in a library type set up for promotion for the target promotion level.
Type	Displays the library type of the listed component.
Status	Displays the status of listed components that are currently in this package promoted. **Blank:** Component has never been promoted in this package or has not changed since it was last promoted. **RESTAGED:** Component was staged again in this package since it was last promoted in this package. These components are treated as if they are at promotion level 0. **OVERLAID:** Component has been overlaid in the promotion library by another package since the last promotion in this package. These components are treated as if they are at promotion level 0. **DELETED:** Component was promoted in this package but deleted from the package. These components will only show on the demotion list and are only eligible for demotion.
Promotion	Displays the level number and nickname of the last promotion level to which the component was promoted in this package.
Date	Displays the date when the component was last promoted in this package. (yyyymmdd)
Time	Displays the time when the component was last promoted in this package. (hh:mm)
User	Displays the userID of the person who last promoted the component in this package.
REQUEST	Displays the action requested for each listed component. Type S beside a component and press Enter to change the Request field to `*SELECT*`. Type D beside a selected component to change the Request back to blank.

2. Type S beside each component you want to selectively demote.

   a. Press Enter to change the Request field to *SELECT*.

   b. Type D beside any selected component to deselect it and exclude it from selective promotion.

   c. Press PF3 or type End on the Command line and press Enter to execute the selective demote.

3. ChangeMan ZMF builds selective demote job JCL and submits the first job.

If the batch promotion process is successful, MVS sends a message telling you that your package was demoted and indicating the site, nickname and promotion level from which it was demoted and the date and time that the process was completed.

Displaying Promotion/Demotion History For A Site

The Promotion History panel shows the current status of all promote and demote actions initiated for a change package in a site.

There are three ways to access this panel:

- From the Promote Options panel in Promote/Demote
- From the Demote Options panel in Promote/Demote
- From the Package Information panel in Query Package.

When you have navigated the required panels, the (site) Promotion History panel is displayed.

```
CMNRPM02          SERT6 Promotion History              Row 1 to 6 of 6
Command ===>                                          Scroll ===> CSR

      Package: ACTP000038     Status: DEV     Install date: 20150218
Count   Promotion    Date         Time   User      Status     History
10      10 S6P1UT    2015/02/01  22:16  USER015  Completed  Full Promote
4       20 S6P1IT    2015/02/01  22:15  USER015  Completed  Full Demote
4       20 S6P1IT    2015/02/01  21:40  USER015  Completed  Full Promote
4       10 S6P1UT    2015/02/01  19:53  USER015  Completed  Full Promote
2       10 S6P1UT    2015/02/01  19:52  USER015  Completed  Full Demote
2       10 S6P1UT    2015/02/01  19:35  USER015  Completed  Full Promote
******************************* Bottom of data ********************************
```

This table describes the fields on the Common Components panel.

Field	Description
Package	Displays the Package ID of the current change package
Status	Displays the current status of the change package.
Install date	Displays the date when this package is proposed for installation. (yyyymmdd).
Count	Displays the number of package components promoted or demoted.
Promotion	Displays the level number and nickname of the target promotion level.
Date	Displays the date when the when the promote or demote action was taken. (yyyymmdd)
Time	Displays the time when the promote or demote action was taken. (hh:mm)
User	Displays the UserID of the person who initiated the promote or demote action.

Field	Description
Status	Displays the current status of the batch promotion process. **SUBMITTED:** The batch process was submitted, but all jobs did not complete successfully. A job may be waiting for an initiator, a job may still be running, or a job may have failed. **COMPLETED:** The batch promote or demote process completed successfully.
History	Displays the type of action in the last promote or demote request. **Full Promote:** Full package promote. **Part Promote:** Selective promote of package components. **1st Promote:** First selective promote of a package component while the package is at Level 0. **Full Demote:** Full package demote. **Part Demote:** Selective demote of a package component

16. Approving or Rejecting a Package

In Approve Package, predefined approvers review package information, components, and test results, and then approve or reject the package for installation.

About Approving or Rejecting a Package

The Approve Package function guarantees that changes to production components are fully authorized.

An approval list is assigned to a package when the package is created. The approval list for planned packages may be different from the approval list for unplanned packages. Other approvals may be automatically assigned to the package later in the package lifecycle, but no required approval is removed.

There can be multiple approvals required for a package, and multiple individuals can be granted authorization in your security system to enter a particular approval. Your application administrator can define approvals in a hierarchy so that some approvals must be entered before others, or your administrator can allow approvals to be entered in any order.

Your application administrator can set up notifications for approvals. The notification list is separate from the list of userids authorized to enter an approval. Notifications can be sent using MVS SEND broadcast messages or other methods such as email. If your application administrator defines approvals in a hierarchy, notification for an approval is held until the approval can be entered. Otherwise, all notifications are sent when a package is frozen.

A package approver has four choices when they take action on an approval:

- Approve the package for installation.
- Reject the package and require that the package be reverted to DEV status for changes.
- Flag the package as Under Review, indicating that the package will be approved or rejected at a later time.
- Add notes to a free form text checkoff list panel without approving or rejecting the package.

Your administrator can configure your application to build package installation JCL when you freeze your package or when the final package approval is entered. Installation JCL is created in a file tailoring server that runs in an address space separate from your ChangeMan ZMF session.

When the final approval for a package is entered, the status of the package is immediately changed to APR. If installation JCL is built after final approval in your application, the package is not distributed to production sites or inserted into the scheduler, and you cannot revert the package, until file tailoring for installation JCL completes successfully.

Rules for Approving and Rejecting Packages

These rules and restrictions apply to the approve package function.

- Either the Planned Approval List or the Unplanned approval list is assigned when a package is *created*. The Unplanned Approval List is assigned to unplanned packaged that are created outside of Normal Business Hours defined in application administration. The planned approval list is assigned to all other packages.
- The assignment of the Planned or Unplanned Approval List cannot be changed after the package is created except through the use of user exits. See Exit Programs for Approve Package Functions for details.
- Planned approvals defined as Interfacing Approvals in other applications are assigned to your participating package when it is frozen if you have designated other applications as Affected Applications in your package.
- If your application administrator changes the Planned Approval List or the Unplanned Approval List (including approver notifications) after your package is created, those changes are not reflected in your package unless you revert your package from a status of FRZ or higher.
- After a package with an Unplanned Approval List is installed and baselined, the Planned Approval List is added to the package. Approvals that are on both lists are not added again. ChangeMan ZMF does not enforce the added approvals, but Report 120 Unplanned Packages Pending Post Approval can be used in your post-installation procedures to make sure approvers examine these packages and enter their approvals.
- When an approver rejects a package, they must enter free form text (up to 10 lines) in Reject Reasons, which usually explain why the approver rejected the package and prevented installation.

- If an approver rejects a package, the package must be reverted to DEV status. No other approver can approve or reject the package until the package is frozen again.
- When a package is reverted, approvals that were entered previously are removed from the package.
- If an approver flags a package as Under Review, approvals with the same sequence number (at same hierarchical level) can be entered, but approvals at higher sequence numbers cannot be entered.

Authorization for Approve Package Functions

Package approvers must have these privileges in your security system.

- READ access to the security entity for the application.
- UPDATE access to the security entity for the approval specified in the planned or unplanned approval list.

Administration Settings for Approve Package Functions

Your Global and Application Administrators make settings in ChangeMan ZMF administration that control how the approve package function works in your application. Ask your administrator if any of the following apply to your application.

- Enforce the hierarchy of approval list definitions, or allow approvals to be entered in any order. (Application Administration Parameters: Hierarchical Approval Process)
- Prohibit package creators from approving their own change package. (Application Administration Parameters: Package Creator Cannot Approve)
- Prohibit anyone who has worked on a component in a package from approving the package. (Application Administration Parameters: Package Worker Cannot Approve)
- Prohibit the same person from entering more than one approval for a change package. (Application Administration Parameters: Only 1 Approval Per User Id)
- Create installation JCL at freeze or when the final approval is entered. (Application Administration Parameters: Build Install JCL At Approve)

Exit Programs for Approve Package Functions

Your ChangeMan ZMF installer or administrators may implement exit programs to alter the behavior ChangeMan ZMF. Ask your administrator if any of the following apply to your application.

- Based on package conditions, automatically add more approvals to the Planned Approval List for planned packages or to the Unplanned Approval List for unplanned packages when a planned package is frozen or when an unplanned package is frozen and installed (exit program CMNEX009).
- Restrict who can approve a package. (Exit program CMNEX010).

Accessing Approve Package Functions

To get to the **Approve/Reject Options** menu, you must use the **Approve Package Parameter** panel. Use one of these methods to display **Approve Package Parameter** panel:

- Using the Menu Hierarchy:

  On the **Primary Option Menu**, select **4 Approve**.
- Using Direct Panel Access:

On a ChangeMan ZMF panel, type **=4** in the **Command** or **Option** line and press **Enter**.

```
CMNAPPRV                            Approve Package Parameters
Command ===>

Specify selection criteria:
Package . . . . . . . . . . ACTP000027      (Full name or pattern, blank for list)
Approval entity name . . .                  (Approvals for this security entity)
Notified user . . . . . . .                 (Packages for which user was notified)
Work request . . . . . . .
Department . . . . . . . .
Site affected . . . . . . .
Package level . . . . . . .                 (1-Simple, 4-Participating)
Package type . . . . . . .                  (Planned or Unplanned)
Package time span . . . . .                 (Permanent or Temporary)
Install date: from . . . .                  (yyyymmdd)
 to . . . . .                               (yyyymmdd)
Creation date: from . . . .                 (yyyymmdd)
 to . . . . .                               (yyyymmdd)

Enter "/" to select option
_ Other parameters
```

> **Note**
>
> Fields named here for ’Work request’ and ’Department’ can have their names customized by Administrators.

Approve functions are accessed from the **Approve/Reject Options** menu (CMNAPPOP).

```
CMNAPPOP                      Approve/Reject Options
Option ===>_________________________________________________________

      Package: ACTP000027    Status: FRZ    Install date: 20150228
1   Approve       Approve or reject a change package
2   Reasons       Display reasons a package was rejected
3   Query         Query change package
4   Reset         Reset approval in progress indicator
5   Re-Build      Re-submit install JCL build request
6   Compare       Compare staging to baseline or promotion
```

Using the Change Package List to Access Approve Functions

You can also access all the options listed on the **Approve/Reject Options** menu directly from the **Change Package List**.

On the **Change Package List** panel, type one of the following commands in the line command field for a change package and press **Enter**.

- **A1** Approve/Reject a change package
- **A2** Display reject reasons
- **QP** Query package
- **A4** Reset approval in progress indicator
- **A5** Re-submit installation JCL build request (post approve)
- **CC** Compare staging to baseline or promotion

Selecting Packages for Approve Functions

Follow these steps to select packages for approval functions.

1. When you access the approve package function, you are presented first with the **Approve Package Parameters** panel (CMNAPPRV).

```
CMNAPPRV                         Approve Package Parameters
Command ===>______________________________________________________

Specify selection criteria:
Package . . . . . . . . . . ACTP000027  (Full name or pattern, blank for list)
Approval entity name . . .              (Approvals for this security entity)
Notified user . . . . . . .             (Packages for which user was notified)
Work request . . . . . . .
Department . . . . . . . .
Site affected . . . . . . .
Package level . . . . . . .             (1-Simple, 4-Participating)
Package type . . . . . . .              (Planned or Unplanned)
Package time span . . . . .             (Permanent or Temporary)
Install date: from . . . .              (yyyymmdd)
              to . . . . .              (yyyymmdd)
Creation date: from . . . .             (yyyymmdd)
               to . . . . .             (yyyymmdd)

Enter "/" to select option
   Other parameters
```

Type selection criteria on the **Approve Package Parameters** panel to display packages with approvals pending.

This table describes the fields on the **Approve Package Parameters** panel.

Field	Description
Package	Type one of the following: **Package ID:** Select one package regardless of any other selection criteria entered on the Approve Package Parameters panel. (You can omit leading zeros in the package number.) If you specify a Package ID, you can specify a package in FRZ, REJ, INS, BAS or TCC status **Application:** Select packages in this application. **Package ID Pattern:** Use `*` or `?` as wildcard characters to specify Package IDs to select. See Building Lists Using Patterns. **blank:** Select packages from all application.
Approval entity name	Type an approval entity to select. You can see what approval entity is associated with a package approval by querying the package, choosing Approval List, then selecting an approver. Your application administrator or security administrator will know who is associated with an approval entity in your security system.

Field	Description
Notified user	Type 1-8 character ID to select an approver that was notified. **Note:** You cannot select on IDs longer than 8 characters, even though some notification vehicles use longer IDs, such as e-mail addresses.
Work request	Type a work request to select. This is a 12-character free-form field.
Department	Type a department to select. This is a 4-character free-form field.
Site affected	Type a site to select package to be installed at that site.
Package level	Type a package level to select. **1:** Simple packages **4:** Participating packages
Package type	Type a package type to select. **PLANNED:** Planned packages **UNPLANNED:** Unplanned packages **Note:** If you type an entry in this field, you must also type an entry in **Package Time Span**.
Package time span	Type a package time span to select. **PERM:** Permanent packages **TEMP:** Temporary packages **Note:** If you type an entry in this field, you must also type an entry in **Package Type**.
Install date: from	Type the low end of a range of package install dates to select. This date is inclusive. Blank is the same as 00000000.
Install date: to	Type the high end of a range of package install dates to select. This date is inclusive. Blank is the same as 99999999.
Creation date: from	Type the low end of a range of package create dates to select. This date is inclusive. Blank is the same as 00000000.
Creation date: to	Type the high end of a range of package create dates to select. This date is inclusive. Blank is the same as 99999999.
Other parameters	Select to display the **Extended Search Criteria** panel where you can enter more parameters to be used to select packages for the Approve Package List.

If you select the **Other Parameters** field on the **Approve Package Parameters** panel, the **Extended Search Criteria** panel (CMNQRY01) is displayed.

```
CMNQRY01                      Extended Search Criteria
Command ===>____________________________________________________

Enter "/" to select:
  Component records      _ Nonsource     _ Source      _ Load
                         _ Rename        _ Scratch     _ Forms

  Scheduling system      _ CMN     _ Manual     _ Other

Component type . . . .

Component name list    (Blank, Full name or Mask separated by ;)
_______________________________________________________________
Component userid list (Blank, Full name or Mask separated by ;)
_______________________________________________________________
Release id list (Blank, Full name or Mask separated by ;)
_______________________________________________________________
```

The following table describes the fields on this panel. All field entries are optional. If you leave any field blank, all change packages in the system that satisfy the specified selection criteria are presented.

Field	Description
Component records	Select for each package component type to select.
Scheduling system	Select for each installation scheduler to select.
Component type	Type a component type or pattern to select.
Component name list	Type a list of component names or patterns to select, separated by semicolons (;). The maximum length of each component name or mask is 16 characters.
Component userid list	Type a list of TSO IDs or patterns to select who worked on a package component, separated by semicolons (;).
Release id list	Type a list of release IDs or patterns to select, separated by semicolons (;). (only appears if you are licensed for ERO).

When you have entered your selection criteria on the **Approval Package Parameters** panel and the **Extended Search Criteria** panel, press **Enter**. The information you type on these panels is processed as follows:

- Fields on these panels are combined with logical AND operators to select change packages with approvals pending. Field values within a specific field, such as COMPONENT RECORDS, are combined with logical OR operators; the results of using patterns in such fields as RELEASE ID LIST are also combined with logical OR operators.
- Packages in APR status that are in Monitor/Limbo are also selected.
- If you specify a specific package ID, you can select packages in FRZ, REJ, INS, BAS or TCC status.

- All fields on these panels are optional. Blank fields are treated as select all.
- Patterns and wildcard characters are not allowed except in the Package ID field. See Building Lists Using Patterns.
- Package selection criteria that you type are stored and displayed the next time these panels display.

2. If there are packages that fit your selection criteria, the **Approve Package List** panel (CMNAPPL1)is displayed.

```
CMNAPPL1                    Approve Package List                  Row 1 to 1 of 1
Command ===>                                                     Scroll ===> csr

   package    sta   install     lvl  type    dept   promotion    Aud     Creator
   ACTP000027 FRZ   20150228    SMPL PLN/PRM IDD    00 STAGING   00      USER015
****************************** Bottom of data ********************************
```

On the **Approve Package List** panel, you can enter one of these line commands on a package row:

- S (Select) - Display the **Approve/Reject Options** menu.
- Q (Query) - Display the **Package Information** panel of the query package function.
- A (Approve) - Automatically enter all approvals that you are authorized to enter for the package.

Caution

The A (Approve) line command does not provide the opportunity for you to see either the Approve/Reject Options menu, which offers you other options besides approval, or the Approval List panel, which shows you all of the approvals for the package.

3. On the **Approve package List** panel, if you type **S** in the line command for a package, the **Approve/Reject Options** menu (CMNAPPOP) is displayed.

```
CMNAPPOP Approve/Reject Options
Option ===>
    Package: ACTP000027 Status: FRZ Install date: 20150228
1 Approve   Approve or reject a change package
2 Reasons   Display reasons a package was rejected
3 Query     Query change package
4 Reset     Reset approval in progress indicator
5 Re-Build  Re-submit install JCL build request
6 Compare   Compare staging to baseline or promotion
```

Approve or Reject a Package

Follow these steps to approve or reject a change package.

1. On the **Approve/Reject Options** menu, select option **1 Approve** to display the **Approval List** panel (CMNAPPLS).

```
CMNAPPLS                Approval List                           Row 1 to 3 of 3
Command ===>_____________________________________________      Scroll ===> CSR

        Package: ACTP000027     Status: FRZ     Install date: 20150228

Approver Description                                   User
                                                       Date  Time  Seq  Status
_ Development Team Lead
                                                                   10
_ Quality Assurance
                                                                   20
_ Program Manager
                                                                   30
******************************* Bottom of data ********************************
```

This table describes the fields and valid entries on the **Approval List** panel.

Field	Description
Package	Displays the ID of the package you are acting on.
Status	Displays the current package status.
Install date	Displays the package install date.
Line Command	Type an action in the line command for an approval. **A:** Approve the package for installation. **R:** Reject the package. **C:** Enter free form text on the Checkoff List panel for this approval. Anyone authorized to enter this approval can view and update the text on the Checkoff List panel. **V:** Flag this approval as “in review”. This approval can be approved or rejected later. Other approvals at the same hierarchical level can be entered, but approvals at a higher level cannot be entered.
Approver description	Displays the description of the approver.
User Date	Displays the userid of the approver who last took action for this approval and the date that the action was taken.
Time	Displays the time that the last action was taken for this approval.

Field	Description
Seq	Displays the sequence number of the approval. If hierarchical approvals are enabled in application administration, sequence numbers determine the order that approvals can be entered.
LP	Indicates that this entry is a linked package approval. These approvals have a sequence number of 00, and may be approved in any sequence.
Status	Displays the status of the approval. **Approve:** A approve action was taken for this approval. Reject This approval was rejected. No further approve or reject actions are permitted for this package. Check The Checkoff List panel for this approval was displayed, but not necessarily updated. Review This approval is flagged as “in review”. This approval can be approved or rejected later. Other approvals at the same hierarchical level can be entered, but approvals at a higher level cannot be entered.

2. Use line commands to act on the package approvals shown on the **Approval List** panel.

Approve Package

1. On the **Approval List** panel, type **A** in the line command for an approval and press **Enter**. The approval status is updated. You can enter other approvals if you are authorized.

```
CMNAPPLS                    Approval List                     Row 1 to 3 of 3
Command ===> _____________________________________________   Scroll ===> CSR

         Package: ACTP000027     Status: FRZ     Install date: 20150228

   Approver Description                 User
                                        Date    Time    Seq    Status
 _ Development Team Lead
                                                        10

 _ Quality Assurance
                                                        20

 _ Program Manager
                                                        30
******************************* Bottom of data ********************************
```

2. Press **PF3** to return to the **Approve/Reject Options** menu.

Reject Package

1. On the **Approval List** panel, type **R** in the line command for an approval and press **Enter**. The **Reject Reasons** panel (CMNREJR1) is displayed.

```
CMNREJR1                        Reject Reasons
Command ===>

         Package: ACTP000027    Status: FRZ     Install date: 20150228

Approver: Development Team Lead
User:

    Unable to justify need for this code change. Rejected.__________
    ______________________________________________________________________
    ______________________________________________________________________
    ______________________________________________________________________
    ______________________________________________________________________
    ______________________________________________________________________
    ______________________________________________________________________
```

2. Type up to 10 lines of free form text. This text usually explains why you rejected the package approval.
3. Press **Enter** to record your text and return to the **Approve/Reject Options** menu where the package status is updated to REJ.

Checkoff List

1. On the **Approval List** panel, type **C** in the line command for an approval and press **Enter**. The **Checkoff List** panel (CMNCHKLS) is displayed.

```
CMNCHKLS                                                  Checkoff List
Command ===>

  Package: ACTP000027      Status: REJ        Install date: 20150228

Approver: Development Team Lead

Approval withheld until:
1. Code must satisfy standards_______________________________________
2. Testing must be independently verified by user department_________
_____________________________________________________________________
_____________________________________________________________________
_____________________________________________________________________
_____________________________________________________________________
_____________________________________________________________________
```

2. Type up to 14 lines of free form text. You can use this text for any purpose. The text can be viewed and updated until this approval is approved or rejected. Press **Enter** to record your text

and return to the **Approval List** panel, where the approval status is updated. You can enter other approvals if you are authorized.

3. Press **PF3** to return to the **Approve/Reject Options** menu.

Review Package

1. On the **Approval List** panel, type **V** in the line command for an approval and press **Enter**. The approval status is updated. You can enter other approvals if you are authorized.

```
CMNAPPLS                  Approval List                   Row 1 to 3 of 3
Command ===>____________________________________________ Scroll ===> CSR

       Package: ACTP000027       Status: FRZ       Install date: 20150228

   Approver Description                   User
                                          Date        Time  Seq   Status
 _ Development Team Lead                  USER019
                                          20150115    1916  10    Review
 _ Quality Assurance
                                                            20
 _ Program Manager
                                                            30
******************************* Bottom of data ********************************
```

2. Press **PF3** to return to the **Approve/Reject Options** menu.

Displaying Reject Reasons

After a package has been rejected by an approver, you can view the reject reasons to see why the package was rejected.

Follow these steps to display the reject reasons.

1. On the **Approve/Reject Options** menu, select option **2 Reasons** to display the **Reject Reasons Selection List** panel (CMNREJR0).

```
CMNREJR0               Reject Reasons Selection List                Row 1 to 1 of 1
Command ===>_______________________________________________________ Scroll ===> CSR

         Package: ACTP000027     Status: REJ     Install date: 20150228

   Approver Description
s Development Team Lead
******************************* Bottom of data ********************************
```

2. On the **Reject Reasons Selection List** panel, type **S** in the line command for an approval and press **Enter** to display the **Reject Reasons** panel (CMNREJR1).

```
CMNREJR1                      Reject Reasons                       BROWSE ONLY
Command ===>_____________________________________________

        Package: ACTP000027      Status: REJ       Install date: 20150228

Approver: Development Team Lead
User:     USER019

  Rejected due to time frames. Try again next calendar month.
```

3. Press **PF3** repeatedly to return to the **Approve/Reject Options** menu.

Resetting the Approval-In-Progress Indicator

Option **4 Reset** on the **Approve/Reject Options** menu (CMNAPPOP) is obsolete and performs no function.

Resubmitting Install JCL Build Request

If your package is in APR status, and you want to rebuild the package install JCL, you can resubmit the request to build install JCL by using option **5 Re-build** on the **Approval Options** menu.

Authorization to use this option depends on whether file tailoring for install JCL was previously executed successfully.

- If package freeze processing successfully changes the status of your package to FRZ, but file tailoring for install JCL fails, anyone authorized to update the application can resubmit the JCL build request.
- If the package is in FRZ status, and file tailoring for install JCL has completed successfully, global or application administration authority is required to resubmit a request to build install JCL.

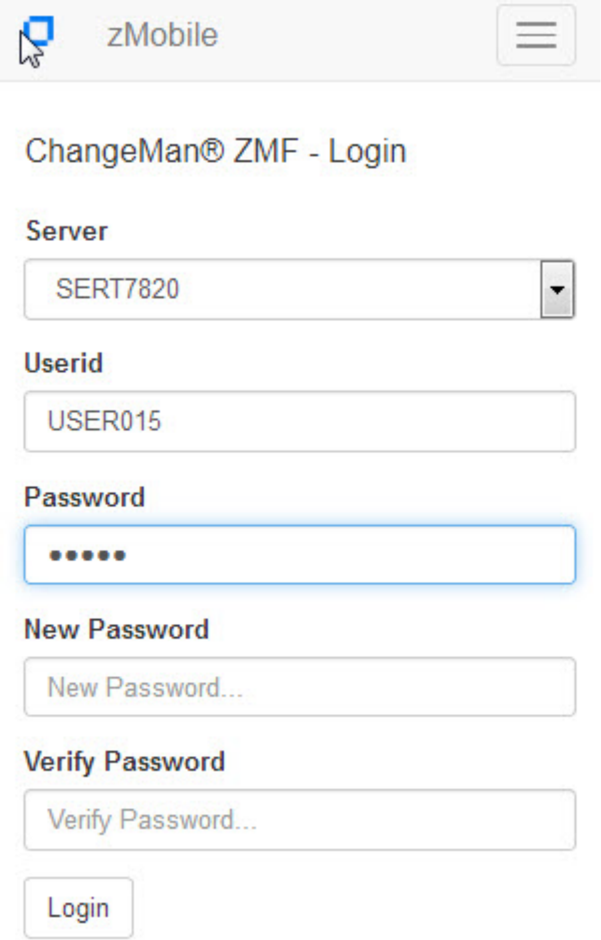
Using zMobile

zMobile is a Web Application for ZMF Approvers. Package Approvers may access zMobile from any device without signing on to TSO/ISPF. If your site has customized ChangeMan ZMF to sent emails with an embedded link to request your approval, then you can click on the link in the email, or copy the link and paste it into your browser. Here is an explanation of the structure of zMobile, and some of the screens you will see in that process.

A sample of the email you will receive if enabled:

```
From: ChangeMan@mail.serena.com <ChangeMan@mail.serena.com>
Sent: Monday, July 30, 2018 5:25 PM
To: USER015
Subject: ChangeMan ZMF package ACTP000004 awaits your approval.
Package: ACTP000004
Title: Test Package
Level: smpl
Type: pln/prm
Dept: 1000
Work request: WORK#000
Requestors name: user025
Requestor phone: 12345
Creators userid: USER025
Install date: 20180817
Approve ACTP000004
or copy and paste the following link to your browser:
http:\\RM8553.serena.com:8080\zmobile\approvepackage\SERT7820\ACTP000004
Approve ACTP000004
or copy and paste the following link to your browser:
https:\\RM8553.serena.com:8080\zmobile\approvepackage\SERT7820\ACTP00000
```

Click on either the http or the https link shown, depending upon your site configuration, or copy the link desired and paste it into your browser. The browser will then give you the logon screen:



ChangeMan® ZMF - Login

Server: SERT7820

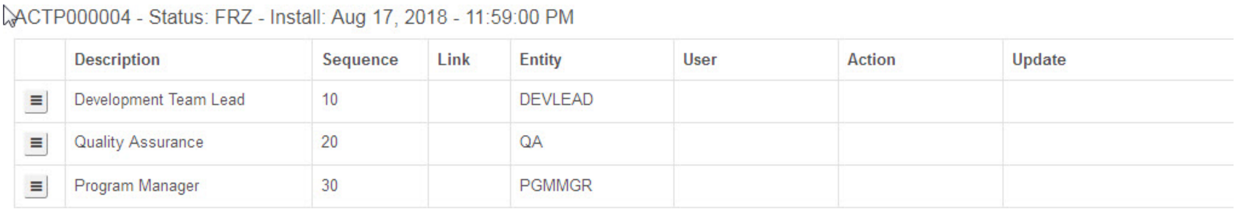
Userid: USER015

Password: •••••

New Password: New Password...

Verify Password: Verify Password...

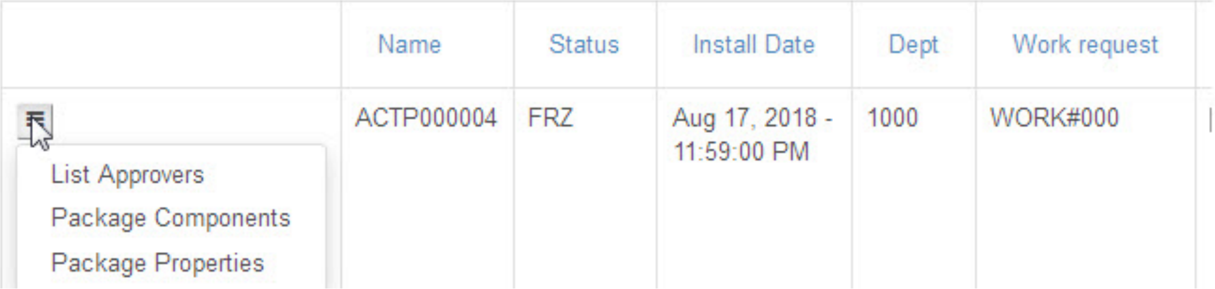
When you logon this way, the next screen will be a list of approvers for the specified package in the email:



ACTP000004 - Status: FRZ - Install: Aug 17, 2018 - 11:59:00 PM

	Description	Sequence	Link	Entity	User	Action	Update
≡	Development Team Lead	10		DEVLEAD			
≡	Quality Assurance	20		QA			
≡	Program Manager	30		PGMMGR			

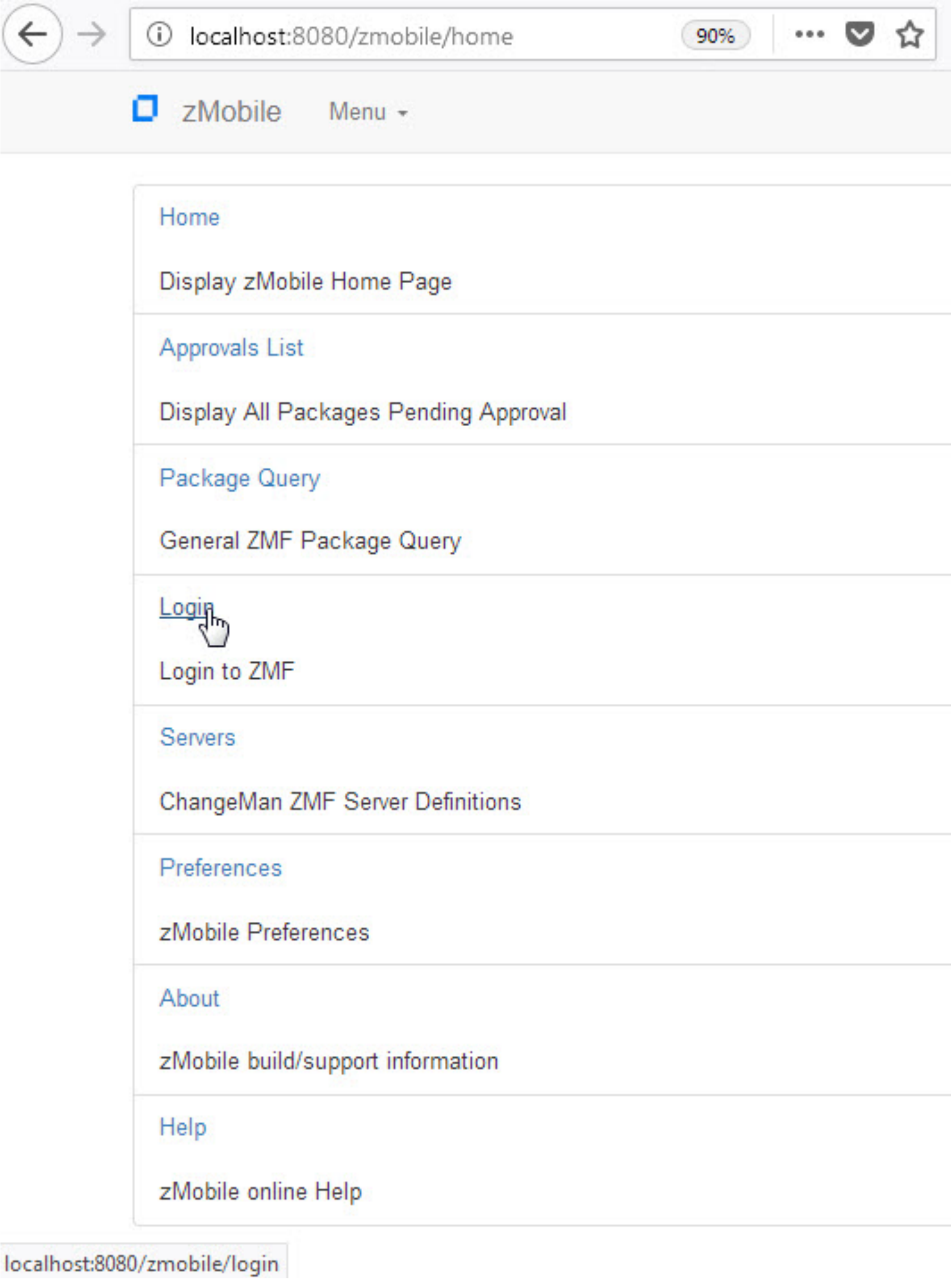
Put the cursor on the menu to the left of the desired description:



	Name	Status	Install Date	Dept	Work request
≡	ACTP000004	FRZ	Aug 17, 2018 - 11:59:00 PM	1000	WORK#000

Follow through with the option(s) desired from the menu to act upon the approval request.

Alternatively you can go directly to zMobile in your browser, logon from the primary zMobile menu:



localhost:8080/zmobile/home

90%

zMobile

Menu

Home

Display zMobile Home Page

Approvals List

Display All Packages Pending Approval

Package Query

General ZMF Package Query

Login

Login to ZMF

Servers

ChangeMan ZMF Server Definitions

Preferences

zMobile Preferences

About

zMobile build/support information

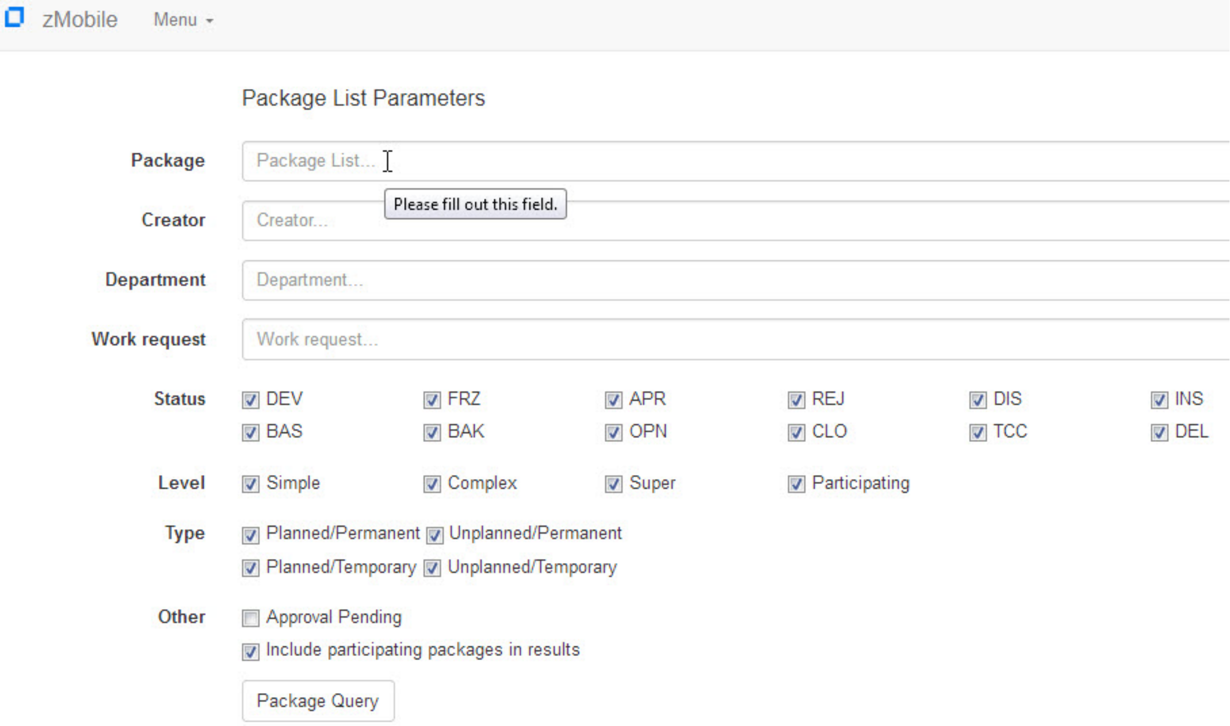
Help

zMobile online Help

localhost:8080/zmobile/login

From the Home Page shown above, the ZMF Functions available are 'Approvals List' and 'Package Query'. Select 'Approvals List' to display all packages that have a pending approval indicator. If you select 'Package Query' you will next be presented with a form to filter the list of packages displayed.

An example is:

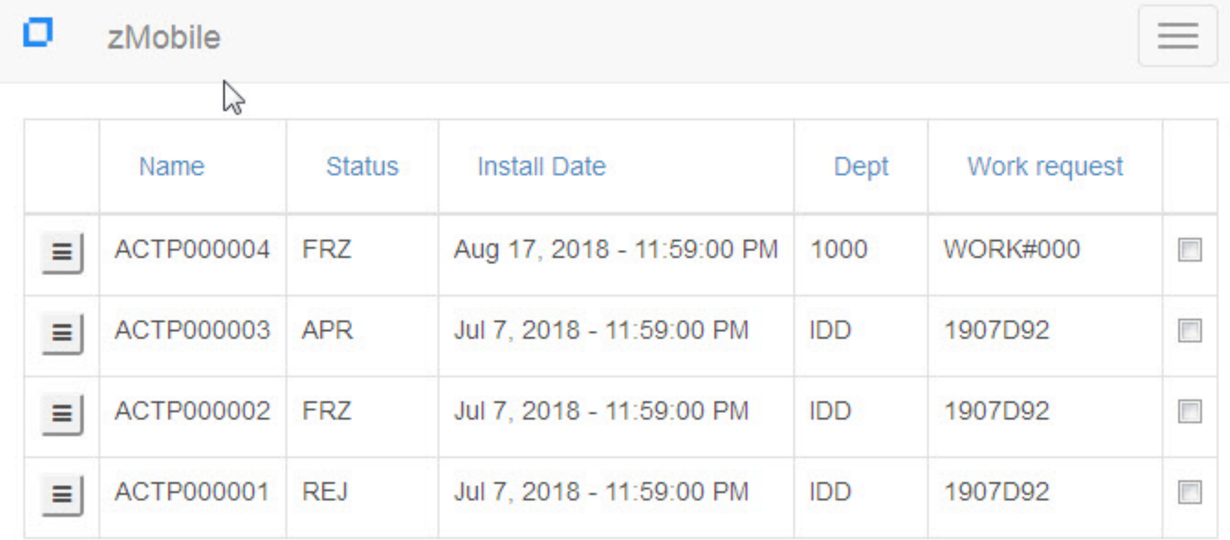


Package List Parameters

Package	Package List...
Creator	Creator...
Department	Department...
Work request	Work request...
Status	☑ DEV ☑ FRZ ☑ APR ☑ REJ ☑ DIS ☑ INS ☑ BAS ☑ BAK ☑ OPN ☑ CLO ☑ TCC ☑ DEL
Level	☑ Simple ☑ Complex ☑ Super ☑ Participating
Type	☑ Planned/Permanent ☑ Unplanned/Permanent ☑ Planned/Temporary ☑ Unplanned/Temporary
Other	□ Approval Pending ☑ Include participating packages in results

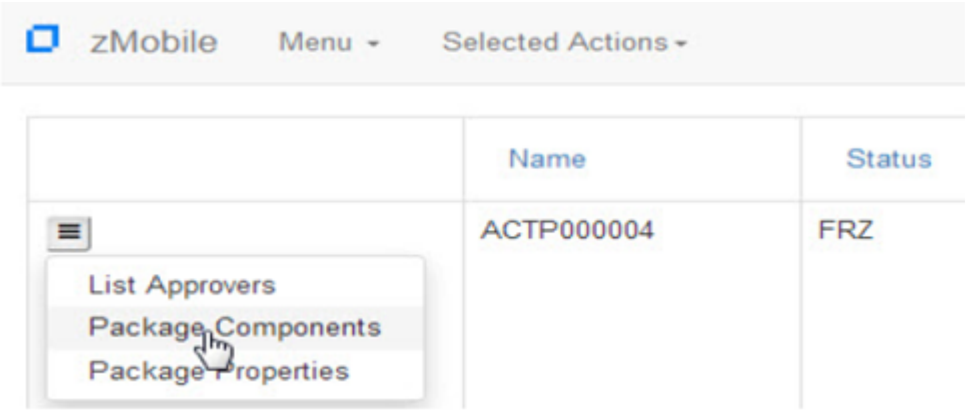
Please fill out this field.

Both functions, 'Approvals List' and 'Package Query' will result in a package list similar to this (Approvals List won't show the package with a Status of APR):



	Name	Status	Install Date	Dept	Work request	
≡	ACTP000004	FRZ	Aug 17, 2018 - 11:59:00 PM	1000	WORK#000	□
≡	ACTP000003	APR	Jul 7, 2018 - 11:59:00 PM	IDD	1907D92	□
≡	ACTP000002	FRZ	Jul 7, 2018 - 11:59:00 PM	IDD	1907D92	□
≡	ACTP000001	REJ	Jul 7, 2018 - 11:59:00 PM	IDD	1907D92	□

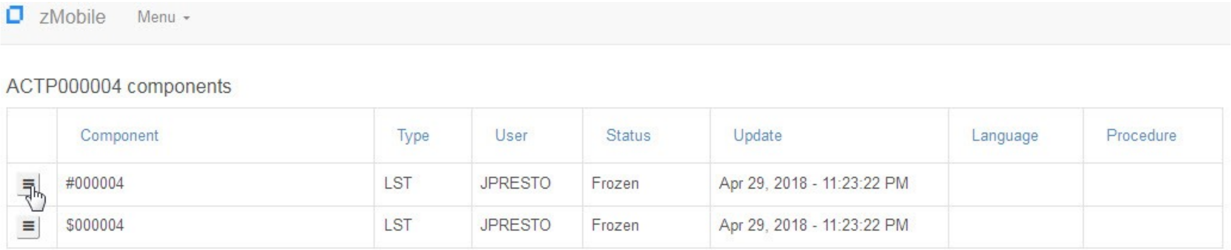
Available Functions are displayed when the menu button on the left of each package is selected:



	Name	Status
	ACTP000004	FRZ

The first option, List Approvers, gives you the list of approvers, shown earlier, and you can Approve, Reject, Checkoff, Review and Package Query.

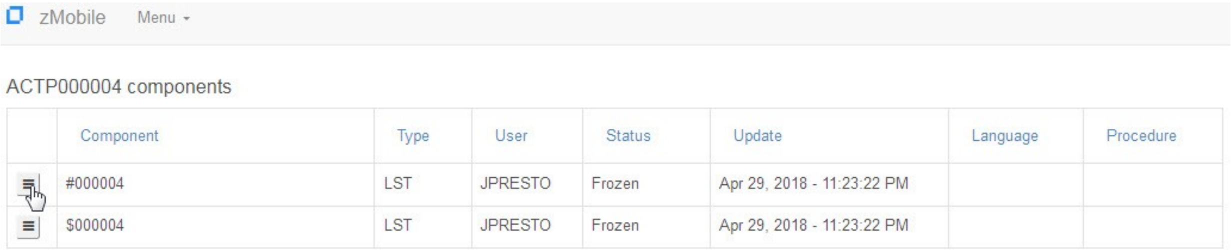
The second option gives you a component list selection to filter a list of the components in the package:



ACTP000004 components

	Component	Type	User	Status	Update	Language	Procedure
	#000004	LST	JPRESTO	Frozen	Apr 29, 2018 - 11:23:22 PM		
	$000004	LST	JPRESTO	Frozen	Apr 29, 2018 - 11:23:22 PM		

Add filters if desired, and click the ’Package Components’ button and the next display will show the components:

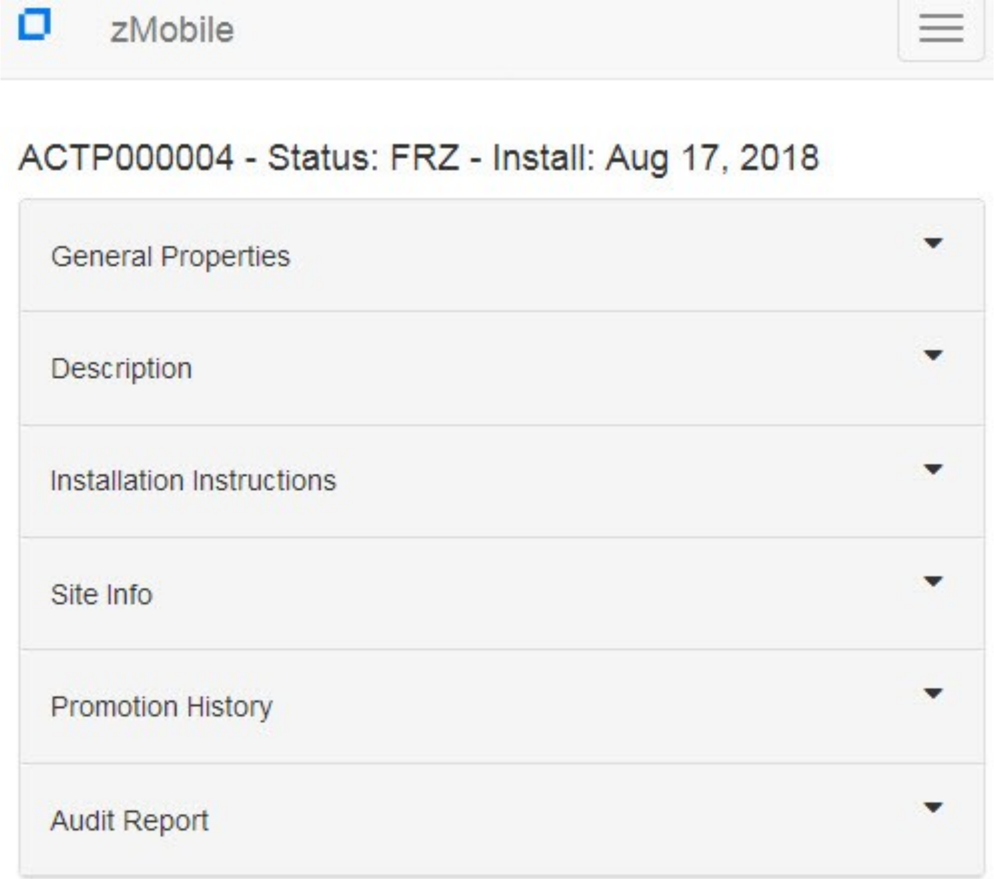


ACTP000004 components

	Component	Type	User	Status	Update	Language	Procedure
	#000004	LST	JPRESTO	Frozen	Apr 29, 2018 - 11:23:22 PM		
	$000004	LST	JPRESTO	Frozen	Apr 29, 2018 - 11:23:22 PM		

The Menu button on the left allows you to browse the component or look at the history.

