CA Endevor® Software Change Manager

Integration for the Natural Environment User Guide

Version 17.0.00

CA Technologies Product References

This document references the following CA products:

- CA Endevor® Software Change Manager (CA Endevor SCM)
- CA Endevor® Software Change Manager Integration for the Natural Environment (CA Endevor Integration for the Natural Environment)

Contact CA Technologies

Contact CA Support

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

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

Providing Feedback About Product Documentation

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

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

Documentation Changes

Note: This edition of the documentation includes changes that were added after Version 16.0 General Availability. In the list below, the edition number refers to the edition of this guide and indicates when the change was added to this guide.

Version 17.0

- Saved SCL Objects (see page 46), Submit Actions in Batch (see page 63)— Updated to add a note about the Add and Retrieve actions not being processed by the Saved SCL submit function.

Version 16.0 Second Edition

For Version 16.0 PTF RO65655, December 2013:

- Main Menu Options and Processing Menu Objects (see page 19)— Added to discuss menu customization.
- Save an API Message Report (see page 19), Transfer an Element (see page 44), Submit Saved SCL Objects (see page 48), Submit Actions in Batch (see page 63)— Updated various topics to add that the EINE administrator may have restricted the user's ability to modify the job card when using the Batch Job Submittal window.
- How Retrieving an Element from an Inventory Location Work (see page 23)— Updated to add how renaming the object affects Retrieve action processing.
- Retrieve an Element from an Inventory Location (see page 24)— Updated to add the Y option on the selection list, which lets you rename the element being retrieved. Updated to describe the results of a batch request.
- Add/Update an Object from Natural or Predict (see page 28)— Updated to add the Y option on the selection list, which lets you rename the element being added or updated. Updated to describe the results of a batch request.
- Submit Actions in Batch (see page 63)— Updated to describe load step processing for Natural programming elements selected for Retrieve processing, including the effects of the SYSOBJH LOAD options.

Version 16.0

- Display Online Help (see page 18)— Updated regarding how to use the question mark (?) to access help.
- Submit Actions in Batch (see page 63)— Updated to add that the JCL for submitting actions in batch includes Retrieve check steps.
- vvll (see page 88)— Updated this glossary term to change the version and level format to VVLL,

Version 15.0

- How Transferring an Element Works (see page 44)–Added a topic to describe what effects the transfer action causes.
- Transfer an Element (see page 44)–Added a topic to describe the transfer action procedure in foreground or batch.
- Saved SCL Objects (see page 46)–Added a topic to describe the enhancement that lets you build, save, and edit SCL.
  - Access Saved SCL Objects (see page 46)–Added a topic to describe the procedure to access saved SCL objects on the new Endevor Saved SCL panel.
  - Delete SCL Objects (see page 47)–Added a topic to describe the procedure to delete saved SCL objects from the Endevor Saved SCL panel.
  - Edit SCL Objects (see page 47)–Added a topic to describe the procedure to edit saved SCL objects from the Endevor Saved SCL panel.
  - Submit Saved SCL Objects (see page 48)–Added a topic to describe the procedure to submit saved SCL objects from the Endevor Saved SCL panel.
- Submit Actions in Batch (see page 63)–Added information about new options (PF11 SCL and PF4 Save) on the Batch Processing Menu that let you build SCL from scratch and save SCL.
- Build Package SCL (see page 53)–Added information on how to import saved SCL into current SCL while in Edit mode.
- Package Processing–Added the following topics to describe the procedures for the package actions that were added to the CA Endevor Integration for the Natural Environment:
  - Reset a Package (see page 54)
  - Review a Package (see page 55)
  - Execute a Package (see page 57)
  - Commit a Package (see page 58)
  - Delete a Package (see page 59)
  - Display Package Approvers (see page 61)
  - Display Cast Reports (see page 62)
  - Display SCL (see page 63)

Contents

Chapter 1: Understanding the Software

This section contains the following topics:

What You Need to Know

To use CA Endevor Integration for the Natural Environment, you need a working knowledge of the mainframe environment, the z/OS mainframe operating system, Time Sharing Option facility (TSO), the Interactive System Productivity Facility (ISPF), and the Natural programming environment. It is also assumed that CA Endevor SCM and CA Endevor Integration for the Natural Environment have been properly installed and configured at your site.

CA Endevor SCM implementations are site-specific. Your application development manager or CA Endevor SCM administrator should provide you with an explanation of how the development lifecycle and inventory structure are implemented at your site.

CA Endevor SCM Functions

For each development task done in CA Endevor Integration for the Natural Environment, CA Endevor SCM performs the following functions:

Source management

Maintains the source base and delta libraries.

Inventory management

Maintains the Master Control File (MCF) definitions.

Output management

Creates and maintains outputs, including source and object code of Natural programming objects, listings and so on.

How CA Endevor SCM Classifies Elements

CA Endevor Integration for the Natural Environment allows application developers working in the Natural environment to perform CA Endevor SCM actions on Natural and Predict objects (known as elements).

CA Endevor SCM identifies elements according to the logical structure implemented at your site. All elements are identified by a fully-qualified name consisting of environment, stage, system, subsystem, type, and element name. Each element is defined by the following:

- Location in the software lifecycle. This is determined by the environment and stage where the element resides. As an application is modified, elements are moved through the software development lifecycle, with those elements residing in different functional locations at different times. Those functional areas, for example, Test, QA, Production, Backup, and so on, are defined as an environment and stage combination. The location of an element is part of the identification of that element.
- Inventory classification. This is determined by the system, subsystem, and type with which the element is associated. Systems are functional groupings of the Natural code applications, such as Finance applications, Tax applications, and so on. A subsystem is a specific application or release. The type classification identifies the element type such as a Natural or Predict object.

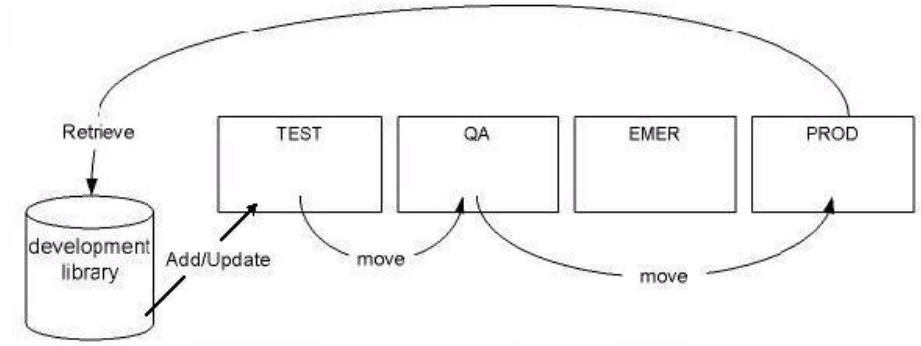
Your Natural administrator and CA Endevor SCM administrator have also implemented a physical data set structure to support your site's logical structure.

A Typical Software Lifecycle

A typical software lifecycle that CA Endevor SCM manages consists of the following *stages*:

- Test - Applications are unit tested in this stage.
- Quality Assurance - Applications are system tested in this stage.
- Emergency - Fixes are applied to production code in this stage.
- Production - Production applications are stored in this stage.

The following diagram shows the typical change procedures in a software lifecycle.



In this lifecycle, new or existing code is developed in the development library. After the code has been unit tested by the application developer, the code is added to the Test stage for unit testing. When the code is ready for system testing, it is moved to the Quality Assurance stage. Finally, when the code is ready to be used by customers, the code is moved to the Production stage. Any fixes are applied to production code in the Emergency stage.

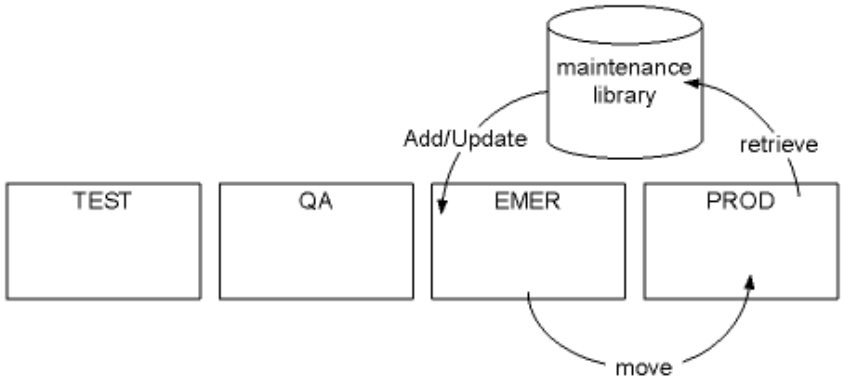
If your company requires approvals to move your software code changes through the lifecycle, you must use packages to perform the moves.