Package Properties gives you a screen with detailed information about the package. Each item may be expanded/collapsed by selecting the toggle buttions to the right of each package category:



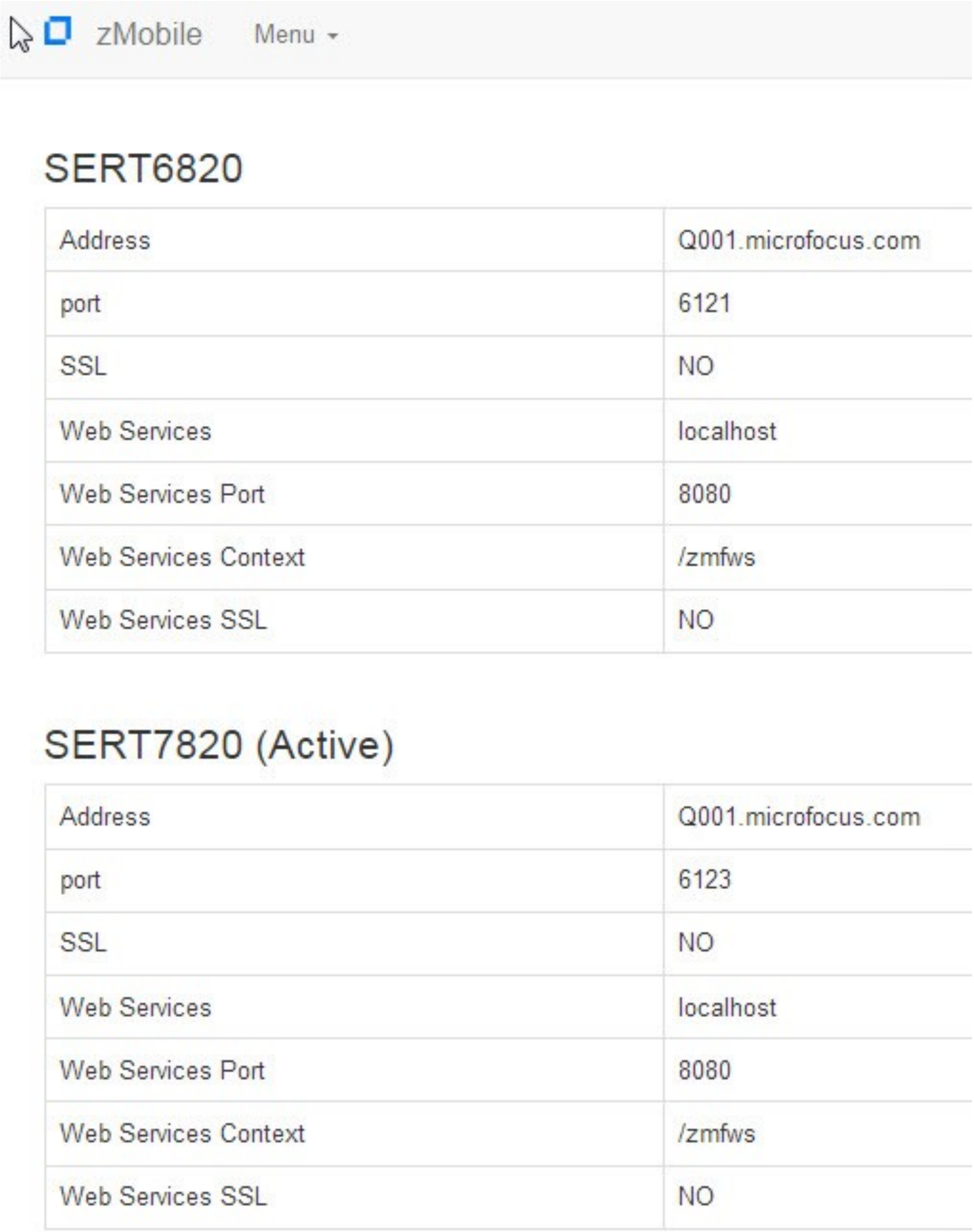
zMobile

ACTP000004 - Status: FRZ - Install: Aug 17, 2018

Detailed information is available about the Package, by selecting the toogle buttons including:

- General Properties
- Description
- Installation Instructions
- Site Information
- Promotion History
- Participating Packages
- Audit Report
- Other Primary Menu options include:

Servers available and the settings for each. See this example:



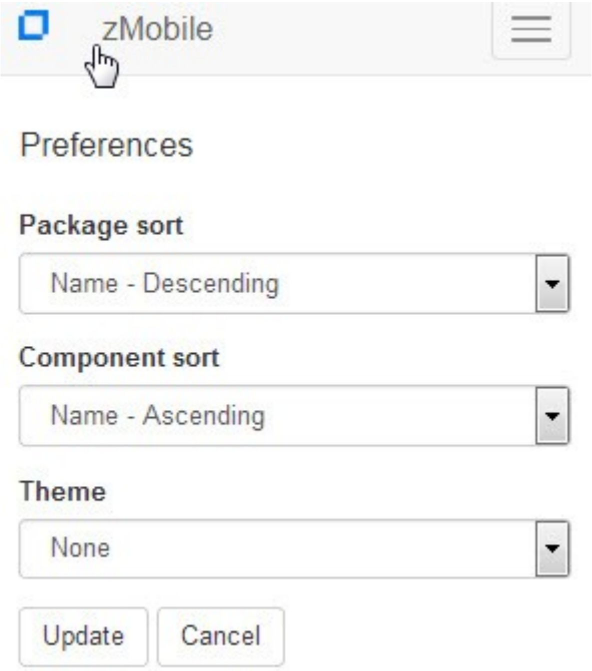
SERT6820

Address	Q001.microfocus.com
port	6121
SSL	NO
Web Services	localhost
Web Services Port	8080
Web Services Context	/zmfws
Web Services SSL	NO

SERT7820 (Active)

Address	Q001.microfocus.com
port	6123
SSL	NO
Web Services	localhost
Web Services Port	8080
Web Services Context	/zmfws
Web Services SSL	NO

The next option on the Primary Menu allows you to change your zMobile Preferences:


zMobile

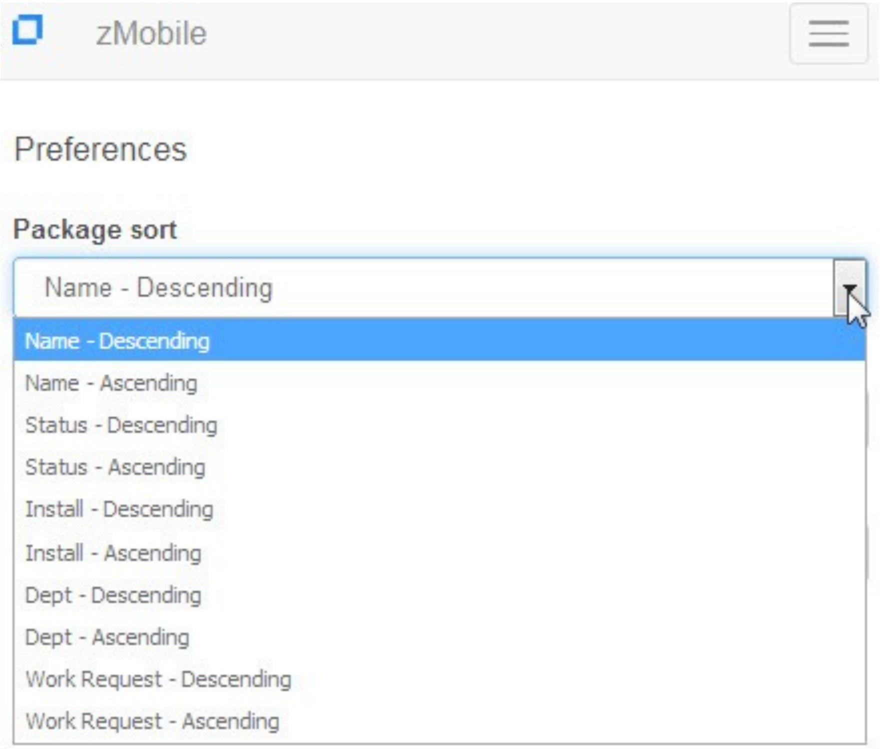
Preferences

Package sort: Name - Descending

Component sort: Name - Ascending

Theme: None

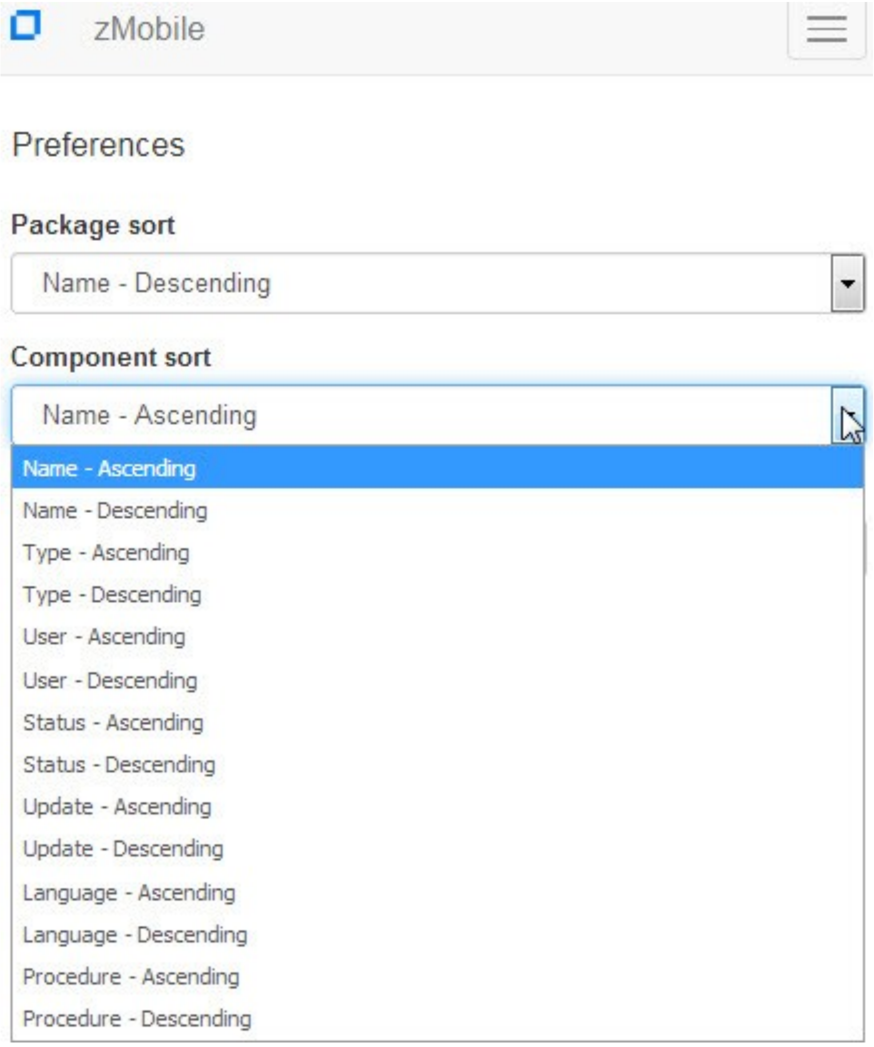
Package sort allows you to sort by these fields:


zMobile

Preferences

Package sort: Name - Descending

Component sort has these options:

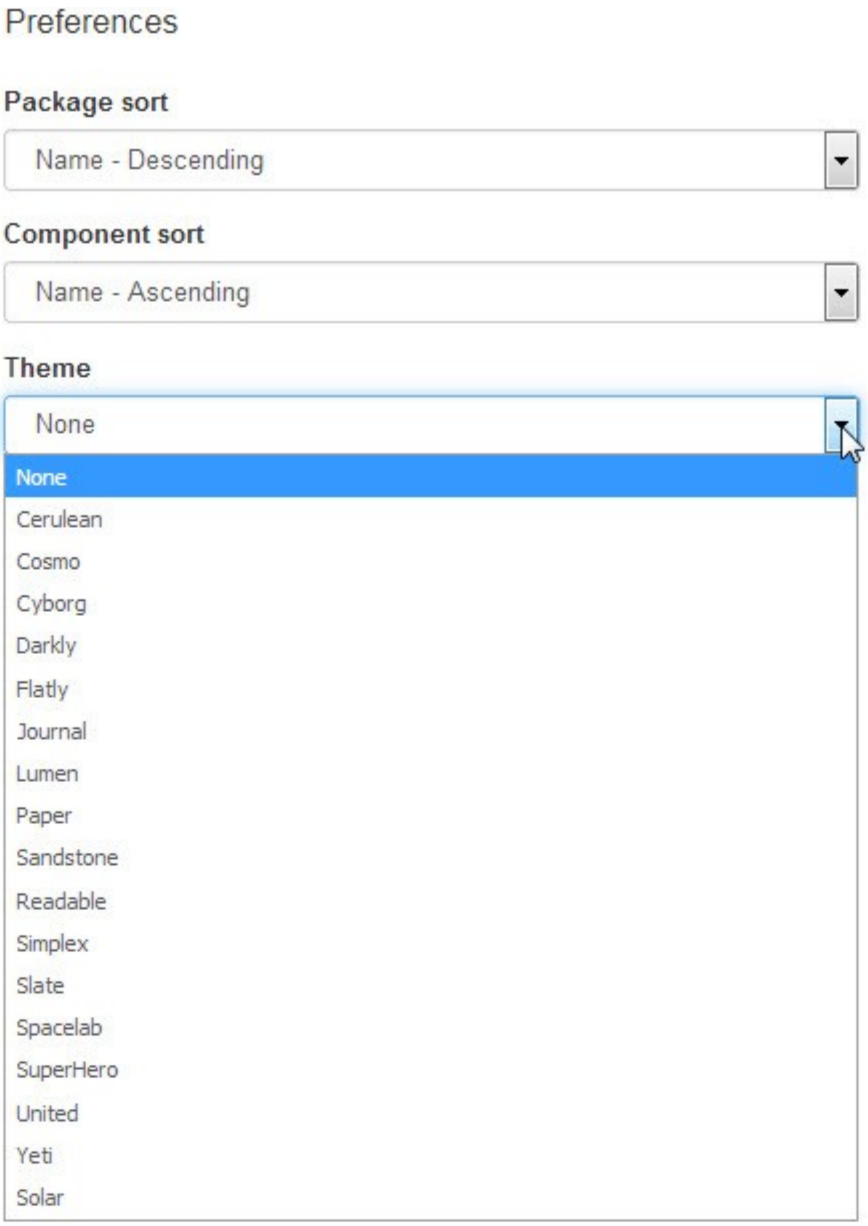


Preferences

Package sort: Name - Descending

Component sort: Name - Ascending

Also the Theme can be altered:



Preferences

Package sort

Name - Descending

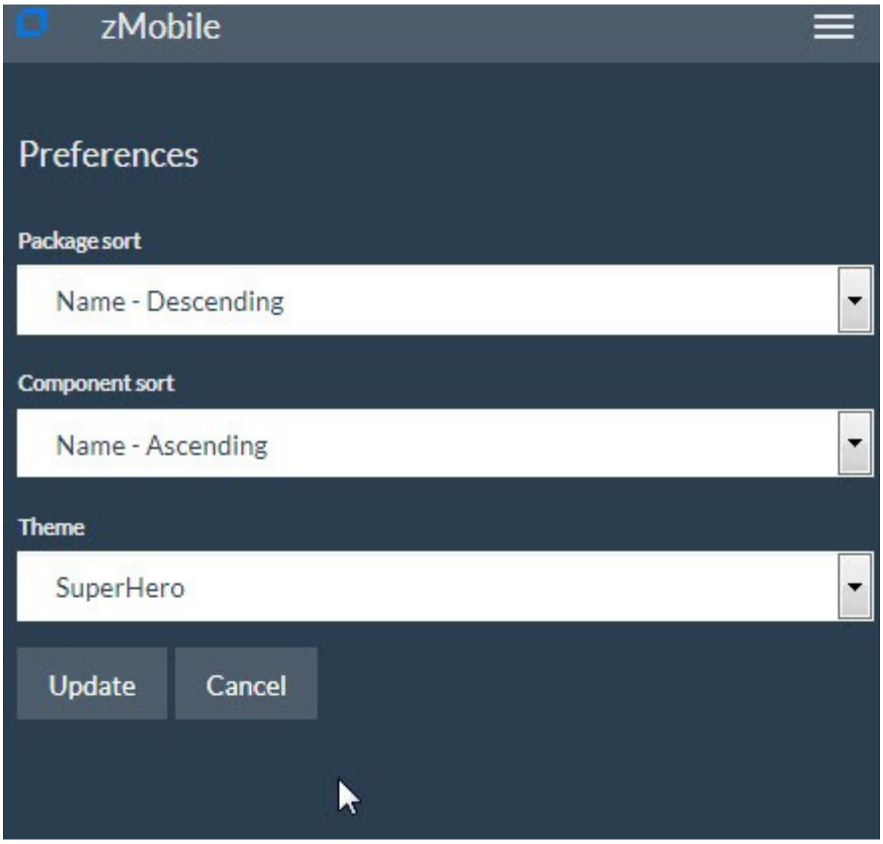
Component sort

Name - Ascending

Theme

None

An example of selecting the SuperHero Theme looks like this:



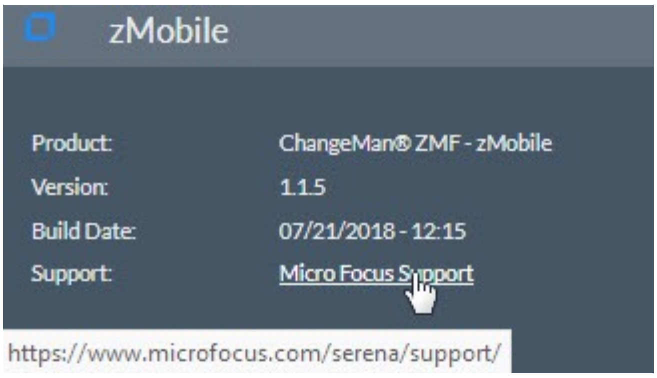
Preferences

Package sort: Name - Descending

Component sort: Name - Ascending

Theme: SuperHero

The next option on the Primary Menu, 'About', will display a screen with the Product, Version, Build Date, and a link to Micro Focus Support:



Product: ChangeMan® ZMF - zMobile

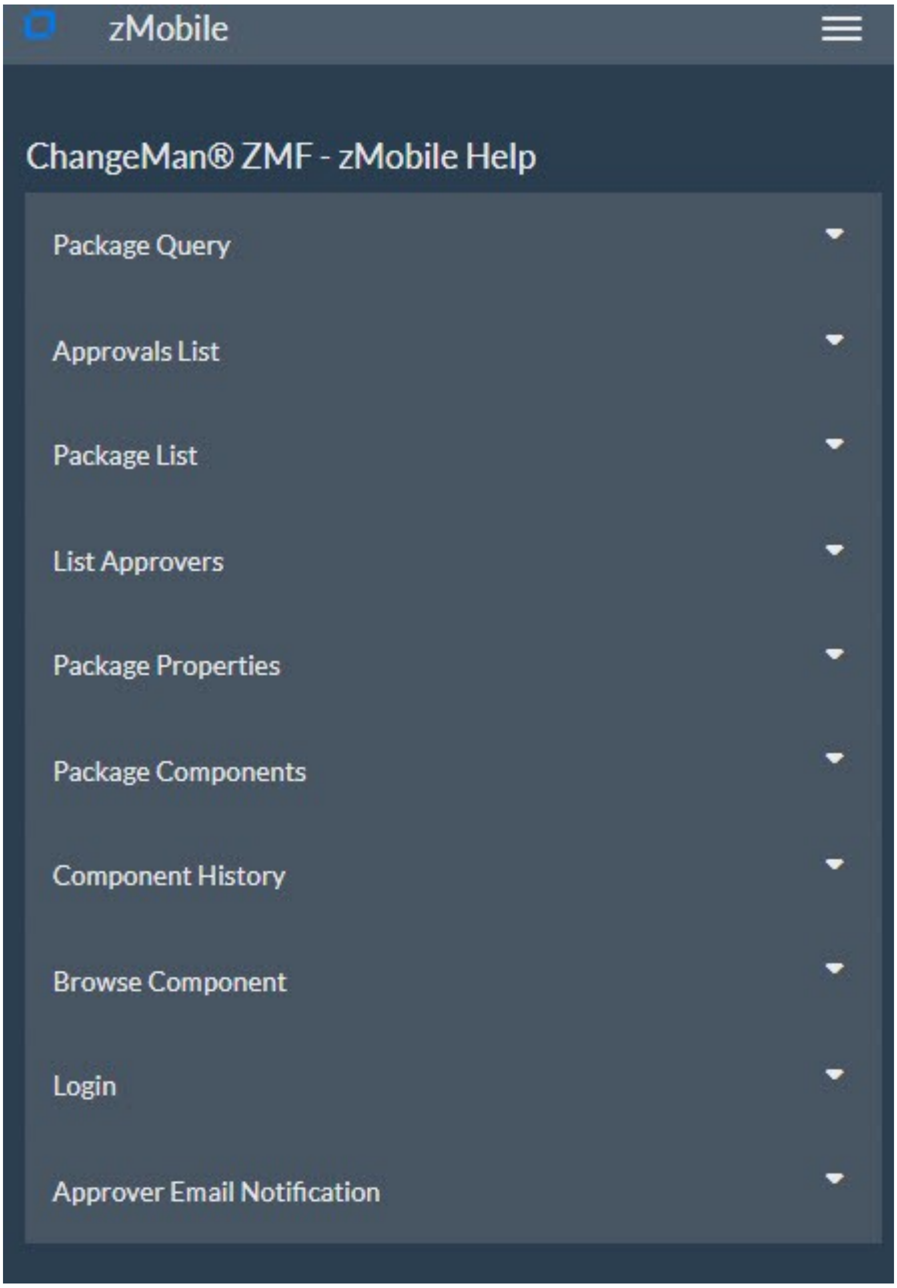
Version: 1.1.5

Build Date: 07/21/2018 - 12:15

Support: Micro Focus Support

https://www.microfocus.com/serena/support/

The last option on the Primary Menu, 'Help', gives you a comprehensive help for zMobile with 10 sections, each of which can be expanded via the toggle button on the right of each section.



zMobile

ChangeMan® ZMF - zMobile Help

17. Installing and Baselining a Package

This chapter describes the automated process that installs a change package into your production environment.

About Installation and Baseline Ripple

Package installation is a fully automated process executed by a series of batch jobs submitted by ChangeMan ZMF. The term "package installation" is used as a general term that includes two distinct processes:

1. Production Installation - Components are copied to application production libraries after the old production versions are copied to production backup libraries. If no production libraries are defined for an application, no package components are copied. When production installation is successfully completed, the package status is changed to INS (installed)
2. Baseline Ripple - All existing baseline versions of package components are copied over the next oldest version, and components in package libraries are copied to the level 0 baseline library. Baseline ripple is executed after package components are installed in production libraries at all sites specified in the package. When baseline ripple is successfully completed, the package status is changed to BAS (baselined)

The JCL for package installation jobs is created when a change package is frozen or when the package is approved, depending on how your administrator configures the application.

How the package installation process is initiated is determined by the Scheduler parameter chosen for your change package:

- **CMN** - The first package installation job is submitted by the ChangeMan ZMF internal scheduler when it finds that the package Install Date/Time has arrived.
- **Other** - The first package installation job is submitted by an external job scheduling system when it determines that the date, time, and dependencies of a job schedule are satisfied.

- **Manual** - The first package installation job is submitted by ChangeMan ZMF when the last approval for the package is entered, regardless of the Install Date/Time specified in the change package information.

After the package installation process is initiated, the installation and baseline ripple jobs communicate success or failure with ChangeMan ZMF instances where the package was created and with other ZMF instances where the package is scheduled for production installation. Each successful job triggers the submission of the next batch job in the installation and baseline ripple series.

Overview of File Tailoring for Installation JCL

File tailoring for installation JCL builds these jobs:

1. Builds applicable skeleton jobs, by site configuration, as indicated in the following tables:

   a. At an ALL site

```
| Package Type | Job Name (CMNxx)            |
|:---------------- |:--------------------------------- |
| PERMANENT        | 20`*`, 30, 50, 55                 |
| TEMPORARY        | 20T, 31T, 50T, 55T                |
| PERMANENT/Db2    | 20, 21, 32, 30, 37, 49, 50, 55, 57 |
| TEMPORARY/Db2    | 20T, 21, 49, 31T                  |
```

   b. At a DP site with IEBCOPY (need CMN65 for revert)

```
| Package Type | Job Name (CMNxx)  |
|:--------------- |:----------- |
| PERMANENT        | 10, 11, 15, 19, 20, 25, 29, 30, 50, 51, 55, 59             |
| TEMPORARY        | 10, 11, 15, 19, 20T, 25, 29, 31T, 35, 39                   |
| PERMANENT/Db2    | 10, 11, 15, 19, 21, 20, 25, 29, 30, 32, 49, 50, 51, 56, 55, 59 |
| TEMPORARY/Db2    | 10, 11, 15, 19, 21, 20T, 25, 29, 49, 31T, 35, 39           |
```

   c. At a DP site with OTHER (need CMN64-65 for Revert)

```
 | Package Type | Job Name (CMNxx) |
 |:------------------ |:---------------- |
 | PERMANENT        | 10, 11, 14, 15, 18, 19, 20, 24, 25, 28, 29, 30, 50, 51, 54, 55,
58, 59                   |
 | TEMPORARY        | 10, 11, 14, 15, 18, 19, 20T, 24, 25, 28, 29, 31T, 34, 35, 38,
39                         |
 | PERMANENT/Db2    | 10, 11, 14, 15, 18, 19, 21, 22, 20, 24, 25, 28, 29, 32, 30, 49,
50, 51, 54, 56, 55, 58, 59 |
 | TEMPORARY/Db2    | 10, 11, 14, 15, 18, 19, 21, 20T, 24, 25, 28, 29, 49, 31T, 34,
35,38, 39                  |
```

   The following jobs are created in all configurations, in the situations listed.

   - CMN17 is present if an external scheduler is used.

- CMN30 and CMN55 are present for all permanent packages. (baseline ripple and backout reverse ripple), respectively
- CMN37and CMN57 are present for all permanent packages. (baseline ripple and backout, respectively, with CMNDSPTM)

2. Submits, through CMNSCHED, the skeleton jobs at the appropriate time. You can defer splitting the skeleton jobs until the final approval of the package.
3. The CMN37 or CMN57 jobs are not run if the impact analysis update is successful in the CMN30 job (baseline ripple) or the CMN55 job (reverse baseline ripple). If the impact analysis update is unsuccessful, transactions are written to the CMNDELAY file. The next time that ZMF is started, the transactions in the CMNDELAY file are processed to submit the CMN37 job or the CMN57 job to update impact analysis data.

The scheduling record for a package is created at approve time. This means that updates to the install date of a package in FRZ status being implemented with the ChangeMan ZMF internal scheduler are captured. At approve time, the scheduling record is built with the latest install date originally entered or updated by the user. If your internal scheduler is not ChangeMan ZMF, you can change the scheduled installation date even after a package is frozen.

Distribution and Installation of the Change Package

After the change package is approved, it is either installed, or first distributed and then installed. The specific distribution and installation process depends on the following:

- The type of scheduler set up by the global administrator
- Whether or not the Installation job has been modified
- The site type (ALL, DP, and so on).

Distribution to Remote Sites

If the site is D or DP, and has remote sites (P), the following occurs:

- Specific staging libraries, the installation JCL, and a copy of the package master record pertaining to this change are distributed (copied) to all the remote sites specified in the package control information (it is specified in the creation/update package process).
- A record of this event (package distribution) is placed in the activity log.
- A distribution acknowledgment is sent back to the development center.
- Package status is changed from APR to DIS.
- The change packages are installed at those sites.

If the site is ALL (no remote sites exist), the change package is ready to install. See the following discussion.

If the site is DP or D, the installation job CMN20 job runs at the corresponding remote sites, according to the scheduler. Once the change packages are installed to the remote sites, the CMN30 job runs at the DP or at the D site.

If the site is ALL, ChangeMan ZMF executes installation jobs CMN20 and CMN50 directly. The status of the change package changes from APR to BAS.

Jobs to Distribute and Install the Change Package

In the following examples, here are some sample snippets of job logs you may see depending upon what is in your package. Here package ACTP000039 is installed into

production libraries at DP site SERT6 and P site SERT6P1. Install into production is specified on the BASELINE CONFIGURATION PART 1 OF 2 panel (CMNCBAS1).

```
CMNCBAS1       ACTP - Baseline Configuration Part 1 of 2 Row 1 to 21 of 21
Command ===> _____________________________________________Scroll ===> CSR


                                 Baseline
                     Install     storage
     Type   Levels   in prod     means
 ___ CPY      10        N        SD
 ___ CP2      10        N        SD
 ___ CTC      10        Y        SD
 ___ DBR      3         N        H
 ___ DOC      10        N        SD
 ___ HTH      3         N        H
 ___ JAR      3         N        H
 ___ JAV      3         N        H
 ___ JCF      3         N        H
 ___ JCL      10        Y        SD
 ___ LCT      10        N        SD
 ___ LOD      3         Y        P
 ___ LOS      3         N        P
 ___ LSH      3         N        H
 ___ LST      3         N        P
 ___ PRC      10        Y        SD
 ___ SRC      10        N        SD
 ___ SRS      10        N        SD
 ___ TST      10        N        SD
 ___ WAR      3         N        H
 ___ WCT      3         N        H
******************************* Bottom of data ********************************
```

Production libraries for both sites have been defined to ZMF and allocated by the user:

```
CMNRMTSL                 ACTP - Site Selection List                  Row 1 to 2 of 2
Command ===>                                                        Scroll ===> CSR

    Site Name
 __ SERT6
 __ SERT6P1
******************************* Bottom of data ********************************
```

SERT6 production libraries (SERT6P1 libraries are identical except they have a different HLQ.):

```
CMNCPRDL                ACTP - SERT6 Production Libraries               Row 1 to 4 of 4
Command ===>                                                           Scroll ===> CSR

     Type Production dataset name    +
          Temporary dataset name     +
          Backup dataset name        +
 ___ CTC CMNTP.S6.V810.PROD.CTC
         CMNTP.S6.V810.PROD.CTC.TEMP
         CMNTP.S6.V810.PROD.CTC.BKUP
 ___ JCL CMNTP.S6.V810.PROD.JCL
         CMNTP.S6.V810.PROD.JCL.TEMP
         CMNTP.S6.V810.PROD.JCL.BKUP
 ___ LOD CMNTP.S6.V810.PROD.LOD
         CMNTP.S6.V810.PROD.LOD.TEMP
         CMNTP.S6.V810.PROD.LOD.BKUP
 ___ PRC CMNTP.S6.V810.PROD.PRC
         CMNTP.S6.V810.PROD.PRC.TEMP
         CMNTP.S6.V810.PROD.PRC.BKUP
******************************* Bottom of data ********************************
```

Installation JCL is built in two libraries, one for SERT6 and one for SERT6P1:

```
ISRUDSM BROWSE        CMNTP.S6.ACTP.STG6.#000039.X.SERT6          Row 0000001 of 0000016
Command ===>                                                            Scroll ===> CSR
           Name   Prompt          Size    Created              Changed          ID
_________ ACTP1039                50     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP1139                56     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP1539                47     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP1939                46     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP2039                105    2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP2539                48     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP2939                46     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP3039                239    2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP3739                80     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP5039                84     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP5439                47     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP5539                277    2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP5739                65     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP5939                46     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP6439                58     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP6539                49     2015/01/28    2015/01/28 22:17:41     USER015
          **End**
```

SERT6P1 (dataset name is 1 char too long to display fully in browse):

```
ISRUDSM BROWSE        CMNTP.S6.ACTP.STG6.#000039.X.SERT6         Row 0000001 of 0000016
Command ===>                                                           Scroll ===> CSR
           Name   Prompt          Size    Created             Changed           ID
_________ ACTP1039                50     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP1139                56     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP1539                47     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP1939                46     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP2039                105    2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP2539                48     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP2939                46     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP3039                239    2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP3739                80     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP5039                84     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP5439                47     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP5539                277    2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP5739                65     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP5939                46     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP6439                76     2015/01/28    2015/01/28 22:17:41     USER015
_________ ACTP6539                49     2015/01/28    2015/01/28 22:17:41     USER015
          **End**
```

For this example, JOBS ACTP1039, ACTP1139, ACTP1539, ACTP2039, and ACTP2539 are executed for both sites. JOB ACTP3039 is executed at the DP site SERT6 in conjunction with a demotion cleanup job will be executed on the appropriate site by CMN20, CMN25 or CMN30 depending upon configuration.

JOB ACTP1039 copies the package and staging datasets from SERT6 to SERT6P1:

```
22.18.08  J0551065   ---- WEDNESDAY, 28 JAN 2015 ----
22.18.08  J0551065   IRR010I   USERID SERT       IS ASSIGNED TO THIS JOB.
22.18.22  J0551065   ICH70001I SERT      LAST ACCESS AT 22:18:21 ON WEDNESDAY, JANUARY 28, 2015
22.18.22  J0551065   $HASP373 ACTP1039 STARTED - INIT 3 - CLASS A - SYS C001
22.18.22  J0551065   -                                       --TIMINGS (MINS.)--            -----PAGING COUNTS----
22.18.22  J0551065   -STEPNAME PROCSTEP    RC  EXCP  CONN  TCB  SRB  CLOCK  SERV  WORKLOAD  PAGE  SWAP  VIO  SWAPS
22.18.22  J0551065   -PKG2PRS              00  70    76    .00  .00  .0     4939  BATCH     2     0     0    0
22.18.24  J0551065   -DEV2PRS              00  136   248   .00  .00  .0     9242  BATCH     2     0     0    0
22.18.35  J0551065   -CMN00                00  674   487   .00  .00  .1     33134 BATCH     0     0     0    0
22.18.35  J0551065   -CMN99                00  14    10    .00  .00  .0     651   BATCH     0     0     0    0
22.18.35  J0551065   -FAILURE           FLUSH   0     0    .00  .00  .0     0     BATCH     0     0     0    0
22.18.35  J0551065   -ACTP1039  ENDED.  NAME-             TOTAL TCB CPU TIME=     .00   TOTAL ELAPSED TIME= .2
22.18.35  J0551065   $HASP395  ACTP1039 ENDED
...
ChangeMan(R) ZMF       CMNBATCH - 8.1.0 2015/01/28 22:18:25
Attempting to initiate dialog with ChangeMan ZMF subtask
Session established with ChangeMan ZMF subtask
SYSIN: ACTP000039 10 NOD=SERT6P1
STATUS UPDATED TO DIS AT DEV         ACTP000039
DISTRIBUTION LOGGED                  ACTP000039
SYSIN: ACTP000039 05 NOD=CMNTP.S7.ACTP.STG7P1.#000039.X.SERT6P1
SYSIN: ACTP000039 05 SUB=ACTP1139
SUBMITTED JOB ACTP1139 FROM CMNTP.S7.ACTP.STG7P1.#000039.X.SERT6P1                                     ACTP000039
END OF DATA ON SYSIN - TERMINATING
Session terminated with ChangeMan ZMF started task
******************************** BOTTOM OF DATA *****************************************************************************
```

The CMN10 job sends the package to the remote site. The CMN11 job inserts the package into the P site, and the CMN15 job logs the distribution back at the DP site:

```
                       J E S 2  J O B  L O G  --  S Y S T E M  C 0 0 1  --  N O D E  M P 3 J E S 2

22.18.21  J0551066  ---- WEDNESDAY, 28 JAN 2015 ----
22.18.21  J0551066   IRR010I    USERID   SERT            IS ASSIGNED TO THIS JOB.
22.18.21  J0551066   ICH70001I   SERT        LAST ACCESS AT 22:18:09 ON WEDNESDAY, JANUARY 28, 2015
22.18.21  J0551066  $HASP373 ACTP1139 STARTED - INIT 2         - CLASS A        - SYS C001
22.18.35  J0551066  -                                           --TIMINGS (MINS.)--                  -----PAGING COUNTS----
22.18.35  J0551066  -STEPNAME PROCSTEP    RC   EXCP   CONN   TCB   SRB   CLOCK   SERV   WORKLOAD   PAGE   SWAP   VIO   SWAPS
22.18.35  J0551066  -CMN00                00   674    499    .00   .00    .2     32079  BATCH      0      0      0      0
22.18.36  J0551066  -CMN99                00    15     10    .00   .00    .0       653  BATCH      0      0      0      0
22.18.36  J0551066  -FAILURE           FLUSH   0        0    .00   .00    .0         0  BATCH      0      0      0      0
22.18.36  J0551066  -ACTP1139  ENDED. NAME-SERT6 SITE          TOTAL TCB CPU TIME=      .00 TOTAL ELAPSED TIME=     .2
22.18.36  J0551066  $HASP395 ACTP1139 ENDED
...
ChangeMan(R) ZMF           CMNBATCH - 8.1.0 2015/01/28 22:18:23
Attempting to initiate dialog with ChangeMan ZMF subtask
Session established with ChangeMan ZMF subtask
SYSIN: ACTP000039 11 PKG=CMNTP.S6.ACTP.STG6.#000039.PACKAGE
SYSIN: ACTP000039 11 NOD=SERT6
STATUS UPDATED TO DIS AT PROD            ACTP000039
DISTRIBUTION LOGGED                      ACTP000039
SUBMITTED JOB ACTP2039 FROM SERT6        ACTP000039
SYSIN: ACTP000039 05 NOD=CMNTP.S6.ACTP.STG6.#000039.X.SERT6
SYSIN: ACTP000039 05 SUB=ACTP1539
SUBMITTED JOB ACTP1539 FROM CMNTP.S6.ACTP.STG6.#000039.X.SERT6                                                    ACTP000039
END OF DATA ON SYSIN - TERMINATING
Session terminated with ChangeMan ZMF started task
******************************** BOTTOM OF DATA ******************************************************************************
```

ACTP1539 distribution acknowledgement:

```
2.18.49   J0551072 ---- WEDNESDAY, 28 JAN 2015 ----
22.18.49   J0551072   IRR010I USERID SERT IS ASSIGNED TO THIS JOB.
22.18.49   J0551072   ICH70001I SERT         LAST ACCESS AT 22:18:49 ON WEDNESDAY, JANUARY 28, 2015
22.18.49   J0551072   $HASP373 ACTP1539 STARTED   - INIT 4         - CLASS A         - SYS C001
22.18.51   J0551072   -                                       --TIMINGS (MINS.)--                     -----PAGING COUNTS----
22.18.51   J0551072   -STEPNAME PROCSTEP   RC   EXCP   CONN   TCB   SRB   CLOCK   SERV   WORKLOAD   PAGE   SWAP   VIO   SWAPS
22.18.51   J0551072   -CMN00               00   675    495    .00   .00    .0     18937  BATCH       1      0      0      0
22.18.51   J0551072   -CMN99               00    14     10    .00   .00    .0       617  BATCH       0      0      0      0
22.18.51   J0551072   -FAILURE          FLUSH     0      0    .00   .00    .0         0  BATCH       0      0      0      0
22.18.51   J0551072   -ACTP1539 ENDED.  NAME-                  TOTAL TCB CPU TIME=       .00     TOTAL ELAPSED TIME= .0
22.18.51   J0551072   $HASP395  ACTP1539  ENDED
...
ChangeMan(R) ZMF CMNBATCH - 8.1.0 2015/01/28 22:18:50
Attempting to initiate dialog with ChangeMan ZMF subtask
Session established with ChangeMan ZMF subtask
SYSIN: ACTP000039 15 SUP=NO
SYSIN: ACTP000039 15 NOD=SERT6P1
DIS ACKNOWLEDGED AT DEV                     ACTP000039
DIS ACKNOWLEDGMENT LOGGED                   ACTP000039
END OF DATA ON SYSIN - TERMINATING
Session terminated with ChangeMan ZMF started task
```

The CMN20 job, ACTP2039 preserves the existing production version of the package components in backup libraries before installing the contents of the staging libraries into the production libraries:

```
22.18.49  J0551070  ---- WEDNESDAY, 28 JAN 2015 ----
22.18.49  J0551070   IRR010I   USERID   SERT        IS ASSIGNED TO THIS JOB.
22.19.03  J0551070   ICH70001I  SERT       LAST ACCESS AT 22:18:50 ON WEDNESDAY, JANUARY 28, 2015
22.19.03  J0551070   $HASP373  ACTP2039   STARTED - INIT 3 - CLASS A            - SYS C001
22.19.04  J0551070   -                                                  --TIMINGS (MINS.)--             -----PAGING COUNTS----
22.19.04  J0551070   -STEPNAME PROCSTEP    RC   EXCP   CONN   TCB   SRB   CLOCK   SERV   WORKLOAD   PAGE   SWAP   VIO   SWAPS
22.19.04  J0551070   -CMN00                00   671    486    .00   .00    .0     19310  BATCH        0      0     0      0
22.19.04  J0551070   -LOD2BAK              04    98     85    .00   .00    .0      5268  BATCH        0      0     0      0
22.19.05  J0551070   -LOD2PRD              00   125    101    .00   .00    .0      6312  BATCH        0      0     0      0
22.19.16  J0551070   -CMN00                00   672    490    .00   .00    .1     33607  BATCH        0      0     0      0
22.19.16  J0551070   -CMN99                00    14     10    .00   .00    .0       675  BATCH        0      0     0      0
22.19.16  J0551070   -FAILURE           FLUSH     0      0    .00   .00    .0         0  BATCH        0      0     0      0
22.19.17  J0551070   -PRINT                00    38     37    .00   .00    .0      1680  BATCH        0      0     0      0
22.19.17  J0551070   -ACTP2039 ENDED.   NAME-SERT6P1 SITE          TOTAL TCB CPU TIME=    .00    TOTAL ELAPSED TIME=  .2
22.19.17  J0551070   $HASP395 ACTP2039 ENDED
...
ChangeMan(R) ZMF CMNBATCH - 8.1.0 2015/01/28 22:19:06
Attempting to initiate dialog with ChangeMan ZMF subtask
Session established with ChangeMan ZMF subtask
SYSIN: ACTP000039 20 SUP=NO
SYSIN: ACTP000039 20 NOD=SERT6P1
STATUS UPDATED TO INS AT PROD                     ACTP000039
INSTALL LOGGED IN PROD                            ACTP000039
STATUS UPDATED TO INS AT PROD                     ACTP000039
SYSIN: ACTP000039 05 NOD=CMNTP.S7.ACTP.STG7P1.#000039.X.SERT6P1
SYSIN: ACTP000039 05 SUB=ACTP2539
SUBMITTED JOB ACTP2539 FROM CMNTP.S7.ACTP.STG7P1.#000039.X.SERT6P1 ACTP000039
END OF DATA ON SYSIN - TERMINATING
Session terminated with ChangeMan ZMF started task
```

ACTP2539 performs promotion and acknowledges:

```
22.19.16  J0551073  ---- WEDNESDAY, 28 JAN 2015 ----
22.19.16  J0551073  IRR010I USERID SERT         IS ASSIGNED TO THIS JOB.
22.19.16  J0551073  ICH70001I SERT     LAST ACCESS AT 22:19:06 ON WEDNESDAY, JANUARY 28, 2015
22.19.16  J0551073  $HASP373 ACTP2539 STARTED - INIT 1     - CLASS A         - SYS C001
22.19.30  J0551073  -                                                  --TIMINGS (MINS.)--              -----PAGING COUNTS----
22.19.30  J0551073  -STEPNAME PROCSTEP      RC   EXCP   CONN   TCB   SRB   CLOCK   SERV   WORKLOAD  PAGE    SWAP   VIO   SWAPS
22.19.30  J0551073  -CMN00                  00    667    488   .00   .00    .2     34960  BATCH        0      0     0      0
22.19.31  J0551073  -CMN99                  00     14      9   .00   .00    .0       728  BATCH        0      0     0      0
22.19.31  J0551073  -FAILURE             FLUSH      0      0   .00   .00    .0         0  BATCH        0      0     0      0
22.19.31  J0551073  -ACTP2539 ENDED. NAME-                          TOTAL TCB CPU TIME=   .00     TOTAL ELAPSED TIME=   .2
22.19.31  J0551073  $HASP395 ACTP2539 ENDED
...
ChangeMan(R) ZMF CMNBATCH - 8.1.0 2015/01/28 22:19:17
Attempting to initiate dialog with ChangeMan ZMF subtask
Session established with ChangeMan ZMF subtask
SYSIN: ACTP000039 25 SUP=NO
SYSIN: ACTP000039 25 NOD=SERT6P1
INSTALLED AT DEV                           ACTP000039
INSTALL LOGGED IN DEV                      ACTP000039
SUBMITTED JOB ACTP3039 FROM SERT6          ACTP000039
END OF DATA ON SYSIN - TERMINATING
Session terminated with ChangeMan ZMF started task
```

ACTP3039 performs a BASELINE RIPPLE of package ACTP000039. This installs the package (staging datasets) into the baseline libraries, and also performs Impact Analysis maintenance:

```
22.19.30  J0551074   ---- WEDNESDAY, 28 JAN 2015 ----
22.19.30  J0551074   IRR010I USERID SERT        IS ASSIGNED TO THIS JOB.
22.19.31  J0551074   ICH70001I SERT     LAST ACCESS AT 22:19:17 ON WEDNESDAY, JANUARY 28, 2015
22.19.31  J0551074   $HASP373 ACTP3039 STARTED - INIT 2      - CLASS A       - SYS C001
22.19.32  J0551074   -                                                  --TIMINGS (MINS.)--            -----PAGING COUNTS----
22.19.32  J0551074   -STEPNAME PROCSTEP    RC   EXCP   CONN   TCB   SRB   CLOCK   SERV   WORKLOAD   PAGE   SWAP   VIO   SWAPS
22.19.32  J0551074   -UPDCPY               00    225    175   .00   .00      .0   19776  BATCH        0      0     0      0
22.19.32  J0551074   -SELCLOD              00     57     87   .00   .00      .0    4550  BATCH        0      0     0      0
22.19.33  J0551074   -BASN122              04    112     89   .00   .00      .0    6953  BATCH        0      0     0      0
22.19.34  J0551074   -BASN121              04    107     86   .00   .00      .0    6930  BATCH        0      0     0      0
22.19.34  J0551074   -LOD2BAS              00    145    119   .00   .00      .0    8402  BATCH        0      0     0      0
22.19.35  J0551074   -SELCLST              00     47     38   .00   .00      .0    4584  BATCH        0      0     0      0
22.19.35  J0551074   -BASN122              04    103     90   .00   .00      .0    6645  BATCH        0      0     0      0
22.19.36  J0551074   -BASN121              04    102     88   .00   .00      .0    6586  BATCH        0      0     0      0
22.19.36  J0551074   -LST2BAS              00     98     92   .00   .00      .0    6584  BATCH        0      0     0      0
22.19.37  J0551074   -UPDSRC               00    140    114   .00   .00      .0   11059  BATCH        0      0     0      0
22.19.37  J0551074   -BASCOND              00     14      9   .00   .00      .0     743  BATCH        0      0     0      0
22.19.40  J0551074   -DSPTM                00    804    568   .00   .00      .0    123K  BATCH        0      0     0      0
22.19.40  J0551074   -PRINT                00     33     24   .00   .00      .0    1931  BATCH        0      0     0      0
22.19.40  J0551074   -RIPPLIA           FLUSH      0      0   .00   .00      .0       0  BATCH        0      0     0      0
22.19.41  J0551074   -CMN00                00    680    493   .00   .00      .0   22734  BATCH        0      0     0      0
22.19.41  J0551074   -CMN99                00     14      9   .00   .00      .0     726  BATCH        0      0     0      0
22.19.41  J0551074   -FAILURE           FLUSH      0      0   .00   .00      .0       0  BATCH        0      0     0      0
22.19.42  J0551074   -ACTP3039 ENDED. NAME-                     TOTAL TCB CPU TIME=      .00     TOTAL ELAPSED TIME=  .1
22.19.42  J0551074   $HASP395 ACTP3039 ENDED
...
                     ChangeMan(R) ZMF (8.1.0 - 20141010) Impact Analysis Dataspace Index Integrity Report
                     A return code of zero indicates that the I/A dataspace is structurally correct.
2015012822194023 Data space at Level 1.0.5
2015012822194023 Usage___ _Starts_ __Next__ __Size__ __Free__
2015012822194023 BUN       00001000 00001528 00007000 00006AD8
2015012822194023 BUN  X    00008000 000080DC 00001000 00000F24
2015012822194023 DSN   X   00009000 000090DC 00001000 00000F24
2015012822194023 ATB   X   0000A000 0000A0DC 00001000 00000F24
2015012822194023 TAB   X   0000B000 0000B0DC 00002000 00001F24
2015012822194023 B FREE    0000D000 0000D000 00002000 00002000

2015012822194023 COMP T    0000F000 0000F980 00005000 00004680
2015012822194023 CMPX X    00014000 00014260 00001000 00000DA0

2015012822194023 RLN T     00015000 00016110 0000E000 0000CEF0
2015012822194023 RLN P X   00023000 0002316C 00002000 00001E94
2015012822194023 RLN B X   00025000 0002516C 00002000 00001E94
2015012822194023 RLN T F   00027000 00027000 00002000 00002000

2015012822194023 UPDB      00029000 00029000 00002000 00002000
2015012822194023 NMBLK     0002B000 0002BAD6 00002000 0000152A

2015012822194023 BUN index check starts
2015012822194023 BUN index check ends
2015012822194023 DsName index check starts
2015012822194023 DsName index check ends
2015012822194023 Appl / LibType index check start
2015012822194023 Appl / LibType index check ends
2015012822194023 LibType / Appl index check start
2015012822194023 LibType / Appl index check ends
2015012822194023 Component Name index check starts
2015012822194023     152 component table indexes examined
2015012822194023 Component Table Index name/BUN check ends
2015012822194023 Relation Table SupComp index chck starts
2015012822194024     91 relation supcomp indexes examined
2015012822194024 Relation Table SupComp index check ends
2015012822194024 Relation Table SubComp index chek starts
2015012822194024     91 relation subcomp indexes examined
2015012822194024 Relation Table SubComp index check ends
 All checks completed, return code is 00

ChangeMan(R) ZMF CMNDSPTM - 8.1.0 2015/01/28 22:19:38
Baseline ripple I/A dataspace maintenance for package ACTP000039

      Component                                                   Copybook/Subroutine
      Name                Type    Rln     ToWhat              Appl   Libtype   Identifier        Action
      ACTCOB01             SRC    CPY     ACTCPY01            ACTP     CPY   DCACBC250000004F       I
      ACTCOB01             SRC    CPY     ACTCPY02            ACTP     CPY   BCACBC250000004F       I
      ACTCOB01             SRC    CPY     ACTCPY03            ACTP     CPY   FCACBC250000004F       I
Session terminated with ChangeMan ZMF started task
I/A dataspace update processing complete; rc=0000
I/A dataspace index check has been requested (see DD CMNIADSX).
1ChangeMan(R) ZMF CMNDSPTM - 8.1.0 2015/01/28 22:19:38
 Baseline ripple - SCRATCH/RENAME - Potential Component Restage Report
 No potential restage requirements were uncovered.
 Session terminated with ChangeMan ZMF started task
 I/A dataspace update processing complete; rc=0000
ChangeMan(R) ZMF CMNBATCH - 8.1.0 2015/01/28 22:19:41
Attempting to initiate dialog with ChangeMan ZMF subtask
Session established with ChangeMan ZMF subtask
SYSIN: ACTP000039 30 SUP=NO
SYSIN: ACTP000039 30 NOD=SERT6
BASELINE RIPPLED AT DEV                 ACTP000039
BASELINE RIPPLE LOGGED                  ACTP000039
END OF DATA ON SYSIN - TERMINATING
Session terminated with ChangeMan ZMF started task
```

If necessary (i.e. if the package is promoted) then you will see a job to demote package ACTP000039:

```
02.33.11  J0551300    ---- THURSDAY, 29 JAN 2015 ----
02.33.11  J0551300   IRR010I   USERID   SERT         IS ASSIGNED TO THIS JOB.
02.33.12  J0551300   ICH70001I SERT       LAST ACCESS AT 02:32:53 ON THURSDAY, JANUARY 29, 2015
02.33.12  J0551300   $HASP373 CMNSTART STARTED - INIT 1      - CLASS A         - SYS C001
02.33.12  J0551300   -                                                  --TIMINGS (MINS.)--            -----PAGING COUNTS----
02.33.12  J0551300   -STEPNAME PROCSTEP    RC   EXCP   CONN   TCB   SRB   CLOCK   SERV   WORKLOAD   PAGE   SWAP   VIO   SWAPS
02.33.12  J0551300   -DEL1LOD              00     42     29   .00   .00      .0   2958   BATCH        0      0     0      0
02.33.14  J0551300   -SUCCESS              00    687    486   .00   .00      .0  26113   BATCH        0      0     0      0
02.33.14  J0551300   -CHKCOND              00     14      9   .00   .00      .0    726   BATCH        0      0     0      0
02.33.14  J0551300   -FAILURE           FLUSH      0      0   .00   .00      .0      0   BATCH        0      0     0      0
02.33.14  J0551300   -PRINT                00     66     38   .00   .00      .0   5391   BATCH        0      0     0      0
02.33.14  J0551300   -CLNLCL            FLUSH      0      0   .00   .00      .0      0   BATCH        0      0     0      0
02.33.14  J0551300   -CMNSTART ENDED.   NAME-SERT6 SITE              TOTAL TCB CPU TIME= .00     TOTAL ELAPSED TIME= .0
02.33.14  J0551300   $HASP395 CMNSTART ENDED
...
ChangeMan(R) ZMF CMNBATCH - 8.1.0 2015/01/29 02:33:13
Attempting to initiate dialog with ChangeMan ZMF subtask
Session established with ChangeMan ZMF subtask
SYSIN: ACTP000039 85 FUN=CLEANUP,NOD=SERT6
SYSIN: ACTP000039 85 LVL=10,LNM=S6P1UT,CID=USER015
SYSIN: ACTP000039 85 SUP=YES,SSI=6799935B
SYSIN: ACTP000039 85 TYP=LOD
SYSIN: ACTP000039 85 CMP=ACTCOB01
Component History has been updated.
Component Promotion History has been updated
Demotion logged ACTP000039
SYSIN: ACTP000039 85 FUN=END
Package Promotion history has been updated
Package Promotion Unlocked
Package ACTP000039         DEMOTE
Package General record has been updated.
END OF DATA ON SYSIN - TERMINATING
Session terminated with ChangeMan ZMF started task

<SIZE: RECS=26 BYTES=1008>
```

18. Backing Out a Change Package

Backout package removes package changes from production and baseline libraries.