Emergency Operations

CA Endevor SCM can manage the following emergency operations when fixes must be applied to production code.

- Retrieving elements from production
- Making changes to elements
- Adding or updating elements into the emergency stage
- Moving elements to production

The following diagram shows emergency change procedures in a software lifecycle.



In this lifecycle, production code is retrieved from the Production stage, modified in the maintenance library, and then added back to the Emergency stage. After the changes are approved in the Emergency stage, the code is moved back to the Production stage.

If your company requires approvals to move your software code changes through the lifecycle, you must use packages to perform the moves.

Chapter 2: Getting Started Using the Software

This section contains the following topics:

Start a User Session

You can start CA Endevor Integration for the Natural Environment at any time to develop, manage, and control your software changes.

To start CA Endevor Integration for the Natural Environment from within your Natural session

1. Enter EINE on the command line and press Enter.

   The logon dialog opens.

2. Enter your user ID and password and press Enter.

   The CA Endevor Integration for the Natural Environment Main Menu opens.

3. Select an option from the menu.

Exit a User Session

You can exit CA Endevor Integration for the Natural Environment at any time from the Main Menu by pressing PF3 or entering an option code of a period (.).

To require a password to enter the next session, a user can override the user password timeout value by logging off using the option X on the Main Menu. After logging off on the Main Menu, the user is presented with the password screen when attempting to enter the next session, even if the password timeout period has not expired.

Field Default Values

To help you more easily and accurately enter field data, default values are used. When you enter a value in an Object/Element Name, Type, Environment, System, Subsystem, Stage, FUSER, or FDIC field, that value becomes the default when the same field appears on a different screen. Using default values saves you time and ensures that the same value is used when the field appears on multiple screens. If you need to specify a new default value for a field, enter the new value and submit your request.

Name Masking

To help you more easily find information and process requests, you can use name masking. By substituting a name with the asterisk (*) wildcard character, a character with the percent (%) sign placeholder, or by using both together, it is much faster and easier to find information and process requests.

You can use name masks for the following CA Endevor SCM items:

- Object/Element names
- Environment, system, subsystem, and stage names within FROM clauses

Note: Natural name masking conventions are honored for the Object field on the Natural Programming Object Add screen. The Natural name masking convention utilizes the question mark symbol (?) in place of the percent sign (%) as a placeholder character.

Wildcard (*)

Using an asterisk (*) as a wildcard character in a search string lets you find all elements or all elements that begin with the same characters you specify. You can use wildcards in the following ways:

- When the wildcard is used as the only character of a search string, all members of the search field are returned.
- When the wildcard is used as the last character of a search string, all members of the search field beginning with the characters in the search string preceding the wildcard are returned. For example, entering UPD*, returns all elements beginning with UPD, such as UPDATED or UPDATE.
- You cannot use more than one wildcard in the same search string. For example, entering U*PD* would result in an error.

Placeholder (%)

Using a percent sign (%) as a placeholder character in a search string lets you represent a wildcard for one character. You can use placeholders in the following ways:

- When the placeholder is used as the last character in a string, the search result includes all members of the search field, beginning with the characters in the search string preceding the placeholder, but which have no more characters than were used in the search string. For example, UPD% would return all elements beginning with UPD that are four characters in length such as UPD1 or UPDA.
- More than one placeholder can be used in the same search string. For example, U%PD% would return all elements with five-character names that have the letter U as the first character and PD as the third and fourth character.

Wildcard (*) and a Placeholder (%) Usage

You can use both a wildcard and one or more placeholders within the same search string, as long as the wildcard appears at the end of the search string and is used only once. For example, a search string of U%D* would return elements with names of any length that have U as the first character, any one character as the second character, and D as the third character.

Function Keys

There are several standard function keys that can help you move through CA Endevor Integration for the Natural Environment. The following function keys are available from many screens and windows:

ENTER (function key)

Processes the information on a detail screen or window. Scrolls down on a summary listing screen or window.

PF1 (function key) - Help

Accesses screen-level help or accesses selection-list help. See the Display Online Help section for details.

PF3 (function key) - End

Returns to the previous logical screen. From the Main Menu, returns to Natural.

PF6 (function key) - Top

Scrolls to the top of a summary listing.

PF7 (function key) - Up

Scrolls up one page in a summary listing.

PF8 (function key) - Down

Scrolls down one page in a summary listing.

PF9 (function key) - Bottom

Scrolls to the bottom of a summary listing.

Display Online Help

You can display CA Endevor Integration for the Natural Environment online help for more information about features, fields, and so on. The following types of help are available:

- To display value-selection list help, on any screen on which Element Type, Environment, System, Subsystem, or Stage values can be entered, enter a question mark (?) on a field and press Enter.
- To display screen-level help, make sure that the cursor is not positioned on a value-selection field such as Element Type, Environment, System, Subsystem, or Stage; and enter a question mark (?) on a field and press Enter, or press PF1.

API Message Reports

CA Endevor Integration for the Natural Environment processes all foreground action requests by invoking the CA Endevor SCM API. After you execute one or more foreground action requests, you will see this message at the top of the screen:

n total actions processed; press PF12 for API-message info

The message shows the total number (*n*) of actions that have been processed.

View an API Message Report

To view the report for a completed foreground action request, press PF12. A summary of all completed action requests appears, allowing you to enter any character next to an item to view its API Message report.

As you view the API Message report, you can scroll left or right using the PF10 and PF11 function keys.

For cases in which detailed tracing of the API structures is required, press PF2 (Trace) to view the content of the API control block, request block, and response block associated with the foreground action request.

Save an API Message Report

All API Message reports are automatically deleted when you return to the Foreground Processing Menu, so the PF4 (Output) option is provided to allow you to externally save an API Message report prior to its automatic deletion.

To send the API Message report to your system's batch output queue, press PF4 (Output). The Batch Job Submittal window appears, allowing you to customize the job card, if job card modifications have not been restricted by the EINE Administrator. Press Enter to submit the job or press PF3 to cancel the submission. Your API Message report will be written to the CMPRT01 dataset of the batch job.

Main Menu Options and Processing Menu Objects

The CA Endevor Integration for the Natural Environment administrator can customize the CA Endevor Integration for the Natural Environment Main Menu to remove the Packages option, or Saved SCL Objects option, or both from the menu. The administrator can also customize the Foreground Processing Menu and Batch Processing Menu to remove the DDMs, or Predict Objects, or both from the menus.

If any of these options or objects is missing from a menu, you can contact your local CA Endevor Integration for the Natural Environment administrator to confirm that the administrator intentionally customized the menu.

Chapter 3: Performing User Tasks

This section contains the following topics:

Development Tasks

When you perform development tasks in CA Endevor Integration for the Natural Environment, you perform CA Endevor SCM actions on Natural and Predict objects (known as elements). For example, to manage a program from a Natural library, you use the Add/Update Natural Programming Objects interface to submit an add action to CA Endevor SCM. The process that you use to perform a development task is similar for each element action. You specify the action (for example, add an element, move an element), enter information associated with the task you want to perform (for example, specify the element (object) name, system, and subsystem), and CA Endevor Integration for the Natural Environment passes the request to CA Endevor SCM to execute the action.

The following actions are supported:

- Retrieve
- Add/Update
- Move
- Delete
- Generate
- Signin
- Transfer
- Display element information
- Create package
- Cast package
- Review package
- Execute package
- Commit package
- Reset package
- Delete package

How Retrieving an Element from an Inventory Location Works

When you use CA Endevor Integration for the Natural Environment to retrieve an element from a CA Endevor SCM inventory location to a specified FUSER/Library (in the case of Natural programming objects or error messages) or to a specified FDIC (in the case of Predict objects or Natural DDMs) using the Retrieve Elements action, CA Endevor SCM performs the following actions and causes these effects:

1. Determines whether the element is signed out. If the element is signed out to someone else, and if you do not set the SIGNOUT ELEMENT option to N, you must set the OVERRIDE SIGNOUT option to Y to retrieve the element. However, your CA Endevor SCM administrator must have set up permissions for you to use the OVERRIDE SIGNOUT option.
2. Searches for the specified element, in the following order:
   a. At the location you specify in the RETRIEVE elements action
   b. In each stage in the environments on the map route

   Note: If CA Endevor SCM finds the element at a subsequent stage that is not part of the map, you will see a warning message in the API Message report. In addition, if you do not know the exact location of the element you want to retrieve, specify the lowest stage in the map route. CA Endevor SCM searches the entire map route for the element.
3. Copies the current level of the element to the specified FUSER/Library or FDIC.

   If the object currently exists in the target FUSER/Library or target FDIC, the Retrieve action fails unless the REPLACE IF PRESENT option was set to Y on the Retrieve Natural Programming Objects panel. If the action fails, you receive an error message.

   When the existence of the object is checked in the Natural library, if a new name for an object was requested, the new name is checked in the Natural library. (A new name can be specified from the selection list or using the New Object Name field on the Retrieve Natural Programming Objects panel.)

   Important: If more than one Retrieve action is requested against the same target object name in a single batch execution of CA Endevor SCM and the REPLACE IF PRESENT option is set to N to indicate no replacement should be done, then the second and all subsequent Retrieve actions will fail to replace the object in the Natural library. The reason being that the first request copied the element to the target so that it is now present at the target location. So, all subsequent requests fail, because of the REPLACE IF PRESENT option set to N restriction. *However, CA Endevor SCM will not issue an error message indicating the subsequent failures* unless the object existed in the target Natural library prior to the first Retrieve, in which case the first Retrieve also will have failed.
4. If you set the SIGNOUT ELEMENT option to Y, CA Endevor SCM updates the Master Control File (MCF) and signs the element out to you. If you set the SIGNOUT ELEMENT option to N, CA Endevor SCM stops processing the action after it creates the copy in step 3.

How Retrieving an Element from an Inventory Location Affects the Signout ID

Your CA Endevor SCM administrator can enable a signout capability on a system-by-system basis. When you retrieve an element from a CA Endevor SCM inventory location to a specified FUSER/Library or FDIC using the Retrieve Elements action, CA Endevor SCM signs the element out to you by recording your user ID in the SIGNOUT ID option on the Master Control File (MCF) for the stage from which the element was retrieved (only when you set the SIGNOUT ELEMENT option to Y).

How Retrieving an Element from an Inventory Location Affects CCIDs and Comments

When you retrieve an element from a CA Endevor SCM inventory location to a specified FUSER/Library or FDIC using the Retrieve Elements action, you can enter a CCID and comment. When you enter a CCID and comment, the Retrieve CCID/Comment is set. When you set the SIGNOUT ELEMENT option to Y, CA Endevor SCM records the CCID and comment you entered in the RETRIEVE CCID and COMMENT options in the Master Control File (MCF). If you set the SIGNOUT ELEMENT option to N, CA Endevor SCM does not update this information.

Retrieve an Element from an Inventory Location

Use the CA Endevor Integration for the Natural Environment Retrieve Elements action (available in foreground and batch) to perform the following actions:

- Retrieve a Natural programming object element or Natural error message element from a CA Endevor SCM inventory location to a specified FUSER/Library.
- Retrieve a Predict object element or Natural DDM element from a CA Endevor SCM inventory location to a specified FDIC.

To retrieve an element from an inventory location

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select either the Foreground or Batch processing option on the CA Endevor Integration for the Natural Environment Main Menu. Press Enter.

   The Foreground Processing Menu or the Batch Processing Menu opens.

3. Select an object type (Natural programming object, Error message text, DDM, or Predict object) and the retrieve action type (R) and press Enter.

   The retrieve interface (for the object type you selected) opens.

4. Choose one of the following steps to select the elements you want to retrieve and submit your request:

   a. To retrieve elements from a Retrieve Selection list, set the Select List option to Y, enter the information for the action, and press Enter.

      The Retrieve Selection screen opens.

      - To select the elements you want to retrieve, enter an X to the left of those element on the Retrieve Selection screen.
      - If you want to retrieve a prior level, enter an S to the left of an element on the Retrieve Selection screen. When the Summary Information for Element window opens, press PF4 to open the Element Level Selection List window. You can select a level for retrieval by entering an X to the left of that level.
      - Enter Y to the left of any object that you want to rename. A dialog window opens showing the old name of the object. You can enter a new name. If you previously specified a value in the New Object Name field on the Retrieve Natural Programming Objects panel, you can change that value in the dialog.

   b. To retrieve elements without a selection list, set the Select List option to N, enter the information for the action, and press Enter.

      In both Foreground and Batch modes, an Action Confirmation window opens telling you how many elements have been selected for retrieval. Press Enter to submit the retrieval requests.

5. In Foreground mode, CA Endevor Integration for the Natural Environment interfaces with CA Endevor SCM to retrieve the selected elements to the designated FUSER/Library or FDIC.

   In Batch mode, press PF5 from the Batch Processing Menu to submit the requests for batch processing. The batch request is processed as follows:

   a. For each Natural and Predict element to be retrieved, a batch Natural step is created to execute the EINE Natural program DNPC. DNPC checks the target location to determine if the Natural object or Predict object already exists in the Natural library or FDIC file, respectively. The program issues a message indicating whether or not the element was found and updates the temporary data created for the Retrieve action with the indicator.

b. When the EINE_ACTIVE=ON option is enabled in the CA Endevor SCM Optional Features table (ENCOPTBL), a CA Endevor SCM UEXIT02 is called that reads the temporary data set updated by DNPC when the CA Endevor SCM SCL is processed.

   - If the REPLACE IF PRESENT option is set to N on the Retrieve Natural Programming Objects panel and the object exists in the target location, CA Endevor SCM fails the Retrieve action and issues an RC = 12 with the following error message:

   ```
   C1U0970E  EXIT 2 - RETRIEVE ACTION DENIED - REPLACE=N SPECIFIED
   AND OBJECT EXISTS IN NATURAL LIBRARY
   ```

   - If the element does not exist in the target location or the REPLACE IF PRESENT option is set to Y, the appropriate level of the Natural or Predict element is written to the temporary data set.

c. If Retrieve action has not been failed by the UEXIT02, the Natural or Predict element is written to the temporary data set specified in the To clause and then loaded to the target location in separate batch Natural load steps.

Note: For more information on the specific details of the Retrieve action Load steps, see Submit Actions in Batch (see page 63).

How Adding/Updating a Member from Natural or Predict Works

When you allow CA Endevor SCM to control an object from Natural or Predict using the Add/Update Elements action, CA Endevor SCM performs the following actions and causes these effects:

1. Verifies that the element is signed out to you and does not exist in the entry stage.
2. Searches beyond the entry stage for an element with the same name.

   Note: Every stage in the mapped route is searched for a match. If the element is found in a stage in the map, CA Endevor SCM copies the current version of the element to the target entry stage, along with the last processor group name used for the element.
3. If you code the new version option, the element is not copied back to the entry stage. CA Endevor SCM assigns the version number you enter.

   Important! If CA Endevor SCM finds the element in a stage that is not included in the map, you will see a warning message in the API Message report. The search continues for an element in a stage that is included in the map.
4. Compares the object being added with the entry stage base. If the element is not found along the map, the element is created in the entry stage with a version number of 01 and base level of 00 (zero).
5. If CA Endevor SCM built an entry stage base, the object being added is compared to that base, and builds a new level with any changes. If no changes are detected, you will see a warning message in the API Message report.

6. Updates the Master Control File (MCF).

   Important! The value specified for Signout Upon Fetch (the SOFETCH parameter) in the CA Endevor SCM Defaults Table (C1DEFLTS) effects how the MCF will be updated for the element copied back (fetched). If Signout Upon Fetch is in effect, the element will be signed out to you unless it is already signed out to someone else. If Signout Upon Fetch is not in effect, the element will not be signed out to you.