About Backing Out a Package

Backout package consists of two processes:

1. Restoration of production libraries - Prior versions are copied to production libraries from backups taken when the package was installed.
2. Reverse baseline ripple - The prior version is copied to the level 0 baseline library, the next oldest is copied to the prior version, and so on until all prior versions have been restored from the next oldest version.

Restoration of production libraries is executed only if production libraries are defined to the application. If production libraries are defined to the application, reverse baseline ripple is executed only after all production libraries are restored at all sites where the package was installed.

When you back out a change package, you must enter free form text (up to 9 lines) in Backout Reasons to provide an audit trail for why the package was backed out. Backout Reasons can be viewed through the query package function.

You cannot selectively back out components in an installed change package. The components in a change package are audited and tested together, so removing part of a package would invalidate audit and test results.

Backout package is initiated and controlled from the ChangeMan ZMF instance where the package was created, even if the package was installed at other sites.

After a package is backed out, the package must be reverted before you can make changes to package components.

Rules for Backing Out Packages

The following rules and restrictions apply to package backout.

- A package must be in BAS status at an A instance for you to back out the package at that instance.
- A package must be in INS status at a P instance for you to back out the package at that instance.
- You can back out a package at a P instance regardless of the status of the package at other P instances or the status of the package at the DP instance where it was created.
- You must install and back out a package at all P instances where it was scheduled for install before you can back out a package that is installed at the DP site where it was created.
- You cannot back out a package if the package staging libraries have been aged and physically scratched by housekeeping. You cannot back out a package if the package records have been aged and physically deleted by housekeeping.
- You cannot back out a package at a P or DP instance if any of the package components in a production library have been overlaid by a subsequent install.

Backout and Temporary Packages

You can use package backout on an installed temporary package to remove package components that are causing a production problem. Backout will remove the members installed into the production override libraries that are concatenated ahead of your regular production libraries, and the package status will be set to BAK. You can revert your temporary package to fix a problem, or you can create a new unplanned temporary package for your changes if you need to use the unplanned approval list.

Do not use backout to remove temporary package components when they have successfully served their purpose in production override libraries. When temporary package components have successfully served their purpose, use the standard temporary package life cycle to remove the components from the override libraries and set the package status to TCC (temporary change complete).

If you cannot use a temporary package duration and the ChangeMan ZMF scheduler to automatically submit the CMN31 job to remove the package components from production override libraries, create your package with a MANUAL scheduler. The CMN31 job will be submitted with TYPRUN=HOLD when the package is installed, and you can release the job(s) manually when you want the temporary package components removed from production override libraries.

Accessing Backout Package Panels

Backout package is executed from the **Backout Change Package** panel (CMNBKOUT).

```
CMNBKOUT                 Backout Change Package
Command ===>___________________________________

Package . . . . ACTP000039
```

Use one of these methods to display the **Backout Change Package** panel.

- **Using the Menu Hierarchy:**

  On the **Primary Option Menu**, select **B Backout**.

- **Using Direct Panel Access:**

  On a ChangeMan ZMF panel, type **=B** in the **Command** or **Option** line and press **Enter**.

Using the Change Package List to Backout Packages

You can access backout panels for a change package directly from the **Change Package List**.

On the **Change Package List** panel, type **B1** in the line command for a package and press **Enter**.

Backing Out an Installed Change Package

The process of backing out an installed change package differs between an All environment and a DP/P environment.

Backing Out in an All Environment

For an A environment, follow these steps.

1. Access the **Backout Change Package** panel.
2. On the **Backout Change Package** panel, type the Package ID of the package you want to back out and press **Enter**. The **site - Backout Reasons** panel (CMNBKRSN) is displayed.

```
CMNBKRSN                       SERT6 - Backout Reasons
Command ===>__________________________________________________________

       Package: ACTP000039     Status: BAS     Install date: 20150225
Backout reasons:______________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
```

3. On the **site - Backout Reasons** panel, type up to nine lines of text explaining the reason why the package is being backed out. You must type at least one character on the panel. Press **Enter**.
4. When package components are backed out of production libraries and baseline libraries are reverse rippled, the package status is changed to BAK, as shown here on the **Change Package List** panel.

```
CMNLIST3            Change Package             List Row 1 to 1 of 1
Command ===>                                       Scroll ===> CSR

   Package    Sta Install  Lvl Type    Work request Dept Promote Aud Creator
   ACTP000039 BAK 20150225 SMP PLN/PRM 100001000106 IDD          USER015
******************************* Bottom of data ********************************
```

Note

You must back out and revert a package before you can change the components in a package. For revert instructions, see Reverting in an All Environment.

Backing Out at a Remote Site

Backing out a package in a P instance at a remote site is performed the same as backing out on an A instance.

Note

You can back out a package at all installed sites from the D or DP instance where the package was created. There is no requirement to logon to a P instance to back out a package there.

Backing Out in a DP/P Environment

For a DP/P instance, follow these steps.

1. Access the **Backout Change Package** panel.
2. On the **Backout Change Package** panel, type the Package ID of the package you want to back out, and press **Enter**.
3. If the package is installed at more than one site, the **Backout: Site Information** panel (CMNBKSTI) is displayed.

```
CMNBKSTI               Backout: Site Information                      Row 1 to 2 of 2
Command ===>                                                         Scroll ===> CSR

          Package: ACTP000039     Creator: USER015       Status: BAS

   Site   Install  from to   Primary/backup contacts  Phone numbers   STATUS
 _ SERT7  20150225 0001 2359 KIKA HALEMANU            808-555-1213      BAS
                             IAN THOMPSON             808-555-1215
 S SERT6  20150225 0001 2359 HUNG NGUYEN              808-555-1214      INS
                             WENWEI HAN               808-555-1212
******************************* Bottom of data ********************************
```

4. On the **Backout: Site Information** panel, select one or more sites from which you want to back the package out, and press **Enter**. The **site - Backout Reasons** panel (CMNBKRSN) is displayed for the first selected site.

```
CMNBKRSN                       SERT6 - Backout Reasons
Command ===>__________________________________________________________

       Package: ACTP000039     Status: BAS     Install date: 20150225
Backout reasons:______________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
______________________________________________________________________
```

5. On the *site* - **Backout Reasons** panel, type up to nine lines of text explaining the reason why the package is being backed out. You must type at least one character on the panel. Press **Enter**.
6. If the site from which you are backing out the package is a remote site, the **Submit Remote Backout Request** panel is displayed. The backout request job is submitted by the D or DP instance, and it connects to the P instance using TCP/IP to request the backout.

```
CMNBKJCD                  Submit Remote Backout Request
Command ===>

    Site: SERT6    Package: ACTP000039    Creator: USER015 Status: INS
Job Statement Information:
    //USER015A JOB (X170,374),'S7.V810',____________________________________
    // CLASS=A,MSGCLASS=X,NOTIFY=USER015____________________________________
    //*_____________________________________________________________________
    //*_____________________________________________________________________
```

7. Update the **Job Statement Information**, if necessary, and press **Enter** to submit the remote backout request job.

8. If you selected additional sites for backout, the **site-Backout Reasons** panel is displayed for each site. Type backout reasons for each site, and submit the remote backout request if required.
9. When backout is completed for a site, the **Site Status** on the **Backout: Site Information** panel is changed to BAK.

```
CMNBKSTI               Backout: Site Information                      Row 1 to 2 of 2
Command ===>                                                         Scroll ===> CSR

          Package: ACTP000039     Creator: USER015       Status: BAS

   Site   Install  from to   Primary/backup contacts  Phone numbers   STATUS
 _ SERT7  20150225 0001 2359 KIKA HALEMANU            808-555-1213      BAS
                             IAN THOMPSON             808-555-1215
 S SERT6  20150225 0001 2359 HUNG NGUYEN              808-555-1214      BAK
                             WENWEI HAN               808-555-1212
******************************* Bottom of data ********************************
```

10. When all sites are in BAK status (package components are backed out of production libraries at all sites, and baseline libraries are reverse rippled), the package status is changed to BAK.

```
CMNBKSTI               Backout: Site Information                      Row 1 to 2 of 2
Command ===>                                                         Scroll ===> CSR

          Package: ACTP000039     Creator: USER015       Status: BAS

   Site   Install  from to   Primary/backup contacts  Phone numbers   STATUS
 _ SERT7  20150225 0001 2359 KIKA HALEMANU            808-555-1213      BAK
                             IAN THOMPSON             808-555-1215
 S SERT6  20150225 0001 2359 HUNG NGUYEN              808-555-1214      BAK
                             WENWEI HAN               808-555-1212
******************************* Bottom of data ********************************
```

> **Note**
>
> You must back out and revert all sites before you can change a package component. For revert instructions, see Reverting in a DP/P Environment.

Backing Out Scratch and Rename Utility Requests

Components that have been scratched (deleted) or renamed with utility requests are restored when the package that contains the requests is backed out.

However, when utility requests are backed out, impact analysis data is restored only for renamed components. Although components deleted by a scratch utility request are restored to baseline libraries when the request is backed out, the impact analysis relationship data for the restored components is not restored.

Backout and Reinstall at P Instance

You can reinstall a package that has been backed out, but not reverted, at a P instance without going through the process of backing out, reverting, auditing, approving, and installing the change package at all sites. Use the following procedure:

1. Logon to the P instance, access the Monitor function (Option M), and select the Limbo function (Option 1).
2. On the **Monitor Packages in Limbo** panel, type the package ID of the backed out package, and press ENTER.
3. On the Limbo Package List, type S in the line command for the package.
4. On the Application - Site Selection List panel, type S in the line command for the site where the package is backed out.
5. On the Limbo Package List panel, see that the Request field says SUBMIT, and then press PF3.
6. On the list of package installation jobs, type S in the line command for the 20 job, and press ENTER. See that the Status column says `*`SELECT, and then press PF3.
7. On the Monitor Packages In Limbo panel, see the short message JOB SUBMITTED.
8. Go to the Change Package List and verify that the package status is INS.
9. Logon to the D or DP instance and use the Backout function to verify that the status of the package is INS at the P instance where the package was backed out and reinstalled.

> **Note**
>
> You cannot use the process described here for a P instance to back out package components from production libraries at an A or DP instance.

19. Memo Deleting and Undeleting Packages

This chapter describes the memo-delete and undelete function.

About Memo-Delete and Undelete

The Housekeeping function in global administration physically deletes change packages according to these criteria:

- A package has a status of BAS (baselined), INS (installed at all sites), or TCC (temporary change cycle completed), and a minimum number of package aging days (specified in application administration) have passed since the package was installed.
- A package has been memo deleted using the Delete Package function, and has a status of DEL.

  Because a memo deleted package is not physically deleted until housekeeping jobs are run, you can use the Undelete Package function to change the DEL status back to DEV.

  When housekeeping jobs physically delete a memo deleted change package, package records in the package master file are deleted, and package staging libraries are scratched.

Memo-Delete and Undelete Package Rules

These rules and restrictions apply to the memo delete and undelete package function:

- Only simple and participating packages can be memo deleted.
- Only packages in DEV status can be memo deleted.
- Packages with promoted components cannot be memo deleted.
- A package must be memo deleted with a status of DEL to be undeleted.
- If housekeeping has physically deleted a memo deleted package, the package cannot be undeleted.

Administration Settings for Memo-Delete Package

Your global and application administrators make settings in ChangeMan ZMF administration that control how the memo-delete and undelete package function works in your application. Ask your administrator if any of the following apply to your application.

- Prohibit memo delete if a package contains components or utility requests (scratch or rename requests). (Application Administration Parameters: Memo Delete Empty Packages Only)
- Are you allowed to delete Complex/Super packages that have participating packages? (ZMF can be set so that it does not allow del or memo-del to delete these packages if they have any attached packages.)

Note

If ZMF is set to allow del or memo-del to delete super packages, all of the participating packages must have one of the following statuses: DEV, DEL, REJ, or BAK.

Accessing Memo-Delete and Undelete Functions

Memo delete and undelete package functions are accessed from the **Delete Options** menu (CMNDELT0).

```
CMNDELT0                      Delete Options
Option ===> __________________________________________________
1   Delete       Memo delete a change package
2   Undelete     Restore memo deleted change packages
```

Use one of these methods to display the **Delete Options** menu:

- Using the Menu Hierarchy:

  On the **Primary Option Menu**, select **D Delete**.

- Using Direct Panel Access:

  On a ChangeMan ZMF panel, type **=D** in the **Command** or **Option** line and press **Enter**.

Using the Change Package List to Access Delete Functions

You can access the options listed on the **Delete Options** menu directly from the **Change Package List**.

On the **Change Package List** panel, type one of the following commands in the line command field for a change package and press **Enter**.

- **D1** Memo delete a change package.
- **D2** Undelete a memo-deleted change package.

Memo Deleting a Change Package

Follow these steps to memo delete a change package:

1. On the **Delete Options** menu, select **1 Delete** to display the **Memo Delete Change Package** panel (CMNDELT1).

```
CMNDELT1             Memo Delete Change Package
Command ===> _________________________________________________

Package . . . . . . . ACTP000028

Enter "/" to select option
    / Confirm request
```

2. Type the change package ID in the **Package** field.
3. Select the **Confirm Request** field to display the **Confirm Request** panel (CMNDELT2) before the package status is changed to DEL.
4. Press **Enter**. The **Confirm Delete** panel (CMNDELT2) is displayed.

```
CMNDELT2                        Confirm Delete
Command ===> ____________________________________________

Package:        ACTP000028
Status:         DEV
Install date:   20150228
```

5. If this is *not* the package you want to memo delete, press **PF3** to exit the panel. If this is the package you want to memo delete, press **Enter**. You are returned to the **Memo Delete Change Package** panel (CMNDELT1) where the delete is confirmed by a short and long ISPF message.

```
CMNDELT1        Memo Delete Change Package      PACKAGE MEMO DELETED
Command ===> ________________________________________
Package . . . . . . . ACTP000028

Enter "/" to select option
  / Confirm request
...
    +--------------------------------------------------------------+
    | CMN3400I - ACTP000028 change package has been memo deleted. |
    +--------------------------------------------------------------+
```

6. Press **PF3** repeatedly to return exit the delete package function.

Undelete a Memo-Deleted Change Package

Follow these steps to undelete a memo-deleted change package.

1. On the **Delete Options** menu, select option **2 Undelete** to display the **Restore Memo Deleted Package** panel (CMNDELT3).

```
CMNDELT3            Restore Memo Deleted Packages
Command ===> ____________________________________________

Package . . . ________________ (Blank for list)
```

2. If you know the package you want to undelete, type the change package ID in the **Package** field and press **Enter**. If you leave the **Package** field blank and press **Enter**, the **Memo Deleted Package List** panel (CMNDELT4) is displayed.

```
CMNDELT4            Memo Deleted Package List       Row 1 to 2 of 2
Command ===> _____________________________________ Scroll ===> CSR

   Package  Status
__ ACTP000027
__ ACTP000028
******************************* Bottom of data ********************************
```

3. Type **U** in the line command field for one or more packages and press **Enter**. The **Status** field is changed to *Undeleted.

```
CMNDELT4            Memo Deleted Package List           Row 1 to 2 of 2
Command ===> _________________________________________ Scroll ===> CSR

   Package    Status
__ ACTP000027 *Undeleted
__ ACTP000028 *Undeleted
******************************* Bottom of data ********************************
```

4. Press **PF3** twice to return the **Delete Options** panel.

20. Querying Packages and Components

This chapter describes various query functions in ChangeMan ZMF.

- About Query
- Accessing the Query Options Menu
- Querying Change Packages
- Querying Components
- Querying Approved Packages Not in the Scheduler

About Query

ChangeMan ZMF includes several query functions that show you information about packages and components. You can use the query functions to:

- Display information about active change packages and packages that have been baselined but not aged and deleted.
- Display a list of active and baselined packages that include a component.
- List other components that will be affected when you change a component.
- List and display information about packages that have been approved but were not successfully inserted into the ChangeMan ZMF installation scheduler.

Accessing the Query Options Menu

Select Option **Q Query** on the **Primary Option Menu** to access the **Query Options** (CMNQDMNU) menu.

```
CMNQDMNU            Query Options
Option ===> __________________________________________

P Package       Query package information
C Component     Query component information
I Impact        Impact analysis of subordinate components
B BofM          Component bill of materials
A Approve       Approve In Limbo packages
```

The following query functions are available:

- Package - Query Package Information. See Querying Change Packages.
- Component - Query Component Information. See Querying Components.
- Impact - Impact Analysis of Subordinate Components. This query is explained in the next chapter. See Impact Analysis of Subordinate Components.
- B of M - Component Bill of Materials. This query is explained in the next chapter. See Getting a Component Bill of Materials.
- Approve - Approve In Limbo Packages. See Querying Approved Packages Not in the Scheduler.

Querying Change Packages

When you query change packages, complete the following tasks:

1. Specify package selection criteria on the **Package Parameters** (CMNQRY00) panel.
2. Select a package from the filtered **Package List** (CMNQRY02) panel.
3. Use the **Package Information** (CMNQRY03) panel to request different types of query information for the package you selected.

Specifying Selection Criteria

Access the **Package Parameters** (CMNQRY00) panel, to specify selection criteria that will be used in ChangeMan ZMF to create a filtered change package list for you.

1. Select Option **P Package** on the **Query Options** menu to display the **Package Parameters** panel.

```
CMNQRY00                    Package Parameters
Command ===> _____________________________________________

Package list            (Blank, full name or mask separated by ;)
______________________________________________________________________
Enter "/" to select:
  Package status        / Dev  _ Frz   _ Apr   _ Rej   _ Dis   _ Ins
                        _ Bas  _ Bak   _ Del   _ Opn   _ Clo   _ Tcc
                        _ Aip  _ Xds   _ Xin   _ Xbk   _ Xrv
  Package level         / Simple _Super _ Complex _ Participating
  Package type          / Planned Permanent   _ Planned Temporary
                        _ Unplanned Permanent _ Unplanned Temporary

Work request . . . . . .  ____________________________
Department . . . . . . .  ___
Install date: From . . .  ________  To . . ________ (YYYYMMDD)
Creation date: From . .   ________  To . . ________ (YYYYMMDD)
Site . . . . . . . . . .  ________
Approver entity . . . .   ________
Creator's id list         (Blank, full name or mask separated by ;)
____________________________________________________________________
Enter "/" to select option
    _ Other parameters
```

2. Complete information in the selection criteria fields. This information limits the range of packages displayed in the **Package List**. Fill in any combination of these fields to limit your search. To limit the range of packages displayed by specific applications, leave the *package list* field blank (or specify a pattern).

   The following table describes the fields on this panel. All field entries are optional. If you leave any field blank, all change packages in the system that satisfy the specified selection criteria are presented.

Field	Description
Package list	Enter a list of package IDs or patterns to select, separated by semicolon (;).
Package status	Select status for the package(s) you want to query.
Package level	Select to include package level; only packages with selected level will be listed.
Package type	Select to include package type; only packages with selected type will be listed.
Work request	Enter a Work request to review change packages with that identifier (if entered at package creation time).
Department	Enter a Department to review change packages with that identifier (if entered at package creation time).

Field	Description
Installation date range: Specify a date range. Only change packages with an installation date during the time period specified will be listed. Both dates are inclusive:	
Install date: From	Enter the start date for a scheduled installation date selection period in YYYYMMDD format. Leaving the date blank is the same as typing 00000000.
(install date) To	Enter the end date for a scheduled installation date selection period in YYYYMMDD format. The date is inclusive. Leaving the date blank is the same as typing 99999999.
Creation date range: Specify a date range. Only change packages with a creation date during this time period specified will be listed. Both dates are inclusive:	
Create date: From	Enter the start date for a package create date selection period in YYYYMMDD format. Leaving the date blank is the same as typing 00000000.
(creation date) To	Enter the end date for a package create date selection period in YYYYMMDD format. The date is inclusive. Leaving the date blank is the same as typing 99999999.
Site	Enter an installation site name to review change packages.
Approver entity	Enter an approver entity to review change packages with that identifier. **Note:** Entities in the planned approval list are added to unplanned packages after they are baselined.
Creator id list	Enter a tso-id or a pattern. Wild-card characters are * for masking multiple characters and? for a place-holder for one character.
Enter more parameters	Select to display the EXTENDED SEARCH CRITERIA (CMNQRY01) panel.

Using the Extended Search Criteria Panel

The **Extended Search Criteria** (CMNQRY01) panel is displayed if you select **Enter More Parameters** field on the **Query Package Parameters** panel. It lets you further limit the display of change packages by the kinds of components the change package contains.

```
CMNQRY01                Extended Search Criteria
Command ===> ___________________________________________

Enter "/" to select:
    Component records       / Nonsource     __Source    __Load
                              Rename        __Scratch   __Form

    Scheduling system       / CMN   __Manual    __Other

Component type . . . . . _______________
Promoted Component . . . ____       (1-YES, 2-No)

Component name list     (Blank, Full name or Mask separated by ;)
___________________________________________________________________

Component userid list   (Blank, Full name or Mask separated by ;)

___________________________________________________________________
Release id list         (Blank, Full name or Mask separated by ;)

___________________________________________________________________
```

The following table describes the fields on this panel. All field entries are optional. If you leave any field blank, all change packages in the system that satisfy the specified selection criteria are presented.

Field	Description
Component records	Select package component category.
Scheduling system	Select installation scheduler.
Component type	Select component type or pattern.
Promoted Component	Select whether to include packages that have promoted components.
Component name list	Enter a list of component names or patterns to select, separated by semicolons (;). The maximum length of each component name or mask is 16 characters.
Component user id list	Enter TSO id or pattern separated by semicolons (;).

Field	Description
Release id list	Enter release id or patterns to select, separated by semicolons (;).

Field values within a specific field, such as **Component Records**, are combined with logical OR operators; the results of using patterns in such fields as **Release ID List** are also combined with logical OR operators.

Press Enter. The **Package List** (CMNQRY02) panel is displayed.

Browsing the Query Package List

When ChangeMan ZMF finishes processing your filter criteria, it displays the **Package List** (CMNQRY02) panel.

```
CMNQRY02           Package List                Row 1 to 20 of 29
Command ===> __________________________________ Scroll ===> CSR

   Package    Sta Install  Lvl Type    Work request Dept Promote Aud Creator
__ ACTP000001 BAS 20141121 SMP PLN/PRM              IDD  00          USER016
__ ACTP000002 BAS 20150104 SMP PLN/PRM 123          BOO  00       00 USER016
__ ACTP000003 BAS 20150105 SMP PLN/PRM 123          BOOK 00       00 USER016
__ ACTP000004 BAS 20141121 SMP PLN/PRM 123          BOOK 00          USER016
__ ACTP000005 BAS 20150105 SMP PLN/PRM 123          BOOK 00       00 USER016
__ ACTP000006 BAS 20141121 SMP PLN/PRM 123          BOOK 00          USER016
__ ACTP000007 BAS 20141121 SMP PLN/PRM 100001000106 TEST 00          USER016
__ ACTP000008 BAS 20141121 SMP PLN/PRM 100001000106 TEST 00       20 USER017
__ ACTP000013 BAS 20150105 SMP PLN/PRM              IDD  00       04 USER017
__ ACTP000016 BAS 20150104 SMP PLN/PRM 100001000106 TEST 00       00 USER016
__ ACTP000018 BAS 20150105 SMP PLN/PRM              IDD  00       00 USER016
__ ACTP000020 BAS 20150104 SMP PLN/PRM 100001000106 IDD  00       04 USER016
__ ACTP000021 BAS 20150106 SMP PLN/PRM              IDD  00       00 USER016
__ ACTP000023 BAS 20150105 SMP PLN/PRM 100001000106 IDD 00        04 USER016
__ ACTP000024 BAS 20141218 SMP PLN/PRM 100001000106 IDD 00        00 USER016
__ ACTP000025 BAS 20150105 PAR PLN/PRM 100001000106 IDD 00        00 USER016
__ ACTP000026 CLO          CMP PLN/PRM 100001000106 IDD 00        08 USER016
__ ACTP000027 FRZ 20150228 SMP PLN/PRM 100001000106 IDD 00        00 USER016
__ ACTP000028 DIS 20150228 SMP PLN/PRM 100001000106 IDD 00        00 USER016
__ ACTP000029 OPN          CMP PLN/PRM 100001000106 IDD 00        00 USER016
```

Note

If only one package satisfies the filtering criteria that you have specified,ChangeMan ZMF displays the Package Information panel instead of the Query Package List. See Displaying Detailed Package Information.

When the **Package List** panel is displayed, the panel shows a shortened list of change package information. You can:

- Enter the S line command in the line command field of selected package IDs to display detailed information about those packages. See Displaying Detailed Package Information.
- Switch between the short and long viewing modes in the **Package List**:
  - Enter **LONG** at the command line if you want to display the package description in addition to the other information that is displayed on the short list.
  - Enter **SHORT** if you want to condense the information in the long list. The package description is not displayed on the sort list.
- Sort on any column heading (SHORT viewing mode only) by specifying the following command on the command line:

  SORT *column_heading*

  where *column_heading* is the column heading. For example, specify SORT INSTALL to display packages in ascending sequence by install date.
- If you have sorted the entries in a column, you can use the **LOCATE** command to locate an entry in that column. For example, if you have previously sorted on the **INSTALL** column, you can specify **LOCATE** 20061215 to locate the first entry with an install date of December 15, 2006.

Displaying Detailed Package Information

If you enter **S** to the left of one or more **Package IDs** on the preceding **Query Package List**, the **Package Information** (CMNQRY03) panel is displayed.

```
CMNQRY03                            Package Information
Option ===> ____________________________________________________________ More: +

           Package: DEMO000001     Status: DEV         Install date: 20220101
           Creator: USER001    Audit RC: 00

1 General           General
2 Non-Source        Non-Source
3 Source            Source
4 Source to Load    Source to Load Relationship
5 Work List         Component Userid Work List
6 Utilities         Renames and Scratches
7 Approvals         Approval List
8 Install Dates     Site/Install Date Information
9 Site Activity     Site Activities Date and Time
10 Online Forms     Online Forms
11 Participating    Participating Packages
12 Status           Status Start Date and Time
13 Revert Reasons   Revert Reasons
14 Backout Reason   Backout Reasons
15 Promo History    Promotion History
16 Promo Libs       Promotion Libraries
17 Dev Stg Libs     Development Staging libraries
18 Prod Stg Libs    Production Staging libraries
```

```
CMNQRY03                Package Information
Option ===> _________________________________________________More: -

2 Non-Source        Non-Source
3 Source            Source
4 Source to Load    Source to Load Relationship
5 Work List         Component Userid Work List
6 Utilities         Renames and Scratches
7 Approvals         Approval List
8 Install Dates     Site/Install Date Information
9 Site Activity     Site Activities Date and Time
10 Online Forms     Online Forms
11 Participating    Participating Packages
12 Status           Status Start Date and Time
13 Revert Reasons   Revert Reasons
14 Backout Reason   Backout Reasons
15 Promo History    Promotion History
16 Promo Libs       Promotion Libraries
17 Dev Stg Libs     Development Staging libraries
18 Prod Stg Libs    Production Staging libraries
19 Production       Production Libraries
20 Baseline         Baseline Libraries
21 IMS              IMS Information
22 Reject Reasons   Reject Reasons
```

This table describes the fields on the Package Information panel (CMNQRY03).

Select an option to display the desired information.

Field	Description
Package	The current package id.
Status	The current package status.
Install date	The scheduled installation date (yyyymmdd format) of the package. Note: If the package has been installed, the ACTUAL installation date can be viewed within the Site Activities Date/Time or Status Start Date/Time categories of the Query Package Option. Package installation date will vary depending on the scheduler used.

Options: Selecting any of the following options displays information specific to your package.

General	Display general package information, such as description, install date and time, requestor name, phone number, and department.
Non-Source	Display non-source components in the package: copybook members, cataloged JCL procedures, linkage editor control statements, control cards, and documentation.
Source	Display source component staging information.
Source to Load Relationship	Display components and their related components.
Component Userid Work List	Display a list of userids that have worked on a component.
	Note:** This option is available only if your administrator has enabled the ChangeMan ZMF Component Work Record facility.
Renames and Scratches	Display package utility requests.
Approval List	Display package approval list.
Site/Install Date Information	Display site information.
Site Activities Date and Time	Display starting date and time stamp of activities that were performed at the site.
Online Forms	Display online forms information.
	Note:** This option is available only if you license the ChangeMan ZMF Online Forms option.
Participating Package(s**)	Display individual application packages (participating packages) that make up this complex/super package.
Status Start Date and Time	Display all possible statuses for the package, as well as the date and time that the package was assigned the status.
Revert Reasons	Display the revert history, all of the reasons the package was reverted.
Backout Reasons	Display the backout history, all of the reasons the package was backed out.

Promotion History	Display promotion/demotion history of the package, listed in reverse chronological order.
Promotion Libraries	Display the level and data set names of the promotion libraries for the application.
Development Staging Libraries	Display the data set name of the development environment staging libraries for the package.
Production Staging Libraries	Display the data set name of the production environment staging libraries for the package.
Production Libraries	Display the data set name of the production libraries affected by the package.
Baseline Libraries	Display the data set name of the baseline libraries affected by the package.
IMS Information	Display IMS Option information. **Note:** This category is available only if you license the ChangeMan ZMF IMS Option.
Reject Reasons	Display the reasons the package was rejected.

> **Note**
>
> Options may not be available if they are not licensed, are disabled or there is no information to display for the package.

See Package Category Panels for an example of each Package Category panel.

Querying Components

To search and display component history information, take the following steps:

1. From the **Primary Option Menu**, select Option **Q Query**. The **Query Options** (CMNQDMNU) panel is displayed.
2. From the **Query Options** panel, select Option **C Component**. The **Query Component Parameters** (CMNQCMP1) panel is displayed.

```
CMNQCMP1        Query Component Parameters
Command ===> __________________________________________________

Specify selection criteria:
Component name . . . . . ___________________________________________ +
                              (Full component name or pattern)
Component type . . . . . . SRC  (Full library type or pattern)
Application . . . . . . .  _____
Package . . . . . . . . .  ____________

Enter "/" to select:
    Package status . . . .  _ Dev _ Frz _ Apr _ Rej _ Dis _ Ins
                            _ Bas _ Bak _ Del _ Opn _ Clo _ Tcc _ D/A
                            _ Aip _ Xds _ Xin _ Xbk _ Xrv
    Package type . . . . .  _ Planned Permanent   _ Planned Temporary
                            _ Unplanned Permanent _ Unplanned Temporary

Procedure name . . . . . . __________________
User . . . . . . . . . . . __________________
Checkout/staging
    From date . . . . . . .    __________________ (YYYYMMDD)
    To date . . . . . . . .    __________________ (YYYYMMDD)

Enter "/" to select option:
    _ Include deleted components
    / Mixed case
```

The QUERY function allows you to obtain a list of components and review the component information. You can list all of the components in the system, or you can specify "search" criteria that allows you to list only those components that meet the search criteria.

Fill in any of the following fields to specify which components you wish to review. If you want to see all the components in the system, enter asterisks in component name and library type fields, and leave blanks in the rest of the selection criteria fields.

This table describes the fields on the **Query Component Parameters** panel (CMNQCMP1).

Field	Description
Component name	Enter a full component name to select, or type a pattern or a wildcard * to display a selection list. Blank is not valid.
Component type	Enter a full library type to select, or type a pattern or a wildcard * to display a selection list. Blank is not valid.
Application	Enter a full 3- or 4-character application mnemonic to select, or leave blank for all applications.
Package	Enter a full 10-character package ID to select, or leave blank for all packages.
Package Status	Enter / to select the package status.
Package Type	Enter / to select the package type.
Procedure name	Enter a full build procedure name to select, or leave blank for all build procedures. Build procedure names are associated with like-source components and all build processing output types.

Field	Description
User	Enter a full TSO ID to select, or leave blank for all IDs. The ID in component history is the last person to check out, stage, recompile, or relink the component in the package, or the last person to submit a SCRATCH/RENAME utility request for the component in the package.
Checkout/staging From date To date	Enter inclusive selection FROM and/or TO dates YYYYMMDD, or leave blank for all dates. The date being selected is the date of the last checkout, stage, relink, recompile, or SCRATCH/RENAME utility request for the component in the package.
Include baseline/ archive packages	Select to view all historical information including baselined and archived packages. NOTE: Historical information for components that have been deleted within a package are not reported for either display mode.
Include deleted components	If this option is selected, entries on panel CMNCMPH1 for components deleted from a package display with "*DELETED" in the Setssi column.
Mixed case	Select to process **COMPONENT NAME** input exactly as you type it, upper and lower case. If not selected, by default, all fields are folded to upper case.

3. Complete the selection criteria and press Enter. If you use a wildcard or pattern in the **COMPONENT NAME** field or the **COMPONENT TYPE** field, the **Component List** (CMNQCMP2) panel is displayed.

```
CMNQCMP2        Component List          Row 22 to 24 of 24
Command ===> ______________________________ Scroll ===> CSR

    Lib Name                                                  +
__ SRS ACPSRS1B
__ SRS ACPSRS1C
__ SRC ACPSRC91
******************************* Bottom of data ********************************
```

4. To display the package history for one or more components, enter **S** in the line command for each component that you want to query, and press **Enter**. The **Component History** (CMNCMPH1) panel is displayed, showing component history for the first component that you selected.

```
CMNCMPH1        Component history              Row 1 to 1 of 1
Command ===>___________________________________ Scroll ===> CSR

Component: ACPSRC91.SRC


    Package    Sta P Promotion vv.mm Last action Size Setssi  User
__ ACTP000032 DEV R                2015/01/12 21:29 *RENAME USER016
******************************* Bottom of data ********************************
```

Notice that some panel fields depend on the library type. This panel displays component history for library type LOD for the same SRC component above.

```
CMNCMPH1        Component history          Row 1 to 5 of 5
Command ===> ____________________________  Scroll ===> CSR

Component: ACPSRC91.LOD

    Package    Sta P Promotion vv.mm Last action      Size  Setssi   User
__ ACTP000033 DEV             01.00 2015/01/09 21:04 00012 678039CE USER016
__ ACTP000028 DIS             01.01 2015/01/20 21:47 00012 678EC447 USER016
__ ACTP000020 BAS             01.00 2015/01/04 23:44 00012 6779C776 USER016
__ ACTP000013 BAS             01.00 2015/01/05 15:01 00012 677A9E06 USER016
__ ACTP000001 BAS             01.01 2014/11/19 03:10       673C0181 USER016
******************************* Bottom of data ********************************
```

This table describes the fields on the **Component History** panel (CMNCMPH1).

Field	Description
Package	All other packages that have referenced this component. 'NO PACKAGE' indicates that the component was listed by Impact Analysis of the Baseline libraries.
Sta	The current status of the change package:
	DEV: Development
	FRZ: Frozen, awaiting approval
	APR: Approved by all required approvers
	REJ: Rejected by an approver
	DIS: Distributed to site(s)
	D/A: Deleted or archived during housekeeping or maintenance
	INS: Installed at production site(s)
	TCC: Temporary change cycled (temporary package removed)
	BAS: Baselined BAK Backed out
	BAK: Backed out
	AIP: Approval In Process
	XDS: Package distribution has failed
	XIN: Package installation has failed
	XRV: Package revert has failed
	XBK: Package backout has failed
P	Processing type for the specified component in the package:
	O: Checked out
	I: Checked in
	Built: (staged) Recompiled component (like-source library types only)

Field	Description
	L: Relinked component (like-load library types only)
	S: Scratch request created
	R: Rename request created
Promotion	The common name for the promotion library, and the level reached, if the package has been promoted. 'STAGING' denotes the package has been subsequently demoted. After full demote, the level is 00. If a library type is not defined to promotion (for example, a like-source type), no data is displayed.
vv.mm	Version number (01 to 99 and then back to 00) and modification number (01 to 99) represent the same "VV.MM" headers of the Staging Libraries when browsing the directory entries - PDF Browse of a PDS.
Last action	Date and time the component was last acted upon (checked out, staged, frozen, etc..)
Size	Lines of code for most component types. Otherwise LOD module size.
Setssi	SETSSI for the component in the package; the 4-byte binary stamp placed on the module by ChangeMan (or taken from a prior placement).
User	Identification of the last person to check out or stage the component.

5. Line commands:

   The **Q** (query) command will display the Package Information in the same manner as if the user had entered the Q.P. (query package) function.

   The **P**(Process) command will place the user in the staging panel allowing them to enter staging commands against the selected component. This is the same as if the user had entered the staging option 3 and the component name and type will be filled in for you.

   The **C**(Checkout) command will place the user at the checkout options panel. From there they can select the checkout option they wish to perform. The package name is passed as well as the component name and type selected. The package name is available to be changed just in case they wish to checkout to another package.

   Regardless of the checkout option specified the panels following the option selection will automatically fill in the component name and type. If this is a checkout from package the source package will be the selected package of the component from the history list.

   The **S**(select) command will display component build data for this component taken from the component history information. To display detailed information about the component in one or more packages, enter **S** in the line command for each package you want displayed, and press **Enter**. In this example, the **Compile and Binder Options** (CMNCMPH2) panel is displayed.

```
CMNCMPH2            Compile and Binder Options
Command ===> ___________________________________________

    Package: ACTP000033     Status: DEV     Install Date: 20150118

Stager's userid . . . . . USER016
Component name . . . . .  ACPSRC91 +
Component type . . . . .  LOD
Language . . . . . . . .  COBOL2
Compile proc . . . . . .  CMNCOB2
Compile parms . . . . . .
Binder parms . . . . . .
Db2 processing . . . . .  NO

Enter "/" to select option:
__Additional user options
```

Querying Approved Packages Not in the Scheduler

If file tailoring for package install JCL is initiated when the final approval is entered, and the file tailoring process fails, the package status is changed to APR but the package will not distribute or install as expected. The package cannot be seen in the Monitor facility, but you can list and query the package in the Query Approve facility.

> **Note**
>
> The Query Approve facility will not list or display packages that have a package scheduler of OTHER. These packages are not inserted into the ChangeMan ZMF scheduler when the package is approved.

Follow these steps to list and query packages in APR status that are not in the ChangeMan ZMF scheduler:

1. On the **Query Options** menu (CMNQDMNU), select **A Approve**.
2. If there are two or more packages in APR status that are not in the scheduler, the **Query Package List** panel (CMNQRY02) is displayed.

```
CMNQRY02 --------------------- QUERY PACKAGE LIST ------------ Row 1 to 2 of 2
COMMAND ____________________________________________ ===> SCROLL ===> PAGE
PACKAGE ID STA INSTALL  LVL  TYPE    WORK REQUEST DEPT PROMOTE AUD CREATOR
__ ACTP000050 APR 20100430 SMPL PLN/PRM CR30009 3033 00 STAGING      USER239
__ ACTP000051 APR 20100430 SMPL PLN/PRM CR30009 3033 00 STAGING      USER239
****************************** Bottom of data ********************************
```

   Enter line command **S** to select the package you want examine, and the **Package Information** (CMNQRY03) panel is displayed.

3. If there is only one package in APR status that is not in the scheduler, the **Package Information** panel (CMNQRY03) is displayed for that package.

For instructions for using the **Query Package List** panel and the **Package Information** panel to browse package information, see Browsing the Query Package List and Displaying Detailed Package Information.

21. Querying Impact Analysis

This chapter describes how to query impact analysis data to discover relationships between components in ChangeMan ZMF.

About Impact Analysis

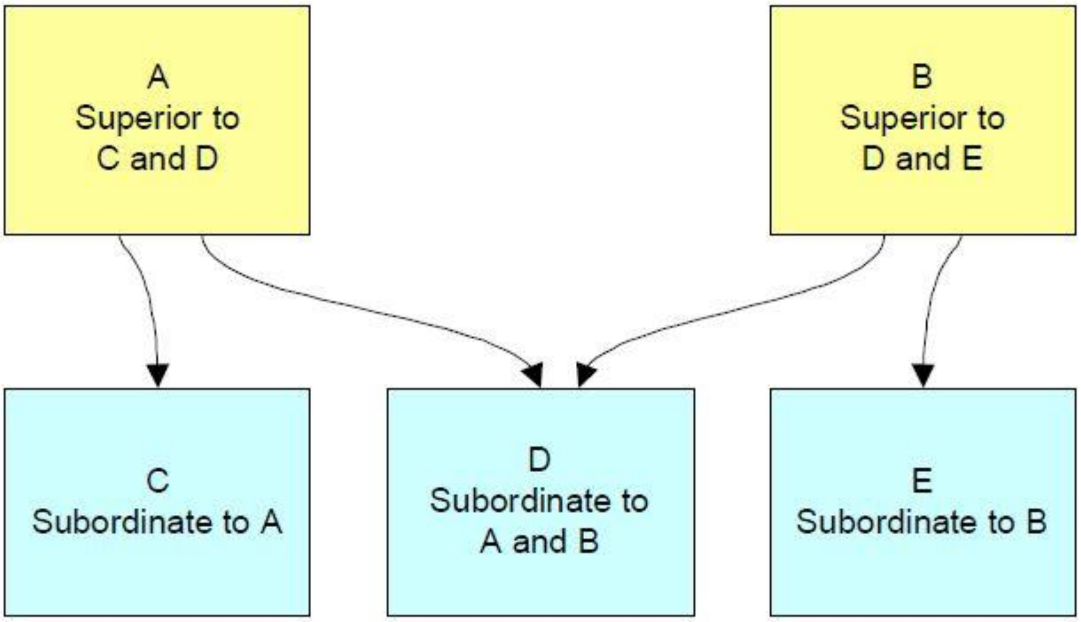
Impact analysis is a ChangeMan ZMF facility that shows relationships and dependencies among baseline library components. By querying impact analysis data for selected components you can do the following:

- Assess the effort involved in completing an intended change. That is, you can find out what and how many other components will be affected if you change a selected component.
- Avoid product failures that occur when you install a change without considering related components.

Relationships in Impact Analysis Data

Impact analysis lets you evaluate the effect of changing a particular component. A *superior component* is defined as one that references one or more subordinate components. Likewise, a *subordinate component* is defined as a component that is referenced by one or more superior components.

The following diagram shows the relationship between superior and subordinate components in impact analysis.



Potentially, multiple superior components can reference the same subordinate component, and multiple subordinate components can be referenced by a superior component.

Impact analysis enables you to examine the following types of relationships and dependencies among baseline library components:

Relationship	Superior Component	Subordinate Component
Source to copybook	Like-source component	Like-copy component
Composite load to linked load	Composite like-load	Statically linked like-NCAL or like-load subprogram
Execution JCL to cataloged procedure	Execution JCL	Cataloged procedure
JCL/Procedure to program name	Execution JCL or cataloged procedure	Program name in PGM=
JCL/Procedure to data set name	Execution JCL or cataloged procedure	Data set name in DSN=

Note

The relationship between a subprograms that is dynamically called by a main program is not included in impact analysis.

Consider carefully the subordinate-superior position as well as the type of relationship that exists among the target components when you formulate impact analysis queries. The way that you formulate a query affects the results that are returned.

Questions Answered by Impact Analysis

There are two types of impact analysis data queries. Each query answers a different question about the effect of a proposed change to a component that is managed by ChangeMan ZMF.