7. Continues executing the action based on the value in the GENERATE ELEMENT option. If this option is set to N, CA Endevor SCM does not generate the element. If this option is set to Y, CA Endevor SCM performs the following actions:

   - Determines which processor group to use and executes the generate processor in that group (if one has been specified). After the generate processor has been run for the element, CA Endevor SCM updates the processor information in the Master Control File.
   - For Natural or Predict elements, the generate processor loads a Natural or Predict object into the FUSER/Library or FDIC that corresponds to the current CA Endevor SCM stage and subsystem. In the case of Natural programming objects, the generate processor also executes the CATALL utility to generate Natural executables.
   - The CATALL execution output is written to the list library defined for that subsystem.

How Adding/Updating an Object from Natural or Predict Affects the Signout Status

Your CA Endevor SCM administrator can enable a signout capability on a system-by-system basis. When you allow CA Endevor SCM to control an object from Natural or Predict using the Add/Update Elements action, CA Endevor SCM performs the following actions and causes these effects:

- If the element exists up the map, a fetch is performed.
  - If your CA Endevor SCM administrator has enabled Signout Upon Fetch, the fetched (source) element is signed out to you, if the element is not signed out to someone else. In addition, the target element is signed out to you.
  - If your CA Endevor SCM administrator has not enabled Signout Upon Fetch, there is no change in the signout status, and the target element is signed out to you.
- If the element does not exist up the map, a fetch is not performed.

Note: For more information, see Actions and Signout Status in the chapter "Action Processing" in the *CA Endevor Software Change Manager User Guide*.

How Adding/Updating an Object from Natural or Predict Affects CCIDs and Comments

When you allow CA Endevor SCM to control an object from Natural or Predict using the Add/Update Elements action, you can enter a CCID and comment. When you enter a CCID and comment, CA Endevor SCM sets the following:

If you manage a new element:

- Current Source CCID/Comment
- Generate CCID/Comment (if changed)
- Last Action CCID/Comment
- Source Delta CCID/Comment
- Component Delta CCID/Comment (if generated)

If you manage an existing element:

- Current Source CCID/Comment (if changed)
- Generate CCID/Comment (if generated)
- Last Action CCID/Comment
- Source Delta CCID/Comment
- Component Delta CCID/Comment (if the generate creates a delta)

Note: If you set the GENERATE ELEMENT option to N, the Add/Update Elements action does not set the generate or component delta CCIDs and comments.

Add/Update an Object from Natural or Predict

To control and manage an object from Natural or Predict (Natural programming object, Natural error message, Natural DDM, or Predict object), you use the Add/Update Elements action.

Important! When you enter information for the Add/Update Elements action, if you enter or select an object name that already exists, the action is rejected or the previous object is replaced with the one you are currently adding (based on the value in the UPDATE IF PRESENT option).

To add/update an object from Natural or Predict

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select either the Foreground or Batch processing option and press Enter.

   The Foreground Processing Menu or the Batch Processing Menu opens.

3. Select an object type (Natural programming object, Error message text, DDM, or Predict object) and the add/update action type (A) and press Enter.

   The add/update interface (for the object type you selected) opens.

4. Choose one of the following steps:

   a. To add/update objects from an Add/Update Selection list, set the Select List option to Y, enter the information for the action, and press Enter.

      The Add/Update Selection screen opens.

      - Enter X to the left of any object you want to add/update.
      - Enter Y to the left of any object that you want to rename. A dialog window opens showing the name of the object. You can enter a new name for the element. If you previously specified a value in the Element Name field on the Add/Update Natural Programming Objects panel, you can change that value in the dialog.

   b. To add/update objects without a selection list, set the Select List option to N, enter the information for the action, and press Enter.

      In both Foreground and Batch modes, an Action Confirmation window opens telling you how many objects have been selected for add/update. Press Enter to submit the add/update requests.

      Note: If you are adding an element for the first time, the comment becomes the element description in the Master Control File.

5. In Foreground mode, CA Endevor Integration for the Natural Environment interfaces with CA Endevor SCM to add/update the selected objects.

   In Batch mode, press PF5 from the Batch Processing Menu to submit the requests for batch processing. The batch request is processed as follows:

   a. For each Natural and Predict object to be added, a batch Natural unload step is created that unloads the Natural or Predict object source to a temporary data set.

   b. A single CA Endevor SCM step processes all the SCL submitted. For the Add actions, the Natural or Predict element is read from the temporary data set specified in the From clause and source is added to CA Endevor SCM.

   Note: For more information about the specific steps CA Endevor SCM performs during an Add, see How Adding/Updating a Member from Natural or Predict Works (see page 26).

Considerations When Adding/Updating an Object from Natural or Predict

Before you allow CA Endevor SCM to control an object from Natural or Predict using the Add/Update Elements action, consider the following information to ensure your success.

Important! If you do not follow these considerations and continue adding or updating the element, the Add/Update Elements action will fail.

- Make sure that an element with the same name is not already in the target entry stage. If so, you should first contact the current element owner, or your CA Endevor SCM administrator, to discuss the most appropriate action. Then, consider these workarounds:
  - Wait until the element has been moved out of the entry stage, and then add the member.
  - Add the member using the UPDATE IF PRESENT option.
- Make sure that the element corresponding to the object you want to add is signed out to you. If the element is signed out to another person on your team, you should first contact the current element owner, or your CA Endevor SCM administrator, to see if they can sign the element out to you. Then, consider this workaround:

  Add the object using the UPDATE IF PRESENT option combined with the OVERRIDE SIGNOUT option.

  Note: If your CA Endevor SCM administrator has restricted permissions for the OVERRIDE SIGNOUT option, you may not be able to use the option.

How Moving an Element Between Inventory Locations Works

To move an element from one inventory location (environment, stage) to the next location on a map route, you use the Move Elements action (available in foreground and batch). Consider the following information when moving elements:

- You can move elements either with history or without history.
- You can only move elements from one environment to another if the elements start in stage 2 of the source environment. For example, to move an element from stage 1 of the Development environment into stage 1 of the QA environment, you have to move the element to stage 2 in Development, and then move it into stage 1 in QA.

How Move Requests Are Processed

When you move an element from one inventory location (environment, stage) to the next location on a map route, CA Endevor SCM performs as many of the following processes as necessary.

- Source management, which involves the actual move of the element and the Master Control File (MCF) updates.
- Processor management, which involves executing (at the target location) either the Move processor of the source processor group or the Generate processor at the target processor group. This is determined by the processor group setting at the source location. Move processing includes invocation of appropriate Natural or Predict utilities to move objects between the FUSER/Library locations or FDIC locations associated with the source and target stages.
- Delete processing, which involves executing the Delete processor and deleting the element at the source location of the move.

Note: If a Move action fails, you should first respond to the error, and then resubmit your request. When you restart a Move action after a source management failure, CA Endevor SCM performs source management, processor management, and delete processing. After a processor management failure, CA Endevor SCM performs processor management and delete processing. After a delete failure, CA Endevor SCM performs delete processing.

How Moving Elements with History Works

When you move an element with history using the Move Elements action, CA Endevor SCM performs the following actions and causes these effects:

1. Determines if the element exists at the source and target location, if the ACKNOWLEDGE ELEMENT JUMP option applies, if source management is necessary, and if processor management is necessary (meaning, a restart situation). If necessary, CA Endevor SCM issues messages in the API Message report.
2. Performs *source management* by moving the element to the target stage. The processing that CA Endevor SCM performs depends on whether the elements exists at the TO location.
   - If the element does *not* exist at the target, CA Endevor SCM performs the following steps:
     - Searches the map for subsequent occurrences of the element. If the element is found farther along the map, the base level of the element to be moved is compared with the current level of the subsequent element.
     - If the two levels are in sync, CA Endevor SCM copies back (or *fetches*) the element up the map to the target location with all delta levels intact.
     - If the two levels are not in sync, CA Endevor SCM issues a warning message and does not perform the copy (as long as the SYNC=N option was specified on the move action). If the SYNC=Y option was specified for the move action and the two are not in synch, CA Endevor SCM issues a warning message and copies the element back to the target location and creates a sync level at the target. The new sync level reflects the differences between the base level of the source element and the current level of the fetched element.

     Note: You can restrict the SYNC=Y option to prevent it from copying the element back to the target by activating the feature DO_NOT_SYNC_AT_TARGET=ON in the optional features table (ENCOPTBL).

     Moves the element with the current level of the source element being moved to the target location, and the source element delta levels are appended to the target.