- If I change a component, what other components contain this component? To answer this question, see Impact Analysis of Subordinate Components.
- If I change a component, what other components are contained in this component? To answer this question, see Getting a Component Bill of Materials.

Impact Analysis of Subordinate Components

The Impact Analysis of Subordinate Components takes a bottom-up view of component relationships. You ask a question about a subordinate component, and it tells you about superior components.

In other words, it tells you what components contain a component that you want to change.

Accessing the Impact Analysis of Subordinate Components Panel

Use one of these methods to display the **Impact Analysis of Subordinate Components** panel (CMNIMPSU):

- Using the Menu Hierarchy:
    1. On the **Primary Option Menu**, select **Q Query**.
    2. On the **Query Options** menu (CMNQDMNU), select **I Impact**.
- Using Direct Panel Access:

  On a ChangeMan ZMF panel, type **=Q.I** in the **Command** or **Option** line and press **Enter**.

Specifying Search Criteria And Results Filters

You specify two types of criteria on the **Impact Analysis of Subordinate Components** panel (CMNIMPSU):

- Search criteria
- Results filter criteria

```
CMNIMPSU        Impact Analysis of Subordinate Components
Command ===> ______________________________________________

Specify search criteria:

Subordinate component name . . . ________________________________ +
Library type . . . . . . . . . . _____
Application . . . . . . . . . .  _______

Type of relationship . . . . . ._________________________


Specify results filter criteria:

Superior component name . . . .  ________________________________ +
Library type . . . . . . . . . . _____
Application . . . . . . . . . .  ________

Short or Long form . . . . . . . L

Enter "/" to select:
__Mixed case
```

You specify component search criteria on the top half of the panel.

Impact analysis returns a list of all superior components in all library types in all applications that include the target or wildcarded **Subordinate Component Name** in the specified type of relationship.

You specify filter criteria on the lower half of the **Impact Analysis of Subordinate Components** panel to limit the results that are displayed.

> **Important**
>
> This panel asks the question, "What components contain the component that I specify in the relationship that I specify?" The panel lets you limit the scope of the answer.

This table describes the fields on the **Impact Analysis of Subordinate Components**

panel:

Field	Required	Description
SPECIFY SEARCH CRITERIA	Specify the subordinate component or components you want to investigate.	
Subordinate component name	Required	Enter the name of a subordinate component. Valid values Include:
		Like-copy component name

Field	Required	Description
		Statically linked like-NCAL or like-load subprogram name
		Cataloged procedure name
		Program name or string in JCL PGM= statement
		Data set name or string in JCL DSN= statement
		Valid formats include:
		Full component name or string
		Component name pattern or string ending in \
Library type	Optional	Enter the library type of the subordinate component. Valid formats include:
		Full three-character library type
		Asterisk (*) for all library types
		Blank, which defaults to \
		A library type pattern (partial wild-card) is invalid in this field. Enter asterisk (*) if the subordinate component is one of the following, which are stored in impact analysis as character strings rather than as components with library types:
		Cataloged procedure name
		Program name or string in JCL PGM= statement
		Data set name or string in JCL DSN= statement
Application	Optional	Enter the application of the subordinate component. Valid formats include:
		The full three- or four-character full application mnemonic
		Asterisk (*) for all applications
		Blank, which defaults to \
		A pattern (partial wild-card) is invalid in this field.

Field	Required	Description
Type of relationship	Required	Enter the relationship between the superior and subordinate components that you want to evaluate. Valid values are listed below. Valid formats include either the abbreviation on the left or the full relationship name. Either may be entered in upper or lower case.
		C: COPYBOOK (Source to copybook)
		S: SUBROUTINE (Composite load to linked load)
		J: JCL-PROCEDURE (Execution JCL to cataloged procedure)
		P: PGM NAME/SYMBOL (JCL/Procedure to program name)
		D: DSN NAME/SYMBOL (JCL/Procedure to data set name)
Mixed case	Optional	Select to use the exact text entered in the subordinate component name to search for relationships. If not selected, it will fold the text entered for subordinate component name into upper case before searching for relationships.
SPECIFY RESULTS FILTER CRITERIA	Specify filter criteria if you want to see a subset of the superior components that have a relationship to the component you are investigating.	
Superior component name	Optional	Enter the name of a superior component. Valid formats Include:
		Full component name
		Component name pattern
		Asterisk (*) for all names
		Blank, which defaults to \
Library type	Optional	Enter the library type of the superior component. Valid formats include:
		Full three-character library type
		Asterisk (*) for all library types
		Blank, which defaults to \

Field	Required	Description
Application	Optional	Enter the application of the subordinate component. Valid formats include:
		The full three- or four-character full application mnemonic
		Asterisk (*) for all applications
		Blank, which defaults to \
Short or Long form	Required	Enter the option for how many lines may be displayed for each superior component that matches the specified search criteria and results filters.
		S: Display only one line for each superior component name. A plus sign (+) at the end of the line indicates that the list of appl:lib type combinations is truncated.
		L: Display as many lines for each superior component as it takes to list all appl:lib type combinations where the relationship exists.

Displaying Impact Analysis Results

After you have specified the desired search and filter criteria on the **Impact Analysis of Subordinate Components** (CMNIMPSU) panel, press **Enter** to display the **Impact Analysis Results Selection List** (CMNIMPI1).

```
CMNIMPI1            I/A Results Selection List        Row 1 to 2 of 2
Command ===> ________________________________________Scroll ===> CSR

Components which reference:
  Appl: GENL Type: LOS Rel:  SUBROUTINE    Name: GNLSRS00          +
and satisfy these criteria:
  Appl: *     Type: * Name: *                                       +

   Component Name       Found in Appl:Libtype (common baselines)    +
__ GNLSRC1A             GENL:LOD
__ GNLSRC50             GENL:LOD
******************************* Bottom of data ********************************
```

The top half of the panel echoes the search criteria and results filter criteria for the *subordinate* component that you specified on the preceding **Impact Analysis of Subordinate Components** panel (CMNIMPSU).

The bottom half of the panel shows a list of *superior* components that satisfy the search and filter criteria that you specified.

> **Important**
>
> This panel shows you what components contain the component you asked about in the relationship you asked for.

The two superior components listed on the bottom half of the sample **Impact Analysis Results Selection List** panel above all contain component GNLSRS00.

This table describes the fields on the **Impact Analysis Results Selection List** panel

Field	Description
Command	Enter one of the following commands, or leave the command line blank and type a line command next to a component name.
	CANCEL: Cancel selections and exit this panel. Abbreviation: C.
	LOCATE *component*: Locate a specified component in the listed results. Abbreviation: L
	REFRESH: Display updated information on this panel. Abbreviation: R
List of components which reference	
Appl:Type	Application **and** Library type **that you specified under** SEARCH CRITERIA **on the** Impact Analysis Of Subordinate Components** panel (CMNIMPSU).
Name	Subordinate component name **that you specified under** SEARCH CRITERIA **on the** Impact Analysis Of Subordinate Components** panel (CMNIMPSU).
with a relationship of: which satisfy these criteria:	Type of relationship **that you specified under** SEARCH CRITERIA **on the** Impact Analysis Of Subordinate Components** panel (CMNIMPSU).
appl:Type	Application **and** Library type **that you specified under** RESULTS FILTER CRITERIA **on the** Impact Analysis Of Subordinate Components** panel (CMNIMPSU).
Name	Superior component name that you specified under **RESULTS FILTER CRITERIA** on the **Impact Analysis Of Subordinate Components** panel (CMNIMPSU).
line command	S Show the **Subordinate Component List** panel (CMNIMPD4) for the selected component.
Component Name	Displays the name of superior components that satisfy the selection and filter criteria that you specified on the **Impact Analysis of Subordinate Components** panel.

Field	Description
Found in Appl:Libtype (common baselines)	Displays the application and library type of the library where the superior component is baselined. See **Short or Long form** (in previous table).

If you type line command **S** on a superior component listed on the **Impact Analysis Results Selection List** panel (CMNIMPI1), the **Subordinate Component List** panel (CMNIMPD4) is displayed. This panel lists subordinate components contained in the selected superior component that have the same relationship type as your original query.

```
CMNIMPD4        Subordinate Component List              Row 1 to 3 of 3
Command ===> _________________________________________ Scroll ===> CSR

Superior Component Name . GNLSRC1A                                   +
Application:Type . . . . .GENL:LOD
Relationship . . . . . . .SUBROUTINE

Subordinate Components              Found in Appl:Libtype            +
GNLSRS00                            GENL:LOS
GNLSRS1B                            GENL:LOS
GNLSRS1C                            GENL:LOS
******************************* Bottom of data ********************************
```

This example shows that source component GNLSRC1A, which was selected on the preceding **Impact Analysis Results Selection List** panel, contains the following subordinate components:

- GNLSRS00, GNLSRS1B and GNLSRS1C in the LOS baseline library in application GENL.

  That is, the panel shows all of the subordinate SUBROUTINE relationships that exist for superior source component GNLSRC1A, which is library type LOD in application GENL.

> **Note**
>
> This is the same result panel that you see when you use the Component Bill of Materials query to see what subordinate components are contained in a superior component.

Getting a Component Bill of Materials

The Component Bill of Materials takes a top-down view of component relationships. You ask a question about a superior component, and it tells you about subordinate components.

In other words, it tells you what components are contained in a component that you want to change.

Accessing the Component Bill of Materials Panel

Use one of these methods to display the **Component Bill of Materials** panel (CMNIMPBM):

- Using the Menu Hierarchy:

  1. On the **Primary Option Menu**, select **Q Query**.
  2. On the **Query Options** menu (CMNQDMNU), select **B BofM**.

- Using Direct Panel Access:

  On a ChangeMan ZMF panel, type **=Q.B** in the **Command** or **Option** line and press **Enter**.

Specifying Search Criteria

You specify component search criteria on the **Component Bill of Materials** panel (CMNIMPBM).

```
CMNIMPBM Component Bill of Materials
Command ===>
Specify search criteria:
Component name . . . . . . +
Library type . . . . . . .
Application . . . . . . .
Type of relationship . . .
Short or Long form . . . . L
Enter "/" to select option:
Mixed case
```

The query bill of materials function returns a list of all subordinate components in all library types in all applications that are included in the target or wildcarded superior **Component Name**.

> **Important**
>
> This panel asks the question, "What components are contained in the component that I specify in the relationship that I specify?"

This table describes the fields on the **Component Bill of Materials** panel:

Field	Required	Description
Component name	Required	Enter the name of a superior component. Valid values Include:
		Like-source component name
		Statically linked like-load component name
		JCL component name
		Cataloged procedure name

Field	Required	Description
		Valid formats include:
		Full component name or string
		Component name pattern or string ending in \
Library type	Optional	Enter the library type of the superior component. Valid formats include:
		Full three-character library type
		Asterisk (*) for all library types
		Blank, which defaults to \
		A library type pattern (partial wild-card) is invalid in this field.
Application	Optional	Enter the application of the superior component. Valid formats include:
		The full three- or four-character full application mnemonic
		Asterisk (*) for all applications
		Blank, which defaults to \
		A pattern (partial wild-card) is invalid in this field.
Type of relationship	Required	Enter the relationship between the superior and subordinate components that you want to evaluate. Valid values are listed below. Valid formats include either the abbreviation on the left or the full relationship name. Either may be entered in upper or lower case.
		C: COPYBOOK (Copybook to source)
		S: SUBROUTINE (Linked load to composite load)
		J: JCL-PROCEDURE (Cataloged procedure to execution JCL)
		P: PGM NAME/SYMBOL (Program name to JCL/Procedure)
		D: DSN NAME/SYMBOL (Data set name to JCL/Procedure)
Short or Long form	Required	Enter the option for how many lines may be displayed for each subordinate component that matches the specified search criteria and results filters.
		S: Display only one line for each subordinate component name. A plus sign (+) at the end of the line indicates that the list of appl:lib type combinations are truncated.
		L: Display as many lines for each subordinate component as it takes to list all appl:lib type combinations where the relationship exists.

Field	Required	Description
Mixed case	Required	Select this option to use the exact text entered in the superior component name to search for relationships. If not selected fold the superior component name text into upper case before searching for relationships.

Displaying Bill of Materials Results

After you have filled in the **Component Bill of Materials** panel (CMNIMPBM), press **Enter** to display the **Bill of Materials Results Selection List** panel (CMNIMPB1).

```
CMNIMPB1     Bill of Materials Results Selection List   Row 1 to 1 of 1
Command ===> ________________________________________ Scroll ===> CSR

List of components with
Name . . . . . GNLSRC1A                                        +
Appl:Type . . .GENL : SRC
Relation . . ..COPYBOOK

Component Name    Found in Appl:Libtype (common baselines)          +
__GNLSRC1A        GENL:SRC
******************************* Bottom of data ********************************
```

The top half of the panel echoes the search criteria for the *superior* component that you specified on the preceding **Component Bill of Materials** panel (CMNIMPBM).

The bottom half of the panel shows a list of *superior* components that match the search criteria you specified. If you specified * in the **Library type** and/or **Application** fields on the **Component Bill of Materials** panel, this panel may list more than one component.

The superior components listed on the bottom half of the sample **Bill of Materials Results Selection List** panel above conform to the search criteria for component name GNLSRC1A in a COPYBOOK relationship in any library type in any application.

This table describes the fields on the **Bill of Materials Results Selection List** panel

Field	Description
Command	Enter one of the following commands, or leave the command line blank and type a line command next to a component name.
	CANCEL: Cancel selections and exit this panel. Abbreviation: C.
	LOCATE *component*: Locate a specified component in the listed results. Abbreviation: L
	REFRESH: Display updated information on this panel. Abbreviation: R
List of components with	

Field	Description
Name	Component name **that you specified under** SEARCH CRITERIA **on the** Bill of Materials Results Selection List** panel (CMNIMPB1)
Appl:Type	Application **and** Library type **that you specified under** SEARCH CRITERIA **on the** Bill of Materials Results Selection List** panel (CMNIMPB1)
Relation	Type of relationship **that you specified under** SEARCH CRITERIA **on the** Bill of Materials Results Selection List** panel (CMNIMPB1)
line command	S: Show the **Subordinate Component List** panel (CMNIMPD4) for the selected component.
Component Name	Displays the names of superior components that satisfy the selection criteria that you specified on the **Bill of Materials Results Selection List** panel (CMNIMPB1)
Found in Appl:Libtype (common baselines)	Displays the application and library type of the library where the superior component is baselined. See **Short or Long form**(in a previous table).

If you type line command **S** on a superior component listed on the **Bill of Materials Results Selection List** panel (CMNIMPB1), the **Subordinate Component List** panel (CMNIMPD4) is displayed.

```
CMNIMPD4        Subordinate Component List          Row 1 to 4 of 4
Command ===> _____________________________________ Scroll ===> CSR

Superior Component Name . GNLSRC1A                                       +
Application:Type . . . . .GENL:SRC
Relationship . . . . . . .COPYBOOK

Subordinate Components          Found in Appl:Libtype                    +
GNLCPY00                        GENL:CPY
GNLCPY1A                        GENL:CPY
GNLCPY1B                        GENL:CPY
GNLCPY1X                        GENL:CPY
******************************* Bottom of data ********************************
```

This panel lists subordinate components contained in the selected superior component that have the relationship that you specified on the **Bill of Materials Results Selection List** panel (CMNIMPB1).

> **Important**
>
> This panel shows you what components are contained in the component you asked about.

This table describes the fields on the **Subordinate Component List** panel

Field	Description
Command	Enter one of the following commands, or leave the command line blank and type a line command next to a component name.
	CANCEL: Cancel selections and exit this panel. Abbreviation: C.
	Locate *component*: Locate a specified component in the listed results. *component* Abbreviation: L
Superior Component Name	Component name **that you selected on the** Bill of Materials Results Selection List** panel
Application:Type	Application **and** Library type **that you specified under** SEARCH CRITERIA **on the** Bill of Materials Results Selection List** panel (CMNIMPB1)
Relationship	Type of relationship **that you specified under** SEARCH CRITERIA **on the** Bill of Materials Results Selection List** panel (CMNIMPB1)
Subordinate Components	Displays the names of subordinate components that are contained in the selected superior component and that have the **Type of relationship** that you specified on the **Bill of Materials Results Selection List** panel (CMNIMPB1)
Found in Appl:Libtype	Displays the application and library type of the library where the subordinate component is baselined. See Short or Long form (in a previous table).

Use Case Scenarios

The use cases in this section illustrate how to use the Query Impact and Query BofM functions to answer questions about the effect of changes you want to make to application components.

What copybooks are contained in a source component?

You want to change the source code for program GNLSRC1A. You want to know what copybooks are included in the source code because you might have to change them too.

Use the Query Bill of Materials function to list like-copy components (subordinate components) that are referenced in like-source component GNLSRS1... (a superior component).

1. Bring up the **Component Bill Of Materials** panel (CMNIMPBM) by selecting option **Q.B** from the **Primary Option Menu**.
2. Fill in the **Component Bill Of Materials** panel (CMNIMPBM) as follows.

```
CMNIMPBM                Component Bill of Materials
Command ===> __________________________________________________

Specify search criteria:

Component name . . . . . . GNLSRC1A                      +
Library type . . . . . . . SRC
Application . . . . . . .  GENL

Type of relationship . . . COPYBOOK

Short or Long form . . . . L
Enter "/" to select option:
__ Mixed case
```

3. Press **Enter** to display the **Bill of Materials Results Selection List** panel (CMNIMPB1).

```
CMNIMPB1        Bill of Materials Results Selection List   Row 1 to 1 of 1
Command ===> ___________________________________________ Scroll ===> CSR

List of components with
Name . . . .  GNLSRC1A                                        +
Appl:Type . . GENL : SRC
Relation . .  COPYBOOK

Component Name          Found in Appl:Libtype (common baselines)      +
GNLSRC1A                GENL:SRC
******************************* Bottom of data ********************************
```

   This panel lists one component with name GNLSRC1A in library type SRC in application GENL. (More components might have been listed if you had used a pattern for the **Component Name** or if you had left **Library type** or **Application** blank.)

4. On the **Bill of Materials Results Selection List** panel (CMNIMPB1), type **S** in the line command for component and press **Enter** to display the **Subordinate Component List** panel (CMNIMPD4).

```
CMNIMPD4            Subordinate Component List          Row 1 to 4 of 4
Command ===> _________________________________________ Scroll ===> CSR

Superior Component Name . GNLSRC1A                               +
Application:Type . . . .  GENL:SRC
Relationship . . . . . .  COPYBOOK

Subordinate Components              Found in Appl:Libtype        +
GNLCPY00                            GENL:CPY
GNLCPY1A                            GENL:CPY
GNLCPY1B                            GENL:CPY
GNLCPY1X                            GENL:CPY
******************************* Bottom of data ********************************
```

This panel lists four like-copy components that are referenced in like-source component GNLSRC1A.

What source components contain a particular copybook?

You want to change copybook GNLCPY00, which is in the baseline library for library type CPY in application GENL. You want to know what source components include the copybook because you might have to change them too. (You must at least recompile those source components.)

Use the Query Impact function to see what like-source components (superior components) reference like-copy component GNLCPY00 (a subordinate component).

1. Bring up the **Impact Analysis of Subordinate Components** panel (CMNIMPSU) by selecting option **Q.I** from the **Primary Option Menu**.
2. Fill in the **Impact Analysis of Subordinate Components** panel (CMNIMPSU) as follows:

```
CMNIMPSU            Impact Analysis of Subordinate Components
Command ===>___________________________________________________

Specify search criteria:

Subordinate component name . . . GNLCPY00                       +
Library type . . . . . . . . . . CPY
Application . . . . . . . . . . .GENL
Type of relationship . . . . . . C

Specify results filter criteria:

Superior component name . . . . .______________________________+
Library type . . . . . . . . . . ________
Application . . . . . . . . . . ._________

Short or Long form . . . . . . . L

Enter "/" to select:
__ Mixed case
```

   Do not specify any **Results Filter Criteria** because you want to see all like-source results.

3. Press Enter to display the **Impact Analysis Results Selection List** panel (CMNIMPI1).

```
CMNIMPI1            I/A Results Selection List      Row 1 to 7 of 7
Command ===> _____________________________________ Scroll ===> CSR

Components which reference:
  Appl: GENL Type: CPY Rel: COPYBOOK        Name: GNLCPY00      +
and satisfy these criteria:
  Appl: *    Type: *   Name: *                                  +

Component Name       Found in Appl:Libtype (common baselines)   +
GNLSRC1A             GENL:SRC
GNLSRS00             GENL:SRS
GNLSRS1B             GENL:SRS
GNLSRS1C             GENL:SRS
GNLSRS5A             GENL:SRS
GNLSRS5B             GENL:SRS
GNLSRS5C             GENL:SRS
******************************* Bottom of data ********************************
```

This panel lists all like-source components that reference the selected like-copy component.

4. The Query Impact function has answered your original question, but if you want to see what other copybooks are referenced in component listed on the **Impact Analysis Results Selection List** panel, type **S** in the line command for that component and press **Enter** to display the **Subordinate Component List** panel (CMNIMPD4).

```
CMNIMPD4         Subordinate Component List         Row 1 to 4 of 4
Command ===> ____________________________________ Scroll ===> CSR

Superior Component Name . GNLSRC1A                               +
Application:Type . . . .  GENL:SRC
Relationship . . . . . .  COPYBOOK

Subordinate Components              Found in Appl:Libtype        +
GNLCPY00                            GENL:CPY
GNLCPY1A                            GENL:CPY
GNLCPY1B                            GENL:CPY
GNLCPY1X                            GENL:CPY
******************************* Bottom of data ********************************
```

Notice that you have arrived at the same result as in use case scenario What copybooks are contained in a source component?.

What subprograms are statically linked in a load module?

You want to change a main program (component name) which is statically link edited and baselined in library type LOD in the selected application. You want to know what subprograms are included in the selected GNLSRC1A composite load because you might have to change them too.

Use the Query Bill of Materials function to list subprogram load modules (subordinate components) that are statically linked in the composite load component (a superior component).

1. Bring up the **Component Bill Of Materials** panel (CMNIMPBM) by selecting option **Q.B** from the **Primary Option Menu**.
2. Fill in the **Component Bill Of Materials** panel (CMNIMPBM) as follows.

```
CMNIMPBM                    Component Bill of Materials
Command ===> _____________________________________________

Specify search criteria:

Component name . . . . . . GNLSRC1A                     +
Library type . . . . . . . *
Application . . . . . . .  *
Type of relationship . . . SUBROUTINE

Short or Long form . . . . L

Enter "/" to select option:
__ Mixed case
```

Leave **Library type** and **Application** blank to see if component name is used in any other library type and application.

3. Press **Enter** to display the **Bill of Materials Results Selection List** panel (CMNIMPB1).

```
CMNIMPB1    Bill of Materials Results Selection List       Row 1 to 1 of 1
Command ===> ____________________________________________ Scroll ===> CSR

List of components with
Name . . . . GNLSRC1A                                          +
Appl:Type . . * : *
Relation . . SUBROUTINE

Component Name      Found in Appl:Libtype (common baselines)         +
__GNLSRC1A          GENL:LOD
******************************* Bottom of data ********************************
```

This panel lists selected component in library type LOD in application GENL.

Note

The named component can be used in different library types in different applications. The baseline libraries may not be shared across applications and there will be no indication whether the content of the components are the same in the two applications.

4. On the **Bill of Materials Results Selection List** panel (CMNIMPB1), type **S** in the line command for component name GNLSRC1A in application GENL and press **Enter** to display the **Subordinate Component List** panel (CMNIMPD4).

```
CMNIMPD4     Subordinate Component List             Row 1 to 4 of 4
Command ===> _____________________________________ Scroll ===> CSR

Superior Component Name . GNLSRC50                               +
Application:Type . . . .  GENL:LOD
Relationship . . . . . .  SUBROUTINE

Subordinate Components              Found in Appl:Libtype        +
GNLSRS00                            GENL:LOS
GNLSRS5A                            GENL:LOS
GNLSRS5B                            GENL:LOS
GNLSRS5C                            GENL:LOS
******************************* Bottom of data ********************************
```

This panel lists **Subordinate component(s)** in like-NCAL library type LOS that is statically linked in composite like-load selected **Superior component**.

What load modules contain a particular statically linked subprogram?

You want to change common subprogram GNLSRS00 in application GENL. The like-NCAL load for this program in the baseline library for library type LOS. You want to know what statically linked load modules include the subprogram because you might have to change them too. (You must at least relink those composite load modules.)

Use the Query Impact function to see what statically linked composite load components (superior components) include subprogram load GNLSRS00 (a subordinate component).

1. Bring up the **Impact Analysis of Subordinate Components** panel (CMNIMPSU) by selecting option **Q.I** from the **Primary Option Menu**.
2. Fill in the **Impact Analysis of Subordinate Components** panel (CMNIMPSU) as follows. Leave the results filter criteria blank to show all results.

```
CMNIMPSU            Impact Analysis of Subordinate Components
Command ===> ____________________________________________________

Specify search criteria:

Subordinate component name . . . GNLSRS00                       +
Library type . . . . . . . . . . LOS
Application . . . . . . . . . .  GENL

Type of relationship . . . . . . SUBROUTINE

Specify results filter criteria:

Superior component name . . . . . *                             +
Library type . . . . . . . . . .  *
Application . . . . . . . . . . . *

Short or Long form . . . . . . . L

Enter "/" to select:
__ Mixed case
```

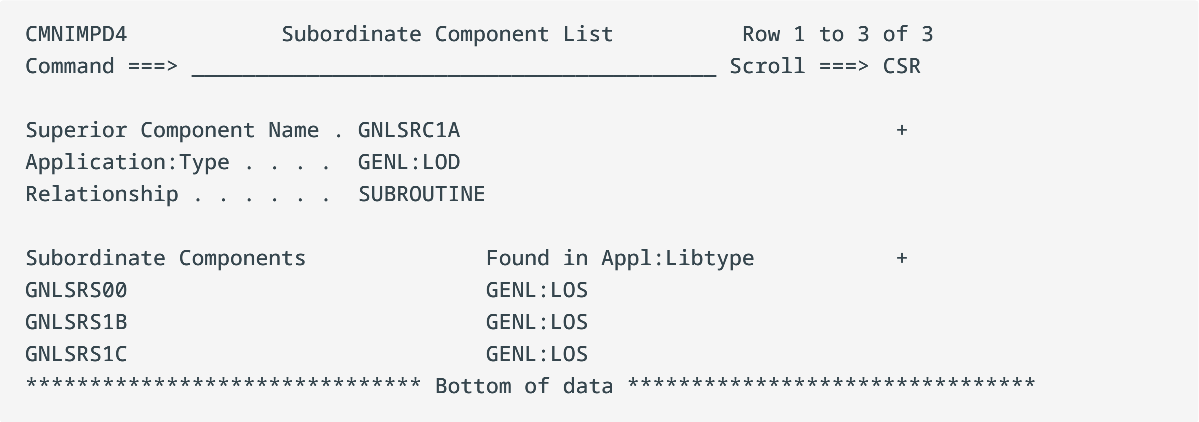
3. Press Enter to display the **Impact Analysis Results Selection List** panel (CMNIMPI1).

```
CMNIMPI1        I/A Results Selection List              Row 1 to 2 of 2
Command ===> __________________________________________ Scroll ===> CSR

Components which reference:
Appl: GENL  Type: LOS   Rel: SUBROUTINE     Name: GNLSRS00          +
and satisfy these criteria:
Appl: *     Type: *     Name: *                                     +

Component Name      Found in Appl:Libtype (common baselines)        +
GNLSRC1A            GENL:LOD
GNLSRC50            GENL:LOD
******************************* Bottom of data ********************************
```

This panel lists two composite like-load components that contain like-NCAL component GNLSRS00.

> **Note**
>
> Notice that the composite like-load components are in baseline libraries for various library types in various applications. In this scenario, GENL subprogram GNLSRS00 was included in builds for these composite load components by using participating packages.

4. The Query Impact function has answered your original question, but if you want to know what subprogram load modules are statically linked in component GNLSRC1A listed on the **Impact Analysis Results Selection List** panel, type **S** in the line command for that component and press **Enter** to display the **Subordinate Component List** panel (CMNIMPD4).



```
CMNIMPD4            Subordinate Component List          Row 1 to 3 of 3
Command ===> _________________________________________ Scroll ===> CSR

Superior Component Name . GNLSRC1A                                 +
Application:Type . . . .  GENL:LOD
Relationship . . . . . .  SUBROUTINE

Subordinate Components              Found in Appl:Libtype          +
GNLSRS00                            GENL:LOS
GNLSRS1B                            GENL:LOS
GNLSRS1C                            GENL:LOS
******************************* Bottom of data ********************************
```

This panel shows you that load module GNLSRC1A contains three statically linked subprograms, GNLSRS00, GNLSRS1B and GNLSRS1C from application GENL.

22. Browsing Compressed Listings:

This chapter describes compressed listings and tells you how to view them.

About Compressed Listings

One of the last steps in a ChangeMan ZMF build job collects sysout data sets, combines them into a single file, and stores the file in compressed format as a member in a staging library for library type LST. When you want to view the saved build job sysout, utility program SERCOPY decompresses the file into its original readable format and copies the listing to a temporary data set.

The build job establishes a source-to-load relationship between the source member and the compressed listing member. When the source member is staged again, the compressed listing is deleted to make way for a new listing. If the source member is deleted from the package, the compressed listing is also deleted.

Compressed listings are stored in baseline libraries like other application components. Prior baseline versions are baseline rippled so that old listings are available. Prior versions of baselined compressed listings are stored as full PDS members, so there are often fewer versions of build listings than there are versions of the source, which are usually stored in Stacked Reverse Delta format.

Compressed listing members can be installed in production libraries to make the listings available at remote production sites where they can be used to diagnose program problems.

Accessing Browse Listing Panels

Use one of these methods to display the **Browse Compressed Listing** panel:

- Using the Menu Hierarchy:

  a On the **Primary Option Menu**, select **1 Build**.

  b On the **Build Option** panel, select **L Listing**.

- Using Direct Panel Access:

You can also access the **Browse Compressed Listings** (CMNBLST0) panel by entering **=1.L** on the **Command** line of any ChangeMan ZMF panel. Shortcuts to navigating panels is described in Chapter 2, Navigating Panels.

- Using the Change Package List:

  On the **Change Package List** (CMNLIST3) panel, type **BL** in the line command field for a change package and press **Enter**.

Browsing Compressed Listings

Follow these steps to browse a compressed listing stored in a staging or baseline library. To complete this process you will access two separate **Browse Compressed Listings** panels (CMNRBLS0 and CMNRBLS1).

1. Access the first **Browse Compressed Listings** panel (CMNRBLS0) by selecting **L Listing** from the **Build Option** (CMNBUILD) panel or entering **=1.L** from any ChangeMan ZMF panel. On this panel you will establish the criteria used to display a filtered list of compressed listings.

```
CMNRBLS0         Browse Compressed Listings
Command ===> ___________________________________________
Specify selection criteria:
Package . . . . . . . . GENL000006
Application . . . . . . GENL    (Required if package not supplied)
Component name . . . .  _________________________________________ +
Library type . . . . . LST      (Blank for list)
From date . . . . . . .         (yyyymmdd)
To date . . . . . . . .         (yyyymmdd)

Enter "/" to select option
__ Area listings
__ Prompt for listing disposition
```

This table describes the fields on the first **Browse Compressed Listings** (CMNRBLS0) panel.

Specify selection criteria: You will be able to enter a combination of Package ID and/or Application to determine whether listings are obtained from package staging libraries,baseline libraries, or both. If only the Application field is selected, the listings from Baseline will be displayed. If only the Package ID field is selected the listings from Staging are displayed. If both are selected, Baseline and Staging listings are displayed.

Field	Description
Package	Enter the package ID for the listings you've selected to browse.
Application	Enter the application for the listings you've selected to browse.
Component name	Enter the component name you wish to browse or view for the listings you've selected.
	Name: Select compressed listings for the named component.
	Note: To select a filtered list of compressed listings you will also be able to enter a component name by specifying a pattern See Building Lists Using Patterns for information about using wild card characters to specify a pattern. Leave blank to display a list of all compressed listings.

Field	Description
Library type	Enter the library type for compressed listings selected. Leave this field blank to display the **Library Type Selection List** (CMNBLST3) panel to choose a specific library type.
From date	Enter the creation start date range for the compressed listing you have specified. The date in YYYYMMDD format is inclusive. Leaving the date blank is the same as typing 00000000. Note: The from date **Created** heading will be shown on the second Browse Compressed Listing Panel.
To date	Enter the creation end date range for the compressed listing you have specified. The date in YYYYMMDD format is inclusive. Leaving the date blank is the same as typing 99999999. Note: The to **Changed** date heading will be shown on the second Browse Compressed Listing Panel.
Area listings	Select if you would like to include the listings that are in release area libraries for the package. The package must be supplied if this option is selected. Omit if you do not want to include the area listings. **Note:** When you view the saved build job sysout, utility program SERCOPY will decompress the file into its original readable format and copy the listing to a temporary dataset.
Prompt for listing disposition	Select this option to display the **Specify Listing Disposition** (CMNBLST2) panel after displaying the uncompressed listing and disposition of the temporary file. Omit to delete the temporary file after it has been browsed.

2. On the first **Browse Compressed Listings** panel, enter selection criteria and press **Enter**. If there are listings that satisfy your selection criteria, the second **Browse Compressed Listings** panel (CMNRBLS1) is displayed.

```
CMNRBLS1    Browse Compressed Listings          Row 1 to 18 of 27
Command ===>____________________________________ Scroll ===> CSR

Package: GENL000006     Status: DEV     Install date: 20150228

   Name    +Level   vv.mm Created    Changed         Size Init User
__ GNLSRC1A BASE-00 01.00 2015/01/05 2015/01/05 01:22 630 630  SERT
__ GNLSRC1A STAGING 01.00 2015/01/06 2015/01/06 19:54 630 630  SERT
__ GNLSRC50 BASE-00 01.00 2015/01/05 2015/01/05 01:00 452 452  SERT
__ GNLSRC50 BASE-01 01.00 2015/01/04 2015/01/04 19:59 449 449  SERT
__ GNLSRC50 STAGING 01.00 2015/01/06 2015/01/06 21:16 453 453  SERT
__ GNLSRC99 BASE-00 01.00 2015/01/05 2015/01/05 00:55 549 549  SERT
__ GNLSRS00 BASE-00 01.00 2015/01/05 2015/01/05 00:56 490 490  SERT
__ GNLSRS00 BASE-01 01.00 2015/01/04 2015/01/04 19:35 495 495  SERT
__ GNLSRS00 BASE-02 01.00 2014/12/23 2014/12/23 18:39 492 492  SERT
__ GNLSRS00 STAGING 01.00 2015/01/06 2015/01/06 19:48 499 499  SERT
__ GNLSRS01 BASE-00 01.00 2014/12/23 2014/12/23 18:09 493 493  SERT
__ GNLSRS1B BASE-00 01.00 2015/01/05 2015/01/05 00:56 524 524  SERT
__ GNLSRS1B BASE-01 01.00 2015/01/04 2015/01/04 19:48 529 529  SERT
__ GNLSRS1B BASE-02 01.00 2014/12/23 2014/12/23 18:01 526 526  SERT
__ GNLSRS1B STAGING 01.00 2015/01/06 2015/01/06 19:21 526 526  SERT
__ GNLSRS1C BASE-00 01.00 2015/01/05 2015/01/05 00:56 506 506  SERT
__ GNLSRS1C BASE-01 01.00 2014/12/23 2014/12/23 18:09 509 509  SERT
__ GNLSRS1C STAGING 01.00 2015/01/06 2015/01/06 19:21 509 509  SERT
```

This table describes the fields on the second **Browse Compressed Listings** panel (CMNRBLS1):

Field	Description**
Command	Enter one of the following commands, or leave the command line blank and type a line command **B** (Browse) or **V** (View) next to a component name.
	REFRESH: Display updated information on this panel. Abbreviation: R
	SORT *heading*: Sort listed components by information under the specified column heading.
	LOCATE *member: Locate a specified entry, based on the current sort. For example, enter 'LOCATEmember' to scroll to the first occurrence of component*member*.
	Note: You are able to sort a specific column heading (package, status or install dates) on the second **Browse Compressed Listing** panel to locate a member from the last sorted column or by Name if the components were not sorted. Abbreviation: L
	LONG: Display long component name on the first line, and display other component information on a second line.
	SHORT: Suppress the long component name displayed with the LONG command.
	CANCEL: Cancel the function and return to the previous panel. Abbreviation: C
Package Status Install date	The Package, Status and Install date column heading fields are displayed on the second **Browse Compressed Listing** (CMNRBLS1) panel if you specified a Package ID in your selection criteria.
Line Command	
	B: Display the uncompressed listing in browse mode.
	V: Display the uncompressed listing in view (edit in stage) mode.
Name	Displays the name of the component.
Level vv.mm Created Changed Size Init User	These fields display the ISPF statistics for the listed members. The **LEVEL** field reflects where the listing currently resides. A listing residing within a package's staging library will display the LEVEL as STAGING a listing residing within an application's baseline library will display the level as BASE - # (where # represents the baseline level). A listing residing within an ERO release area library will display the area name as the level. All release area levels that contain the listing for the selected package will be displayed. The **User** field displays the userid of the SERNET started task where ChangeMan ZMF runs because the member was created by the build job submitted from the started task.

3. On the second **Browse Compressed Listings** (CMNRBLS1) panel, enter **B** (Browse) or **V** (View) in the command line field for a component, and press **Enter**. The compressed listing member is then expanded into a ChangeMan ZMF utility data set and displayed.

```
Menu Utilities Compilers Help
sssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss
BROWSE CMNTP.A009D.#CE61754.#4BFB561.OUTLIST    Line 00000000 Col 001 132
Command ===>                                            Scroll ===> PAGE
*********************************************************** Top of Data ************************************************************

********************************************************************************
* DDNAME: SERCOPY.SYSPRINT                                               *
********************************************************************************

SER9403I SERCOPY options: BSAM EXPAND
SER9405I Input dsname: CMNTP.S6.V810.BASE.GENL.SRS
SER9406I Output dsname: SYS15005.T005627.RA000.USER015D.SOURCE.H09
SER9407I Begin copy: INFILE=SYS15005SYS00001 OUTFILE=SYSUT2
SER9425I Copy completed successfully

********************************************************************************
* DDNAME: WRITE.SYSPRINT                                                 *
********************************************************************************

ChangeMan(R) ZMF    CMNWRITE - 8.1.0 MONDAY JANUARY 5, 2015   00:56:34
    PARM='SUBSYS=6,USER=USER015,'
    PARM interpretation: ChangeMan ZMF subsystem "6"
      Expansion of uncovered COPY/Include variations

SYSIN: TYP=CPY/CMNTP.S6.GENL.STG6.#000005.CPY
SYSIN: TYP=ZCP/CMNTP.S6.V810.BASE.GENL.ZCP
SYSIN: TYP=CPY/CMNTP.S6.V810.BASE.GENL.CPY
SYSIN: CMP=GNLSRS5A.SRS
SYSIN: LNG=COBOL2
SYSIN: PKG=GENL000005
CMN7500I - Attempting to initiate dialog with started task.
CMN1400I - Session established with ChangeMan ZMF started task.
Library search order: PDS/CMNTP.S6.GENL.STG6.#000005.CPY            (CPY)
PDS/CMNTP.S6.V810.BASE.GENL.ZCP                                     (ZCP)
PDS/CMNTP.S6.V810.BASE.GENL.CPY                                     (CPY)
LV -MEMBER--- VV.MM --CREATE-- --LAST CHANGED-- SIZE INIT -USERID- -- L I B R A R Y  N A M E ---------------- ---FINGERPRINT---
                                                      --- C O M P O N E N T  D E S C R I P T I O N ---(IF ANY)-------------
00 GNLSRS5A   --.-- *** NO ISPF/PAN/LIB STATISTICS *** SYS15005.T005627.RA000.USER015D.SOURCE.H09                 FC04ECCA-0000030C
01 GNLCPY00   03.01   2002/05/07 2015/01/01 21:43 5   1   USER015     CMNTP.S6.GENL.STG6.#000005.CPY            D345511F-000000C7
01 GNLCPY5A   01.02   2002/05/07 2014/12/23 17:56 5   1   USER015     CMNTP.S6.V810.BASE.GENL.CPY               F953A35C-000000BE
02 GNLCPY1X   01.02   2002/05/07 2014/12/23 17:56 4   1   USER015     CMNTP.S6.V810.BASE.GENL.CPY               23E79EA6-000000A3
01 GNLCPY5B   01.02   2002/05/07 2014/12/23 17:57 4   1   USER015     CMNTP.S6.V810.BASE.GENL.CPY               23F7CCA0-000000A3
CMN1410I - Session terminated with ChangeMan ZMF started task.
CMN5400I - Time of day at end of job: 00:56:35 - Condition Code on exit: 0
Record count in ====> 00029
Record count out ===> 00071

********************************************************************************
* DDNAME: COBOL2.SYSPRINT                                                      *
********************************************************************************
PP 5648-A25 IBM COBOL for OS/390 & VM 2.1.2         Date 01/05/2015 Time 00:56:37   Page 1
Invocation parameters:
OBJECT,LIB,
Options in effect:
NOADATA
  ADV
NOANALYZE
  QUOTE
NOAWO
  BUFSIZE(4096)
NOCMPR2
NOCOMPILE(S)
NOCURRENCY
  DATA(31)
NODATEPROC
NODBCS
NODECK
NODLL
NODUMP
NODYNAM
```

Each sysout data set kept for the job is headed by a flower box that is labeled with **DDNAME: stepname.dsname** where *stepname* is the job step name and *dsname* is the sysout data set name where the sysout data set was written. See compressed listing member above for example **DDNAME: SERCOPY.SYSPRINT**.

If you want to page through the sysout data sets in a listing, type the following in the command line:

```
find ddname:
```

then press **PF5** until you find the sysout data set that you want. (Remember to type the colon after ddname.)

4. Press **PF3** to exit the browse session.
5. The **Specify Listing Disposition** panel (CMNBLST2) is displayed if you select the **Prompt For Listing Disposition** field on the first **Browse Compressed Listings** (CMNRBLS0) panel.

```
CMNBLST2            Specify Listing Disposition
Command ===> _________________________________________________
Listing dataset name:       CMNTP.A009D.#CE6171C.#1468839.OUTLIST

Listing disposition . . . 3         1. Print dataset and delete
                                    2. Print dataset and keep
                                    3. Delete dataset without printing
                                    4. Keep dataset without printing

Job statement information if printing:
//USER016 JOB (0000),'CHANGE MAN',
//      CLASS=A,NOTIFY=USER016,MSGCLASS=X
//*
//*
```

This panel shows you the name of the utility dataset that contains the uncompressed listing, and the panel gives you various choices to delete or keep the dataset and to print or not print the listing.

6. Press **PF3** repeatedly to exit the browse compressed listing function.

23. Using Baseline Browse Functions:

This chapter describes the various functions available through the baseline browse facility.

About Baseline Browse

The baseline browse/print facility provides access to components in baseline and promotion libraries without requiring you check out the components to a change package. Access restrictions of member level security are enforced.

Options under the baseline browse/print facility include:

- List libraries that contain a component
- Browse a component
- View (browse in edit) a component
- Print a component to a user-defined print class
- Print a component with expanded copybooks
- Copy a component to a user specified dataset
- List history by package that contain a component
- Browse a stacked reverse delta (SRD) member for a prior version of a component
- View a stacked reverse delta (SRD) member for a prior version of a component

Accessing the Baseline Browse/Print Facility

Baseline browse/print functions are executed from the **Baseline Browse/Print Facility** (CMNBRWB0) panel.

```
CMNBRWB0            Baseline Browse/Print Facility
Command ===> _____________________________________________________

Application . . . . . . GENL    (Blank for list)
Library type . . . . .  ___     (Blank for list)
Library level . . . . . 0       (Baseline 0 to -n; Promotion 1 to n)
Component name . . . .  _______________________________________________

Enter "/" to select option
__ Build member list from scan results
__ Case sensitive

Records to select . . . 0       (0 = all)
From column . . . . . . 0       (0 = all)
To column . . . . . .   0       (Max = 80)

Data string _________________________________________________
```

Use one of these methods to display the **Baseline Browse/Print Facility** panel:

- Using the Menu Hierarchy:

  a On the **Primary Option Menu**, select **1 Build**.

  b On the **Build Option** panel, select **B Browse**.

- Using Direct Panel Access:

  On a ChangeMan ZMF panel, type **=1.B** in the **Command** or **Option** line and press Enter.

On the **Change Package List** (CMNLIST3) panel, type **BB** in the line command for a change package and press **Enter**.

Using Baseline Browse

The top half of the **Baseline Browse/Print Facility** (CMNBRWB0) panel provides panel fields for you to select files and components, and it provides options to act on components that you select.

The bottom half of the **Baseline Browse/Print Facility** panel provides a scan facility to select components based on their content.

1. These are the panel options and the file and component selection fields on the top half of the **Baseline Browse/Print Facility** panel (CMNBRWB0).

```
CMNBRWB0                Baseline Browse/Print Facility
Command ===> ________________________________________________________

Application . . . . . . GENL    (Blank for list)
Library type . . . . .  ____    (Blank for list)
Library level . . . . . 0       (Baseline 0 to -n; Promotion 1 to n)
Component name . . . .  _______________________________________________
```

This table describes the fields on the top half of the **Baseline Browse/Print Facility** panel (CMNBRWB0).

Field	Description
Panel Options	Enter an option, or leave the option blank if you use a pattern in the Component Name field.
	L: List all baseline and promotion libraries that contain a component
	B: Browse a component in a baseline library
	C: Copy a member from a baseline library to a user- specified dataset.
	V: View (browse in edit) a component in a baseline library
	X: Print-expanded option allows the user to expand all COPY and INCLUDE statements prior to printing.
	P: Print a member to a user-defined print class.
	H: Provides a detailed history by package for the member specified.
	BS: Browse stack of the stacked reverse delta (SRD). Allows the user to display the number of levels associated with the specified component. It does NOT expand the differences between levels or versions.
	VS: View stack of the stacked reverse delta (SRD).
Application	Enter the three or four-character code for the application your user ID is authorized. You may leave this field blank to display a selection list of valid applications defined to ChangeMan ZMF.
Library type	Enter the library type of the component. For a list of valid library types and their descriptions, leave this field blank. This required field works in conjunction with the Application field to determine the target library for the request.
Library level	Enter the numeric value of the baseline version or promotion level library associated with the version of the component you want to access.
	0: Current or 0-level baseline library.
	-1 to -999: Prior version baseline library. The numeric value following the minus sign represents a relative prior version (-1 means the version immediately prior to the current version).

Field	Description
	+1 to +99: Future version from a promotion library. The numeric value following the plus sign is a promotion level number in a promotion site. If there are multiple sites, the Promotion Library Selection List panel is displayed where you select one of the listed promotion libraries.
	Blank: Leave this field blank if you type L in the Option field.
Component name	Enter the component name or pattern you want to access for a selection list. This field may be left blank to display all members in the selected library.

2. These are the scan function fields.

```
Enter "/" to select option
__ Build member list from scan results
__ Case sensitive

Records to select . . . 0 (0 = all)
From column . . . . . . 0 (0 = all)
To column . . . . . .   0 (Max = 80)

Data string . . . . . .
```

This table describes the fields on the bottom half of the **Baseline Browse/Print Facility** panel (CMNBRWB0).

If you use a pattern or blank in the Component Name field, you can use the following fields to build a selection list of components that contain a data string.

Field	Description
Build member list from scan results	Select to initiate a scan on the baseline library to produce a member list for the option you selected. Omit to indicate that no scan criteria is to be used. If not selected, all fields listed below will be ignored. **Note:** Member List Scan does not support zFS files or data sets with RECFM=U.
Case sensitive	Select if the DATA STRING entered should be used exactly as entered (with regard to upper and lower case characters). Omit to scan data string without case sensitivity.
Records to select	Enter a numeric value to determine when the scan is to be discontinued. If '0' is entered, the scan will display a list of all members in the baseline library that meet the DATA STRING search criteria.
From column	Enter the start column to begin scan for each record. Entering a '0' value will scan each record beginning in column one.
To column	Enter the column the scan is to end. Entering a '0' value will scan to the 80th column.

Field	Description
Data string	Enter a data string that will be used to build a member selection list for the requested option. If the Member List Scan field is not selected the data String will be ignored.

Baseline Browse Component list

If a pattern or blank component name is specified, the component list is displayed.

```
CMNBRWB1 Browse/Print from SRC/lvl(0) Row 1 to 17 of 26
Command ===> Scroll ===> CSR
Input dataset:
CMNTP.S6.V810.BASE.ACTP.SRC
Name + Status vv.mm Created Changed Size Init User
ACPSRCCA 01.03 2002/05/07 2015/01/05 00:44 34 23 JPRESTO
ACPSRCCC 02.00 2002/05/07 2015/01/05 14:56 29 23 JPRESTO
ACPSRCCE 02.00 2002/05/07 2015/01/05 14:56 29 23 JPRESTO
ACPSRCD1 04.01 2002/05/07 2015/01/06 01:43 53 15 JPRESTO
ACPSRCSA 02.00 2002/05/07 2015/01/05 14:56 27 23 JPRESTO
ACPSRC1A 01.03 2014/12/18 2014/12/18 02:14 33 34 JPRESTO
ACPSRC2A 02.00 2010/07/14 2015/01/05 14:56 28 24 JPRESTO
ACPSRC4A 02.00 2012/09/26 2015/01/05 14:57 44 27 JPRESTO
ACPSRC6A 02.00 2012/09/26 2015/01/05 14:57 43 27 JPRESTO
ACPSRC80 02.00 2002/05/07 2015/01/05 14:57 24 1 JPRESTO
ACPSRC90 02.00 2002/05/07 2015/01/05 14:57 28 1 JPRESTO
ACPSRC91 02.00 2002/05/07 2015/01/05 14:57 29 15 JPRESTO
ACPSRC92 02.00 2002/05/07 2015/01/05 14:58 29 15 JPRESTO
ACPSRC93 02.01 2002/05/07 2015/01/04 22:22 29 15 JPRESTO
ACPSRC94 02.00 2002/05/07 2015/01/05 14:58 24 1 JPRESTO
ACPSRC95 02.03 2002/05/07 2014/12/16 19:46 25 15 JPRESTO
ACPSRC96 02.00 2002/05/07 2015/01/05 14:58 24 15 JPRESTO
******************************* Bottom of data ********************************
```

Valid primary commands are:

Command	Description
CANCEL	Cancel processing and exit.
DCD	Display component general descriptions.
LOCATE	Locate an entry.
REFRESH	Refresh the panel display.
SHORT	Display abbreviated component list.
SORT	Sort the list.

Valid line commands are:

Command	Description
B	Browse a component in a baseline library.

Command	Description
BS	Browse stack of the stacked reverse delta (SRD). This allows the user to display the number of levels associated with the specified component. It does NOT expand the differences between levels or versions.
C	Copy a component from a baseline library to a user-specified dataset.
D	Deselect a previously selected component.
H	Provide a detailed history by package for the component specified.
P	Print a component to a user-defined print class.
V	View a component in a baseline library.
VS	View stack of the stacked reverse delta (SRD).
X	Use the Print-expanded option. This allows the user to expand all COPY and INCLUDE statements prior to printing.

24. Cross-Application Scan:

This chapter describes the library scan facility.

About Cross Application Scan

The library scan facility searches *baseline level 0 libraries* for character strings and component dependencies:

For any character string, you can further specify:

- Whether the search is case sensitive.
- The record positions to examine for the string.
- Two strings with a boolean OR or AND relationship.

Component dependencies include:

- The copybook members that like-source components or other copybooks reference.
- The names of subroutines that like-source components call.
- Program names, procedure names, and data set names that are referenced in JCL and procedure libraries.

You can include multiple library types and multiple applications in a single scan request.

> **Note**
>
> Unlike the impact analysis function, which obtains its information from relationship records in the impact analysis data space, the scan utility scans the members in target libraries looking for characters strings such as COPY or CALL to find copybook or subprogram references respectively.

Requesting a Scan

You can use the Library Scan utility to scan baseline libraries or a specific package’s staging libraries.

To request a scan:

1. Select Option 1 Build from the Primary Option Menu. In response, the **Build Options**

```
CMNBUILD                Build Options
Option ===> _______________________________________________

0 Dates         Display the installation calendar
1 Create        Create a new package
2 Update        Update package information
3 Forms         Create, update, approve or review online forms
4 Utility       Rename and scratch information
5 Checkout      Check out components from baseline or promotion
6 Stage         Stage, edit, browse and delete components
7 Audit         Audit a package
8 Recompile     Recompile source code from baseline or promotion
9 Rebind        Rebind load modules
B Browse        Browse, print, copy baseline or promotion
C Compare       Compare staging to baseline or promotion
L Listing       Browse compressed listings
S Scan          Scan baseline for character strings
Z Compress      Compress change package staging libraries
```

2. Select Option S Scan from the Build Options menu. The **Library Scan Utility** (CMNSCN01) panel appears.

```
CMNSCN01                    Library Scan Utility
Command ===> _______________________________________________________

Package . . . . . . . . . TEST001852_
Application . . . . . . . ___       (Blank for list)
Library type . . . . . .  JCL_      (Blank for list)
Scan mode . . . . . . . . 1_        (1-Online, 2-Batch, 3-Alt Batch)
Records to select . . . . 0_        (0 = all)

Enter "/" to select option
___ List member names only
___ Scan for dependencies
___ Case sensitive

Displacement:
    From . . . 0_ (0=all)           To . . . . . 0 (max=80)
Components to scan:                 (Blank for all components)
    Begin . . ._______________      End . . . .     ________________
Data strings:
cobol ________________________              OR_     (and/or)
  __________________________
```

The panel has the following fields:

Package

Enter the package name.

Application

Fill in the application name. You can:

- Specify a complete application name to restrict the scan to that application.
- Specify a pattern ending in an asterisk to request that all applications that match the pattern be included in the scan. For example, APP* will include all applications whose names begin with APP.
- Leave this field blank and press Enter to display the Application Selection List panel from which you can select the applications that you want to include in the scan.

Library type

Fill in the library type mnemonic. You can:

- Specify a specific library type to restrict the scan to that library type.
- Specify a pattern ending in an asterisk to request that all library types that match the pattern be included in the scan. For example, S* will include all library types whose mnemonics begin with the letter S.
- Leave this field blank and press Enter to display the Library Type Selection List panel from which you can select the library types that you want to include in the scan.

You cannot scan a like-load library type, even though they appear on the Library Type Selection List panel.

Scan mode

Specify whether the scan is executed online or in a batch job: See Online and Batch Execution Modes

- Enter `1` to perform the scan and display the data in an online mode.
- Enter `2` to submit a batch job for the scan and display the data in the SYSPRINT DD.
- Enter `3` to submit an alternative batch job where the scan activity takes place in the local machine rather than the server. |

Records to select

Enter the number of output records to display:

0: Display all records that match the scan criteria.

n: Display up to *n* records that have been retrieved.

This value determines the number of target records that are included in the displayed output. (It does not limit the number of record hits that scan retrieves.)

List member names only

Select this to specify to display only the library member names and associated data that meet the scan criteria, or omit to display the names of the library members and the data that meets the scan criteria.

Scan for dependencies

Specify whether or not to scan members for component dependencies or data strings.

Scan for component dependencies. The dependencies that are displayed depend on the library type, as follows:

Libtype	Dependencies Displayed
Like-copy	Copybook members with embedded copy statements.
Like-source	Copybooks that like-source members reference and the subroutines that like-source components call.
JCL or proc	Displays program names, procedure names, and data set names that are referenced in JCL and procedure library members.

You can scan an LCT library for data strings but not component dependencies.

Scan for a data string or strings. You need to supply a value for Data string as a minimum.

Selecting Dependencies is mutually exclusive with specifying a value for the Data Strings.

Case sensitive?

Select this to specify the scan for a data string is to be case sensitive exactly as typed, otherwise any case will be reported, upper, lower or mixed case.

Displacement

Enter the starting (From) and ending (To) character positions to search in each record. Type 7 for column 7, 36 for column 36, and so on. Type 0 in the *From and To* fields to search all positions in the record.

Components to scan

Enter the *Begin* and *End* member name to specify a range of members to scan:

- To scan all members, leave the *Begin* and *End* fields blank.

- To scan a single member, type its name in the *Begin* field. Leave the *End* field blank.
- To limit the search to members that match a pattern, type a pattern (such as `ABC*`) in the *Begin* field. ChangeMan ZMF ignores the *End* field value.

You can type a range, such as `MEMBERA` for *Begin* and `MEMBERZ` for *End* even if the members do not exist in the data set. All members that fall within the specified name range will be scanned.

Data strings

Enter the string or strings to search for.

Do not enclose a string containing embedded blanks or nonalphabetic characters in single quotes.

Enclose a string that contains leading or trailing blanks in single quotes.

These fields are ignored if you select Scan for dependencies.

and/or

If you specify a value for the second Data string, specify if you want the two Data strings to be ANDed or ORed.

Selecting Applications

The Application Selection List (CMNSCN04) panel is displayed if you leave the Application field blank on the preceding Library Scan Utility panel:

~~~~
```
CMNSCN04        Application Selection List             Row 1 to 3 of 3
Command ===>                                           Scroll ===> CSR

   Appl Description                     Request
__ ACTP ACTP Accounts Payable (Base ZMF)
__ COMM COMM Common Components (Base ZMF)
__ GENL GENL General Ledger (Db2 Option)
******************************* Bottom of data ********************************
```
~~~~

Note

Only the applications that your security system has authorized for your TSO userid are included in the application list.

Use this panel as follows:

1. Type the letter S in the selection field to the left of each application that you want to include in the scan. (If you make a selection error, you can type the letter D (deselect) in the selection field to deselect the application.)
2. When you have selected all desired applications, press Enter. (If you press PF3 at this point before you press Enter, no applications are selected and you are returned to the Library Scan Utility panel). After you press Enter, the text *SELECT* will appear in the REQUEST column to indicate that you have selected the corresponding application. The following example indicates that application ACTP has been selected.

```
CMNSCN04            Application Selection List          Row 1 to 3 of 3
Command ===> _________________________________________ Scroll ===> CSR

   Appl Description                             Request
__ ACTP ACTP Accounts Payable (Base ZMF)        *SELECT*
__ COMM COMM Common Components (Base ZMF)
__ GENL GENL General Ledger (Db2 Option)
******************************* Bottom of data ********************************
```

3. Press PF3 to return to the Library Scan Utility panel after you have selected the desired applications.

Selecting Library types

The Library Type Selection List (CMNSCN05) panel is displayed if you leave the Library Type field blank on the preceding Library Scan Utility panel:

```
CMNSCN05            Library Type Selection List         Row 1 to 26 of 26
Command ===> ___________________________________________   Scroll ===> CSR

   Lib Description                                  Request
__ CPY Copybooks
__ CP2 Copybooks for Utilities
__ CTC Control Statements
__ DBR DBRM
__ DOC Documentation
__ HTH zFS resident HTML
__ JAR Java Archives
__ JAV Java source
__ JCF Java Class files
__ JCL Execution JCL
__ JCT Java JAR Build Control
__ LCT Linkedit Control Cards
__ LOD Executable Load Modules
__ LOS Load for Subprograms to be Linked NCAL
__ LSH zFS Listings
__ LST Compressed Stage listings
__ OBJ Object module library
__ PRC Cataloged Procedures
__ SRC Source for Programs to be Linked Executable
__ SRS Source for subprograms to be Linked NCAL
__ TST Test Library type
__ WAR Java Web Archives
__ WCT Java WAR Build Control
__ ZCP Shared Baseline Components
__ ZLS Shared Baseline Subprogram Load (NCAL)
__ ZSS Shared Baseline Subprogram Source
******************************* Bottom of data ********************************
```

This panel lists, in alphanumeric order, all library types that are defined for all the applications you have selected. If the same library type is defined in more than one application, only one entry (the first one encountered) for the library type is included in the list.

Use this panel as follows:

1. Type the letter S in the selection field to the left of each library type that you want to include in the scan.
2. When you have selected all desired library types, press Enter. (If you press PF3 at this point before you press Enter, no library types are selected and you are returned to the Library Scan Utility panel.) After you press enter, the text *SELECT* appears in the REQUEST column to indicate that you have selected the corresponding library type. The following example indicates that all library types have been selected.

```
CMNSCN05           Library Type Selection List            Row 1 to 26 of 26
Command ===>_______________________________________________ Scroll ===> CSR

   Lib Description                                        Request
__ CPY Copybooks                                          *SELECT*
__ CP2 Copybooks for Utilities
__ CTC Control Statements
__ DBR DBRM
__ DOC Documentation
__ HTH zFS resident HTML
__ JAR Java Archives
__ JAV Java source
__ JCF Java Class files
__ JCL Execution JCL
__ JCT Java JAR Build Control
__ LCT Linkedit Control Cards
__ LOD Executable Load Modules
__ LOS Load for Subprograms to be Linked NCAL
__ LSH zFS Listings
__ LST Compressed Stage listings
__ OBJ Object module library
__ PRC Cataloged Procedures
__ SRC Source for Programs to be Linked Executable        *SELECT*
__ SRS Source for subprograms to be Linked NCAL           *SELECT*
__ TST Test Library type
__ WAR Java Web Archives
__ WCT Java WAR Build Control
__ ZCP Shared Baseline Components
__ ZLS Shared Baseline Subprogram Load (NCAL)
__ ZSS Shared Baseline Subprogram Source
******************************* Bottom of data ********************************
```

Use Case Scenarios

The use cases in this section illustrate how to use the library scan facility.

Scanning for a Data String

In this scenario, assume you know that several jobs access a data set named PROD.DATA.SET.NAME for input. You want to change the name of the data set that these jobs use for input. Therefore, you use the scan facility to scan all JCL and procedure libraries for references to this data set. Take the following steps:

1. Select Option 1 Build from the Primary Option Menu. Then, select Option S Scan on the Build panel to bring up the Library Scan Utility (CMNSCN01) panel.
2. Fill in the Library Scan Utility panel as shown below:
   - Leave the Application field blank because you want to include the libraries of multiple applications in the scan.
   - Leave the Library Type field blank because you want to include multiple library types in the scan.

- Do not select Dependencies, because you want to scan the selected library types for a data string ACPCPYC. (You do not want to scan for component dependencies.)
- Specify ACPCPY for the first Data string.

```
CMNSCN01                  Library Scan Utility
Command ===> ____________________________________________________

Package . . . . . . . . . ________
Application . . . . . . . ______  (Blank for list)
Library type . . . . . .  _____   (Blank for list)
Scan mode . . . . . . . . 1_      (1-Online, 2-Batch, 3-Alt Batch)
Records to select . . . . 0____   (0 = all)

Enter "/" to select option
 __ List member names only
 __ Scan for dependencies
 /_ Case sensitive

Displacement:
From . . . 0_   (0=all)            To . . . . . 0_ (max=80)
Components to scan:                (Blank for all components)
  Begin . . . ______________________  End . . . .__________________
Data strings:
  ACPCPYC                                    OR (and/or)
_________________________________
_________________________________
```

3. Press Enter to bring up the Application Selection List (CMNSCN04) panel.
4. Type the letter S in the line command field to select the ACTP and GENL applications.

```
CMNSCN04                Application Selection List          Row 1 to 3 of 3
Command ===>_______________________________________________ Scroll ===> CSR


   Appl Description                             Request
s  ACTP ACTP Accounts Payable (Base ZMF)
__ COMM COMM Common Components (Base ZMF)
s  GENL GENL General Ledger (Db2 Option)
******************************* Bottom of data ********************************
```

5. Press Enter to complete the selection. The word *SELECT* will appear in the Request column for the selected applications.
6. Press PF3 (END) to display the Library Type Selection List (CMNSCN05) panel.
7. Type the letter S in the line command field to select the SRC and SRS libraries:

```
CMNSCN05            Library Type Selection List            Row 1 to 37 of 37
Command ===> ______________________________________________  Scroll ===> CSR

   Lib Description                     Request
__ CPY Copybooks
__ CP2 Copybooks for Utilities
__ CTC Control Statements
__ DBB Db2 BIND PLAN Commands
__ DBR DBRM
__ DOC Documentation
__ HTH zFS resident HTML
__ JAR Java Archives
__ JAV Java source
__ JCF Java Class files
__ JCL Execution JCL
__ JCT Java JAR Build Control
__ LCT Linkedit Control Cards
__ LDB Db2 Program Load
__ LOD Executable Load Modules
__ LOS Load for Subprograms to be Linked NCAL
__ LSH zFS Listings
__ LST Compressed Stage listings
__ OBJ Object module library
__ PKG Db2 Bind Package Commands
__ PRC Cataloged Procedures
__ SDB Db2 Program Source
__ SPD Db2 Stored Proc Definitions - Non-SQL
__ SPN Db2 Stored Proc Source - Native SQL
__ SPQ Db2 Stored Proc Source - SQL Language
s  SRC Source for Programs to be Linked Executable
s  SRS Source for subprograms to be Linked NCAL
__ STL Db2 Stored Proc Load Modules
__ STP Db2 Stored Proc Source - External Lan
__TRG Db2 Trigger Definitions
__ TST Test Library type
__UDF Db2 User-Defined Function Definitions
__ WAR Java Web Archives
__WCT Java WAR Build Control
__ ZCP Shared Baseline Components
__ZLS Shared Baseline Subprogram Load (NCAL)
__ ZSS Shared Baseline Subprogram Source
*******************************Bottom of data********************************
```

8. Press Enter to complete the selection. The word *SELECT* will appear in the Request column for the selected library types, and the last one selected will be at the top of the screen.
9. Press PF3 (END) to initiate the scan.

Here’s an excerpt from the scan report that is displayed online:

```
CMNSCN03        SCAN RESULTS                        ROW 1 TO 39 OF 47
Command ===>______________________________________    Scroll ===> CSR
********************************* TOP OF DATA **********************************
Appl=ACTP LibType=SRC Baseline=CMNTP.S6.V810.BASE.ACTP.SRC

<***MEMBER ACPSRCCA***>
        COPY ACPCPYCA.                                          000023
        COPY ACPCPYCB.                                          000024
- CALL 'ACPSRSCB' USING ACPCPYCB.                         000031

<***MEMBER ACPSRCCC***>
        COPY ACPCPYCC.                                          000018
        COPY ACPCPYCD.                                          000019
- CALL 'ACPSRSCD' USING ACPCPYCD.                         000026

<***MEMBER ACPSRCCE***>
        COPY ACPCPYCE.                                          000018
        COPY ACPCPYCF.                                          000019
- CALL 'ACPSRSCF' USING ACPCPYCF.                         000026

<***MEMBER DENISE***>
        COPY ACPCPYCA.                                          000023
        COPY ACPCPYCB.                                          000024
- CALL 'ACPSRSCB' USING ACPCPYCB.                         000031

<***MEMBER TESTCO01***>
- COPY ACPCPYCB.                                          000024
- CALL 'ACPSRSCB' USING ACPCPYCB.                         000031

***26 members scanned***
***5 member hits***
***14 record hits***

Appl=ACTP LibType=SRS Baseline=CMNTP.S6.V810.BASE.ACTP.SRS

***Scan found no matches for this criteria***

Appl=GENL LibType=SRC Baseline=CMNTP.S6.V810.BASE.GENL.SRC

***Scan found no matches for this criteria***

Appl=GENL LibType=SRS Baseline=CMNTP.S6.V810.BASE.GENL.SRS

***Scan found no matches for this criteria***

***1.74 seconds elapsed time
******************************** BOTTOM OF DATA ********************************
```

A separate section for each application/libtype is displayed in the scan output in application/libtype sequence.

Each section has the following components:

- Header that names the application, library type, and data set name of the target baseline library.
- Name of each member that has a match for the specified scan criteria.

- The statement or statements within each library member in which the specified data string was found. (These details are omitted if you selected the option to *List member names only* on the Library Scan Utility panel).
- Statistics that show:
- The number of members in the target application/library type that were scanned.
- The number of members that have a match for the specified scan criteria.
- The number of matching records displayed in the report output (not shown in the above excerpt).

The excerpt in the above example shows that six members of the SRC library type for the ACTP application reference the data string ACPCPYC.

If a specified library type does not exist for an application or if there are no matches in the target library type for the specified scan criteria, the following message is displayed for that application/libtype:

```
    *** Scan found no matches for this criteria ***
```

The end of the report shows the elapsed time that the scan utility used to complete the scan request.

Listing Members Only in Scan Output

The example in this scenario is the same as in the preceding scenario except that you request that the scan output list only member names and not the associated data.

1. Fill in the Library Scan Utility panel as shown below:
   - Leave the Application field blank because you want to select the libraries of multiple applications.
   - Leave the Library Type field blank because you want to request that multiple library types be included in the scan.
   - Select the option *List member names only*.
   - Do not select *Dependencies*, because you want to scan the selected library types for the data string ACPCPYC. (You do not want to scan for component dependencies.)
   - Specify ACPCPYC for the first Data string.

```
CMNSCN01                Library Scan Utility
Command ===> ________________________________________________________

Package . .. . . . . . . ______
Application . . . . . . .______ (Blank for list)
Library type . . . . . . ______ (Blank for list)
Scan mode . . . . . . .  1_     (1-Online, 2-Batch, 3-Alt Batch)
Records to select . . .  0_____ (0 = all)

Enter "/" to select option
 /_ List member names only
 __ Scan for dependencies
 /_ Case sensitive

Displacement:
   From . . . 0__(0=all)       To . . . . . 0__ (max=80)
Components to scan:            (Blank for all components)
   Begin . . .________________ End . . . .__________________
Data strings:
ACPCPYC___________________________    OR__ (and/or)
__________________________________
```

2. Press Enter to bring up the Application Selection List (CMNSCN04) panel.

3. Enter the letter S in the line command field to select the ACTP application. (If you select the wrong application by mistake, you can type the letter D in the line comand field to deselect the application.)

```
CMNSCN04     A  pplication Selection List           Row 1 to 3 of 3
Command ===> _____________________________________ Scroll ===> CSR

Appl   Description Request
s ACTP ACTP Accounts Payable (Base ZMF)
__COMM COMM Common Components (Base ZMF)
__GENL GENL General Ledger (Db2 Option)
*******************************Bottom of data********************************
```

4. Press Enter to complete the selection. The word *SELECT* will appear in the Request column for the selected applications.

5. Press PF3 (END) to display the Library Type Selection List (CMNSCN05) panel.

6. Enter the letter S in the line command field to select the libraries you want.

```
CMNSCN05                Library Type Selection List         Row 1 to 26 of 26
Command ===>________________________________________________ Scroll ===> CSR

   Lib Description                         Request
__ CPY Copybooks
__ CP2 Copybooks for Utilities
__ CTC Control Statements
__ DBR DBRM
__ DOC Documentation
__ HTH zFS resident HTML
__ JAR Java Archives
__ JAV Java source
__ JCF Java Class files
__ JCL Execution JCL
__ JCT Java JAR Build Control
__ LCT Linkedit Control Cards
__ LOD Executable Load Modules
__ LOS Load for Subprograms to be Linked NCAL
__ LSH zFS Listings
__ LST Compressed Stage listings
__ OBJ Object module library
__ PRC Cataloged Procedures
s  SRC Source for Programs to be Linked Executable
__ SRS Source for subprograms to be Linked NCAL
__ TST Test Library type
__ WAR Java Web Archives
__ WCT Java WAR Build Control
__ ZCP Shared Baseline Components
__ ZLS Shared Baseline Subprogram Load (NCAL)
__ ZSS Shared Baseline Subprogram Source
******************************* Bottom of data ********************************
```

7. Press Enter to complete the selection. The word *SELECT* will appear in the Request column for the selected library types.

8. Press PF3 (END) to initiate the scan.

Here’s the output from the scan report that is displayed online:

```
CMNSCN03             SCAN RESULTS           ROW 1 TO 20 OF 20
Command ===>_______________________________  Scroll ===> CSR
********************************* TOP OF DATA **********************************
Appl=ACTP LibType=SRC Baseline=CMNTP.S6.V810.BASE.ACTP.SRC

***Member list only***

ACPSRCCA
ACPSRCCC
ACPSRCCE
TEST01
TESTC001
```

```
Appl=ACTP LibType=SRS Baseline=CMNTP.S6.V810.BASE.ACTP.SRS

***Scan found no matches for this criteria***

***0 members scanned***
***0 member hits***

*** 0.72 seconds elapsed time

******************************** BOTTOM OF DATA********************************
```

Note that only the members with the specified data string (but no data records) are listed in the scan output.

Scanning for Dependencies

The following scenario shows how to scan baseline libraries for copybook-to-copybook, source-to-copybook, and source-to-subprogram dependencies. In the scenario, you restrict the scope to the scan to the CPY, SRC, and SRS (like-source) library types.

1. Fill in the Library Scan Utility panel as follows:
   - Specify GENL for the Application.
   - Leave the Library Type blank to display a list of library types from which you can select.
   - Select Scan for dependencies.
   - Omit any Data string values because the data string scan is mutually exclusive with the component dependency scan:

```
 CMNSCN01                     Library Scan Utility
 Command ===>______________________________________________

 Package . . . . . . .. .. _____________
 Application . . . . . . . GENL      (Blank for list)
 Library type . . . . . .  _____     (Blank for list)
 Scan mode . . . . . . . . 1         (1-Online, 2-Batch, 3-Alt Batch)
 Records to select . . . . 0____     (0 = all)

 Enter "/" to select option
__ List member names only
/_ Scan for dependencies
/_  Case sensitive

 Displacement:
From . . . 0 (0=all)        To . . . . . __(max=80)
 Components to scan:           (Blank for all components)
Begin . . . __________      End . . . . _____________
 Data strings:
 ___________________________________ OR (and/or/bth)
 ___________________________________
```

2. Press Enter. The Library Type Selection List (CMNSCN05) panel appears. You select the CPY, SRC, and SRS library types because you want to see the copybooks that are referenced by

members of these library types and the subprograms that members of the SRC and SRS library types call.

```
            Library Type Selection List             Row 1 to 26 of 29
Command ===> ______________________________________   Scroll ===> CSR

   Lib Description                                      Request
s  CPY Copybooks
__ CTC Control Statements
__ DBB Db2 BIND PLAN Commands
__ DBR Db2 DBRM
__ DOC Documentation
__ JCL Execution JCL
__ LCT Linkedit Control Cards
__ LDB Db2 Program Load
__ LOD Executable Load Modules
__ LOS Load for Subprograms to be LinkedNCAL
__ LST Compressed Stage listings
__ PKG Db2 Bind Package Commands
__ PRC Cataloged Procedures
__ SDB Db2 Program Source
__ SPD Db2 Stored Proc Definitions - Non-SQL
__ SPN Db2 Stored Proc Source - Native SQL
__ SPQ Db2 Stored Proc Source - SQL Language
s  SRC Source for Programs to be Linked Executable
s  SRS Source for subprograms to be Linked N
__ STL Db2 Stored Proc Load Modules
__ STP Db2 Stored Proc Source - External Lan
__ TRG Db2 Trigger Definitions
__ UDF Db2 User-Defined Function Definitions
__ ZCP Shared Baseline Components
__ ZLS Shared Baseline Subprogram Load (NCAL)
__ ZSS Shared Baseline Subprogram Source
******************************* Bottom of data *****************************
```

3. Press Enter. The text *SELECT* will appear in the Request column for the selected library types.
4. Press PF3 (END) to initiate the scan request.

Here is an example of scan output that is displayed online:

```
Appl=GENL LibType=CPY Baseline=CMNTP.S6.V810.BASE.GENL.CPY

<***MEMBER GNLCPY1A***>
COPY        GNLCPY1X                                            000005

<***MEMBER GNLCPY5A***>
COPY        GNLCPY1X                                            000005

***11 members scanned***
***2 member hits***
***2 record hits***

Appl=GENL LibType=SRC Baseline=CMNTP.S6.V810.BASE.GENL.SRC
```

```
<***MEMBER COBOL002***>
COPY        CPY001                                              000011

<***MEMBER GNLSRC1A***>
COPY        GNLCPY00                                            000016
COPY        GNLCPY1A                                            000017
COPY        GNLCPY1B                                            000018
CALL        GNLSRS00                                            000024
CALL        GNLSRS1B                                            000025

<***MEMBER GNLSRC99***>
COPY        GNLCPY99                                            000016

***4 members scanned***
***3 member hits***
***7 record hits***

Appl=GENL LibType=SRS Baseline=CMNTP.S6.V810.BASE.GENL.SRS

<***MEMBER GNLSRS00***>
COPY        GNLCPY00                                            000018

<***MEMBER GNLSRS01***>
COPY        GNLCPY01                                            000018

<***MEMBER GNLSRS1B***>
COPY        GNLCPY00                                            000019
COPY        GNLCPY1C                                            000020
COPY        GNLCPY1B                                            000022
CALL        GNLSRS00                                            000026
CALL        GNLSRS1C                                            000027

<***MEMBER GNLSRS1C***>
COPY        GNLCPY00                                            000018
COPY        GNLCPY1C                                            000020
CALL        GNLSRS00                                            000024

<***MEMBER GNLSRS5A***>
COPY        GNLCPY00                                            000017
COPY        GNLCPY5A                                            000018
COPY        GNLCPY5B                                            000019
CALL        GNLSRS00                                            000025
CALL        GNLSRS5B                                            000026

<***MEMBER GNLSRS5B***>
COPY        GNLCPY00                                            000017
COPY        GNLCPY5C                                            000018
COPY        GNLCPY5B                                            000020
CALL        GNLSRS00                                            000024
CALL        GNLSRS5C                                            000025

<***MEMBER GNLSRS5C***>
COPY        GNLCPY00                                            000017
COPY        GNLCPY5C                                            000019
CALL        GNLSRS00                                            000023

***7 members scanned***
***7 member hits***
***23 record hits***

***1.31 seconds elapsed time
*
```

The grouping shown in **bold** in the above example shows that:

> Member GNLSRC1A of the SRC library type for the GENL application references the GNLCPY00, GNLCPY1A and GNLCPY1B copybooks, and calls two subprograms GNLSRS00 and GNLSRS1B.

Requesting a Batch Scan

This scenario simply looks for SRS dependencies except that you submit the request for execution as a batch job. You can use the Library Scan utility to scan baseline libraries or package staging libraries.

1. Fill in the Library Scan Utility panel as per example below, ensuring that you change the Scan mode from 1 to 2:

```
CMNSCN01                    Library Scan Utility
Command ===>

Package . . . . . . . . .   _____________
Application . . . . . . .   ACTP_    (Blank for list)
Library type . . . . . .    SRS_     (Blank for list)
Scan mode . . . . . . . .   2_       (1-Online, 2-Batch, 3-Alt Batch)
Records to select . . . .   0____    (0 = all)

Enter "/" to select option
  __ List member names only
  /_ Scan for dependencies
  /_ Case sensitive

Displacement:
  From . . . 0_ (0=all)                To . . . . 0__ (max=80)
Components to scan:                   (Blank for all components)
  Begin . . .                         End . . . .
Data strings:
    ________________________________       OR_ (and/or)
    ________________________________
```

2. Press Enter to bring up the Batch Baseline Scan Job Statements panel (CMNSCN02). This panel shows:

   - The application that you selected in the Application field (If you selected more than one then only the first one will be shown).
   - The library type that you selected in the Library Type field. (If you selected more than one then the first one will be shown).
   - Job statements for the Batch Baseline Scan job.

```
                Batch Baseline Scan Job Statements
Command ===>__________________________________________

Application: ACTP
Library type: CPY

Job statement information:
//USER015A JOB (SM-1IKF-SM),'SCAN',
//      CLASS=A,MSGCLASS=X,NOTIFY=USER015
//*
//* JOB TO SCAN
```

3. Press Enter to submit the job.

Scan writes the output for the job to the SYSPRINT DD. You can use a facility such as the Spool Display and Search Facility (SDSF) to view the output.

Batch Output

The output from a batch scan appears in the SYSPRINT data set. Here’s an example:

```
ChangeMan(R) ZMF        CMNBSCAN - 8.1.0 2015/01/30 03:38:18
Session established with ChangeMan ZMF Started task.

Criteria for this Scan:
Application              :ACTP
Library Type            :SRS
Scan for Dependencies
Case Sensitive
Records Selected        :ALL
Starting displacement   :1
Ending displacement     :80
Starting member         :First Member
Ending member           :Last Member
Dataset Organization    :PDS
Dataset Name to Scan    :CMNTP.S6.V810.BASE.ACTP.SRS

<***MEMBER ACPSRS00***>
COPY            ACPCPY00                                    000018

<***MEMBER ACPSRS1B***>
COPY            ACPCPY00                                    000016

COPY            ACPCPY1C                                    000017

COPY            ACPCPY1B                                    000019

CALL            ACPSRS00                                    000023

CALL            ACPSRS1C                                    000024

<***MEMBER ACPSRS1C***>
COPY            ACPCPY00                                    000014

COPY            ACPCPY1C                                    000016
CALL            ACPSRS00                                    000019

***3 members scanned***
***3 member hits***
***9 record hits***
***0.07 seconds elapsed time***

Session terminated with ChangeMan ZMF Started task.
******************************** BOTTOM OF DATA********************************
```

The batch output contains a separate section for each application/library type that is included in the scan. Each section includes:

- Header that identifies the parameters used to scan the target application/library type.
- Name of each library member where a hit occurs.
- The details of the hit (omitted if List Members Only is selected).
- Statistics that identify:
- The number of members scanned in the target application/library type.
- The number of member hits.
- The total number of record hits in the members.

- The elapsed time taken to complete this portion of the scan request.

Reviewing Batch SYSIN Input

You can review the SYSIN input to the scan utility to confirm the parameters used when the job executes. For example, you can use the SJ line command in SDSF to display the job stream,

```
Display Filter View Print Options Search Help
------------------------------------------------------------------------------

SDSF STATUS DISPLAY ALL CLASSES                 LINE 1-5 (5)
COMMAND INPUT ===>____________________________________ SCROLL ===> CSR
PREFIX=USER015* DEST=(ALL) OWNER=* SORT=Pos/D SYSNAME=
NP JOBNAME  JobID    Prty Queue C Pos Max-RC   Owner SAff ASys
sj USER015A J0553728 1   PRINT A 3908 CC 0000 SERT
__ USER015A J0551219 1   PRINT A 1514 CC 0000 SERT
__ USER015A J0551201 1   PRINT A 1503 CC 0000 SERT
__ USER015A J0551062 1   PRINT A 1400 CC 0000 SERT
```

and page down to the SYSIN input for the job. (This excerpt shows the single step BATSCAN and the SYSIN input data for the job described in "Requesting a Batch Scan".

```
//BATSCAN EXEC PGM=CMNBSCAN, ***DATASET SCAN
//          PARM='SUBSYS=6,USER=USER015'
//*)IM CMN$$SPR
//SER#PARM DD   DISP=SHR,DSN=CMNTP.SER810.C6.TCPIPORT
//ABNLIGNR DD   DUMMY
//SYSABEND DD   SYSOUT=*
//SYSPRINT DD   SYSOUT=*
//SYSIN DD *
APL=ACTP
LIB=SRS
DPN=YES
LST=NO
REC=0
CAS=YES
DPS=0
DPE=0
MB1=
MB2=
ST1=
AOR=OR
ST2=
**************************** Bottom of Data********
```

SYSIN data corresponds to the information you supply on the scan utility panels when you request the scan, as follows:

SYSIN Parameter	Corresponding Field on Library Scan Utility Panels	**Description
APL	Application	The application you specified on the Library Scan Utility (CMNSCN01) panel or that you selected from the list of applications on the Application Selection List (CMNSCN04) panel. A separate APL parameter is included in the SYSIN input stream for each application that you select. The above example shows that the ACTP, COMM, and GENL applications are included in the scan.
LIB	Library Type	The library type you specified on the Library Scan Utility (CMNSCN01) panel or that you selected from the list of library types on the Library Type Selection List (CMNSCN05) panel. A separate LIB parameter is included in the SYSIN input stream for each library type that you select. The example above shows that the JCL and PRC library types are included in the scan. The baseline libraries associated with the JCL and PRC library types for each application are scanned.
DPN	Scan for dependencies	Whether to scan for dependencies (YES or NO).
LST	List member names only	Whether to list members only (YES or NO).
REC	Records to select	Number of output records to display in the scan report: *n* The first *n* output records. 0 All output records.
CAS	Case Sensitive	Specifies whether the search for the specified data strings is case sensitive: YES Search for a string exactly as you typed it. NO Find all occurrences of a string, whether in upper, lower, or mixed case.
DPS	Displacement From	The column in which to begin the search for the specified string.
DPE	Displacement To	The column in which to end the search for the specified string.
MB1	Components To Scan Begin	The name of the beginning member in a range of members to be scanned.
MB2	Components To Scan End	The name of the ending member in a range of members to be scanned.
ST1	Data Strings	The first string to scan for.
AOR	AND/OR	AND or OR boolean operator.
ST2	Data strings (second line)	The second string to scan for.

25. Comparing Package Components to Promotion or Baseline

This chapter describes the compare function that shows differences between a package component and same component in a promotion or baseline library.

About Compare Component

The compare component function produces a report that shows the differences between a component in a package staging library and the same component in a promotion or baseline library.

You can use this function to:

- See what changes you have made to a component since you checked it out from baseline.
- Help you decide whether to overlay a component with the same name in a promotion library.
- Help resolve a SYNCH10! condition where the baseline version of a component has changed since you checked the component out.

Accessing Compare Component Panels

Use one of these methods to display the **Compare** panel:

- **Using the Menu Hierarchy:**
  1. On the **Primary Option Menu**, select **1 Build**.
  2. On the **Build Option** panel, select **C Compare**.
- **Using Direct Panel Access:**

  On a ChangeMan ZMF panel, type **=1.C** in the **Command** or **Option** line and press **Enter**.
- **Using the Change Package List:**

  On the **Change Package List** panel, type **CC** in the line command for a change package and press **Enter**.

Specifying Components to Compare

To compare one or more package components to the same component in a promotion or baseline library, bring up the **Compare** panel (CMNCOMP1). See Accessing Compare Component Panels.

```
CMNCOMP1                            Compare
Command ===> _______________________________________________________

Compare mode . . . . 1_           (1-Online; 2-Batch)
Package . . . . . .  TEST001852_
Component . . . . . ___________________________________________________________ +
                                  (* for all members; blank for list)
Text type . . . . . ____          (text type to be compared)
Library type . . .  JCL_          (Blank for list)
Source library . .  0_            (Baseline 0 to -n; Promotion 1 to n)
Data Encoding . . . __            (1-ASCII, 2-UTF-8)

Enter "/" to select option
  _ Prompt for report disposition

Job statement information for batch compare:
 //WSER73T1 JOB (0),'L TEST CMP .  ',___________________________________________
 // NOTIFY=WSER73,CLASS=D,MSGCLASS=X____________________________________________
 //* ___________________________________________________________________________
 //* ___________________________________________________________________________
```

This table describes the fields on the **Compare** panel.

Field	Description
Command	C: **Compare the component specified in the** COMPONENT NAME **field.** Blank:** Display selection list of components in the staging library.
Compare mode	1:** Execute compare online.
	2:** Execute compare in a batch job.
Package	Type the ID of the package that contains the components to be compared to a promotion or baseline library.
Component	Type a component name. Leave this field blank or type a pattern for a selection list of components in the staging library. If the specified **LIBRARY TYPE** is an zFS type, then the text in the **COMPONENT NAME** field is processed exactly as entered; otherwise the text is folded into upper case before processing. **Note:** Component names are truncated to 160 characters in the compare component function.
Data Encoding	This option can be used to view components stored using a CCSID representing ASCII or UTF-8 characters. This option has no effect when using browse.
	1 - Enables ASCII conversion for ISPF view.
	2 - Enables UTF-8 conversion for ISPF view.

Field	Description
Text type	Type a value to specify how the text in the component should be compared. Default: **$**. (dollar sign and period) This is usually sufficient to get valid compare results. General Types: **$:** Prefix to any **TEXT TYPE** to force a position-by-position comparison and to flag lines as different even if the only differences are the positions of spaces and commas. Example: $JCL . (period) Automatically set **TEXT TYPE** from an analysis of the first four records. Differences in spacing and commas are ignored as described in Specific Types below. **Specific Types:** If **TEXT TYPE** is ALC, BAL, C, CLIST, JCL, FORTRAN, NATURAL, PASCAL, PL/1, PL/I, PL1, PLI, REXX, or RPG, then spaces are squeezed out as not significant. If **TEXT TYPE** is COBOL, then spaces and commas are squeezed out as not significant. Only positions 7 through 72 of the record are compared. If **TEXT TYPE** is PANEL, REPORT, or SCRIPT, then no characters are squeezed out.
Library type	Type the library type of the components you want to compare. Leave this field blank for a selection list of library types in the package.
Source library	Type a number to indicate the baseline version or promotion level to compare to. **-1 to -999:** Compare to a prior version in a baseline library. The negative number represents a relative prior baseline version. For example, -1 means the version immediately prior to the current version. **0:** Compare to the current (0-level) baseline library. <br. **+1 to +99:** Compare to a version in a promotion library. The positive number is a promotion level number in any promotion site. For example, +10 means promotion level 10 in any promotion site. **Blank:** Display a selection list of baseline and promotion libraries that contain the specified component. **Note:** The **COMPONENT NAME** field cannot be blank or a pattern if you want a list of libraries.

Field	Description
Prompt for report disposition	If you specify **1** Online for **COMPARE MODE:** Select this option to display the **Specify Report Disposition** panel after the comparison report is displayed in an ISPF panel. If not selected, then **suppress** the **Specify Report Disposition** panel after the comparison report is displayed in an ISPF panel.
Job statement information for batch compare	If you specified **2** Batch for **COMPARE MODE**, type job statement information for the batch job that ChangeMan ZMF submits to compare components.

If you type **C** on the **Command** line, a full component name in the **Component** field, and type an entry in every other field on the **Compare** panel, the compare component function executes directly (unless you are comparing a package component to a promoted component). Go to Compare Component Results.

Otherwise more panels are displayed so you can choose the library to compare to and choose the staging library component to compare:

- If you specify a promotion level number (+nn) in the **Source library** field, the **Promotion Library Selection List** panel (CMNCPLSL) is displayed so that you can select the promotion library that you want. See Choosing a Promotion Library.
- If you specify a full **Component** name (not a pattern) and leave the **Source library** field blank, the **component.type Library List** panel (CMNCMLSL) is displayed so that you can select the baseline level or promotion library that you want. See Choosing a Promotion or Baseline Library.
- If you leave the **Command** line blank or if you type a pattern in the **Component** field, the **Compare From libtype - stagelib** panel (CMNCOMPL) is displayed, which lists components in the package staging library for the library type you specified. This panel is displayed after you select a promotion or baseline library if your entry in the **Source library** field sends you to other panels to make that choice. See Choosing a Package Component to Compare.

Choosing a Promotion Library

On the **Compare** panel, if you specify a promotion level number (+nn) in the **Source library** field, the **Promotion Library Selection List** panel (CMNCPLSL) is displayed.

```
CMNCPLSL               Promotion Library Selection List             Row 1 to 2 of 2
Command ===>                                                       Scroll ===> CSR

   Library type: SRC          Promotion level: +10
   Dataset                                     Name             Site
__ CMNTP.S6.V810.PROM.S6P1UT.SRC             + S6P1UT           SERT6
__ CMNTP.S6.V810.PROM.S6P1UT1.SRC            + S6P1UT1          SERT6P1
******************************* Bottom of data ********************************
```

This panel displays all of the promotion libraries that are defined in application administration for the specified library type and promotion level number, even if the package or the component you want to compare is not promoted to this level. More than one library is listed if the specified promotion level number is used in multiple promotion sites.

This table describes the fields on the **Compare From libtype - stagelib** panel.

Field	Description
Command	Type one of the following commands, or leave the command line blank and type a line command next to a library name. **Cancel:** Cancel the function and return to the previous panel. Abbreviation: C or CAN **Refresh:** Refresh the list of libraries that are displayed. Abbreviation: R
Library type	Displays the library type of the promotion library.
Promotion level	Displays the promotion level number.
Line Command	Type **S** to the left of the **DATASET** column to select a library.
Dataset	Displays the library name.
Name	Displays the promotion level nickname.
Site	Displays the promotion site.

Type **S** in the line command for a library and press **Enter**.

Choosing a Promotion or Baseline Library

On the **Compare** panel, if you specify a full **Component** name (not a pattern) and leave the **Source library** field blank, the **component.type Library List** panel (CMNCMLSL) is displayed.

```
CMNCMLSL               ACPSRC6A.SRC Library List                Row 1 to 4 of 4
Command ===>                                                  Scroll ===> CSR

   Lvl    Dataset/pathname                     + Prom.name           Site
__ 0000   CMNTP.S6.V810.BASE.ACTP.SRC            BASELINE
__ -001   CMNTP.S6.V810.BASE.ACTP.SRC.DELTA      BASDELTA
__ -002   CMNTP.S6.V810.BASE.ACTP.SRC.DELTA      BASDELTA
__ +010   CMNTP.S6.V810.PROM.S6P1UT.SRC          S6P1UT               SERT6
******************************* Bottom of data ********************************
```

This panel displays all of the application baseline and promotion libraries that contain the specified component.

This table describes the fields on the *component*.**type Library List** panel.

Field	Description
Command	Type one of the following commands, or leave the command line blank and type a line command next to a library name. **CANCEL:** Cancel the function and return to the previous panel. Abbreviation: C or CAN **REFRESH:** Refresh the list of libraries that are displayed. Abbreviation: R
Line Command	Type **S** to the left of the **LVL** column to select a library.
Lvl	Displays the relative baseline level or the promotion level number for the library.
Dataset/ pathname	Displays the library name or zFS path and file name.
Prom.name	Displays one of the following: - pppppppp: Promotion level nickname - BASELINE: Indicates a baseline library - BASEDELTA Indicates a stacked reverse delta baseline library
Site	Displays the promotion site.

Type **S** in the line command for a library and press **Enter**.

Choosing a Package Component to Compare

On the **Compare** panel, if you leave the **Command** line blank or if you type a pattern in the **Component** name field, the **Compare From libtype - stagelib** panel (CMNCOMPL) is displayed.

This panel lists components in the package staging library for the library type you specified.

```
CMNCOMPL Compare from SRC Row 1 to 4 of 4
Command ===> _________________________________________Scroll ===> CSR
    Staging library:
    CMNTP.S6.ACTP.STG6.#000038
___ Name Function vv.mm Created Changed Size Init User
___ ACPSRCCE 03.04 2002/05/07 2015/01/21 22:45 30 23 USER016
___ ACPSRC1A 02.02 2014/12/18 2015/01/21 20:59 33 34 USER017
___ ACPSRC6A 03.02 2012/09/26 2015/02/01 19:45 44 27 USER017
___ ACPSRC92 03.01 2002/05/07 2015/01/15 00:06 30 15 USER017
******************************* Bottom of data *******************************
```

This table describes the fields on the **Compare From** *libtype - stagelib* panel.

Field	Description
Command	Type one of the following commands, or leave the command line blank and type a line command next to a component name. **BROWSE:** Browse the specified component. Abbreviation: B *component* **CANCEL:** Cancel the function and return to the previous panel. Abbreviation: C or CAN **HIST:** Display component history for the specified *component* component. Abbreviation: H LOCATE Locate a component by information in the last sorted *component* column or by Name if the components were not sorted. Abbreviation: L **REFRESH:** Display updated information on this panel. Abbreviation: R **SELECT:** Select the specified component for compare. *component* Abbreviation: SE **SORT:** Sort the listed components by information under the *heading* specified column heading. Abbreviation: SO **VIEW:** View (browse in edit) the specified component. *component* Abbreviation: V **Note:** Component functions such as Browse, Hist, and View are executed against the staging library component, not the component in the promotion or baseline library.

Field	Description
Line Command	Type a line command to the left of the **Name** row. **B:** Browse the component. **D:** Deselect the component from compare. H Display component history. S Select the component for compare. V View (browse in edit) the component. **Note:** Component functions such as Browse, Hist, and View are executed against the staging library component, not the component in the promotion or baseline library.
Name	Displays the component name.
Function	Displays the function that was executed for each component: * BROWSE DESELECT * HISTORY * VIEW * COMPARE * ERROR
vv.mm Created Changed Size Init User	These fields display the ISPF statistics in the staging library directory for each listed member.