     If the element is found at a stage that is not included in the map, a warning message is issued, and the search continues for the element in a stage that is included in the map.

- If the element exists at the target location, CA Endevor SCM performs level-matching. During level-matching, CA Endevor SCM determines the sync point of the source and target elements by comparing the level timestamp of the base level of the source element with the current level of the target element. If the two timestamps do not match, CA Endevor SCM checks the next oldest level of the source element, and so on.
  - If the sync point is found, CA Endevor SCM moves the element from the FROM location to the TO location, appending the FROM location delta levels after the sync-point element.
  - If the two levels are different and SYNC=Y, CA Endevor SCM first creates a sync level at the target reflecting the differences between the base level of the FROM element and the target, and then moves the element to the TO location and appends the FROM location delta levels to the target.
  - If the element does not exist at the target, CA Endevor SCM moves the element from the source to the target location with all delta levels intact.

3. Completes source management after the element base and change levels have been moved by updating the Master Control File (MCF) to reflect the move.

   Note: The value specified for Signout Upon Fetch (the SOFETCH parameter) in the CA Endevor SCM Defaults table (C1DEFLTS) will effect how the MCF for the element copied back will be updated. If Signout Upon Fetch is in effect, the element will be signed out to you unless it is already signed out to someone else. If Signout Upon Fetch is not in effect, the element will not be signed out to you.

4. Completes processor management by determining the processor group last associated with the element, and then executes the move or generate processor in that group if one has been specified. After the processor has been run for the element, CA Endevor SCM updates the processor information in the Master Control file.

5. If the DELETE FROM ELEMENT option is set to Y, CA Endevor SCM executes standard delete processing for the element.

How Moving Elements Without History Works

You can move an element without history using the Move Elements action by setting the WITH HISTORY option to N in your request. CA Endevor SCM executes the request and attempts to find a sync level between the source and target elements beginning with the first level at the source and working forward through the deltas.

- If CA Endevor SCM finds a sync level, the elements are compared and a new level is created at the target that reflects the differences.
- If CA Endevor SCM cannot find a sync level and you set the SYNC option to N, a message is issued indicating that the elements are "out of sync" and the MOVE action terminates.
- If CA Endevor SCM cannot find a sync level and you set the SYNC option to Y, an out of sync message is issued. CA Endevor SCM then compares the last level of the source and last level of the target, and creates a new level at the target that reflects the differences.

How Moving Elements Affects the Signout Status

Your CA Endevor SCM administrator can enable a signout capability on a system-by-system basis. When you move an element from one inventory location (environment, stage) to the next location on a map route, CA Endevor SCM performs the following actions and causes these effects:

- If the element exists up the map, a *fetch* is performed.
  - If your CA Endevor SCM administrator has enabled Signout Upon Fetch, the fetched (source) element (the element at the FROM location of the action) is signed out to you, if the element is not signed out to someone else. In addition, the target element is signed in.
  - If your CA Endevor SCM administrator has not enabled Signout Upon Fetch, the fetched (source) element is signed out to you, if the element is not signed out to someone else. In addition, the target element is signed in.
- If the element does *not* exist up the map, a fetch is not performed.
  - If your CA Endevor SCM administrator has enabled Signout Upon Fetch, the target element is signed in.
  - If your CA Endevor SCM administrator has not enabled Signout Upon Fetch, the target element is signed in.

  Note: For more information, see Actions and Signout Status in the chapter "Action Processing" in the *CA Endevor Software Change Manager User Guide*.

How Moving Elements Affects CCIDs and Comments

When you move an element from one inventory location (environment, stage) to the next location on a map route using the Move Elements action (with history and without history), you can specify a CCID and comment. If you specify a CCID and comment, CA Endevor SCM performs the following actions and causes these effects:

- If you use the Move Elements Action (with history):
  - The Source CCID/Comment is set from the start location value (Stage 1 value).
  - The Generate CCID/Comment is set from the start location value (Stage 1 value).
  - The Last Action CCID/Comment is set.
  - The Retrieve CCID/Comment is cleared.
  - The Source Delta CCID/Comment is carried with delta levels.
  - The Component Delta CCID/Comment is carried with delta levels.
- If you use the Move Elements Action (without history):
  - The Source CCID/Comment is set from the start location value (Stage 1).
  - The Generate CCID/Comment is set from the start location value (Stage 1).
  - The Last Action CCID/Comment is set.
  - The Retrieve CCID/Comment is cleared.
  - The Source Delta CCID/Comment is set from the last start location delta value.
  - The Component Delta CCID/Comment is set from the last start location delta value.

Move an Element

To move an element from one inventory location (environment, stage) to the next location on a map route, you use the Move Elements action.

To move an element

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select either the Foreground or Batch processing option and press Enter.

   The Foreground Processing Menu or the Batch Processing Menu opens.

3. Select an object type (Natural programming object, Error message text, DDM, or Predict object) and the move action type (M) and press Enter.

   The move interface (for the object type you selected) opens.

4. Complete the following fields:

   Sync

   Enter Y (Yes) or N (No) to indicate whether you want the MOVE action performed when the base level of the element at the source location is different from the current level of the element at the target. When you enter Y, CA Endevor SCM creates a "sync" level at the target that reflects the differences between the base level at the source location and the current level at the target. The move fails if these levels are different and you have specified SYNC = N.

   Important! You must specify SYNC = Y when moving to a location where the out-of-sync condition exists. If the element does not exist at the target of the move, the SYNC=Y option searches up the map for the element and fetches (copies back) the element to the target and creates a sync level at the target. If you do not want this to happen, you must set the DO_NOT_SYNC_AT_TARGET=ON option in the optional features table ENCOPTBL.

   With History

   Enter Y (Yes) or N (No) to indicate whether you want to move the element with history. When you move the element without history, CA Endevor SCM searches through the element levels at the source location to find a matching level at the target location. CA Endevor SCM then compares the two and creates a new level at the target location that reflects the differences.

   Signout To

   Enter the TSO user ID of the person at the target location for which you want to sign out the element. If the RETAIN SIGNOUT option is set to Y, you cannot use this option.

   Delete from Element

   Enter Y (Yes) or N (No) to indicate whether you want CA Endevor SCM to delete the elements at the source location after moving them.

   Important! If you enter N, you must specify SYNC = Y for any subsequent moves of this element.

5. Choose one of the following steps.

   a. To move elements from a Move Selection list, set the Select List option to Y, enter the information for the action, and press Enter.

   b. The Move Selection screen opens, letting you select the elements you want to move by entering an X to the left of an object.

   c. To move elements without a selection list, set the Select List option to N, enter the information for the action, and press Enter.

      In Foreground mode, an Action Confirmation window opens telling you how many elements have been selected for moving. Press Enter to submit the move requests.

6. In Foreground mode, CA Endevor Integration for the Natural Environment interfaces with CA Endevor SCM to move the selected elements to the next location on a map route.

   In Batch mode, press PF5 from the Batch Processing Menu to submit the requests for batch processing.

How Deleting Elements Works

When you delete element levels using the Delete Elements action, CA Endevor SCM performs the following actions and causes these effects:

- Verifies that the element currently exists in the stage you specify. If the element does not exist in the stage, you cannot delete the element.
- Determines whether the element is signed out to you. If the element is signed out to someone else, you must set the OVERRIDE SIGNOUT option to Y to delete the element.
- Removes the list library entries for the element.
- Determines the processor group last associated with the element, and then executes the delete processor in that group if one has been specified.
- Deletes the element base level, and all change levels.
- Updates the Master Control File (MCF) to reflect this processing.

Note: When you delete Predict objects, the delete process only removes CA Endevor SCM information. It does not remove the CA Endevor Integration for the Natural Environment FDIC file entry.

Delete Elements

To delete all levels (base and delta members) of an element from CA Endevor SCM, any associated processors outputs from either stage, and the component list for the element, use the Delete Elements action.

Note: This element action erases base and delta forms of an element and removes related information from the Master Control File (MCF) or component list.

To delete the levels for an element

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select either the Foreground or Batch processing option and press Enter.

   The Foreground Processing Menu or the Batch Processing Menu opens.

3. Select an object type (Natural programming object, Error message text, DDM, or Predict object) and the delete action type (#) and press Enter.

   The delete interface (for the object type you selected) opens.