Type a line command for one or more components and press **Enter**. How the line commands are processed depends on whether you specified **1** Online or **2** Batch in the **Compare mode** field on the **Compare** panel.

Online Processing for Line Commands

If you specified **1** Online on the **Compare** panel, line commands on the **Compare From** *libtype - stagelib* panel are processed as follows:

1. Each line command on the **Compare From libtype - stagelib** panel is processed serially, top to bottom.
2. The result for each line command is displayed in an ISPF panel. See Online Compare Mode Results for an example of an online comparison report.
3. To exit a result panel and show the next result panel, press **PF-3**, or type **END** on the **Command** line and press **Enter**.

Batch Processing for Line Commands

If you specified **2** Batch on the **Compare** panel, line commands on the **Compare From** *libtype - stagelib* panel are processed as follows:

1. Each line command on the **Compare From libtype - stagelib** panel *except* **S** *Select* is processed serially, top to bottom.
2. The result for each non-**S** line command is displayed in an ISPF panel.
3. To exit a result panel and show the next result panel, press **PF-3**, or type **END** on the **Command** line and press **Enter**.
4. After all non-**S** line command are processed, you are returned to the **Compare From libtype - stagelib** panel. The FUNCTION column shows the function that was requested for each component.
5. When you press **PF-3**, or type **END** on the **Command** line and press **Enter**, all **S** Select line commands are processed, and JCL for a batch comparison job is generated and submitted. See Batch Compare Mode Results for an example of a batch comparison report.

Compare Component Results

The format of the compare component results depends on what you specified in the

COMPARE MODE field of the **Compare** panel (CMNCOMP1).

- If you specified **1** Online, see Online Compare Mode Results.
- If you specified **2** Batch, see Batch Compare Mode Results.

Online Compare Mode Results

On the **Compare** panel, if you specified **1** Online in the **COMPARE MODE** field, the comparison report is displayed in an ISPF panel:

```
 -----------------------------------------------------------------------------------------------------------------
 BROWSE CMNTP.A009E.#CE72391.#590E452.OUTLIST
 Command ===>
************************************************************************ Top of Data *****************************
       S E R C M P A R (MVS - 871 - 20140828) 2 TEXTONLY SUNDAY FEBRUARY 1, 2015 (2015/032) 20:21:09 PAGE 1
SYSUT1=CMNTP.S6.V810.BASE.ACTP.SRC(ACPSRC6A),SYSUT2=CMNTP.S6.ACTP.STG6.#000038.SRC(ACPSRC6A)
               IDENTIFICATION DIVISION.                                                 O N E 1
               PROGRAM-ID. ACPSRC6A.                                                    O N E 2
                                                                                        O N E 3
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>++.++++8+++++++++++++++++++++
I             *PACKAGE ACTP000038 S6.V810 DIF                                           T W O 4 +
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>++.++++8+++++++++++++++++++++
              *PACKAGE ACTP000001 S6.V810                                          O N E 4
              *PACKAGE ACTP000016 S4.V710T19                                       O N E 5
              *PACKAGE ACTP000045 S4.V710T19                                       O N E 6
              *PACKAGE ACTP000007 S4.V711                                          O N E 7
              *PACKAGE ACTP000080 S4.V71201T3                                      O N E 8
              *PACKAGE ACTP000081 S4.V71201T3                                      O N E 9
              *PACKAGE ACTP000082 S4.V71201T4                                      O N E 10
                                                                                   O N E 11
              ENVIRONMENT DIVISION.                                                O N E 12
              CONFIGURATION SECTION.                                               O N E 13
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>++.++++8++++++++++++++++++++
D             SOURCE-COMPUTER. IBM-370.                                        DIF O N E 14 +
--------|---.----1----.----2----.----3----.----4----.----5----.----6----.----7----.----8--------------------
I             SOURCE-COMPUTER. IBM-390.                                        DIF T W O 15 +
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>++.++++8++++++++++++++++++++

              OBJECT-COMPUTER. IBM-370.                                            O N E 15
                                                                                   O N E 16
              INPUT-OUTPUT SECTION.                                                O N E 17
              FILE-CONTROL.                                                        O N E 18
                                                                                   O N E 19
              DATA DIVISION.                                                       O N E 20
              WORKING-STORAGE SECTION.                                             O N E 21
                   COPY ACPCPY00.                                                  O N E 22
                   COPY ACPCPY6A.                                                  O N E 23
              LINKAGE SECTION.                                                     O N E 24
SER71I - END OF TEXT ON FILE SYSUT1
SER72I - END OF TEXT ON FILE SYSUT2
SER75I - RECORDS PROCESSED: SYSUT1(43)/SYSUT2(44),DIFFERENCES(1,0,1)
                            EXPLANATION - 1 RECORD DIFFER THAT SYNCHRONIZED TOGETHER
                                          0 RECORDS WERE CONSIDERED INSERTED ON SYSUT1
                                          1 RECORD WAS CONSIDERED INSERTED ON SYSUT2
        S E R C M P A R (MVS - 871 - 20140828) 2 TEXTONLY SUNDAY FEBRUARY 1, 2015 (2015/032) 20:21:09 PAGE 2
SYSUT1=CMNTP.S6.V810.BASE.ACTP.SRC,SYSUT2=CMNTP.S6.ACTP.STG6.#000038.SRC
SER71I - END OF DIRECTORY ON FILE SYSUT1
SER72I - END OF DIRECTORY ON FILE SYSUT2
SER78I - MEMBERS PROCESSED: SYSUT1(26)/SYSUT2(4),DIFFERENCES(1),REJECTED BY FILTERS: SYSUT1(25)/SYSUT2(3)
SER80I - TIME OF DAY AT END OF JOB: 20:21:09 - CONDITION CODE ON EXIT: 4
*********************************************************************** Bottom of Data ***************************
```

Navigate the compare report with standard PF keys.

To exit the comparison report, press **PF-3**, or type **END** on the **Command** line and press **Enter**. If you specified **Y** in the **Prompt for report disposition** field on the **Compare** panel, the **Specify Report Disposition** panel (CMNCDISP) is displayed.

```
CMNCDISP ----------------- Specify report disposition -------------------------
Command ===>________________________________________________

Report dataset name:      CMNTP.A0171.#C9E48A9.#8D00504.OUTLIST
Report disposition ===> 1      1 Print Dataset and delete
                               2 Print dataset and keep
                               3 Delete dataset without printing
                               4 Keep dataset without printing

Job statement information if printing:
===> //USER015A JOB (X170,374),'S4.V712',____________________________________
===> // CLASS=A,MSGCLASS=Y,NOTIFY=USER015____________________________
===> //*_____________________________________________________________________
===> //*______________________________
```

Follow the instructions on this panel to delete, print, and/or keep the comparison report. See Batch Compare Mode Results for an example of the printed comparison report.

If you selected multiple components for compare on the **Compare From libtype - stagelib** panel (CMNCOMPL), each comparison report is displayed in a separate panel in the order that the components were listed on the **Compare From** *libtype - stagelib* panel.

Batch Compare Mode Results

If you specified **2** Batch for **Compare mode** on the **Compare** panel, or if you specified a print **Report disposition** on the **Specify Report Disposition** panel, the comparison report is printed at the SYSPRINT DD statement in a batch job.

```
 S E R C M P A R   (MVS - 871 - 20140828) 2     TEXTONLY SUNDAY FEBRUARY 1, 2015 (2015/032)         20:28:02       PAGE 1
SYSUT1=CMNTP.S6.V810.BASE.ACTP.SRC(ACPSRC6A),SYSUT2=CMNTP.S6.ACTP.STG6.#000038.SRC(ACPSRC6A)
              IDENTIFICATION DIVISION.                                                   O N E 1
              PROGRAM-ID. ACPSRC6A.                                                      O N E 2
                                                                                         O N E 3
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>++.++++8++++++++++++++++++++
I             *PACKAGE ACTP000038 S6.V810                                            DIF T W O 4 +
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>++.++++8++++++++++++++++++++

              *PACKAGE ACTP000001 S6.V810                                                O N E 4
              *PACKAGE ACTP000016 S4.V710T19                                             O N E 5
              *PACKAGE ACTP000045 S4.V710T19                                             O N E 6
              *PACKAGE ACTP000007 S4.V711                                                O N E 7
              *PACKAGE ACTP000080 S4.V71201T3                                            O N E 8
              *PACKAGE ACTP000081 S4.V71201T3                                            O N E 9
              *PACKAGE ACTP000082 S4.V71201T4                                            O N E 10
                                                                                         O N E 11
              ENVIRONMENT DIVISION.                                                      O N E 12
              CONFIGURATION SECTION.                                                     O N E 13
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>++.++++8++++++++++++++++++++
D             SOURCE-COMPUTER. IBM-370.                                              DIF O N E 14 +
--------|---.----1----.----2----.----3----.----4----.----5----.----6----.----7----.----8--------------------
I             SOURCE-COMPUTER. TRS-80.                                               DIF T W O 15 +
++++++++|+++.+<++1++++.++++2++++.++++3++++.++++4++++.++++5++++.++++6++++.++++7+>++.++++8++++++++++++++++++++

              OBJECT-COMPUTER. IBM-370.                                                  O N E 15
                                                                                         O N E 16
              INPUT-OUTPUT SECTION.                                                      O N E 17
              FILE-CONTROL.                                                              O N E 18
                                                                                         O N E 19
              DATA DIVISION.                                                             O N E 20
              WORKING-STORAGE SECTION.                                                   O N E 21
                  COPY ACPCPY00.                                                         O N E 22
                  COPY ACPCPY6A.                                                         O N E 23
              LINKAGE SECTION.                                                           O N E 24
SER71I - END OF TEXT ON FILE SYSUT1
SER72I - END OF TEXT ON FILE SYSUT2
SER75I - RECORDS PROCESSED: SYSUT1(43)/SYSUT2(44),DIFFERENCES(1,0,1)
                           EXPLANATION - 1 RECORD DIFFER THAT SYNCHRONIZED TOGETHER
                                         0 RECORDS WERE CONSIDERED INSERTED ON SYSUT1
                                         1 RECORD WAS CONSIDERED INSERTED ON SYSUT2
         S E R C M P A R (MVS - 871 - 20140828) 2 TEXTONLY SUNDAY FEBRUARY 1, 2015 (2015/032) 20:28:02 PAGE 2
SYSUT1=CMNTP.S6.V810.BASE.ACTP.SRC,SYSUT2=CMNTP.S6.ACTP.STG6.#000038.SRC
SER71I - END OF DIRECTORY ON FILE SYSUT1
SER72I - END OF DIRECTORY ON FILE SYSUT2
SER78I - MEMBERS PROCESSED: SYSUT1(26)/SYSUT2(4),DIFFERENCES(1),REJECTED BY FILTERS: SYSUT1(25)/SYSUT2(3)
SER80I - TIME OF DAY AT END OF JOB: 20:28:02 - CONDITION CODE ON EXIT: 4
******************************** BOTTOM OF DATA
*******************************************************************************
```

If you selected multiple components for compare on the **Compare From libtype - stagelib** panel (CMNCOMPL), all comparison reports are in the SYSPRINT DD statement in component name order.

26. Using Git with Packages

The contents of a ChangeMan ZMF package can be defined as a remote git repo. You can then use git for managing concurrent local development of package components.

Common git oriented tasks can be performed such as:

- Cloning the remote repo (that is, the package) locally
- Changing source components in your local repo and pushing them to the remote repo
- Pulling and merging changes from the remote repo to your local repo

Setting up to Use Git

Prerequisites

Before you start, make sure you have installed the following components:

- The ZMF REST services
- The zowe zmf cli plugin

See documentation for those features if you don't have them in place already (`https://community.microfocus.com/`).

Download the executables

These consist of two executable files and a sample batch/shell file:

- `git-remote-ZMF` is the git protocol helper which enables communication between git and ZMF.
- `showZmfMeta` is a utility program which can be used (usually at the request of Micro Focus support) to list the contents of the ZMF meta-database in the local git repo.
- zbld is a sample batch (windows) or shell script (Linux) file which shows how the git user can request that ZMF build a source component. It can also be used as a template for using zowe zmf cli commands in general.

  There are also two archive/zip files available:
- For windows you'll need `gitZMF_yyyymmdd.zip`.
- For Linux you'll need `gitZMF_yyyymmdd.tgz`.

Install and activate the facility

The main engine of this support is the executable program git-remote-ZMF. This is a git protocol helper and should onlybe invoked by git itself. However, if you type the name at a command prompt *with no parameters*, it displays a version identifier:

```
C:\>git-remote-ZMF
ChangeMan ZMF git protocol helper 1.0 (2022/01/18) [phase #2]
```

Windows

First, place the executables in a directory on your path and define the environment variable which tells the facility where to write log messages (for example):

`ZMF_GIT_LOGFILE=D:\GitDev\ZMF\logs\git-ZMF.log`

> **Note**
>
> This document assumes the use of zowe v2 'team' profiles. In the examples that follow we have set up a high level node called 'd001' which denotes the lpar on which both of our ZMF subsystems, prodzmf and u825all, are running. Below this we have individual profiles for 'prodzmf' and 'u825all'.

You can change the actual file name to a more friendly name.

Then define a zowe zmf profile to allow you to access your target ZMF subsystem. You may have already done this as part of setting up the zowe zmf explorer vscode plugin. If not, follow instructions in the relevant documentation (e.g. zowe --hw presents structured web browsable documentation).

The name of this profile is the name you will use in the git commands to reference ZMF as a remote repository.

Linux

First, place the executables in a directory on your path. There are several ways to set environment variables. For example, you could add the required commands to the .bashrc script which is executed when the bash shell is invoked (you may use a different shell or do something different).

You can add the following lines to the end of the script.

```
export ZMF_GIT_LOGFILE="$HOME/Git/logs/zmf-git-client.log"
```

> **Note**
>
> This document assumes the use of zowe v2 ‘team’ profiles. In the examples that follow we have set up a high level node called ‘d001’ which denotes the lpar on which both of our ZMF subsystems, prodzmf and u825all, are running. Below this we have individual profiles for ‘prodzmf’ and ‘u825all’.

You can change the actual file name to make it more friendly.

Then define a zowe zmf profile to allow you to access your target ZMF subsystem. You may have already done this as part of setting up the zowe zmf explorer vscode plugin. If not, follow instructions in the relevant documentation (e.g. zowe --hw will present structured web browsable documentation).

The name of this profile is the name you will use in the git commands to reference ZMF as a remote repository.

Restricting package components

Using the normal method of operation, the total contents of the target ZMF package are considered eligible to be part of the git remote repo. However, if you want to only consider a subset of the package components, you'll need to define a component filter file. You can define this file either in the local repo, (that is, below the .git directory), where it applies only locally, or in a more general location which you identify using the environment variable GIT_ZMF_FILTER.

Note

If you want to apply a component filter to an initial clone, the only location available to you is that recorded in the environment variable GIT_ZMF_FILTER.

As you can see, you can filter on the name of the remote repo (d001.prodzmf in this case), application, package number, library types, and component names.

Wildcards can be used. And libType and componentName take the form of a json array.

```
{

    "componentFilter": [
    {
        "targetZMF": "d001.prodzmf",
        "applName": "CZMF",
        "packageNumber": "000125",
        "libType": ["BAT","PYT"],
        "componentName": ["\*"]
    },

    {
        "targetZMF": "d001.prodzmf",
        "applName": "CZMF",
        "packageNumber": "000125",
        "libType": ["SRC"],
        "componentName": ["CBLPGM01", "CBLPGM02"]
    }

  ]

}
```

...

{: .blank-line-full }

Examples of git-ZMF interaction

Before we use git to interact with ZMF, all components taking part in the exchange between a ZMF package and a local git repository must have already been checked out into the ZMF package. The exception to this rule is when the components being pushed up from the local repo are *new* to ZMF.

The ZMF subsystem used for these examples is U825ALL.

A zowe zmf profile called d001.u825all has been created which targets the REST services *base_path*/zmfrest. In the ZMF REST api TomCat stc, the web app called zmfrest has startup parameters that point it to the U825ALL ZMF subsystem.

This is the sample package we will be working with:

```
            STAGE: STEV001590 Components                        Row 1 to 9 of 9
Command ===>                                                    Scroll ===> CSR
Name       + Type   Status      Changed             Procname    User Request
CBLCPY01    CPY     ACTIVE      20210826 080231                     WSER58
CBLCPY02    CPY     ACTIVE      20210812 035540                     WSER58
CBLCPY03    CPY     ACTIVE      20210812 042249                     WSER58
CBLCPY04    CPY     ACTIVE      20210809 034448                     WSER58
CBLCPY05    CPY     ACTIVE      20210729 023109                     WSER58
CBLCPY06    CPY     ACTIVE      20210811 022913                     WSER58
CBLCPY07    CPY     ACTIVE      20210812 035851                     WSER58
CBLPGM01    SRC     ACTIVE      20210826 075600     CMNCOBE         WSER58
CBLPGM02    SRC     ACTIVE      20210715 032617     CMNCOBE         WSER58
******************************* Bottom of data ******************************
```

It only has a handful of components but it’s enough to work through the git/ZMF functions.

Note

Because we will not be using a "component filter" on these components, all of them will be eligible for use by the local git repo.

Case 1: Create a local clone of the contents of a ZMF package

This example creates a new git local repo populated with the contents of the package.

First create a directory into which you will be cloning the package. The action of git cloning will automatically create a git repo in the directory you point the clone at. In this example I will clone this package into a directory called ‘sandbox’ .

At a command prompt, navigate your way to the directory in which you will create the ‘sandbox’ clone.

Then issue the following git command:

`git clone ZMF://d001.u825all/STEV1590 sandbox`

The source location for the clone is ZMF://d001.u825all/STEV1590 (or STEV001590, either will work). Let’s break this down.

Starting this location with ZMF: lets git know that it must use a remote protocol helper to access this repository. It looks for an executable called git-remote-ZMF to do the actual remote access. This is the main executable that we provide.

Everything after the ZMF: is then interpreted by git-remote-ZMF.

The name after the // must be a zowe zmf profile name defined on the local machine. This is the profile that the zowe zmf cli commands issued by the ZMF protocol helper will use to route, firstly, to the ZMF RESt api Tomcat stc, and then on to the U825ALL ZMF subsystem.

After the profile name we have a / delimiter followed by the package name. The output in the command prompt window looks like this:

```
D:\GitDev\ZMF\clonedRepo>git clone ZMF://d001.u825all/STEV1590 sandbox
Cloning into 'sandbox'...
progress : querying ZMF package STEV001590 for eligible components ...
progress : extracting 9 components from ZMF package STEV001590 ...
progress 1 of 9 CBLCPY05.CPY -> D:\GitDev\ZMF\clonedRepo\sandbox\CPY\CBLCPY05.cbl
progress 2 of 9 CBLCPY04.CPY -> D:\GitDev\ZMF\clonedRepo\sandbox\CPY\CBLCPY04.cbl
progress 3 of 9 CBLCPY06.CPY -> D:\GitDev\ZMF\clonedRepo\sandbox\CPY\CBLCPY06.cbl
progress 4 of 9 CBLCPY02.CPY -> D:\GitDev\ZMF\clonedRepo\sandbox\CPY\CBLCPY02.cbl
progress 5 of 9 CBLCPY07.CPY -> D:\GitDev\ZMF\clonedRepo\sandbox\CPY\CBLCPY07.cbl
progress 6 of 9 CBLCPY03.CPY -> D:\GitDev\ZMF\clonedRepo\sandbox\CPY\CBLCPY03.cbl
progress 7 of 9 CBLCPY01.CPY -> D:\GitDev\ZMF\clonedRepo\sandbox\CPY\CBLCPY01.cbl
progress 8 of 9 CBLPGM02.SRC -> D:\GitDev\ZMF\clonedRepo\sandbox\SRC\CBLPGM02.cbl
progress 9 of 9 CBLPGM01.SRC -> D:\GitDev\ZMF\clonedRepo\sandbox\SRC\CBLPGM01.cbl
progress complete for ##ZMF## CMNI STEV001590 2022-01-17 17:14:33.331505

D:\GitDev\ZMF\clonedRepo>
```

If you see something like this...

```
D:\GitDev\ZMF\clonedRepo>git clone ZMF://d001.u825all/STEV1590 sandbox
Cloning into 'sandbox'...
component-filter file has no entries for our target ZMF subsystem
```

It means you have defined a global component filter file in the GIT_ZMF_FILTER location (such as a limited clone to a different repo) and it contains no entries for the target we wish to work with.

Either delete this file (zmf-component-filter) or add an appropriate entry to it.

When the clone has completed you will find a directory called sandbox has been created which has three subdirectories. The .git subdirectory is "hidden", and this is where git places all its tracking and control information. We also use the directory structure below .git for ZMF operations. The other two subdirectories are CPY and SRC – in fact there will be as many of these subdirectories created as there are library types being used in the package.

Note the ‘progress complete’ message from the git clone. This string is used by git to create a commit point for the clone and it contains a number of pieces of information:

- `##ZMF##` makes it easy to spot that this is a ZMF related commit point.
- `CMNI` is the subsystem identifier for U825ALL.
- `STEV001590` is the package which forms the remote repository for this clone .
- A timestamp for the clone operation is displayed below the package.

To issue git commands related to this newly create local repo you must be inside the top level directory, so you can use cd to change to the sandbox directory. Then you can issue git commands. For example, the git log command lists logged activities.

```
D:\GitDev\ZMF\clonedRepo\sandbox>git log
commit a8cd39838ce6e6371ae03e48fa81d5e13b68c0ed (HEAD -> master, origin/master,
    origin/HEAD, refs/ZMF/origin/heads/master)
Author: Me <[me@microfocus.com](mailto:me@microfocus.com)>
Date: Mon Jan 17 18:14:33 2022 +0100

    ##ZMF## CMNI STEV001590 2022-01-17 17:14:33.331505

D:\GitDev\ZMF\clonedRepo\sandbox>
```

The commit point created by the clone operation is given a sha1 hash token (the first few characters are a8cd39…) that uniquely identifies the event and its collection of files to git. It is also given a more readable comment (or "commit message") that is generated by the helper at the end of the clone operation.

When a clone operation is performed, the ZMF git protocol helper supplies git with the components it wants to be ‘cloned’ (via zowe zmf commands) and git takes over from there. We also build rows for each component in the local ZMF meta-database. This information is used when reconciling local components with those up at the package as we will see in following examples.

Case 2: Synch local development on our cloned repo with the target ZMF package

At this time, we have made changes to a source component in our local (cloned) repo.

For example, we have updated CBLPGM01 and performed whatever local building and testing we needed to do. We now want to send the changes back to the ZMF package.

In order to see this at work, you can use whatever desktop editor you like to update CBLPGM01. You then need to add the changes to git and then commit them.

We are now in a position to *push* these changes back to ZMF. In this example we have no concurrent development clashes and this proceeds quite simply:

```
D:\GitDev\ZMF\clonedRepo\sandbox\SRC>git push origin master
To ZMF://d001.u825all/STEV1590
    a8cd398..fc630c6 master -> master
```

> **Note**
>
> The origin is the target of the push (which has been defined by git to be our ZMF: reference). And master is the name of the local git branch we are pushing from.

If you look on ZMF on the mainframe you should see the changes to CBLPGM01 are now in the package and the source component is in INCOMP status. If you are using SSV (and why wouldn't you be ?) you can use VC to see the SSV delta:

```
                                Version Control                            Row 1 to 17 of 17

Command ===>                                                                 Scroll ===> CSR

Type: SRC Package: STEV001590                                                   Staging
view
Component: CBLPGM01 +
    Level Prompt    Changed             User            Description

    STG             2022/01/18 02:37    WSER58          fc630c:change to CBLPGM01

Etc.
```

The SSV description is generated by the protocol helper and is made up of the first 6 chars of the git commit sha1 token (which is usually enough to uniquely identify any commit within a local repo) followed by as much of the 'eyeball' commit message that fits.

The component is currently in INCOMP status (it is a SRC component that needs to be built). This is where the sample zbld file can be used by the desktop user to initiate that build. For example:

```
D:\GitDev\ZMF\clonedRepo\sandbox\SRC>zbld CBLPGM01 SRC d001.u825all STEV001590
zmf is "d001.u825all"
pkg is "STEV001590"
{
    "message": "CMN8700I - Component Build service completed",
    "reasonCode": "8700",
    "returnCode": "00"
}
```

You should look at the zbld.bat file to see how this works. It has requested the build job to be submitted by ZMF.

We can do this again, but this time we will introduce a conflicting change made by a mainframe user directly in the package. So, we update CBLPGM01 using some other ZMF client. Then update, add and commit a conflicting change in the local repo.

We then attempt to push our change back to ZMF:

```
D:\GitDev\ZMF\clonedRepo\sandbox\SRC>git push origin master
conflict detected with target repository - use git fetch/merge or pull to
consolidate latest changes in the remote repo
error: failed to push some refs to 'ZMF://d001.u825all/STEV1590'
```

So the process does not let you overwrite changes made to the component in the ZMF package. As the message says, you need to pull down the changes in the ZMF package to your local repo and merge them before re-attempting to push back to ZMF.

The next case will take a look at doing this.

Case 3: Pull down changes made in the target ZMF package and merge them with our local repo

There are two ways of pulling down and merging changes: fetch&merge, and pull. They both achieve the same objective, i.e. fetching changes from the remote repo (ZMF pkg in our case) and merging them into whatever we have in the local repo. The git pull command attempts to do this in one action. The git fetch command fetches the changed components from ZMF and places them into gits staging area (this is where your changes go when you git add them but before you git commit). The follow on git merge will attempt to merge the staged changes with your local repo and create a commit point.

It is at the merge stage (whether explicit as in fetch&merge, or implied in a pull) that conflicts are notified so you can resolve them.

We will now attempt a fetch and merge for the situation we were left with at the end of case#2, i.e. conflicting change in the CBLPGM01 component.

We start off by fetching updates from the ZMF package:

```
D:\GitDev\ZMF\clonedRepo\sandbox\SRC>git fetch origin
progress : querying ZMF package STEV001590 for eligible components ...
progress : extracting 1 components from ZMF package STEV001590 ...
progress 1 of 1 CBLPGM01.SRC -> D:\GitDev\ZMF\clonedRepo\sandbox\SRC\CBLPGM01.cbl
progress complete for ##ZMF## CMNI STEV001590 2022-01-18 12:07:53.199458
From ZMF://d001.u825all/STEV1590
    fc630c6..977f491 master -> origin/master
```

The fetch operation has decided that the only component that has changed is CBLPGM01. It has fetched the current version of the component from the ZMF package and placed it in the local repo as a separate branch (origin/master) commit point.

When two branches of a git managed development are different, unless you want to throw one of them away, you will need to merge them. So, to get the changes from ZMF into your local repo master branch you now proceed to attempt to merge these two branches.

```
D:\GitDev\ZMF\clonedRepo\sandbox\SRC>git merge origin/master
Auto-merging SRC/CBLPGM01.cbl
CONFLICT (content): Merge conflict in SRC/CBLPGM01.cbl
Automatic merge failed; fix conflicts and then commit the result.
```

Note that it is assumed that the current branch (i.e. master) is being merged into.

In this case, as expected, the merge has failed due to conflicting updates. You need to resolve all indicated conflicts.

Git has actually updated your local repo but it has added indications as to the conflicts in the components affected. You must resolve the conflict, save your changes, and then use git add followed by git commit to finish up the merge of code from the ZMF package into the local repo.

At this point you can push the resulting merged update back to ZMF.

```
D:\GitDev\ZMF\clonedRepo\sandbox\SRC>git push origin master
To ZMF://d001.u825all/STEV1590
977f491..245cf74 master -> master
```

Your merged change will now be in the ZMF package.

```
Version Control Row 1 to 19 of 19
Command ===> Scroll ===> CSR
Type: SRC Package: STEV001590 Staging view
Component: CBLPGM01 +
Level Prompt Changed User Description
STG 2022/01/18 04:33 WSER58 245cf7:resolve conflicts
-001 2022/01/18 03:47 WSER58 ISPF UI change
-002 2022/01/18 03:22 WSER58 fc630c:change to CBLPGM01
```

As already mentioned, the git pull command does the same thing as fetch&merge is a single operation. Unless you are wanting to perform several fetch operations before attempting a merge you are as well to just use pull. For example:

```
git pull origin
```

will fetch and merge from origin (i.e. our remote repo, the ZMF package) into the current branch of the local repo.

Case 4: Hook up a local repo to a ZMF package

We have already created a standard local git repo and we now want to hook it up to a target ZMF package instead of cloning the package.

In case#1 we created the local git repo by cloning the remote repo (i.e. the ZMF package) into it.

In this case we start with a repo that already exists and we now decide we wish to move components up to a ZMF package. Here we will look at sending new components to ZMF (if they already exist in baseline, they need to be checked out to the package first. Then you would pull them from there into your pre-existing local repo, merging conflicts as necessary).

With a pre-existing local repo (i.e. one set up before the need to interact with ZMF was identified), your repo may not be in the correct form. You need subdirectories for each ZMF library type with the relevant components stored beneath their library type subdirectory. In which case it may be easier to start a fresh repo and stage/commit components from your existing repo in that one. There are actually many ways of addressing this within git.

If you have a large pre-existing local repo, it is very possible that you will not want to send all of it to ZMF. If you want to send a certain subset to ZMF, you can use the component filter mechanism to restrict the eligible components.

In this example we will populate the local repo with a handful of copybooks and source components and then restrict the interaction with ZMF to a single source component and the copybooks it uses. (This is not a realistic scenario but good enough to demonstrate how this works.)

Here is the initial population of our new local repo (which is not, as yet, connected to ZMF in any way):

```
D:\GitDev\ZMF\localRepo\SRC>git commit -m "initial population"
[master (root-commit) f2a78fa] initial population
5 files changed, 135 insertions(+)
create mode 100644 CPY/NEWCPY01.cbl
create mode 100644 CPY/NEWCPY02.cbl
create mode 100644 CPY/NEWCPY03.cbl
create mode 100644 SRC/NEWPGM01.cbl
create mode 100644 SRC/NEWPGM02.cbl
```

I also need to create a ZMF package which will be the target for pushing component up from this local repo. I am still using U825ALL and I created package STEV001618 for this purpose.

OK, let’s say that I only want to consider NEWPGM01 and its two copybooks NEWCPY01 and NEWCPY02 as eligible for sending up to ZMF.

We need to create a file that looks like this:

```
{

    "componentFilter": [
        {
          "targetZMF": "d001.u825all",
          "applName": "STEV", "packageNumber": "001618",
          "libType": ["CPY"],
          "componentName": ["NEWCPY01", "NEWCPY02"]
        },
        {
          "targetZMF": "d001.u825all",
          "applName": "STEV",
          "packageNumber": "001618",
          "libType": ["SRC"],
          "componentName": ["NEWPGM01"]
        }
    ]
}
```

If this is placed here:

```
D:\GitDev\ZMF\localRepo\.git>dir
Volume in drive D is DATA
Volume Serial Number is 7440-DCE6
Directory of D:\GitDev\ZMF\localRepo\.git
18/01/2022 14:21 19 COMMIT_EDITMSG
18/01/2022 13:56 130 config
18/01/2022 13:56 73 description
18/01/2022 13:56 23 HEAD
18/01/2022 13:56 <DIR> hooks
18/01/2022 14:21 521 index
18/01/2022 13:56 <DIR> info
18/01/2022 14:21 <DIR> logs
18/01/2022 14:21 <DIR> objects
18/01/2022 13:56 <DIR> refs
18/01/2022 14:30 362 zmf-component-filter
6 File(s) 1,128 bytes
5 Dir(s) 1,977,709,088,768 bytes free
```

Then it will override any filter criteria placed at the ZMF_GIT_FILTER location.

In case#1, where the local repo was initialized via the git clone operation, git assigned the reference ‘origin’ to the url which denoted our target ZMF and package. In this case that won’t happen. We can continue to reference the url longhand or we can define our own reference. We could, of course, call it origin too but I will use a different name to show some variation. To assign the reference, issue:

```
D:\GitDev\ZMF\localRepo\>git remote add U825ALL ZMF://U825ALL/STEV001618
```

You can see the reference using…

```
D:\GitDev\ZMF\localRepo>git remote -v
U825ALL ZMF://d001.u825all/STEV001618 (fetch)
U825ALL ZMF://d001.u825all/STEV001618 (push)
```

Where we used ‘origin’ to denote the remote repo in previous cases, we will use ‘U825ALL’ in this case.

We are now ready to send our (limited subset) local repo components to the ZMF package:

```
D:\GitDev\ZMF\localRepo>git push U825ALL master
To ZMF://d001.u825all/STEV001618
* [new branch] master -> master
```

After this we can see our subset of components in the target package:

```
STAGE:                  STEV001618 Components              Row 1 to 3 of 3
Command ===> Scroll ===> CSR
    Name        + Type  Status  Changed           Procname    User Request
NEWCPY01      CPY   ACTIVE     20220118 075638                WSER58
NEWCPY02      CPY   ACTIVE     20220118 075638                WSER58
NEWPGM01      SRC   INCOMP     20220118 075638                WSER58
****************************** Bottom of data ******************************
```

About Git File Extensions and Git GUIs

Understanding file extensions in the local repo

The file extensions applied to the components in the local repo directories are assigned so that intelligent editors will know what they are editing, e.g. the file extension .cbl is given to source components as we expect them to be COBOL.

The language assigned by ZMF to all components can be seen using the `cmponent.history.listlang` service, the results of this service can be altered using both CMNEX038 and HLLX.

Use of git GUIs

Note that all references so far refer to command line use of git.

There are a number of git apps which provide a gui front end, these will work equally as well *as long as they use standard git commands under the covers*.

Many git apps are geared to work with their own remote repos and require other collections of change (for example, a ZMF package) to work as a local repo feeding into the established remote repo (for example, github etc.). This configuration is not supported.

We support allowing a local git repo to treat ZMF as a remote repo so that ZMF contains the ‘gold’ code and there may be any number of local repos, where independent development is going on, with changed code being pushed back up to ZMF eventually.

27. Running Reports

You can submit batch jobs through the ChangeMan ZMF client to run reports that help you manage components, packages, and applications.

About Reports

ChangeMan ZMF reports display information about global and application administration, change packages, and components managed by ChangeMan ZMF. Information for reports comes from the package master and from component history file.

Batch report jobs are initiated online, or jobs can be submitted manually. Batch reporting jobs communicate with a running ZMF instance to access package and component data, but report jobs can also use copies of the online data.

Rules for Running Reports

These rules and restrictions apply to the ChangeMan ZMF reporting function:

- Global and Application administrators determine what ChangeMan ZMF reports appear on user panels in the ISPF interface.
- A report requestor must have READ access to an application in your security system to collect information for a ZMF report from that application.
- If you request a report for multiple applications and you do not have READ access to one of those applications, the data for that application is omitted from the report.

Accessing Reports

Reports are requested from the **Report Selection List** panel (CMNREPT6).

1. Use one of these methods to access ZMF report functions:

   - **Using the Menu Hierarchy:**

     On the **Primary Option Menu**, select **6 Reports**.

   - Using Direct Panel Access:

     On a ChangeMan ZMF panel, type **=6** in the **Command** or **Option** line and press **Enter**.

2. If you license the ChangeMan ZMF Db2 Option, the **Submit Batch Reports** menu (CMNREPTD) is displayed.

   - Select **1 General** and press **Enter**.
   - Option 2 Db2 is only functional if you license the ChangeMan ZMF Db2 Option. See the *ChangeMan ZMF Db2 Getting Started Guide* for instructions.

3. On the **Generate Batch Reports** panel (CMNREPT5), type your **Job Statement Information** and press **Enter** to display the **Report Selection List** panel (CMNREPTA).

Generating ChangeMan ZMF Batch Reports

This section provides instructions for requesting ChangeMan ZMF general batch reports online and manually submitting batch reporting jobs.

Submitting Batch Report Jobs Online

You request one or more ChangeMan ZMF Reports using the **Report Selection List** panel (CMNREPTA).

1. Type report filter criteria and variables in the fields at the top of the panel. Select the reports you want to run from the list at the bottom of the panel.

```
CMNREPTA                           Report Selection List                         Row 1 to 3 of 3
Command ===>                                                                    Scroll ===> CSR

Application mnemonic . . . . GENL
Package number . . . . . . .
Report variable . . . . . . __________________________________________________________________ +
Date range: from . . . . .  (yyyymmdd)
            to . . . . . .  (yyyymmdd)
Enter "/" to select option
  __ Extended Report
  __ Mixed Case

Id Report description
__ 080 Package Component Content
__ 400 Package Promotion Component
__ 500 Package Information Report
******************************* Bottom of data ********************************
```

Scroll up and down in the report list to select and deselect reports before you submit the request. You can press **Enter** to see what reports you have selected and deselected, as shown on the example above.

Note

The global and application administrators determine the reports that you are authorized to run. Only the reports that you are authorized to run are shown on the **Report Selection List** panel (CMNREPTA).

The following table describes the fields on the **Report Selection List** panel (CMNREPT6).

Field	Description
	ChangeMan ZMF will submit a batch job to produce the reports selected from this list. This panel is scrollable. Enter S to select one or more reports; Enter D to "de-select" a report that you have selected in error.
Application mnemonic	The full 4-character application mnemonic or a pattern that consists of one or more characters of the application mnemonic followed by an asterisk. If you specify a pattern, all applications that match the pattern are included in the report. **Note:** The following reports cannot be filtered by application: CMN900, CMN980, CMN990, CMN992 **GEN`*`:** Identifies application names beginning with GEN **GENL:** Identifies an application named GENL. **Blank:** Assumes `'*'` s
Package number	The 6-character package number or a pattern that consists of one or more characters of the package number followed by an asterisk. If you specify a pattern, all packages that match the pattern are included in the report. A blank assumes `*`. The value of the Application Mnemonic and the Package Number field together determine the package or packages to include in the report. The Package Number is ignored if you specify a pattern for the Application Mnemonic. For example: - If Application Mnemonic is GENL and Package Number is 000022, the report only contains information about package GENL000022. - If Application Mnemonic is GEN`*`, all packages in all applications whose names begin with the characters GEN are included in the report. - If Application Mnemonic is GENL and Package Number is 00002`*`, all packages whose names match the pattern GENL00002`*x*` are included in the report.
Report variable	A variable (other than the Application Mnemonic or the Package Number) or a pattern that consists of one or more characters of the variable value followed by an asterisk. If you specify a pattern, all items that match the pattern are included in the report. Blank assumes `*`.
Date range from	Optionally type a From date in the format yyyymmdd. If you omit this field, the date 19600101 (January 1, 1960) is used by default

Field	Description
Date range to	Optionally type a To date in the format yyyymmdd. If you omit this field, today's date is used by default.
Extended Report	Select to append an 'E' to the selected report ID to execute a version of the REXX report program that expands component names to 256 bytes, and dataset names to 1024 bytes. Use this option when managing zFS components. If not selected it will execute the standard REXX report program that assumes 8 character component names and 44 character dataset names.
Mixed Case	Select to process Name input exactly as you type it, upper and lower case. If not selected, by default, all fields will input to upper case regardless of the case that you type.
Report selection field	Enter one of the following values in the selection field to the left of the Report ID: - **S:** Select this report. - **D:** Deselect a report that you have selected by mistake.
Report Id	The 3-character report identification number.
Report description	Description of report content.

The following table lists all of the reports that are shipped with ChangeMan ZMF. It identifies the input fields and date range that applies to each report.

Rpt #	Report Title	Input Fields	Date Range
010	Summary of Planned and Unplanned Packages	Appl Mnemonic	N/A
050	Installed Package History Report	Appl Mnemonic Package Number	Package Install Date
060	Installed Simple Package History Report	Appl Mnemonic Package Number	Package Install Date
070	Package Staging Versions (see note 1)	Appl Mnemonic Package Number	Package Create Date
080	Package Component Content	Appl Mnemonic Package Number	Package Create Date
090	Super/Complex Packages Report	Appl Mnemonic Package Number	Package Create Date
100	Application Configuration Report	Appl Mnemonic	N/A
110	Work Request Number Report	Appl Mnemonic Package Number Work Chg Req Nbrb (see note 2)	Package Create Date

Rpt #	Report Title	Input Fields	Date Range
120	Unplanned Packages Pending Post Approval	Appl Mnemonic Package Number	Package Install Date
130	Report of Backed Out Packages	Appl Mnemonic Package Number	Package Backout Date
140	Report of Rejected Packages	Appl Mnemonic Package Number	Approver Reject Date
160	Trend Analysis of Planned Packages	Appl Mnemonic Package Number	Package Install Date
180	Component History Report	Appl Mnemonic Package Number Component Name(see note 2)	Changed Date
185	Component Build Report	Appl Mnemonic Package Number Component Name(see note 2)	Changed Date
190	Trend Analysis of Unplanned Packages	Appl Mnemonic Package Number	Package Install Date
200	Copies, Source and Load Component Report	Appl Mnemonic Package Number	Changed Date
210	Utility (Scratch/Rename) Report	Appl Mnemonic Package Number	Package Create Date
240	Activity Summary by Application	Appl Mnemonic	N/A
260	Implementation Schedule	Appl Mnemonic Package Number	Package Install Date
400	Package Promotion Component Report	Appl Mnemonic Package Number	Package Create Date
500	Package Information Report	Appl Mnemonic Package Number	Package Create Date
900	Designated Procedures Report	Component Name(see note 2)	N/A
910	Packages Using the CMN Scheduler	Appl Mnemonic Package Number	Package Install Date
920	Staging Library Aging Report	Appl Mnemonic	(see Note 3)
930	Package Aging Report	Appl Mnemonic	(see note 4)
940	Component Aging Report	Appl Mnemonic	(see note 5)

Rpt #	Report Title	Input Fields	Date Range
950	Report of Delinquent Packages	Appl Mnemonic Package Number	Package Install Date
960	Component Level Security Report	Appl Mnemonic Component Name (see note 2)	N/A
970	Packages Pending Approval by Application	Appl Mnemonic Package Number (see note 6)	Package Install Date
980	Implementation Calendar Report	Site Name (see note 2)	N/A
985	Implementation Calendar Summary Report	Site Name (see note 2)	N/A
990	Global Administration Configuration Report	none	N/A
992	Global Environment Report	none	N/A

Notes

1. The Package Staging Versions Report (CMN070) dynamically allocates the staging libraries. The report may take a long time to run if staging libraries have been migrated.
2. Specify this value in the Report Variable field.
3. Date parameters are set by the AGING - STAGING DATASETS field on the *appl* Parameters Part 3 of 3 panel (CMNGLP03) in application administra- tion.
4. Date parameters are set by the AGING - INSTALLED PACKAGES field on the *appl* Parameters Part 3 of 3 panel (CMNGLP03) in application adminis- tration.
5. Date parameters are set by the AGING - COMPONENT HISTORY field on the *appl* Parameters Part 3 of 3 panel (CMNGLP03) in application administra- tion.
6. Displays all package approver entities and their associated approval status.

Press **PF3** to run the selected reports, or type **End** on the **Command** line and press **Enter**.

Note

If you specify a filter field or variable that does not apply to a selected report, the field is ignored when the report data is collected.

Viewing Report Output

Report output is written to DDname SYSTSPRT.

```
//SYSTSPRT DD SYSOUT=*,DCB=(RECFM=VBA,LRECL=2044,BLKSIZE=2048)
```

This is defined with attributes RECFM=VBA and LRECL=2044. (The LRECL=2044 value allows scrolling left and right in sdsf when reports contain long filenames or dataset names.) The first character of each report record is a print control character.

If you request multiple reports in the same job, each report is executed in a separate job step that is named REPnnn, where nnn is the report number.

Use a facility such as the System Display and Search Facility (SDSF) to view report output online.

Manually Submitting a Batch Report Job

Instead of submitting report jobs online through the ISPF user interface, you may be able to submit your own batch jobs outside of ChangeMan ZMF to generate reports.

Ask your ChangeMan ZMF administrator about:

- Custom ZMF report JCL configured for manual submission
- Whether the ZMF instance must be running or if separate VSAM package master and component history files have been populated for offline batch reporting.

Writing Custom Reports

Reports CMNU00-99 are reserved numbers for user written reports. All parameters are passed to these reports and you must validate or ignore used and unused parameters.

Parameters passed in order are:

1. release name
2. release area
3. optional variable
4. application
5. package number (appended to application with no space)
6. subsys
7. tso id
8. from date

9. to date
10. sort option
11. test option

You must handle unused variables even though they will not be validated because they are positional for subsequent variables.

So for example, a report which does not make use of release variables could have the following parse statement:

```
parse var parmargs null1 null2 optionalvar appl ... srtopt tst
```

where null1 and null2 are just not used.

This statement must match the parse statement in the actual report.

Use Case Scenario

The following **Report Selection List** (CMNREPTA) panel requests the 070 Package Staging Versions Report and the 080 Package Component Content report.

```
CMNREPTA                           Report Selection List                        Row 1 to 4 of 4
Command ===>                                                                   Scroll ===> CSR

Application mnemonic . . . . ACTP
Package number . . . . . . . 00001*
Report variable . . . . . .                                                   +
Date range: from . . . . .            (yyyymmdd)
            to . . . . . .            (yyyymmdd)
Enter "/" to select option
  __ Extended Report
 __ Mixed Case

  Id Report description
s 070 Package Staging Versions Report                         *SELECT
s 080 Package Component Content
  400 Package Promotion Component
  500 Package Information Report
******************************* Bottom of data ********************************
```

All packages with names in the range from ACTP000010 through ACTP000019 are requested for the reports.

This is the first online page of the 070 Package Staging Versions Report displayed in SDSF in job USER016 J0556970 at DDNAME SYSTSPRT and StepName REP070.

```
SDSF OUTPUT DISPLAY USER016    J0556970  DSID     102 LINE 0             COLUMNS 01- 80
COMMAND INPUT ===>                                                      SCROLL ===> CSR
********************************* TOP OF DATA ********************************
1Report CMN070 generated from subsystem:  6 on:  3 Feb 2015 at: 14:48:56 Page: 1
0Date Range: 1 Jan 1960 to 3 Feb 2015
  ------------------------
  Package Staging Versions
  ------------------------
  Package    Component  Component Version Update   Lock     Date Last   Time Last
  Name       Name       Type      Level   Userid   Userid   Modified    Modified
  ---------- ---------  --------- ------- -------- -------- ---------   ---------
  ACTP000013 ACPSRCCC   SRC       STG     USER016  USER015  20150105    145644
  ACTP000013 ACPSRCCC   SRC       BAS     USER016           20150105    145603
  ACTP000013 ACPSRCCE   SRC       STG     USER016  USER015  20150105    145741
  ACTP000013 ACPSRCCE   SRC       BAS     USER016           20150105    145610
  ACTP000013 ACPSRCSA   SRC       STG     USER016  USER015  20150105    145815
  ACTP000013 ACPSRCSA   SRC       BAS     USER016           20150105    145631
  ACTP000013 ACPSRC2A   SRC       STG     USER016  USER015  20150105    145916
  ACTP000013 ACPSRC2A   SRC       BAS     USER016           20150105    145643
  ACTP000013 ACPSRC4A   SRC       STG     USER016  USER015  20150105    145941
  ACTP000013 ACPSRC4A   SRC       BAS     USER016           20150105    145735
  ACTP000013 ACPSRC6A   SRC       STG     USER016  USER015  20150105    145957
```

This is the first online page of the 080 Package Component Content report displayed in SDSF in job USER016 J0556975 at DDNAME SYSTSPRT and StepName REP080.

```
SDSF OUTPUT DISPLAY USER016    J0556975   DSID   102   LINE 0         COLUMNS 02- 81
COMMAND INPUT ===>                                                   SCROLL ===> CSR
********************************* TOP OF DATA *********************************
Report CMN080 generated from subsystem: 6 on: 3 Feb 2015 at: 14:57:38 Page: 1
Date Range: 1 Jan 1960 to 3 Feb 2015
--------------------------
Package Component Contents
--------------------------
Package    Component  Component         Component   Lock     Recompile/
Name       Name       Type     Language Status      Status   Relink
---------- ---------  --------- ------- ---------   ------   ----------
ACTP000020 ACPCPY2B   CPY                Frozen     N
ACTP000020 ACPCPYCA   CPY                Frozen     N
ACTP000020 ACPCPYCB   CPY                Frozen     N
ACTP000020 ACPCPYCC   CPY                Frozen     N
ACTP000020 ACPCPYCD   CPY                Frozen     N
ACTP000020 ACPCPYCE   CPY                Frozen     N
ACTP000020 ACPCPYCF   CPY                Frozen     N
ACTP000020 ACPCPYSA   CPY                Frozen     N
ACTP000020 ACPCPYSB   CPY                Frozen     N
ACTP000020 ACPCPY00   CPY                Frozen     N
ACTP000020 ACPCPY01   CPY                Frozen     N
```

28. Miscellaneous Functions

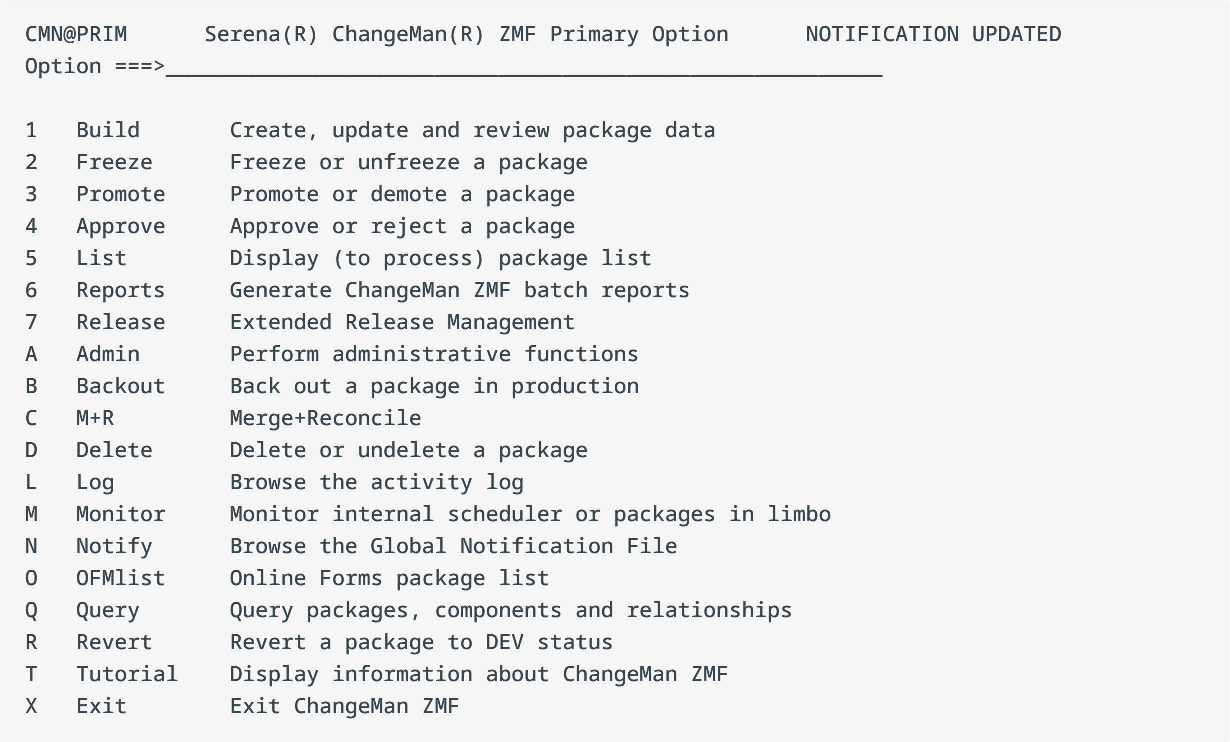
This chapter describes several functions that are not directly involved in managing change packages or package components.

Browsing the Global Notification File

If you see option **N Notify** on the **Primary Option Menu**, the Global Notification File is activated. Administrators use the Global Notification file to pass information to all ChangeMan ZMF users. Your administrator can use this facility to inform you about:

- Scheduled down time.
- Scheduled maintenance changes to ChangeMan ZMF.
- Work-arounds for recently discovered problems.
- Upcoming ChangeMan ZMF education or discussion sessions.

  When a you connect to ChangeMan ZMF after an administrator updates the Global Notification File, the ISPF short message NOTIFICATION UPDATED is displayed in the upper right corner of the **Primary Option Menu**. If the you press **PF1**, a long message is also displayed.

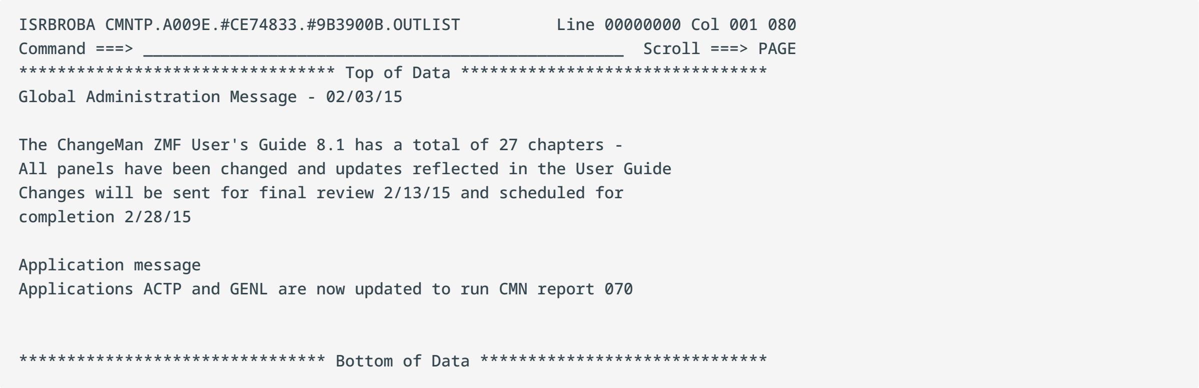


```
CMN@PRIM      Serena(R) ChangeMan(R) ZMF Primary Option      NOTIFICATION UPDATED
Option ===>________________________________________________________

1   Build       Create, update and review package data
2   Freeze      Freeze or unfreeze a package
3   Promote     Promote or demote a package
4   Approve     Approve or reject a package
5   List        Display (to process) package list
6   Reports     Generate ChangeMan ZMF batch reports
7   Release     Extended Release Management
A   Admin       Perform administrative functions
B   Backout     Back out a package in production
C   M+R         Merge+Reconcile
D   Delete      Delete or undelete a package
L   Log         Browse the activity log
M   Monitor     Monitor internal scheduler or packages in limbo
N   Notify      Browse the Global Notification File
O   OFMlist     Online Forms package list
Q   Query       Query packages, components and relationships
R   Revert      Revert a package to DEV status
T   Tutorial    Display information about ChangeMan ZMF
X   Exit        Exit ChangeMan ZMF
```

The short message persists on the **Primary Option Menu** until you browse the Global Notification file.

Follow these steps to browse the Global Notification file.

1. On the **Primary Option Menu**, select **N Notify**. The Global Notification File is opened in browse mode.



```
ISRBROBA CMNTP.A009E.#CE74833.#9B3900B.OUTLIST          Line 00000000 Col 001 080
Command ===> __________________________________________________  Scroll ===> PAGE
********************************* Top of Data ********************************
Global Administration Message - 02/03/15

The ChangeMan ZMF User's Guide 8.1 has a total of 27 chapters -
All panels have been changed and updates reflected in the User Guide
Changes will be sent for final review 2/13/15 and scheduled for
completion 2/28/15

Application message
Applications ACTP and GENL are now updated to run CMN report 070


******************************** Bottom of Data ******************************
```

   Use standard ISPF commands and PF keys to navigate the file and view the messages it contains.

2. Exit the Global Notification File by pressing **PF3**, and the **Global Notification Report Disposition** panel (CMNNOT00) is displayed.

```
CMNNOT00 Global Notification Report Disposition
Command ===>

Outlist dataset:     CMNTP.A009E.#CE74833.#9B3900B.OUTLIST

Report disposition . . . . 3    1. Print dataset and delete
                                2. Print dataset and keep
                                3. Delete without printing
                                4. Keep dataset without printing

Job statement information if printing:
  //USER016 JOB (0000),'CHANGE MAN',
  //             CLASS=A,NOTIFY=USER016,MSGCLASS=X
  //*
  //*
```

The **Global Notification Report Disposition** panel shows the name of the ChangeMan ZMF utility OUTLIST file used to display the Global Notification File. Choose a **Report Disposition** on this panel to print the OUTLIST file, and to keep or delete the file. If you are printing the file, provide **JOB Statement Information** in the four lines provided.

3. Press **PF3** Enter to execute the Report Disposition and return to the **Primary Option Menu**.

Compressing Staging Libraries

ChangeMan ZMF dynamically reallocates PDS package staging libraries when they run out of extents or directory space.

Staging libraries are under the exclusive control of ChangeMan ZMF. If you want to manually recover unused space in PDS staging libraries, use the Compress Staging Libraries facility of ChangeMan ZMF. This function submits a batch job that executes IEBCOPY to compress the PDS libraries.

Follow these steps to compress PDS staging libraries for a change package.

1. On the **Primary Option Menu**, select option **1 Build**.
2. On the **Build Options** menu, select **Z Compress** to display the **Compress Staging Libraries** panel (CMNCPRSS).

```
CMNCPRSS Compress Staging Libraries
Command ===>

Package . . . . ACTP000038

Job statement information:
  //USER016 JOB  (0000),'CHANGE MAN',
  //             CLASS=A,NOTIFY=USER016,MSGCLASS=X
  //*
  //*
```

3. On the **Compress Staging Libraries** panel, enter package in **Package** field, for the staging libraries you want to compress. Update your the **Job Statement Information** and press **Enter**.

4. You should see a **job submitted** message for a batch job. When the **job ended** message is displayed, examine the job output to see that IEBCOPY successfully compressed the package staging libraries.

Displaying the Component User Work List

If your application administrator enables the Component Work Record facility, ChangeMan ZMF keeps a record of each person who changed a component in a change package.

Actions which are recorded include:

- Checkout
- Stage
- Edit in stage
- Recompile
- Relink
- Delete

Component Work Records can be used to restrict who can approve a change package or how many approvals one person can enter. See Rules for Approving and Rejecting Packages for more information.

You can display a list of the userids of people who changed the components in your package. For each component, only the last action taken by each userid is shown, but there is a count of the total actions taken by that userid against that package component.

Displaying the Userid Work List for All Package Components

You can see the userids of anyone that changed any of the components in your change package though the Query Package function.

After you have selected the package you want to query and displayed the **Package Information Categories** panel, select **Component Userid Work List** category. The **Component Userid Work List** panel (CMNUCWK1) is displayed.

```
CMNUCWK1                   Component Userid Work List                     Row 1 to 15 of 25
Command ===>                                                               Scroll ===> PAGE

        Package: ACTP000043      Status: REJ       Install date: 20150131

 Component Name      Type Status     User     Last      Last      Action       Actioned
                  +                           date      time      request      count
 ACPCPY00            CPY             USER239   20100428 231525    STAGED       0005
 ACPCPY1A            CPY             USER239   20100426 115611    STAGED       0002
 ACPCPY1B            CPY  DELETED    USER239   20100426 154156    DELETED      0003
 ACPCPY1C            CPY  DELETED    USER239   20100426 154159    DELETED      0003
 ACPCPY1X            CPY  DELETED    USER239   20100426 154202    DELETED      0003
 ACPCPY5A            CPY  DELETED    USER239   20100426 154205    DELETED      0003
 ACPCPY5B            CPY  DELETED    USER239   20100426 154208    DELETED      0003
 ACPCPY5C            CPY  DELETED    USER239   20100426 154212    DELETED      0003
 ACPCTL10            CTL             USER239   20100501 232640    STAGED       0003
 ACPCTL10            JCL  DELETED    USER239   20100427 002955    DELETED      0002
 ACPJCL10            JCL             USER239   20100501 232643    STAGED       0014
 ACPJCL10            JCL             USER240   20100425 173934    DELETED      0002
 ACPJCL20            JCL  DELETED    USER239   20100426 154215    DELETED      0003
 ACPPRC10            PRC             USER239   20100501 232645    STAGED       0003
 ACPSRCD1            SRC             USER239   20100501 232915    STAGED       0004
```

Displaying the Userid Work List for One Package Component

You can see the userids of any persons that changed a particular component in your change package though the **Stage** (package list) or **Checkout** (package list) functions.

After you have displayed the **Stage:** *package* **Components** panel for your change package, or the **Checkout: package Components** panel, enter line command **UL** on a package component row. The **Userid Work List** panel (CMNUCWK0) is displayed.

```
CMNUCWK0                  Userid Work List                 Row 1 to 2 of 2
Command ===>                                              Scroll ===> PAGE

       Package: ACTP000043      Status: DEV      Install date: 20150131

       Component name . . . ..ACPJCL10 +
       Component type . . . ..JCL

                  Last       Last       Action        Actioned
        User      date       time       request       count
       USER239    20100502   181508     STAGED        0016
       USER240    20100425   173934     DELETED       0002
******************************* Bottom of data ********************************
```

Browsing and Printing the Activity Log

User and administrator actions taken in ChangeMan ZMF are recorded in the Activity Log file, a VSAM KSDS. You can use log records as an audit trail for user and administrator activities in ChangeMan ZMF. The log can also help you remember what you did with a package or component.

You can display records from the log file online or in a batch report. You can use the online log function in ChangeMan ZMF to see log records in the VSAM log file that ChangeMan ZMF is currently using, or you can run batch reports to display records from sequential backups of the VSAM log file.

This table shows the three ways you can display Activity Log records.

Log Report Type	Where Initiated	Where Executed	Output Title
Browse current VSAM LOG file online	Online	Online	Activity Log Entries panel
Run Log Activity Report using one log backup file.	Online	Batch	Log Activity Report
Run Log Activity Report job using one or more log backup files	Batch JCL	Batch	Log Activity Report

When you use any of these methods, you can provide filter criteria to select activity you want to see.

Note

The format of activity log records changed in ZMF 7.1. The ChangeMan ZMF 7.1 Migration Guides tells you to clear the CMNLOG file when upgrading from a prior ZMF version. To report on activity log records that were created before the upgrade, you must use a batch job with a JOBLIB pointing to your pre-upgrade libraries. See Running Log Activity Report Job Using One or More Log Backups.

Accessing Activity Log Panels

The online log browse function and the batch report initiated from online are accessed from the **Browse Activity Log** panel (CMNLOGEX).

Use one of these methods to display the **Browse Activity Log** panel:

- **Using the Menu Hierarchy:**

  On the **Primary Option Menu**, select **L Log**.

- **Using Direct Panel Access:**

  From any ChangeMan ZMF panel, enter **=L** in the **Command** or **Option** line and press **Enter**.

Browsing Current VSAM LOG File Online

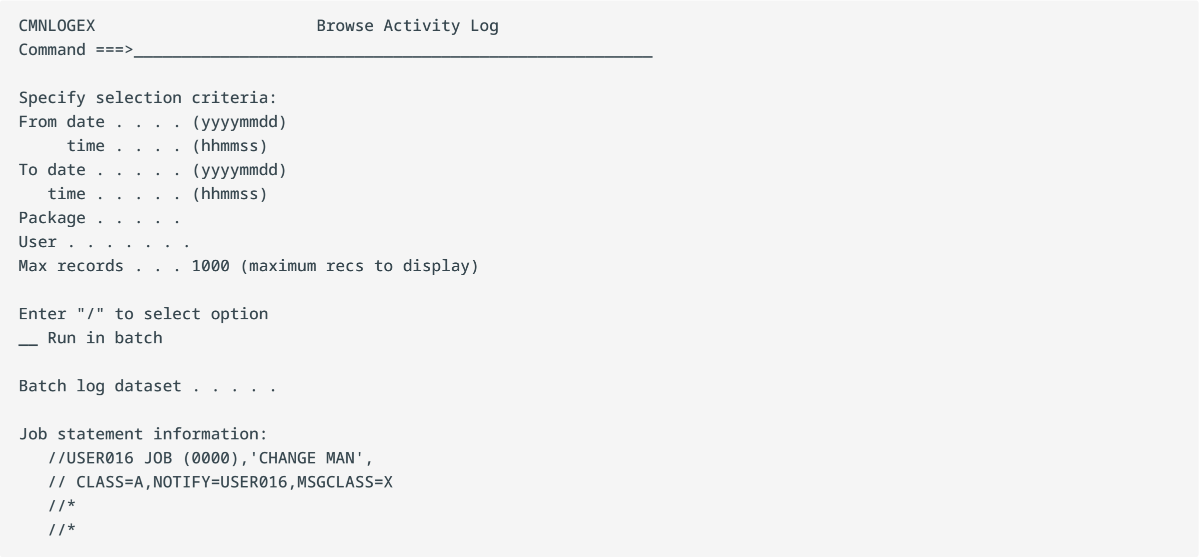
When you browse the current VSAM log file online, you can only see ChangeMan ZMF activity recorded since the last time the log file was cleared by ChangeMan ZMF housekeeping jobs.

If you want to see log activity from earlier periods, see:

- Running Log Activity Report Using One Log Backup
- Running Log Activity Report Job Using One or More Log Backups

Follow these steps to browse the current VSAM Activity Log file online.

1. Using the ChangeMan ZMF menu hierarchy or accessing the panel directly, display the **Browse Activity Log** panel (CMNLOGEX).



```
CMNLOGEX                       Browse Activity Log
Command ===>______________________________________________________

Specify selection criteria:
From date . . . . (yyyymmdd)
     time . . . . (hhmmss)
To date . . . . . (yyyymmdd)
   time . . . . . (hhmmss)
Package . . . . .
User . . . . . . .
Max records . . . 1000 (maximum recs to display)

Enter "/" to select option
__ Run in batch

Batch log dataset . . . . .

Job statement information:
   //USER016 JOB (0000),'CHANGE MAN',
   // CLASS=A,NOTIFY=USER016,MSGCLASS=X
   //*
   //*
```

Use the following field descriptions to set **Browse Activity Log** panel fields:

Field	Description
From date	Start date for selection period in YYYYMMDD format. FROM DATE selection is inclusive. Leaving the date blank is the same as typing 00000101.
From time	Start time for selection period in hhmmss format. FROM TIME selection is inclusive. Leaving the time blank is the same as typing 000000.
To date	End date for selection period in YYYYMMDD format. TO DATE selection is inclusive. Leaving the date blank is the same as typing 99991231.
To time	To time for selection period in hhmmss format. TO TIME selection is exclusive; times are selected up to but not including the specified time. Leaving the time blank is the logical equivalent of specifying 235959+000001.
Package	Application or package ID to select in the format *aaaannnnnn,* where:
	aaaa Application mnemonic to select application administration activity
	aaaannnnnn Package ID to select package activity

Field	Description
	If you code a mask in *nnnnnn*, (like ACTP00001*), only package activity is selected. If you code a mask in *aaaa* (like ACT*), package activity and application administration activity are selected.
User	User ID or mask in activity records to select.
Max. records	This field limits the number of displayed log entries to display online only. Once the specified number of records has been read, the remaining records are ignored. Any valid number may be entered.
Run in batch	If not selected you will browse the current VSAM log file online. Select to run in batch mode to produce the LOG ACTIVITY REPORT based upon the last backup copy of the log file (usually backed up during the nightly Housekeeping Tasks).
Batch log dataset	The full dataset name of the QSAM Log File to which your Administrator backs up the current (VSAM) Log File. Do NOT use apostrophe ('). This field is only required if 'Run in batch' is selected.
Job statement information	Job card information is required if ’Run in batch’ is selected. Not used to browse current VSAM log file online.

2. After you have specified selection criteria on the panel, press Enter to display the **Select Activity Log Codes** (CMNLOGE2) panel.

```
CMNLOGE2                   Select Activity Log Codes                  Row 1 to 20 of 81
Command ===> Scroll ===> CSR
    Code Description
 __ 01 Backout a Package
 __ 02 Install a Package
 __ 03 Temporary Change Cycle
 __ 04 Distribute a Package
 __ 05 Unauthorized Member Access
 __ 07 Generate Package Information
 __ 08 Delete a Package (Physical Delete)
 __ 09 Update Application Information
 __ 10 Revert a Package
 __ 11 Update Global Information
 __ 12 Activate a Component
 __ 13 Memo Delete Package
 __ 14 Undelete a Package
 __ 15 Baseline Ripple
 __ 16 Reverse Baseline Ripple
 __ 18 Age Installed Package
 __ 20 Approve a Package
 __ 21 Re-synch Calendar
 __ 22 Age Staging Libraries
```

This table describes the fields on the **Select Activity Log Codes** panel.

Field	Description
Command	CANCEL: **Cancel processing and return to the Primary Option Menu. Abbreviation: C** LOCATE: **Scroll to specified entry.** REFRESH: **Display updated information on this panel. This clears any S line commands. Abbreviation: R<br.** SELECT: **Select all codes. No abbreviation: S** SORT: **Sort the list.** FIND:** Finds the text entered after the command. Entering the same Find command finds the next occurrence.
	Note:** This command performs a case-sensitive search.
Line command	Enter 'S' to display or print all activity codes available.
Code	Select an internal number identifying each type of log record you want to display or print.
Description	Displays a short description of the activity associated with the activity log code.

3. Enter S in one or more line commands to select activity types for display, or type SELECT on the COMMAND line to select all activities.
4. Press PF3 (END) to display the **Activity Log Entries** panel (CMNLOGDS) that shows the selected log records.

```
CMNLOGDS                        Activity Log Entries                    Row 141 to 160 of 241
Command ===>                                                                 Scroll ===> CSR

Date      Time    User    Package      Description +
20141216  200725  USER016  ACTP000020   CMPONENT/CHECKOUT/SERVICE   CPY PLICPY01
20141216  202728  USER016  ACTP000020   CMPONENT/CHECKOUT/SERVICE   CPY SAMCPY1B
20141217  143908  USER016  COMM000001   PACKAGE/CREATE/SERVICE
20141217  173200  USER016  COMM000002   PACKAGE/CREATE/SERVICE
20141217  210929  USER016  COMM000003   PACKAGE/CREATE/SERVICE
20141217  211213  USER016  ACTP000023   PACKAGE/CREATE/SERVICE
20141217  211506  USER016  ACTP000023   CMPONENT/CHECKOUT/CONFLICT  SRC ACPSRC1A
20141217  211507  USER016  ACTP000023   CMPONENT/CHECKOUT/SERVICE   SRC ACPSRC1A
20141218  011018  USER016  ACTP000024   PACKAGE/CREATE/SERVICE
20141218  014327  USER016  ACTP000025   PACKAGE/CREATE/SERVICE
20141218  014627  USER016  ACTP000025   CMPONENT/CHECKOUT/SERVICE   LOD ACPSRC1A
20141219  012934  USER016  GENL000001   PACKAGE/CREATE/SERVICE
20141223  152927  USER016  ACTP000026   PACKAGE/CREATE/SERVICE
20141223  161358  USER016  GENL000002   PACKAGE/CREATE/SERVICE
20141223  162804  USER016  GENL000002   CMPONENT/CHECKOUT/SERVICE   CPY GNLCPY00
20141223  162825  USER016  GENL000002   CMPONENT/CHECKOUT/SERVICE   SRS GNLSRS00
20141223  190749  USER016  GENL000003   PACKAGE/CREATE/SERVICE
20141223  214249  USER016  GENL000004   PACKAGE/CREATE/SERVICE
20141223  222833  USER016  GENL000003   CMPONENT/CHECKOUT/SERVICE   CPY GNLCPY00
20141223  222834  USER016  GENL000003   CMPONENT/CHECKOUT/SERVICE   CPY GNLCPY01
```

!!! note The **Description** may be truncated on the **Activity Log Entries** panel. Expand the panel field with **PF4** to see the full description.

This table describes the fields on the **Select Activity Log Codes** panel.

Field	Description
Command	REFRESH: **Display updated information on this panel. This clears any S line commands. Abbreviation: R** LONG: **Display the activity on two lines with DESCRIPTION wrapped to the second line. Abbreviation: L** SHORT: **Display the activity on one line, showing only the first 40 bytes of DESCRIPTION. Abbreviation: S** SORT:** Sort the displayed activity by the specified column heading. Example: SORT PACKAGE
Date	The date that the logged activity occurred (YYYYMMDD).
Time	The time that the logged activity occurred (HHMMSS).
User	The user name that initiated or caused the activity to occur. This field may display the User id of the task that submitted the batch job which created the log entry.
Package	This field can display one of the following: - Package ID where this activity occurred - "Global" for global administration activity - Application mnemonic for application administration activity - Component name - Mask or pattern.
Description	A short description of the logged event. The full description may be displayed by placing the cursor in the description and using the 'expand' `<PF4>` function.

Running Log Activity Report Using One Log Backup

When you use the **Browse Activity Log** panel to initiate the batch Log Activity Report, you can specify only one input log backup file. The date range of the report is effectively limited by the records included in that backup.

If you want to see log activity from other periods, see:

- Browsing Current VSAM LOG File Online
- Running Log Activity Report Job Using One or More Log Backups

Follow these steps to use ISPF panels to initiate the batch Log Activity Report using one log backup file. For panel field descriptions, see Browsing Current VSAM LOG File Online.

1. Display the **Browse Activity Log** panel (CMNLOGEX) using the ChangeMan ZMF menu hierarchy or accessing the panel directly. Make these panel entries:

a. Select (/) RUN IN BATCH field.

b. In the BATCH LOG DATASET NAME, type the fully qualified data set name of a QSAM backup for the VSAM log file. Do not enclose the data set name in apostrophes.

c. In the JOB STATEMENT INFORMATION, type a valid JOB statement.

d. If necessary, specify selection criteria in fields FROM DATE, FROM TIME, TO DATE, TO TIME, PACKAGE ID, and USER ID.

2. Press ENTER to display the **Select Activity Log Codes** (CMNLOGE2) panel. Enter S in one or more line commands to select activity codes for display, or type SELECT (S) on the COMMAND line to select all codes.
3. Press ENTER to submit a batch job. When the job ends, select the SYSOUT data set at DD name LOGRPORT to see the **Log Activity Report**.

```
REPORT ID: CMN020                                                                                      AS OF 2010-05-04 08:55

                                          LOG ACTIVITY REPORT PAGE 1
                                            2010-03-02 THRU 2010-05-04

                                                                 PACKAGE
ACTIVITY                      DATE       TIME       USER-ID  NAME        DESCRIPTION
______________________________ __________ __________ ________ __________  _______________________________________________
BUILD A PACKAGE               2010-04-15 18:52      USER239  ACTP000019  PACKAGE/CREATE/SERVICE
CHECKOUT                      2010-04-15 18:55      USER239  ACTP000019  CMPONENT/CHECKOUT/SERVICE JCL ACPJCL20
DELETE A MODULE               2010-04-15 18:56      USER239  ACTP000019  CMPONENT/DELETE /PKG_COMP JCL ACPJCL20
PACKAGE MEMO DELETED          2010-04-15 18:56      USER239  ACTP000019  Package has been memo deleted.
BUILD A PACKAGE               2010-04-15 18:59      USER239  ACTP000020  PACKAGE/CREATE/SERVICE
CHECKOUT                      2010-04-15 19:08      USER239  ACTP000020  CMPONENT/CHECKOUT/SERVICE JCL ACPJCL30
CHECKOUT                      2010-04-15 19:08      USER239  ACTP000020  CMPONENT/CHECKOUT/SERVICE JCL ACPJCL10
CHECKOUT                      2010-04-15 19:08      USER239  ACTP000020  CMPONENT/CHECKOUT/SERVICE JCL ACPJCL20
CHECKOUT                      2010-04-15 19:08      USER239  ACTP000020  CMPONENT/CHECKOUT/SERVICE JCL ACPJCL23
CHECKOUT                      2010-04-15 19:08      USER239  ACTP000020  CMPONENT/CHECKOUT/SERVICE JCL ACPJCL25
DELETE A MODULE               2010-04-15 19:09      USER239  ACTP000020  CMPONENT/DELETE /PKG_COMP JCL ACPJCL20
DELETE A MODULE               2010-04-15 19:09      USER239  ACTP000020  CMPONENT/DELETE /PKG_COMP JCL ACPJCL23
DELETE A MODULE               2010-04-15 19:09      USER239  ACTP000020  CMPONENT/DELETE /PKG_COMP JCL ACPJCL25
DELETE A MODULE               2010-04-15 19:09      USER239  ACTP000020  CMPONENT/DELETE /PKG_COMP JCL ACPJCL30
COPY A COMPONENT              2010-04-15 19:09      USER239  ACTP000020  CMPONENT/COPY /SERVICE JCL ACPJCL10
STAGE A COMPONENT             2010-04-15 19:09      USER239  ACTP000020  CMPONENT/CHECKIN /SERVICE JCL ACPJCL10
AUDIT PACKAGE                 2010-04-15 19:10      USER239  ACTP000020  Submitted audit job for package.
CHANGE AUDIT RETURN CODE      2010-04-15 19:10      USER239  ACTP000020  CMNVPKGE has set audit rc to 00
AUDIT PACKAGE                 2010-04-15 19:10      USER239  ACTP000020  Pass the audit with return code of 00
FREEZE                        2010-04-15 19:10      USER239  ACTP000020  PACKAGE/FREEZE/SERVICE
GENERATE GLOBAL PARAMETERS    2010-04-15 19:15      USER239  Global      PARMS/UPDATE/GBL
GENERATE GLOBAL PARAMETERS    2010-04-15 19:15      USER239  Global      model dsn - prod - ????.GLOBAL.PRD.#######
APPROVE                       2010-04-15 19:36      USER239  ACTP000020  ACTPLEAD PACKAGE/APPROVE/SERVICE
APPROVE                       2010-04-15 19:36      USER239  ACTP000020  QAMGR PACKAGE/APPROVE/SERVICE
APPROVE                       2010-04-15 19:36      USER239  ACTP000020  ACCTPAY PACKAGE/APPROVE/SERVICE
DISTRIBUTE                    2010-04-15 19:37      SERT     ACTP000020  Package distributed
INSTALL                       2010-04-15 19:37      SERT     ACTP000020  Installed
```

Running Log Activity Report Job Using One or More Log Backups

If you need a comprehensive view of activity in ChangeMan ZMF, run the Log Activity Report using the batch JCL delivered in member CMNALRPT in the CMNZMF CNTL library. You can concatenate multiple log backup files in the input to this job.

This report will not show activity recorded in the VSAM log file since the last log backup. If you want to see current ChangeMan ZMF activity, see Browsing Current VSAM LOG File.

Follow these steps to run the batch Log Activity Report with the JCL in CMNALRPT.

1. Copy member CMNALRPT from the vendor CMNZMF CNTL library to your custom CNTL library.
2. Modify member CMNALRPT in your custom CNTL library.
   a. Code your JOB statement information at the top of the JCL.
   b. Change the library names in the STEPLIB statement to your names for the vendor CMNZMF and SERCOMC LOAD libraries.
   c. Code one or more log backup files at DD name LOGQSAM. Concatenate the oldest backup file on top, the newest backup at the bottom.
   d. Code control statements at the SYSIN DD statement to produce the report you want. This table describes control statements for the batch job.

> **Tip**
>
> An easy way to create JCL and control statements for the batch Log Activity Report is to specify the filter criteria and activity codes on the ISPF panels described in Running Log Activity Report Using One Log Backup. Submit the report online, and then use SDSF command SJ on the job output to display the generated JCL. Copy that JCL into your custom CNTL library and modify the log backup files at the LOGQSAM DD statement to include the backup files that contain the activity you want to report.

CTL	Panel Field	Description
FDT=	FROM DATE	Start date for selection period in YYYYMMDD format. FROM DATE selection is inclusive. Leaving the date blank is the same as typing 00000101.
FTM=	FROM TIME	Start time for selection period in hhmmss format. FROM TIME selection is inclusive. Leaving the time blank is the same as typing 000000.
TDT=	TO DATE	End date for selection period in YYYYMMDD format. TO DATE selection is inclusive. Leaving the date blank is the same as typing 99991231.

CTL	Panel Field	Description
TTM=	TO TIME	To time for selection period in hhmmss format. TO TIME selection is exclusive; times are selected up to but not including the specified time. Leaving the time blank is the logical equivalent of specifying 235959+000001.
USR=	USER ID	User ID or mask in activity records to select.
PKG=	PACKAGE ID	Application or package ID to select in the format *aaaannnnnn*, where: - *aaaa:* - Application mnemonic to select application administration activity - *aaaannnnnn:* - Package ID to select package activity If you code a mask in *nnnnnn*, (like ACTP00001 `*`), only package activity is selected. If you code a mask in *aaaa* (like ACT `*`), package activity and application administration activity is selected.
LOG=nn	CODE	01 Backout a Package 02 Install a Package 03 Temporary Change Cycle 04 Distribute a Package 05 Unauthorized Member Access 07 Generate Package Information 08 Delete a Package 09 Update Application Information 10 Revert a Package 11 Update Global Information 12 Activate a Component 13 Package Memo Delete 14 Undelete a Package 15 Baseline Ripple 16 Reverse Baseline Ripple 18 Aged Installed Package 20 Approve a Package 21 Re-Synch Calendar 22 Age Staging Libraries 30 Reject a Package 38 HLLX administration 39 HLLX commands 40 Freeze a Package 42 Selectively Unfreeze a Package 43 Demote a Component

CTL	Panel Field	Description
		44 Demote a Package 47 Approve an online form 48 Promote a Package 49 Promote a Component 50 Audit a Package 51 Alter audit return code 57 Submit package audit autoresolve 59 Comment an online form 60 Link a Package 61 Deactivate an online form 62 Unlink a Package 63 Edit an online form 64 Scratch a Component 65 Reject an online form 66 Rename a Component 67 Relink a Component 68 Component Copied 70 File Tailoring Started 71 File Tailoring Failed 72 File Tailoring Completed 76 Submit an online form 77 Unlock an online form 80 Create a Package 82 Checkout a Component 83 Potential Checkout Conflict 84 Stage a Component 85 Overlay Previous Module 86 Delete Component From Package 88 Copy Forward a Package 90 Monitor Limbo and Internal Scheduler

CTL	Panel Field	Description
		ERO Codes** 23 Backout A Release 24 Install A Release 25 Distribute a Release 26 Delete a Release 27 Revert a Release 28 Approve a Release 29 Reject a Release 45 Promote a Release Area 46 Demote a Release Area 52 Audit a Release Area 53 Approve a Release Area 54 Reject a Release Area 55 Block a Release Area 56 Unblock a Release Area 58 Submit release audit autoresolve 78 Checkin to Release Area Completed 79 Retrieve from Release Area Completed 81 Checkin Component to Release 87 Checkout Component from Release 89 Retrieve Component from Release 91 Update Global Release Approvers 92 Update Release Definitions 93 Update Release Applications 94 Attach a Package to a Release 95 Detach a Package from a Release

3. Submit the job.
4. When the job ends, select the SYSOUT data set at DD name LOGRPORT to see the **Log Activity Report**.

29. Managing Packages with the Package List:

The Change Package List provides you with an alternative method for performing the ChangeMan ZMF functions presented in the previous chapters.

About the Package List

Using the Package List, you can manage one or more packages and their components through the package life cycle without using the multi-level menus described previously. Each package function in the life cycle is invoked by entering a two-character option on the line command for a package on the Package List.

Accessing the Package List

Use one of these methods to display the **Package List Parameters** panel, which is the first panel that leads you to the Package List.

- **Using the Menu Hierarchy:**

  On the **Primary Option Menu**, select **5 List**.

- **Using Direct Panel Access:**

  From any ChangeMan ZMF panel, enter **=5** in the **Command** or **Option** line and press **Enter**.

Selecting Packages for the Package List

Before you act on packages on the Package List, you must specify what packages are to be selected for the list.

1. Access the **Package List Parameters** (CMNLIST0) panel.

```
CMNLIST0                            Package List Parameters
Command ===> _________________________________________________________

 Package . . . . . . . . . ACTP0000*          (Full name or pattern; blank for list,
                                               or '*' for all packages)
 Enter "/" to select:
     Package status . . . . __ Dev    __ Frz    __ Apr    __ Rej    __ Dis    __ Ins
                            __ Bas    __ Bak    __ Del    __ Opn    __  Clo   __ Tcc
                            __ Aip    __ Xds    __ Xin    __ Xbk    __ Xrv
     Package level . . . .  __ Simple  __ Super __ Complex  __ Participating
     Package type . . . .   __ Planned  Permanent     __ Planned Temporary
                            __ Unplanned  Permanent  __ Unplanned Temporary

 Creator . . . . . . . . . ______________
 Work request . . . . . .  ______________
 Department . . . . . . .  ____
 Install date: from . . .  20200606_   (yyyymmdd)
               to . . . . _________   (yyyymmdd)
 Creation date: from . . . _________   (yyyymmdd)
               to . . . . _________   (yyyymmdd)

 Enter "/" to select option
    __ Other parameters
```

This table describes the fields on the **Package List Parameters** panel.

Field	Description
Package	Enter one of the following: - **Package:** Select one package regardless of any other selection criteria entered on the Package List Parameters panel. (You can omit leading zeros in the package number.) If a change package ID is entered and that package level is either SUPER or COMPLEX, its PARTICIPATING packages are listed as well. The PARTICIPATING packages are filtered by the search criteria entered. - **Application:** Select all packages in this application. - **Package Return:** Use `*` or ? as wildcard characters to specify Pattern Package IDs to select. See Building Lists Using Patterns. **blank:** The Application List panel is displayed for you to select an application. When you select an application from the Application List panel, package selection is executed immediately without returning to the Package List Parameters panel.

Field	Description
Package status	Enter a / against a package statuses to select: **Dev:** Development - Change packages where components are still being changed. **Frz:** Frozen - Packages that are frozen. **Apr:** Approved - Packages that are approved and are awaiting distribution for installation and/or baseline ripple. **Rej:** Reject - Packages that are rejected by an approver. **Dis:** Distributed - Packages that are distributed to the sites where they are to be installed. **Ins:** Install - Packages that are installed in the sites specified in the change package. **Bas:** Baseline - Packages that completed their life cycle with components rippled through the application baseline libraries. **Bak:** Back Out - Packages that are installed and baselined, and then whose components have been removed from production and baseline libraries with prior versions restored. **Opn:** Open - Complex or super change packages that have been created to relate participating packages. **Clo:** Closed - Complex or super packages whose participating packages have been baselined. **Tcc:** Temporary Change Cycle - Temporary change packages that have been installed and then deleted from temporary production libraries. **Del:** Delete - Change packages that are memo deleted.**AIP:** Approval In Process**XDS:** Package distribution has failed**XIN:** Package installation has failed**XRV:** Package revert has failed**XBK:** Package backout has failed
Creator	Enter a TSO-ID to select.
Work request	Enter a work request to select. This is a 12-character free-form field.
Department	Enter a department to select. This is a 4-character free-form field.
Package level	Select with / package levels required: - Simple packages - Super packages - Complex packages - Participating packages
Package type	Enter package types to select

Field	Description
From install date	Enter the low end of a range of package install dates to select.This date is inclusive. Blank is the same as 00000000.
To install date	Enter the high end of a range of package install dates to select.This date is inclusive. Blank is the same as 99999999.
From creation date	Enter the low end of a range of package create dates to select.This date is inclusive. Blank is the same as 00000000.
To creation date	Enter the high end of a range of package create dates to select.This date is inclusive. Blank is the same as 99999999.
Other parameters	Select this option to display panel Extended Search Criteria (CMNLIST8) for filtering by package component types.

2. If you select option **Other Parameters** field on the P**ackage List Parameters** panel, the **Extended Search Criteria** panel is displayed.

```
CMNLIST8                           Extended Search Criteria
Command ===> ______________________________________________________

Enter "/" to select:
  Component records         / Nonsource   _Source    _Load
                              Rename      _Scratch   _Form

  Scheduling system        / CMN   _Manual  _Other

Component type . . . . _____
Promoted Component . . ___      (1-YES, 2-No)

Component name list (Blank, Full name or Mask separated by ;)
____________________________________________________________________

Component userid list (Blank, Full name or Mask separated by ;)
____________________________________________________________________

Release id list (Blank, Full name or Mask separated by ;)
____________________________________________________________________
```

The following table describes the fields on this panel.

Field	Description
Component records	Select package component category.
Scheduling system	Select installation scheduler.
Component type	Select component type or pattern.
Promoted Component	Select whether to include packages that have promoted components.
Component name list	Enter a list of component names or patterns to select, separated by semicolons (;). The maximum length of each component name or mask is 16 characters.
Component user id list	Enter TSO id or pattern separated by semicolons (;).

Field	Description
Release id list	Enter release id or patterns to select, separated by semicolons (;).

All field entries on the **Extended Search Criteria** panel are optional. If you select any field, all change packages in the system that satisfy the specified selection criteria are presented.

3. Make your selections on the **Extended Search Criteria** panel and press **Enter**. The **Change Package List** panel is displayed showing the packages that meet your selection criteria.

```
CMNLIST3                       Change Package List                        Row 1 to 17 of 17
Command ===>                                                               Scroll ===> CSR

    Package     Sta Install   Lvl  Type    Work request   Dept  Promote      Aud Creator
 __ ACTP000001  BAS 20141121  SMP  PLN/PRM                IDD   00               USER016
 __ ACTP000002  BAS 20150104  SMP  PLN/PRM 123            BOO   00           00  USER015
 __ ACTP000003  BAS 20150105  SMP  PLN/PRM 123            BOOK  00           00  USER015
 __ ACTP000005  BAS 20150105  SMP  PLN/PRM 123            BOOK  00           00  USER015
 __ ACTP000013  BAS 20150105  SMP  PLN/PRM                IDD   00           04  USER016
 __ ACTP000018  BAS 20150105  SMP  PLN/PRM                IDD   00           00  USER016
 __ ACTP000020  BAS 20150104  SMP  PLN/PRM 100001000106   IDD   00           04  USER015
 __ ACTP000021  BAS 20150106  SMP  PLN/PRM                IDD   00           00  USER016
 __ ACTP000023  BAS 20150105  SMP  PLN/PRM 100001000106   IDD   00           04  USER015
 __ ACTP000024  BAS 20141218  SMP  PLN/PRM 100001000106   IDD   00           00  USER015
 __ ACTP000025  BAS 20150105  PAR  PLN/PRM 100001000106   IDD   00           00  USER015
 __ ACTP000027  DEV 20150228  SMP  PLN/PRM 100001000106   IDD   00               USER015
 __ ACTP000028  DIS 20150228  SMP  PLN/PRM 100001000106   IDD   00           00  USER015
 __ ACTP000032  DEV 20180318  SMP  PLN/PRM                IDD   20 SERT6         USER016
 __ ACTP000033  DEV 20180318  SMP  PLN/PRM                IDD   00           12  USER016
 __ ACTP000038  DEV 20150218  SMP  PLN/PRM                IDD   10 SERT6         USER016
 __ ACTP000039  BAS 20150129  SMP  PLN/PRM 100001000106   IDD   00           00  USER015
****************************** Bottom of data ********************************
```

This table describes the fields on the **Change Package List** panel.

Field	Description
COMMAND	Enter COMMAND to display the Valid Package List Commands panel, which explains all of the other commands that can be typed on the Command Line. Valid commands include:
	COMMAND: Display a list of valid entries you can use in the Command line.
	OPTIONS: Display a list of valid options you can use on the line command for packages.
	REFRESH: Select packages again by the Package List Parameters entered previously and show updated package information.
	LOCATE: Locate a listed package by information in the last sorted column or by Package ID if the packages were not sorted.
	SORT: Sort listed packages by information under the heading specified column heading, then by package ID. Valid headings include: PACKAGE, STA, INSTALL, LVL, TYPE, WORK, DEPT, AUD, CREATOR. You can sort listed packages by several promotion fields:

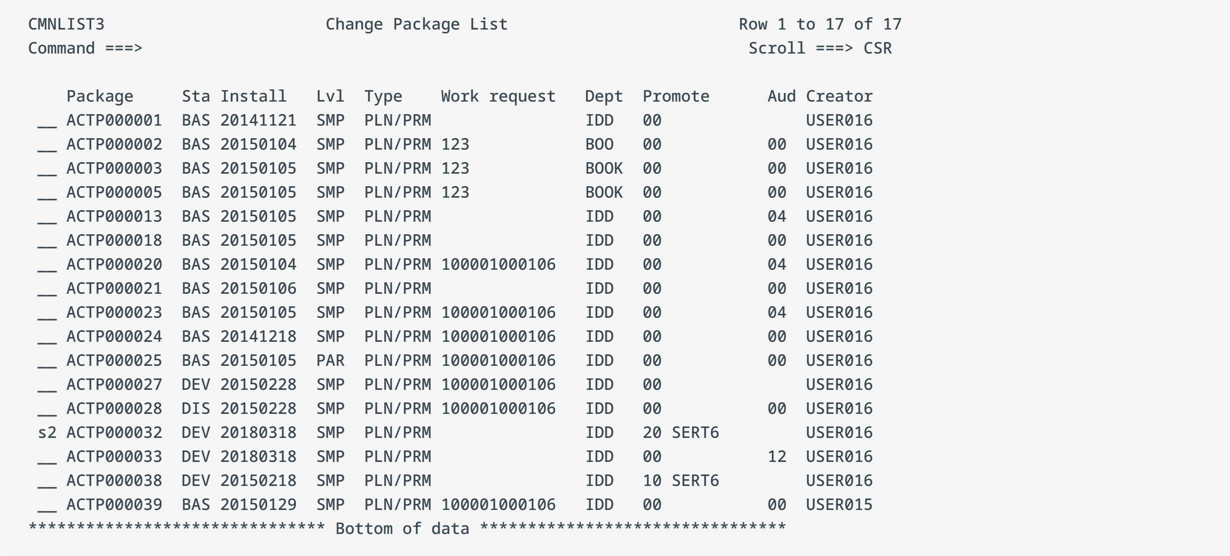
Field	Description
	- **SORT:** Sort by promotion site, promotion PROMOTE level, package ID SORT PROMO SORT Sort by promotion site, package ID. PROMSIT SORT Sort by promotion level, package PROMLVL ID SORT Sort by promotion level nickname, PROMNME package ID LONG Display the package list in long mode, with two lines of information for each package. SHORT Display the Package List in short mode with one line of information per package. DATES Display the Installation Calendar CREATE Bring up the Create A New Package panels. CANCEL Cancel processing and exit from the Package List SORT and LOCATE do not operate on fields displayed under the LONG list.
Line Command	Enter a package list option. To display the Package List Option panel, which shows valid options and allows you to enter the command from that panel, do one of the following: - Enter OPTION in the Command line. - Enter an invalid option, or type nonsense characters such as / or ? or 99 on the line command for a package.
Package id	Displays the package ID of selected packages.
STA	Displays the package status.
Install	Displays the earliest Install Date among all install sites selected for this package.
LVL	Displays the package level. SMPL Simple package CMPX Complex package SUPR Super package PART Participating package
Type	Displays a combination of the package type and package time span. PLN/ Planned permanent package PRM PLN/ Planned temporary package TMP UNP/ Unplanned permanent package PRM UNP/ Unplanned temporary package TMP
Work request	Displays information in the 12-character Work Request field.
Dept	Displays information in the 4-character Department field.
Promote	Displays the promotion level and the first seven characters of the promotion site name where the most recent full promote or “first promote” was successfully completed, without a subsequent full demote. After first promote, only the results of full promote/demote actions are reflected in the Promote field. Selective actions do not affect the value of this field. When you fully demote from any site, a promotion history list is scanned again to find the most recent remaining full promote or “first promote” to update the values displayed in the Promote fields. Special values: Blank This package was never promoted. 00 This package was promoted, then fully demoted from STAGING all sites.
AUD	Displays the last audit return code recorded in the package master for this package. If the package is changed so that another audit is required, the last AUD return code is blanked out.

Field	Description
Creator	Displays the first seven characters of the UserID of the person who created the package.
The following information displays on a second line for each listed change package if LONG is entered on the Command Line:	
PACKAGE TITLE	Displays the package title.

Invoking Package Functions from the Package List

To initiate a ChangeMan ZMF function for a package on the **Change Package List** panel, type a two-character Package List Option on the line command for a package and press Enter. For most options, the first data entry panel for the selected function is displayed, or a confirmation panel is displayed.