4. Choose one of the following steps.

   a. To delete elements from a Delete Selection list, set the Select List option to Y, enter the information for the action, and press Enter.

      The Delete Selection screen opens, letting you select elements you want to delete by entering an X to the left of an element.

   b. To delete elements without a selection list, set the Select List option to N, enter the information for the action, and press Enter.

      In Foreground mode, an Action Confirmation window opens telling you how many elements have been selected for deletion. Press Enter to submit the delete requests.

5. In Foreground mode, CA Endevor Integration for the Natural Environment interfaces with CA Endevor SCM to delete the selected elements.

   In Batch mode, press PF5 from the Batch Processing Menu to submit the requests for batch processing.

How Generating Elements Works

CA Endevor SCM can generate elements in two different ways:

- Automatically. This happens when you add or update elements (unless you have set the GENERATE ELEMENTS option to N).
- Explicitly, when you execute an explicit GENERATE element action.

How Generating Elements with Copyback Works

Generate an element with copyback when you change an element and you want to recompile the affected programs. When you generate an element with copyback using the Generate Elements action, CA Endevor SCM performs the following actions and causes these effects:

1. Searches beyond the current stage, in every stage in the current and subsequent environments in the map.
2. If the element is found, the current version of the element is copied to the target stage, along with the processor group name last used for the element. Consider the following options when generating elements:
   - To restrict the search to the current environment, generate the element by setting the BUILD USING MAP option to N.
   - If CA Endevor SCM finds the element in a stage that is not included in the map, a warning message is issued, and the search continues for the elements in a stage that is included in the map.
3. Updates the Master Control File (MCF).

   Note: The Signout Fetch (SOFETCH) value in the Defaults Table (C1DEFLTS) determines how the MCF is updated for the element copied back (fetched). If Signout Upon Fetch is in effect, the element is signed out to you unless it is already signed out to someone else. If Signout Upon Fetch is not in effect, the element is not signed out to you.
4. Checks the element type definition for a source output library specification, and then writes a copy of the current level of the element to that library.
5. Determines the processor group, and then executes the processor in that group, if one has been specified.
6. After the generate processor is run for the element, updates the information in the Master Control File (MCF).

How Generating Elements without Copyback Works

When you generate an element without copyback using the Generate Elements action, CA Endevor SCM performs the following actions and causes these effects:

1. Determines whether the element is signed out to you.
2. Determines the processor group, and then executes the processor in that group, if one has been specified.
3. After the generate processor is run for the element, updates the information in the Master Control File (MCF).
4. Updates the list library with execution information.

How Generating Elements Affects the Signout Status

Your CA Endevor SCM administrator can enable a signout capability on a system-by-system basis. When you create an executable form of an element using the Generate Elements action, CA Endevor SCM performs the following actions and causes these effects:

- If the element exists up the map (with copyback), a *fetch* is performed.
  - If your CA Endevor SCM administrator has enabled Signout Upon Fetch, the fetched (source) element is signed out to you, if the element is not signed out to someone else. In addition, the target element is signed out to you.
  - If your CA Endevor SCM administrator has not enabled Signout Upon Fetch, there is no change in the signout status, and the target element is signed out to you.
- If the element does *not* exist up the map (without copyback), a fetch is not performed.
  - If your CA Endevor SCM administrator has enabled Signout Upon Fetch, there is no change in the signout status of the target element.
  - If your CA Endevor SCM administrator has not enabled Signout Upon Fetch, there is no change in the signout status of the target element.

  Note: For more information, see Actions and Signout Status in the chapter "Action Processing" in the *CA Endevor Software Change Manager User Guide*.

How Generating Elements Affects CCIDs and Comments

When you create an executable form of an element using the Generate Elements action (with copyback and without copyback), you can enter a CCID and comment. When you enter a CCID and comment, CA Endevor SCM sets the following:

- When you generate an element with copyback:
  - Current Source CCID/Comment (to the copied back value, or the Stage 2 value)
  - Generate CCID/Comment
  - Last Action CCID/Comment
  - Source Delta CCID/Comment (to the copied back value, or the Stage 2 value)
  - Component Delta CCID/Comment
- When you generate an element *without* copyback:
  - Generate CCID/Comment
  - Last Action CCID/Comment
  - Component Delta CCID/Comment (if the generate creates a delta)

Generate an Element

To load a Natural or Predict object into the FUSER/Library or FDIC that corresponds to a CA Endevor SCM stage and subsystem, and, in the case of Natural programming objects, execute the CATALL utility to generate Natural executables, use the Generate Elements action.

To generate an element output

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select either the Foreground or Batch processing option and press Enter.

   The Foreground Processing Menu or the Batch Processing Menu opens.

3. Select an object type (Natural programming object, Error message text, DDM, or Predict object) and the generate action type (G) and press Enter.

   The generate interface (for the object type you selected) opens.

4. Choose one of the following steps.

   a. To generate elements from a Generate Selection list, set the Select List option to Y, enter the information for the action, and press Enter.

      The Generate Selection screen opens, letting you select elements for generation by entering an X to the left of an element.

   b. To generate elements without a selection list, set the Select List option to N, enter the information for the action, and press Enter.

      In Foreground mode, an Action Confirmation window opens telling you how many elements have been selected for generation. Press Enter to submit the generate requests.

5. In Foreground mode, CA Endevor Integration for the Natural Environment interfaces with CA Endevor SCM to generate the selected elements.

   In Batch mode, press PF5 from the Batch Processing Menu to submit the requests for batch processing.

Signin Elements

When you add or update an element in, or retrieve an element from, CA Endevor SCM, the element is signed out to you. A signout is the assignment of a user ID to an element, establishing ownership of the element. Signout is automatic when adding or updating elements in or when retrieving elements from CA Endevor SCM. However, you can use the Signout To option to assign the element to another person.

The signin action removes the current signout for an element. Signin can be implicit or explicit.

How Signin Works

When you use the signin action, you remove the signout user ID associated with the element. CA Endevor SCM performs the following actions and causes these effects:

- Determines whether the element is signed out to you. If the element is signed out to someone else, you must set the OVERRIDE SIGNOUT option to Y to sign in the element.

  Note: Your CA Endevor SCM administrator must set up permissions for you to use the OVERRIDE SIGNOUT option.

- Updates the Master Control File (MCF) for the element by removing the current signout (that is, the user ID associated with the last RETRIEVE action).

Note: After you sign in an element, you can assign the element to another person using the SIGNOUT TO option.

How Signing In an Element Affects CCIDs and Comments

When you remove the signout user ID associated with an element, CA Endevor SCM clears the Retrieve CCID/Comment for the element.

How Signing In an Element Affects the Signout ID

Note: You cannot sign in an element unless it is already signed out to you, unless you set the OVERRIDE SIGNOUT option to Y.

Your CA Endevor SCM administrator can enable a signout capability on a system-by-system basis. When you remove the signout user ID associated with an element using the Signin Elements action, CA Endevor SCM performs the following actions and causes these effects:

- Clears the SIGNOUT ID field at the stage at which the signin is being performed, unless you also specify a user ID in the SIGNOUT TO field.
- If you specify a user ID in the SIGNOUT TO field, CA Endevor SCM sets the SIGNOUT ID at the stage at which the action is being performed to the user ID specified in the action.

Sign In an Element

You use the Signin action to sign in an element, which removes the signout user ID associated with an element. SIGNIN is only available for systems where signin and signout is in effect.

To remove the user signout

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select either the Foreground or Batch processing option and press Enter.

   The Foreground Processing Menu or the Batch Processing Menu opens.

3. Select an object type (Natural programming object, Error message text, DDM, or Predict object) and the sign-in action type (S) and press Enter.

   The sign-in interface (for the object type you selected) opens.

4. Choose one of the following steps.

   a. To sign-in elements from a Sign-in Selection list, set the Select List option to Y, enter the information for the action, and press Enter.

      The Sign-in Selection screen opens, letting you select elements for generation by entering an X to the left of an element.

   b. To sign-in elements without a selection list, set the Select List option to N, enter the information for the action, and press Enter.

      In Foreground mode, an Action Confirmation window opens telling you how many elements have been selected for sign-in. Press Enter to submit the sign-in requests.

5. In Foreground mode, CA Endevor Integration for the Natural Environment interfaces with CA Endevor SCM to sign-in the selected elements.