This example shows the option for Stage Package Components (s2) entered for package ACTP000032.



```
CMNLIST3                       Change Package List                        Row 1 to 17 of 17
Command ===>                                                               Scroll ===> CSR

    Package     Sta Install   Lvl  Type    Work request   Dept  Promote      Aud Creator
 __ ACTP000001  BAS 20141121  SMP  PLN/PRM                IDD   00               USER016
 __ ACTP000002  BAS 20150104  SMP  PLN/PRM 123            BOO   00            00  USER016
 __ ACTP000003  BAS 20150105  SMP  PLN/PRM 123            BOOK  00            00  USER016
 __ ACTP000005  BAS 20150105  SMP  PLN/PRM 123            BOOK  00            00  USER016
 __ ACTP000013  BAS 20150105  SMP  PLN/PRM                IDD   00            04  USER016
 __ ACTP000018  BAS 20150105  SMP  PLN/PRM                IDD   00            00  USER016
 __ ACTP000020  BAS 20150104  SMP  PLN/PRM 100001000106   IDD   00            04  USER016
 __ ACTP000021  BAS 20150106  SMP  PLN/PRM                IDD   00            00  USER016
 __ ACTP000023  BAS 20150105  SMP  PLN/PRM 100001000106   IDD   00            04  USER016
 __ ACTP000024  BAS 20141218  SMP  PLN/PRM 100001000106   IDD   00            00  USER016
 __ ACTP000025  BAS 20150105  PAR  PLN/PRM 100001000106   IDD   00            00  USER016
 __ ACTP000027  DEV 20150228  SMP  PLN/PRM 100001000106   IDD   00               USER016
 __ ACTP000028  DIS 20150228  SMP  PLN/PRM 100001000106   IDD   00            00  USER016
s2 ACTP000032  DEV 20180318  SMP  PLN/PRM                IDD   20 SERT6         USER016
 __ ACTP000033  DEV 20180318  SMP  PLN/PRM                IDD   00            12  USER016
 __ ACTP000038  DEV 20150218  SMP  PLN/PRM                IDD   10 SERT6         USER016
 __ ACTP000039  BAS 20150129  SMP  PLN/PRM 100001000106   IDD   00            00  USER015
******************************* Bottom of data ********************************
```

This example used the line command of s2 (as shown above), and ENTER, resulting in the display of the **Stage: package Components** panel for the package selected (ACTP000032).

```
CMNSTG01                        STAGE: ACTP000032 Components                  Row 1 to 3 of 3
Command ===>                                                                 Scroll ===> CSR

Name               + Type Status      Changed                   Procname  User      Request
ACPSRCEE             SRC  ACTIVE      20150210 151804           CMNCOB2   USER015
COB001               SRC  ACTIVE      20150201 220059           CMNCOB2   USER015
CPY001               CPY  ACTIVE      20150201 215848                     USER015
******************************* Bottom of data ********************************
```

Alternatively you can type options in the line command on the Change Package List panel (CMNLIST3) for more than one package at a time. The packages are processed serially, starting with the package highest on the list.

In many places in ChangeMan, a list of available options is obtained by entering an invalid command, e.g. a '?', on the line command, and in this case (instead of s2), on panel CMNLIST3 Change Package List and ENTER.

This displays the **Package List Options** menu.

```
CMNLIST5 Package List Options INVALID LINE COMMAND
Option ===>

 AP Audit package             A1 Approve/reject           A2 Reject reasons
 A4 Reset apr in progress     A5 Re-submit JCL build      BB Browse baseline
 BL Browse listings           B1 Backout package          CC Compare staging/baseline
 C1 Checkout(baseline)        C2 Checkout(package)        C3 Checkout(release)
 C4 Checkout(from pkg)        DM Demote package           D1 Memo delete
 D2 Undo memo delete          F1 Freeze(online)           F2 Freeze(batch)
 F3 Unfrz/refrz comp          F4 Reset frz indicators     F5 Resubmit JCL build
 OF Online forms              PR Promote package          QP Query package
 RC Recompile                 RL Re-bind                  RP Reports
 RV Revert package            SC Scan baseline            S1 Stage(dev lib)
 S2 Stage(package)            S3 Stage(package select)    UA Rename/scratch(pkg)
 UB Rename/scratch(bas)       UD Update Db2 info          UI Update IMS inf
 UR Update release info       U1 Update control info      U2 Update general info
 U3 Update implementation     U4 Update scheduler         U5 Update aff. appl.
 U6 Update prt. pkg. info     U7 Update site info         U8 Close complex/super
 U9 Open complex/super        UE Update user info         ZP Compress pkg libs
```

You can press **PF3** to return to the **Change Package List** panel, or you can type a two- character option in the **Command** line and press **Enter** to proceed to the desired function.

When you finish with a package function, return to the **Change Package List** by pressing **PF3** or by typing **End** on the **Command** line and pressing **Enter**.

Package List Options

This table lists all of the two-character options that you can use on the **Change Package List** panel.

Option	Description
AP	Audit package
A1	Approve/Reject package
A2	Display approver's reject reasons

Option	Description
A4	Reset Approval-In-Progress indicator
A5	Re-submit installation JCL build request for package in APR status
BB	Browse/Print Baseline/Promotion libraries
BL	Browse compressed listing
B1	Back out package
CC	Compare components in staging library against baseline/promotion library
C1	Check out components (baseline/promotion)
C2	Check out components (package list)
C3	Check out components (from release)
C4	Check out components (from package)
DM	Demote package
D1	Memo delete package
D2	Undelete a memo deleted package
F1	Freeze package online
F2	Freeze package in batch
F3	Selectively unfreeze or refreeze package components
F4	Reset Freeze-In-Progress indicator
F5	Re-submit installation JCL build request for package in FRZ status
OF	Work with Online Forms
PR	Promote package
QP	Query package information
RC	Recompile source code from baseline/promotion libraries
RL	Relink load module
RP	Generate application/package reports
RV	Revert package back to DEVelopment status
SC	Scan baseline for character strings
S1	Stage components from an external data set
S2	Stage components already in package
S3	Specify parameters to select a subset of package components
UA	Manage Rename or Scratch request in package

Option	Description
UB	Create Rename or Scratch request for baseline component
UD	Update package Db2 information
UI	Update package IMS information
UR	Update package Release information
U1	Update package control information
U2	Update package description
U3	Update package installation instructions
U4	Update package install scheduling dependencies for external scheduler
U5	Update participating package affected applications
U6	Update participating package information
U7	Update package site information and install date
U8	Update complex/super package to CLO status
U9	Update complex/super package to OPN status
UE	Update package user information
ZP	Compress package libraries

30. Package Information Panels

This appendix shows the panels that appear when you query a package and then make a selection on the **Package Information** panel (CMNQRY03). See Querying Packages and Components.

General

Control Information

```
CMNQRY10                             Control Information
Command ===> ____________________________________________________________________________

        Package: DEMO000190                Status: DEV             Install date: 20220210
   Complex/super: N/A Status:

Package title
Demo package

Application . . . . . . . . .   DEMO
Requestor name . . . . . . .    John Doe
Requestor phone . . . . . . .   (555) 555-5555
Work request . . . . . . . .
Department . . . . . . . . .    IDD
Package level . . . . . . . .   1         (1-Simple, 2-Complex,
                                           3-Super, 4-Participating)
Package type . . . . . . . .    PLANNED   (Planned or Unplanned)
Package time span . . . . . .   PERM      (Permanent or Temporary)
Unplanned reason code . . . .
Temporary change duration . .             (In days)
Notify user . . . . . . . . .   DOE
```

Package Description

```
CMNPGNL2           QUERY: Package Description             Row 1 to 36 of 46
Command ===>                                                Scroll ===> CSR

         Package: ACTP000032 Status: DEV Install date: 20180318
  Description:
  This package includes changes to the mainframe server components of the
  Accounts Payable system that enable improvements to the Web Services
  Facilities.
```

Installation Instructions

```
CMNPGNL3             QUERY: Installation Instructions           Row 1 to 30 of 46
Command ===>                                                     Scroll ===> CSR

          Package: ACTP000032       Status: DEV        Install date: 20180318
Contingency . . . . 1  1-Hold production and contact analyst
                       2-Backout change, continue production
                       3-Other:

Scheduler . . . . . MANUAL (CMN, Manual or Other)

   Instructions:
   If there are problems with the installation of this package, contact the
   on-call change control analyst. If there are problems with Accounts
   Payable application processing, contact the project lead.
```

Non-Source

```
CMNSTG01            QUERY: ACTP000032 Components               Row 1 to 1 of 1
Command ===>                                                  Scroll ===> CSR

    Name           + Type  Status   Changed           Procname User     Request
 __ CPY001           CPY   ACTIVE   20150201  215848           USER015
******************************* Bottom of data *******************************
```

Valid Commands include DCD, LOCATE, LONG, SHORT, SORT and XLONG.

Valid line commands are: B (Browse), H (History), S (Options), UL (User List), V (View) or VC (Version Control).

Source

```
CMNSTG01             QUERY: ACTP000032 Components             Row 1 to 2 of 2
Command ===>                                                 Scroll ===> CSR

    Name         + Type Status    Changed            Procname  User    Request
 __ ACPSRCEE       SRC  ACTIVE    20150201 215924    CMNCOB2   USER015
 __ COB001         SRC  ACTIVE    20150201 220059    CMNCOB2   USER015
******************************* Bottom of data *******************************
```

Valid Commands include DCD, LOCATE, LONG, SHORT, SORT and XLONG.

Valid line commands are:

B (Browse) BL (Browse Listing) CB (Compare package component with its baseline version) H (Hist) SL (SRC to LOD) S (Comp/Linkopts) UL (Usr list) V (View) VC (Version Control) VL (View Listing)

Source to Load Relationship

```
CMNQRY21              Source to Load Relationship             Row 1 to 4 of 4
Command ===>                                                 Scroll ===> CSR

        Package: ACTP000032      Status: DEV       Install date: 20180318
Source         Load
Name +   Type  Name   +  Type  Status     Promotion     Changed         User
ACPSRCEE SRC   ACPSRCEE LOD    ACTIVE     20 S6P1IT     20150201 215924 USER015
               ACPSRCEE LST    ACTIVE      0 STAGING    20150201 215928 USER015
COB001   SRC   COB001   LOD    ACTIVE     20 S6P1IT     20150201 220059 USER015
               COB001   LST    ACTIVE      0 STAGING    20150201 220104 USER015
******************************* Bottom of data *******************************
```

Component Userid Work List

```
 CMNUCWK1                Component Userid Work List               Row 1 to 15 of 25
 Command ===>                                                      Scroll ===> HALF

          Package: ACTP000043       Status: DEV        Install date: 20100430

   Component  Name    Type  Status   User    Last     Last     Action    Actioned
                     +                       date     time     request   count
__ ACPCPY00   CPY                    USER239 20100426 115639   CHECKOUT  0004
__ ACPCPY1A   CPY                    USER239 20100426 115611   STAGED    0002
__ ACPCPY1B   CPY           DELETED  USER239 20100426 154156   DELETED   0003
__ ACPCPY1C   CPY           DELETED  USER239 20100426 154159   DELETED   0003
__ ACPCPY1X   CPY           DELETED  USER239 20100426 154202   DELETED   0003
__ ACPCPY5A   CPY           DELETED  USER239 20100426 154205   DELETED   0003
__ ACPCPY5B   CPY           DELETED  USER239 20100426 154208   DELETED   0003
__ ACPCPY5C   CPY           DELETED  USER239 20100426 154212   DELETED   0003
__ ACPCTL10   CTL                    USER239 20100426 153859   CHECKOUT  0001
__ ACPCTL10   JCL           DELETED  USER239 20100427 002955   DELETED   0002
__ ACPJCL10   JCL                    USER239 20100426 200519   CHECKOUT  0012
__ ACPJCL10   JCL                    USER240 20100425 173934   DELETED   0002
__ ACPJCL20   JCL           DELETED  USER239 20100426 154215   DELETED   0003
__ ACPPRC10   PRC                    USER239 20100426 114421   STAGED    0002
__ ACPSRCD1   SRC                    USER239 20100426 114246   CHECKOUT  0001
```

Valid Commands include LOCATE, SORT.

Rename/Scratch List

```
CMNQRY12                    Rename/Scratch List                   Row 1 to 3 of 3
Command ===>                                                     Scroll ===> CSR

              Package: ACTP000032     Status: DEV       Install date: 20180318

Req   Name        Rename     Type   Changed            User      Status
REN   ACPCPYCB    ACPCOPCB   CPY    20150112 220021    JPREST2   ACTIVE
REN   ACPCPYCF    ACPCOPCF   CPY    20150112 210056    JPREST2   ACTIVE
REN   ACPSRC91    ACSORC91   SRC    20150112 212904    JPREST2   ACTIVE
******************************* Bottom of data *******************************
```

Approval List

```
CMNAPPLS                        Approval List                Row 1 to 3 of 3
Command ===>                                                Scroll ===> CSR

         Package: ACTP000032       Status: DEV       Install date: 20180318
    Approver Description                User
                                        Date     Time      Seq     Status
 __ Development Team Lead               USER015
                                        20150129 0231      10      Approve
 __ Quality Assurance                   USER015
                                        20150129 0231      20      Approve
 __ Program Manager                     USER015
                                        20150129 0231      30      Approve
******************************* Bottom of data *******************************
```

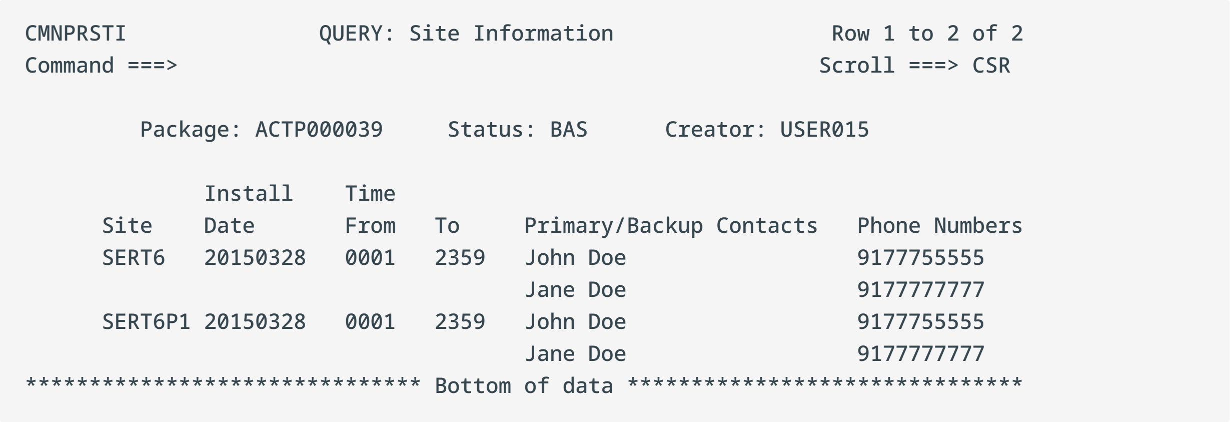
Valid line commands are: S (select).

Select one and the Approver Details panel (CMNQAPL1) is displayed:

```
CMNQAPL1           ACTP000039 - Approver Details          Row 1 to 33 of 35
Command ===>                                                Scroll ===> CSR

Approver . . . Development Team Lead
Entity . . . . DEVLEAD
Order no . . . 10

      Vehicle User(s) to Notify
```

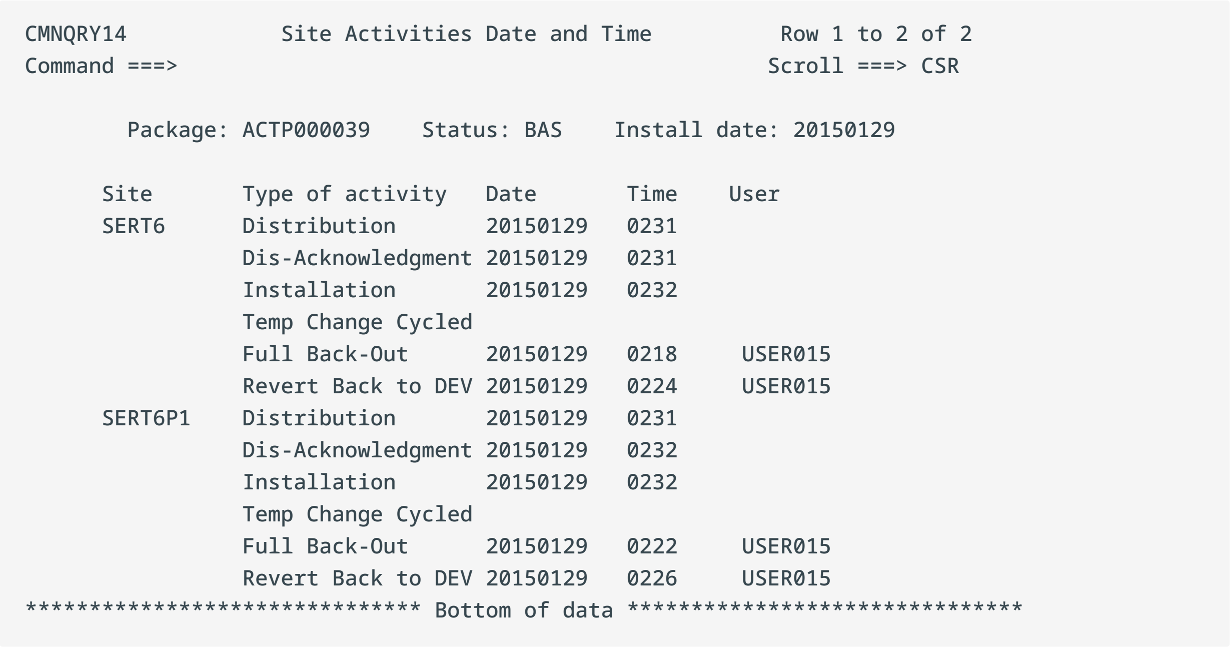
Site and Install Date Information

If your administrator set up your ChangeMan ZMF environment as an All site, the Update Site Information panel is displayed. Otherwise, the Query Remote Site Information panel is displayed.



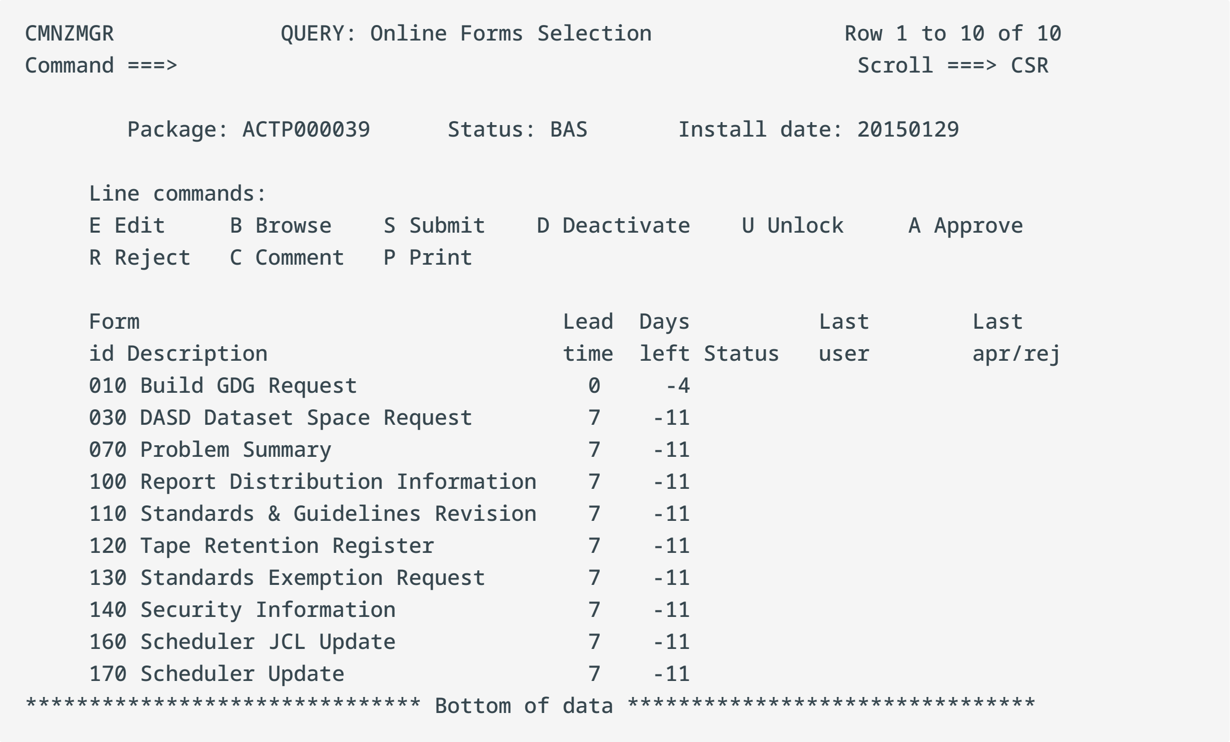
```
CMNPRSTI               QUERY: Site Information                Row 1 to 2 of 2
Command ===>                                                 Scroll ===> CSR

         Package: ACTP000039     Status: BAS      Creator: USER015

              Install    Time
      Site    Date       From   To     Primary/Backup Contacts   Phone Numbers
      SERT6   20150328   0001   2359   John Doe                  9177755555
                                       Jane Doe                  9177777777
      SERT6P1 20150328   0001   2359   John Doe                  9177755555
                                       Jane Doe                  9177777777
******************************* Bottom of data *******************************
```

Site Activities Date and Time



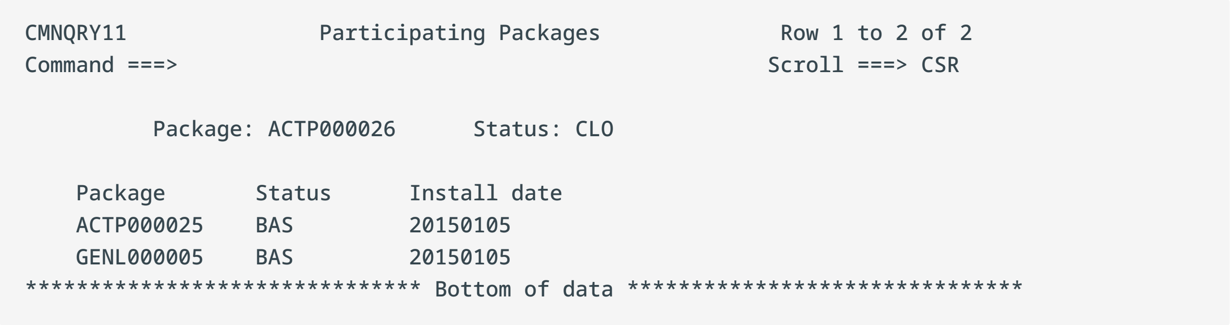
```
CMNQRY14            Site Activities Date and Time          Row 1 to 2 of 2
Command ===>                                              Scroll ===> CSR

        Package: ACTP000039    Status: BAS    Install date: 20150129

      Site       Type of activity   Date       Time    User
      SERT6      Distribution       20150129   0231
                 Dis-Acknowledgment 20150129   0231
                 Installation       20150129   0232
                 Temp Change Cycled
                 Full Back-Out      20150129   0218     USER015
                 Revert Back to DEV 20150129   0224     USER015
      SERT6P1    Distribution       20150129   0231
                 Dis-Acknowledgment 20150129   0232
                 Installation       20150129   0232
                 Temp Change Cycled
                 Full Back-Out      20150129   0222     USER015
                 Revert Back to DEV 20150129   0226     USER015
******************************* Bottom of data *******************************
```

Custom Forms



```
CMNZMGR             QUERY: Online Forms Selection               Row 1 to 10 of 10
Command ===>                                                     Scroll ===> CSR

        Package: ACTP000039      Status: BAS       Install date: 20150129

     Line commands:
     E Edit     B Browse    S Submit    D Deactivate    U Unlock     A Approve
     R Reject   C Comment   P Print

     Form                                 Lead  Days          Last        Last
     id Description                       time  left Status   user        apr/rej
     010 Build GDG Request                  0     -4
     030 DASD Dataset Space Request         7    -11
     070 Problem Summary                    7    -11
     100 Report Distribution Information    7    -11
     110 Standards & Guidelines Revision    7    -11
     120 Tape Retention Register            7    -11
     130 Standards Exemption Request        7    -11
     140 Security Information               7    -11
     160 Scheduler JCL Update               7    -11
     170 Scheduler Update                   7    -11
******************************* Bottom of data ********************************
```

Participating Packages



```
CMNQRY11               Participating Packages              Row 1 to 2 of 2
Command ===>                                              Scroll ===> CSR

          Package: ACTP000026      Status: CLO

    Package       Status      Install date
    ACTP000025    BAS         20150105
    GENL000005    BAS         20150105
******************************* Bottom of data *******************************
```

Status Start Date and Time

```
CMNQRY04                   Status Start Date and Time
Command ===>

         Package: ACTP000038     Status: DEV      Install date: 20150218

                   Status      Date      Time      User
                   Dev         20150114  114822
                   Rev-Dev     20150115  003403    USER015
                   Frz         20150115  003138
                   Apr
                   Rej
                   Dis
                   Dis-Ack
                   Ins
                   Tcc
                   Bas
                   Bak
```

Promotion History

```
CMNQRY26              ACTP - Promotion Site List                Row 1 to 3 of 3
Command ===>                                                   Scroll ===> CSR


         Package: DAVE000192        Status: FRZ         Install date: 20221201

            Demote reqd         Reset after         Promote beyond
    Site    prior sites         demotion            package level
    DAVE        N                   Y                   Y
    LOCAL       N                   N                   N
    MP3000      N                   N                   N

******************************* Bottom of data ********************************
```

Type S next to a site press Enter to display the history of promotion and demotion of this package at the selected site.

```
CMNQRY27          Promotion History for SERT6          Row 1 to 6 of 6
Command ===>                                          Scroll ===> CSR

        Package: ACTP000038     Status: DEV      Install date: 20150218

Count   Promotion   Date        Time    User      Status     History
10      10  S6P1UT  2015/02/01 22:16   USER015   Completed  Full Promote
4       20  S6P1IT  2015/02/01 22:15   USER015   Completed  Full Demote
4       20  S6P1IT  2015/02/01 21:40   USER015   Completed  Full Promote
4       10  S6P1UT  2015/02/01 19:53   USER015   Completed  Full Promote
2       10  S6P1UT  2015/02/01 19:52   USER015   Completed  Full Demote
2       10  S6P1UT  2015/02/01 19:35   USER015   Completed  Full Promote
******************************* Bottom of data *******************************
```

Press Enter again to view the promotion status of package components.

```
CMNQRY28         Component Promotion Status for SERT6         Row 1 to 10 of 10
Command ===>                                                   Scroll ===> CSR

         Package: ACTP000038       Status: DEV       Install date: 20150218
Name      + Type     Promotion     Date        Time     User        Status
ACPSRCCE    LOD     10 S6P1UT     2015/02/01  22:16    USER015
ACPSRCCE    SRC     10 S6P1UT     2015/02/01  22:16    USER015
ACPSRCEE    LOD     10 S6P1UT     2015/02/01  22:16    USER015
ACPSRCEE    SRC     10 S6P1UT     2015/02/01  22:16    USER015
ACPSRC1A    LOD     10 S6P1UT     2015/02/01  22:16    USER015
ACPSRC1A    SRC     10 S6P1UT     2015/02/01  22:16    USER015
ACPSRC6A    LOD     10 S6P1UT     2015/02/01  22:16    USER015
ACPSRC6A    SRC     10 S6P1UT     2015/02/01  22:16    USER015
ACPSRC92    LOD     10 S6P1UT     2015/02/01  22:16    USER015
ACPSRC92    SRC     10 S6P1UT     2015/02/01  22:16    USER015
******************************* Bottom of data *******************************
```

Promotion Libraries

```
CMNQRY26                DEMO - Promotion Site List              Row 1 to 2 of 2
Command ===> ___________________________________________________Scroll ===> CSR

          Package: DEMO000182     Status: DEV        Install date: 20200120

            Demote reqd
  Site      prior sites
__D001          N
__LOCAL         Y
******************************* Bottom of data *******************************
```

Type S next to a *remote site* to view package library information, then press Enter. The following panel is displayed.

```
CMNQRY30            ACTP/SERT6 - Promotion Levels             Row 1 to 3 of 3
Command ===>                                                 Scroll ===> CSR

           Package: ACTP000038      Status: DEV      Install date: 20150218

 Nickname      Entity       Level       Procedure
 S6P1UT        ACTPDEV      10          CMN$$PRM
 S6P1IT        INTCOORD     20          CMN$$PRM
 S6P1AT        UATCOORD     30          CMN$$PRM
******************************* Bottom of data *******************************
```

Type S next to the promotion levels that you want to view, then press Enter.

```
CMNQRY31         ACTP/SERT6 - Promotion Libraries              Row 1 to 5 of 5
Command ===>                                                  Scroll ===> CSR

Promotion name: S6P1UT       Level: 10

          Package: ACTP000038      Status: DEV       Install date: 20150218

      Syslib
Lib   exclude    Target libraries     +
CTC     Y        CMNTP.S6.V810.PROM.S6P1UT.CTC                        Shadow
                 CMNTP.S6.V810.PROM.S6P1UT.CTC                        Library 1
                                                                      Library 2
                                                                      Library 3
JCL     Y        CMNTP.S6.V810.PROM.S6P1UT.JCL                        Shadow
                 CMNTP.S6.V810.PROM.S6P1UT.JCL                        Library 1
                                                                      Library 2
                                                                      Library 3
LOD     Y        CMNTP.S6.V810.PROM.S6P1UT.LOD                        Shadow
                 CMNTP.S6.V810.PROM.S6P1UT.LOD                        Library 1
                                                                      Library 2
                                                                      Library 3
PRC     Y        CMNTP.S6.V810.PROM.S6P1UT.PRC                        Shadow
                 CMNTP.S6.V810.PROM.S6P1UT.PRC                        Library 1
                                                                      Library 2
                                                                      Library 3
SRC     Y        CMNTP.S6.V810.PROM.S6P1UT.SRC                        Shadow
                 CMNTP.S6.V810.PROM.S6P1UT.SRC                        Library 1
                                                                      Library 2
                                                                      Library 3
******************************* Bottom of data *******************************
```

Development Staging Libraries

```
CMNQRY05             Development Staging Libraries              Row 1 to 3 of 3
Command ===>                                                   Scroll ===> CSR

         Package: ACTP000038        Status: DEV        Install date: 20150218
Lib  Development staging library     +
LST  CMNTP.S6.ACTP.STG6.#000038.LST
LOD  CMNTP.S6.ACTP.STG6.#000038.LOD
SRC  CMNTP.S6.ACTP.STG6.#000038.SRC
******************************* Bottom of data *******************************
```

Production Staging Libraries

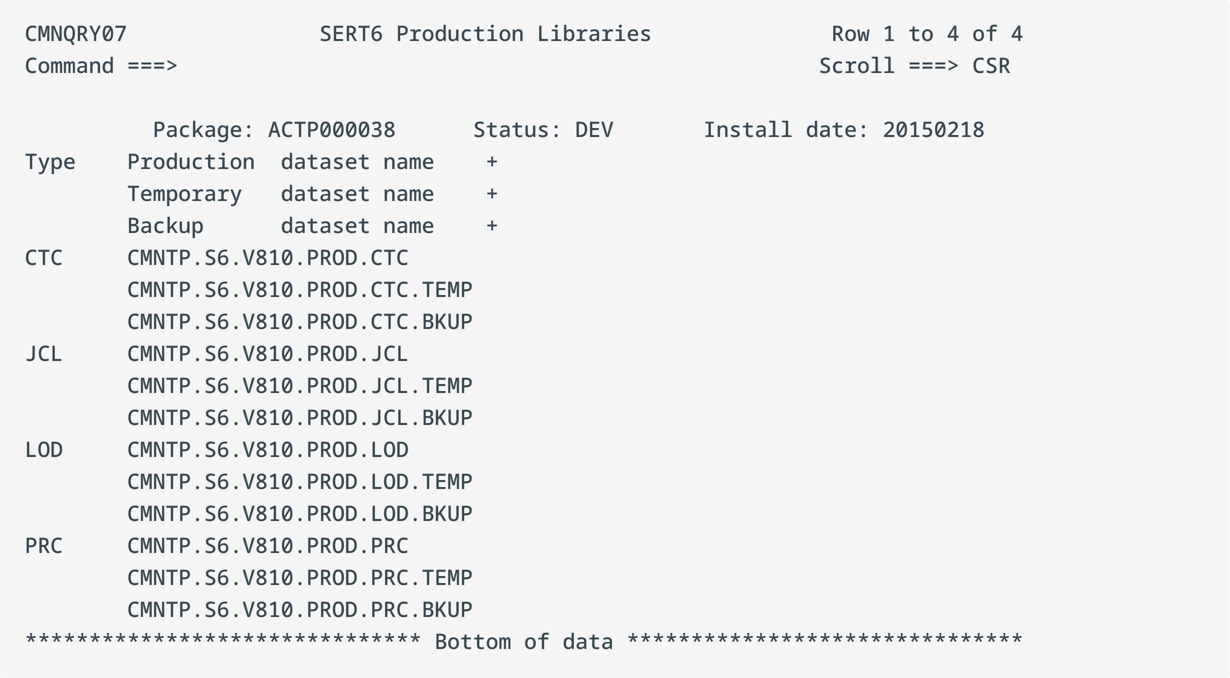
```
CMNRMTSL                ACTP - Site Selection List                Row 1 to 1 of 1
Command ===>                                                     Scroll ===> CSR

    Site Name
 __ SERT6
 __ SERT6P1
******************************* Bottom of data *******************************
```

```
CMNQRY06               Production Staging Libraries             Row 1 to 3 of 3
Command ===>                                                   Scroll ===> CSR

          Package: ACTP000038        Status: DEV       Install date: 20150218
Lib Production staging library name       +
LST CMNTP.S6.ACTP.STG6.#000038.LST
LOD CMNTP.S6.ACTP.STG6.#000038.LOD
SRC CMNTP.S6.ACTP.STG6.#000038.SRC
******************************* Bottom of data *******************************
```

Production Libraries

```
CMNRMTSL                 ACTP - Site Selection List               Row 1 to 1 of 1
Command ===>                                                     Scroll ===> CSR

    Site Name
 __ SERT6
 __ SERT6P1
******************************* Bottom of data *******************************
```

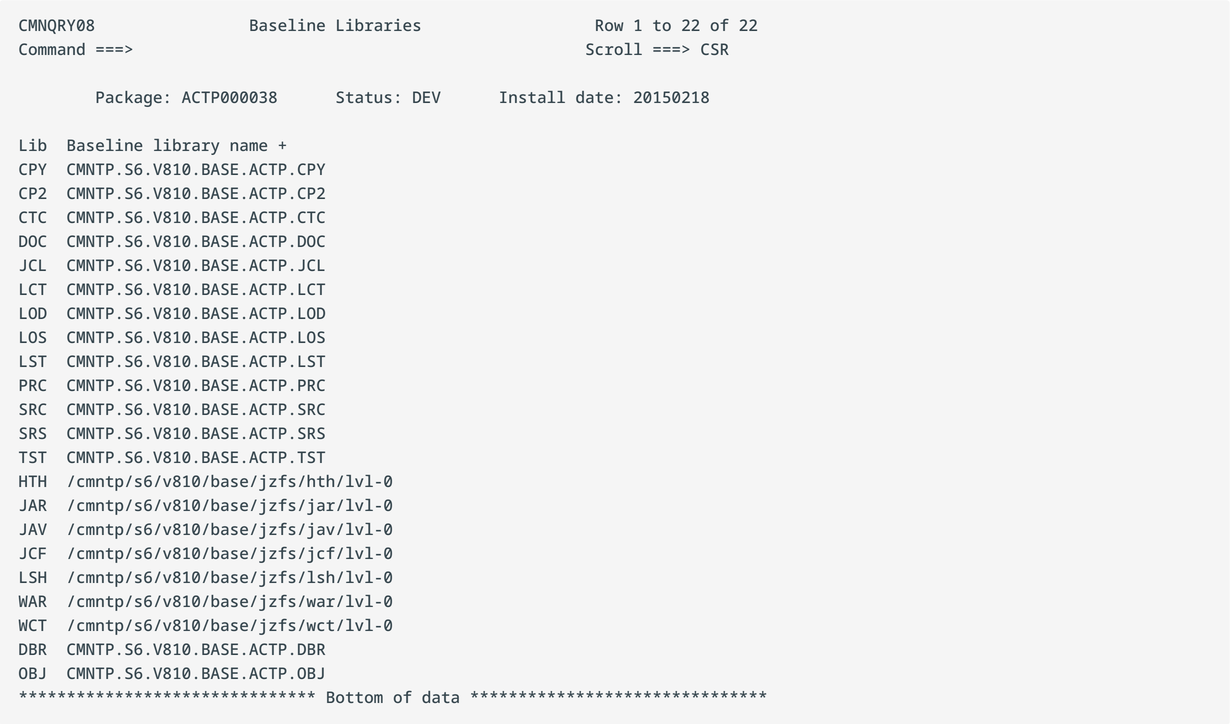
If your administrator set up your ChangeMan ZMF environment as an All site, the Query Production Libraries panel (CMNQRY07) is displayed.



```
CMNQRY07              SERT6 Production Libraries              Row 1 to 4 of 4
Command ===>                                                 Scroll ===> CSR

          Package: ACTP000038      Status: DEV       Install date: 20150218
Type    Production  dataset name    +
        Temporary   dataset name    +
        Backup      dataset name    +
CTC     CMNTP.S6.V810.PROD.CTC
        CMNTP.S6.V810.PROD.CTC.TEMP
        CMNTP.S6.V810.PROD.CTC.BKUP
JCL     CMNTP.S6.V810.PROD.JCL
        CMNTP.S6.V810.PROD.JCL.TEMP
        CMNTP.S6.V810.PROD.JCL.BKUP
LOD     CMNTP.S6.V810.PROD.LOD
        CMNTP.S6.V810.PROD.LOD.TEMP
        CMNTP.S6.V810.PROD.LOD.BKUP
PRC     CMNTP.S6.V810.PROD.PRC
        CMNTP.S6.V810.PROD.PRC.TEMP
        CMNTP.S6.V810.PROD.PRC.BKUP
******************************* Bottom of data *******************************
```

Otherwise, ChangeMan ZMF displays the Remote Site Selection List so you can choose a remote site, and then displays the Query Production Libraries panel.

Baseline Libraries



```
CMNQRY08                Baseline Libraries                Row 1 to 22 of 22
Command ===>                                             Scroll ===> CSR

        Package: ACTP000038      Status: DEV      Install date: 20150218

Lib  Baseline library name +
CPY  CMNTP.S6.V810.BASE.ACTP.CPY
CP2  CMNTP.S6.V810.BASE.ACTP.CP2
CTC  CMNTP.S6.V810.BASE.ACTP.CTC
DOC  CMNTP.S6.V810.BASE.ACTP.DOC
JCL  CMNTP.S6.V810.BASE.ACTP.JCL
LCT  CMNTP.S6.V810.BASE.ACTP.LCT
LOD  CMNTP.S6.V810.BASE.ACTP.LOD
LOS  CMNTP.S6.V810.BASE.ACTP.LOS
LST  CMNTP.S6.V810.BASE.ACTP.LST
PRC  CMNTP.S6.V810.BASE.ACTP.PRC
SRC  CMNTP.S6.V810.BASE.ACTP.SRC
SRS  CMNTP.S6.V810.BASE.ACTP.SRS
TST  CMNTP.S6.V810.BASE.ACTP.TST
HTH  /cmntp/s6/v810/base/jzfs/hth/lvl-0
JAR  /cmntp/s6/v810/base/jzfs/jar/lvl-0
JAV  /cmntp/s6/v810/base/jzfs/jav/lvl-0
JCF  /cmntp/s6/v810/base/jzfs/jcf/lvl-0
LSH  /cmntp/s6/v810/base/jzfs/lsh/lvl-0
WAR  /cmntp/s6/v810/base/jzfs/war/lvl-0
WCT  /cmntp/s6/v810/base/jzfs/wct/lvl-0
DBR  CMNTP.S6.V810.BASE.ACTP.DBR
OBJ  CMNTP.S6.V810.BASE.ACTP.OBJ
******************************* Bottom of data *******************************
```

31. Glossary

ChangeMan ZMF Application

User software project or domain being implemented. Named and defined by the application administrator; can be a mnemonic. User restrictions are set for it. Type in the exact name of the application in an application, or select from a list of applications by typing a pattern, or use masking.

Possible References: APPL, PROJECT, LOCAL

Change Package

A group of changes to user software. Three levels include simple, super/ complex, and participating. Made up of members of partitioned data sets such as JCL, source code, copybooks, and so on.

Compile Procedure

Used to process special components like source (SRC). Related to the language name of a component. Choose from a list of available procedures for a component by leaving the compile procedure field blank. (If there is only one compile procedure defined for a certain language, this field cannot be modified.)

Use compile procedures and options designated by your application administrator by typing a ?. The appropriate procedure and options are displayed in the other fields.

Possible References: PROCEDURE, COMPILE PROC, PROCEDURE NAME

Component

A single member of a change package; usually a member of a library that was brought into ChangeMan ZMF for modification.

Component Library Type

List of available library types set by the application administrator; can be a combination of reserved, like, or custom library types.

- **Reserved** – component types internally defined
- **Like** – components that act similar to ChangeMan ZMF reserved components

- **Custom** – components designed for unique needs

Compile procedures may be designated for components and are associated with the language name associated with the component type.

Possible References: COMPONENT TYPE, LIBRARY TYPE, TYPE.

Component Name

Name of the component selected from a list of components (you may need to provide some qualifying information for the list, such as a library type or application name), or type a pattern (signified by *, following the relevant portion of the application name) or a blank in the component name field.

Possible References: COMPONENT, MEMBER, MODULE

Component Version (*VV.MM*)

Component version and modification number using IBM standards (*mm* starts at 01, increments to 99 and stays there; *vv* starts at 01, increments to 99 and then recycles to 01). The first time the component is processed by ChangeMan ZMF, the *vv.mm* is 01.01. Each successive stage request (in the same change package) increments the *mm* portion (01.02, 01.03, and so on.). The next package that uses the component causes the *vv* portion to be incremented (02.01).

Confirmation Requests

Tell ChangeMan ZMF whether or not major operations prompt before running the function.

- **YES** – a confirmation is required when overlaying a target file when running checkout.
- **NO** – the function should be run without a user confirmation.

An all-or-none parameter. Cannot require confirmation for some packages and components and not for others.

Data Set Organization (DSORG)

A library type of organization:

- **PDS** – partitioned data set (Basic Partitioned Access Method {BPAM})
- **SEQ** – sequential file (Queued Sequential Access Method {QSAM})
- **PAN** – CA Panvalet library
- **LIB** – CA Librarian master
- **OTHER** – any other type of library organization

Not a required field; if left blank, ChangeMan ZMF determines the DSORG.

Possible References: ORGANIZATION, DSORG

ID or TSO ID

TSO USERID that performed a function.

Possible References: TSOID, USERID

Install Job Scheduler

Controls the submission of the package installation batch jobs. The choice of installation scheduler is controlled by the global administrator. This field may also contain a default value if the administrator set the value when setting up ChangeMan ZMF.

The global administrator sets these parameters in part 1 of the Global Parameters Panels (CMNGGP01).

Change this field by typing in the scheduler of choice, if it has been authorized by global administration:

- **CMN** – ChangeMan ZMF internal scheduling.

  The ChangeMan ZMF started task schedules the submission of the package installation jobs. See the *ChangeMan ZMF Administrator Guide* for details on these capabilities.

- **MANUAL** – Manually control the submission of installation jobs.

  The installation process begins as soon as the package is fully approved. Package installation can be inhibited by holding the last approval or installed immediately by approving the package before the scheduled installation date and time.

- **OTHER** – An external scheduler such as CA7, ADC2, CA-Scheduler, Control- M, and so on.

  The ChangeMan ZMF started task performs a batch interface to add the package install job information to the external scheduler's database.

Install Date

Date on which to install your change package.

Language Name (Language Assumption)

Programming language associated with a component that requires special processing such as the source (SRC) component. Language name is associated with each language and is customized predefined compile procedures.

Important when performing checkout or stage of components.

ChangeMan ZMF lets you choose a language (from a list of available ones) or use the one that the administrator designated. Type the language in the Checkout panel.

If component is not new and has a checkout history, history records on the component characteristics were saved. ChangeMan ZMF determines the language name of a component:

- Designation – initiates Designated Procedures force option level 2
- Specified – checks if a specified language name is used for the components
- History – if history exists for the components
- Designation – designated language for the component type
- Last Used Language

ChangeMan ZMF uses the language of the first item in the list for all subsequent components in the list.

If procedure has force level 1, a user who wants to freeze a change package with the selected component, must perform the last stage request with the designated procedure. ChangeMan ZMF allows staging with alternate procedures during package development, but the final stage (before attempting to freeze) must be performed with the designated compile procedure for each component. If force level 2 is chosen, then a user who wants to stage the component must use the designated compile procedure. The application administrator may have used extra compile and/or link options.

See the *ChangeMan ZMF Administrator Guide* for Designated Procedures and Component Level Security.

Mode

An on-line function or a batch mode. In batch mode, ChangeMan prompts for batch job card information in an additional panel; only fill it in the first time unless there a change is necessary.

Possible References: ONLINE BROWSE.

Package ID or Name

ChangeMan ZMF assigns a unique ID, which is a combination of the application name and a unique number. Assigned IDs are needed to update or review the change package. When typing the package IDs, key in the application mnemonic, but truncate the number to only the relevant part:

- CGM 000012 — rendered as CGM 12, or CGM 012, or CGM 00012, or as shown.
- XYZV130056 — type it, as shown.
- UHRD004061 — rendered UHRD4061, or UHRD04061, or as shown.
- @13 045300 — rendered @13 45300, or as shown.
- $B29000246 — rendered $B29246, or $B290246, or $B2900246, or as shown.

This field may be editable. Key in the change package name or access a list of change packages. At other times it is provided for reference.

Possible References: PKG NAME, PACKAGE NAME, PKG ID, PACKAGE ID

Package Level

Change package levels in ChangeMan ZMF indicate the complexity and are designated when the package is created. Different levels have different behavior and creation requirements.

- **Simple** – a component of a complex or a super change package.

  Does not affect any other application, or does not require changes to software or operational procedures.

- **Super and Complex** – You are creating the “parent” change package for two or more change package applications that have related or interdependent changes, or have major changes to application processing which result in a major impact on the data processing environment.

  Super and complex change packages contain only control, general information, and a list of the participating packages. There are no staging libraries associated with super and complex change packages. Remote site and the installation dates for each site are tracked in the participating change packages.

  Once created, super or complex packages are automatically marked for limbo causing them to immediately display in the Monitor Packages in Limbo option. It is from this monitor function that their status is changed to closed.

  A super change package is functionally equivalent to a complex package. Use super to draw attention to, or segregate, packages.

- **Participating** – A participating package is a variation of a simple package that allows an association to one or more additional participating packages. By using participating packages you can:

  - Account for other packages' copybooks or load module staging libraries during compile or link processing.
  - Combine the approvers of different applications (Not applicable to participating packages in the same application).
  - Participating change packages must be created before being listed as participating in the parent change package.

  Your package’s copybook and load module staging libraries are always concatenated before other participating packages in the SYSLIB DDNAME of compile/link steps. The order of the other packages is based upon the order of the packages listed in the complex packages.

Permanent or Temporary

A permanent change package is enduring and is deleted only if requested by a user. It can be rippled into the baseline library. A temporary package is not permanent and is never rippled into the baseline library. It is automatically deleted from production after a specified number of days.

Planned or Unplanned

A planned change package is created during normal business hours (defined in global parameters) in a non-emergency situation. Its implementation is subject to the complete Approval List. Its scheduling for installation is governed by the Planned Installation Calendar.

An unplanned change package is created for an emergency situation, often outside normal business hours. Its implementation is subject to the complete Approval List if it is created during normal business hours. If it is created outside of normal business hours, its approval is based on the abbreviated Approval List. Its installation date is not dependent on the Planned Installation Calendar. Data Set Organization (DSORG).

- Use a planned permanent change package to schedule and install permanent updates to production software.
- Use a planned temporary change package to set up and run special one- time or short-term processing (that is, conversion, parallel processing, special reports, or extractions).
- Use an unplanned permanent change package to make emergency fixes to production software which must be executed for multiple cycles of processing.
- Use an unplanned temporary change package to make short term or one- time-only emergency fixes to production software.

Remote Site

Use to indicate which remote sites are affected by a function. Provide remote site information when creating change packages and specifying where to install the package.

32. Legal Notice

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.

Contains Confidential Information. Except as specifically indicated otherwise, a valid license is required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.