   In Batch mode, press PF5 from the Batch Processing Menu to submit the requests for batch processing.

How Transferring an Element Works

The transfer action migrates the specified objects from the From Environment to the To Environment. In addition, CA Endevor SCM performs the following actions and causes these effects:

- Determines whether the element is signed out to you. If the element is signed out to someone else, you must set the OVERRIDE SIGNOUT option to Y to transfer the element.

  Note: Your CA Endevor SCM administrator must set up permissions for you to use the OVERRIDE SIGNOUT option.

- Updates the Master Control File (MCF) for the element based on the options specified as part of the transfer.
- Updates the CCID and Comment fields for the element with the information provided on the Transfer screen.

Note: As part of the transfer function, the signout can be manipulated using the RETAIN SIGNOUT, SIGNOUT TO, and OVERRIDE SIGNOUT options.

Transfer an Element

To transfer an element, you use the Transfer Elements action.

To transfer an element

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select the Batch processing option and press Enter.

   The Batch Processing Menu opens.

3. Select an object type (Natural programming object, Error message text, DDM, or Predict object) and the transfer action type (T) and press Enter.

   The transfer interface for the object type you selected opens.

4. Choose one of the following steps.

   a. To transfer elements from a Transfer Selection list, set the Select List option to Y, enter the information for the action, and press Enter.

      The Transfer Selection screen opens, letting you to select the elements for generation by entering an X to the left of an element. The batch selection screens indicate your selections by displaying *Written* next to each on you select. After all elements to be Transferred have been selected and written, press PF3 to return to the Batch Processing Menu.

      A message with the total number of batch action requests written is noted at the top of the screen.

   b. To transfer elements without a selection list, set the Select List option to N, enter the information for the action, and press Enter.

      A message with the total number of batch action requests written is noted at the top of the screen

5. Display the summary list of your selections. After elements are selected on any batch selection interface, you can display the summary list of all elements and actions that you selected, by pressing the PF10 key (BReqs). You can delete any items from the summary list that were selected in error.

6. Submit or save the batch requests.

   a. Use PF5 (Submit) to submit the requests for batch processing. The Batch Job Submittal window displays.

   b. Use PF4 (Save) to save the generated SCL to a Natural text member in the library EINESCL. The SCL can then be edited using the Edit SCL functions or by logging on to the EINESCL library and using the standard Natural editor.

7. On the Batch Job Submittal window, you can edit the STOPRC value for generated SCL and customize the job card, which will be submitted at the top of the generated JCL if job card modifications have not been restricted by the EINE Administrator. Press Enter to submit the batch job for processing. If you want to cancel the submission, press PF3.

Saved SCL Objects

Saved SCL objects contain SCL statements for the batch submission of actions or for import into package SCL and are stored in the Natural library EINESCL as text objects. These objects can be created using the batch processing panels accessible from the Batch Processing Menu. You can save SCL generated using the panels or SCL that you built from scratch using the SCL option on the Batch Processing Menu. In addition, template SCL is provided in the EINELIB library for each SCL function. The sample SCL objects are: NATSDIS (display); NATSRET(retrieve); NATSADD (add); NATSMOV (move); NATSDEL (delete); NATSGEN (generate); NATSTRN (transfer); and NATSSGN (signin). Saved SCL objects can be edited outside of CA Endevor Integration for the Natural Environment by copying the required templates from the EINELIB to the EINESCL library and using the standard Natural editor to modify the copied template members in the EINESCL library. In addition, the SCL can be listed, deleted, edited, or submitted from the Saved SCL panel.

Note: Add and Retrieve actions cannot be processed by the Saved SCL submit function.

For more information about saving batch SCL, see Submit Actions in Batch. For more information about package SCL, see Building Package SCL.

Note: The CA Endevor Integration for the Natural Environment administrator can customize the CA Endevor Integration for the Natural Environment Main Menu to remove certain options. If the Saved SCL Objects option is not on the menu, contact your local administrator for further information.

Access Saved SCL Objects

Saved SCL objects are accessible from the Endevor Saved SCL panel. You can delete, edit, or submit saved SCL objects on this panel. All text objects that exist in the Natural library EINESCL are displayed on this panel.

To access saved SCL objects

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select option S from the main menu and press Enter.

   The Endevor Saved SCL panel opens displaying all the SCL objects saved in the Natural library EINESCL.

Delete SCL Objects

You can delete a previously saved SCL object from the Natural library EINESCL.

To delete a previously saved SCL object

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select option S from the main menu and press Enter.

   The Endevor Saved SCL panel opens displaying all the SCL objects saved in the Natural library EINESCL.

3. Enter a D in the C column next to the specified SCL Object on the Endevor Saved SCL panel and press Enter.

   A delete confirmation message displays at the top of the screen.

Edit SCL Objects

You can edit a previously saved SCL object from the Natural library EINESCL.

To edit a previously saved SCL object

1. Enter an E in the C column next to the specified SCL Object on the Endevor Saved SCL Screen and press Enter.

   The data entry screen opens.

   Note: For details on how to access the Endevor Saved SCL screen, see Accessing Saved SCL Objects (see page 46).

2. Add, delete, and modify the SCL lines using standard TSO line commands on this screen.

Submit Saved SCL Objects

You can submit previously saved Display, Move, Delete, Generate, Sign-In and Transfer SCL objects from the Natural library EINESCL.

Note: Retrieve and Add/Update SCL objects cannot be submitted using the Saved SCL submit function. Retrieve and Add/Updates actions must be executed in Foreground or submitted from the Batch Processing menu.

To submit a previously saved SCL object

1. Enter an S in the C column next to the SCL Object you want to save on the Endevor Saved SCL Screen and press Enter.

   The Batch Job Submittal window opens.

   Note: For details on how to access the Endevor Saved SCL screen, see Accessing Saved SCL Objects (see page 46).

2. Customize the job card, which will be submitted at the top of the generated JCL, if job card modifications have not been restricted by the EINE Administrator, and optionally edit the STOPRC value for the generated SCL. Press Enter.

   Note: If you want to cancel the submission, press PF3.

   The saved SCL is submitted for batch processing.

Element Information

You can view the following detailed information about each element in your software inventory:

- Summary of changes made to an element.
- Element information from the Master Control File (MCF).
- All statements in the current level of an element (and the level at which each statement was inserted).
- All changes (insertions and deletions) made to an element up to the current level.
- All statements in all levels of an element, from the base level to the current level (including the level at which each insertion and deletion occurred).

View Element Information

To learn more about individual elements, you can view the element information associated with each element.

To view element information

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Choose one of the following steps.

   a. Select the Foreground processing option and press Enter.

      The Foreground Processing Menu opens.

   b. Select the Batch processing option and press Enter.

      The Batch Processing Menu opens.

   c. Select the Package processing option and press Enter.

      Follow the steps outlined in the section Create and Modify a Package to advance to the Build Package SCL Menu.

   Note: All the display interfaces function the same in the foreground, batch, and package subsystems of CA Endevor Integration for the Natural Environment.

3. Select a specific object type (Natural programming object, Error message text, DDM, or Predict object) and the display action type (D) and press Enter.

   The display interface (for the object type you selected) opens.

4. Enter the information for the action and press Enter.

   The Display Selection screen opens.

5. Select the elements you want to display by entering an appropriate code to the left of an element: 'B' for browse, 'S' for summary information, 'H' for history information, 'M' for master data information, 'C' for change information, or 'L' for listing information.

   Note: The listing information (option 'L') consists of CMPRINT output from Natural utilities executed within the most recent processor that was executed against an element. For example, following execution of a generate processor against a Natural program element, the element's listing information contains CMPRINT output from the step which loaded the Natural program into the appropriate FUSER/Library, and also contains CMPRINT output from the CATALL step which generated the executable Natural program object.

Packages

Packages are a set of SCL requests to perform CA Endevor SCM actions on elements. Packages enable you to group specific actions so they can be maintained and tracked as a single unit, establish formal approval procedures to ensure data integrity through modifications, and centralize specific action groups so you can see them across environments and reuse them. You create or modify a package by defining the package attributes and creating the SCL requests. Package attributes are as follows:

- Package types are standard or emergency
- Packages can be sharable or not sharable
- Packages are promotion Packages or non-promotion packages
- Backout enabled or not backout enabled

After you create or modify a package, you perform a cast package action to begin the review process. If the package does not need to be approved, the cast package action changes the package status to Approved. If the package needs approval, the cast package action changes the package status to In-Approval.

For more information about packages, see the *CA Endevor Software Change Manager Packages Guide*.

Note: The CA Endevor Integration for the Natural Environment administrator can customize the CA Endevor Integration for the Natural Environment Main Menu to remove certain options. If the Packages option is not on the menu, contact your local administrator for further information.

Create a Package

You can create a package, which is a set of CA Endevor SCM actions that require approval before execution. After you create the package, you build the SCL statements that CA Endevor SCM uses to perform actions on elements.

To create a package

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select the Package Processing option, on the CA Endevor Integration for the Natural Environment Main Menu, and press Enter.

   The Endevor Package ID Entry screen opens.

3. Press PF5.

   The Create Package screen opens.

4. Type in a package ID and description and change the package attributes if necessary. Then press Enter.

   The package is saved in IN-EDIT status and the Build Package SCL Menu opens.

   Note: To build the package's SCL statements, see Build Package SCL.

Modify a Package

You can modify a package to change its attributes or change the SCL statements that CA Endevor SCM uses to perform actions on elements.

To modify a package

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select the Package Processing option, on the CA Endevor Integration for the Natural Environment Main Menu. Then press Enter.

   The Endevor Package ID Entry screen opens.

3. Type in a package ID. Name masking is allowed. Change the filter criteria for the package information as necessary to limit the package selection list. Then press Enter.

   Valid values for the package status criteria are Y for yes or N for no.

   Valid values for the other criteria are as follows:

   Enterprise Pkg

   A - Display all packages - Default

   E - Display enterprise packages only

   X - Exclude display of enterprise packages

   Promotion Pkg

   A - Display all packages - Default

   P - Display promotion packages only

   X - Exclude display of promotion packages

   Prom Pkg Tgt Env

   Display promotion packages whose target environment matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

   Prom Pkg Tgt Stage

   Display promotion packages whose target stage matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

   A package selection list opens that matches the selection criteria specified on the Endevor Package ID Entry screen.

4. Type an M to the left of the package you want to modify. Then press Enter.

   The Modify Package screen opens.

5. Change the package information, if necessary. Then press Enter.

   The package is saved in IN-EDIT status and the Build Package SCL Menu opens.

   Note: To modify the package's SCL statements, see Build Package SCL.

Build Package SCL

You can build SCL statements for a new package or modify SCL statements in a package, provided the package has a status of IN-EDIT. The Build Package SCL screen opens after you have either created a package using the Create a Package screen, or you press Enter on the Modify a Package Screen.

To build or modify package SCL statements

1. Select a specific object type (Natural programming object, Error message text, DDM, or Predict object) and the action type you want (Move, Delete, Generate, or Sign-in), from the Build Package SCL screen. Press Enter.

   The appropriate interface (for the object type and action type you selected) opens.

2. Enter action information just as you would for Foreground or Batch action processing, either selecting elements from a selection list (with Selection List set to Y) or selecting elements without a selection list (with Select List set to N). For each element selected, a corresponding SCL statement is generated and appended to the SCL of the package.

3. You can construct a package consisting of different object types and action types by returning to the Build Package SCL Menu and choosing a different object type and action type combination to generate additional SCL.

4. At any time, view or edit your current SCL by returning to the Build Package SCL Menu and pressing PF10 (SCL).

   The SCL for Package window opens. You can enter # next to any line of SCL to delete it, and you can place the entire window into EDIT mode by pressing PF11 (Edit). When in EDIT mode, make any changes you want and press Enter to save. You can import SCL that exists in the Natural library EINESCL by entering a P on any line while in Edit mode. You can toggle the window back to DISPLAY mode by pressing PF11 again.

   If you are building SCL from scratch, template SCL is provided for each SCL function. Specifying these objects using the Import function lets you customize the template. The sample SCL objects are:

   NATSDIS (display); NATSRET(retrieve); NATSADD (add); NATSMOV (move); NATSDEL (delete); NATSGEN (generate); NATSTRN (transfer); and NATSSGN (signin).

5. When you leave the Build Package SCL Menu screen, the Save Package SCL Confirmation opens, giving you the option to press Enter and save all SCL changes to CA Endevor SCM or press PF3 to cancel and discard all SCL changes.

Cast a Package

Some CA Endevor SCM packages must be reviewed and approved before being executed. Casting a package is the first step in the package review process.

To cast a package

1. After creating or modifying a package, return to the Build Package SCL Menu screen and press PF5 (Cast). If you are modifying an already existing package, you may also press PF5 (Cast) while viewing the Modify Package screen.
2. If the cast is not successful, you may press PF12 (Info) to view the API Message Report, which has the Package Cast Report appended to it. You can then correct any invalid aspects of the package's SCL (as reported in the Cast Report) and re-cast the package.

When you cast a package that requires approval, CA Endevor SCM changes its status to IN-APPROVAL. When you cast a package that does not require approval, CA Endevor SCM changes its status to APPROVED. After a package has been cast, it can be approved and then executed.

Reset a Package

The reset a package action erases all package event records, returning the package to In-edit status. A package can be reset at anytime.

To reset a package

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).
2. Select the Package Processing option, on the CA Endevor Integration for the Natural Environment Main Menu. Then press Enter.

   The Endevor Package ID Entry screen opens.
3. Type in a package ID. Name masking is allowed. Change the filter criteria for the package information as necessary to limit the package selection list. Then press Enter.

   Valid values for the package status criteria are Y for yes or N for no.

   Valid values for the other criteria are as follows:

   Enterprise Pkg

   A - Display all packages - Default

   E - Display enterprise packages only

   X - Exclude display of enterprise packages

Promotion Pkg

A - Display all packages - Default

P - Display promotion packages only

X - Exclude display of promotion packages

Prom Pkg Tgt Env

Display promotion packages whose target environment matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

Prom Pkg Tgt Stage

Display promotion packages whose target stage matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

A package selection list opens that matches the selection criteria specified on the Endevor Package ID Entry screen.

4. Type an P to the left of the package you want to reset. Then press Enter.

   The confirmation screen opens.

5. Click Y to reset.

   The package is saved in IN-EDIT status and the Build Package SCL Menu opens.

Review a Package

The review a package action lets you to Approve or Deny a package when the status is IN-APPROVAL.

To review a package

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select the Package Processing option, on the CA Endevor Integration for the Natural Environment Main Menu. Then press Enter.

   The Endevor Package ID Entry screen opens.

3. Type in a package ID. Name masking is allowed. Change the filter criteria for the package information as necessary to limit the package selection list. Then press Enter.

   Valid values for the package status criteria are Y for yes or N for no.

   Valid values for the other criteria are as follows:

   Enterprise Pkg

   A - Display all packages - Default

   E - Display enterprise packages only

   X - Exclude display of enterprise packages

   Promotion Pkg

   A - Display all packages - Default

   P - Display promotion packages only

   X - Exclude display of promotion packages

   Prom Pkg Tgt Env

   Display promotion packages whose target environment matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

   Prom Pkg Tgt Stage

   Display promotion packages whose target stage matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

   A package selection list opens that matches the selection criteria specified on the Endevor Package ID Entry screen.

4. Type a V to the left of the package you want to review. Then press Enter.

   The confirmation screen opens.

5. Enter A to approve or D to deny.

   The package is approved or denied.

Execute a Package

The execute a package action lets you execute a package immediately or submit the package for batch processing. You can execute a package which has a status of APPROVED or EXEC-FAILED.

To execute a package

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select the Package Processing option, on the CA Endevor Integration for the Natural Environment Main Menu. Then press Enter.

   The Endevor Package ID Entry screen opens.

3. Type in a package ID. Name masking is allowed. Change the filter criteria for the package information as necessary to limit the package selection list. Then press Enter.

   Valid values for the package status criteria are Y for yes or N for no.

   Valid values for the other criteria are as follows:

   Enterprise Pkg

   A - Display all packages - Default

   E - Display enterprise packages only

   X - Exclude display of enterprise packages

   Promotion Pkg

   A - Display all packages - Default

   P - Display promotion packages only

   X - Exclude display of promotion packages

   Prom Pkg Tgt Env

   Display promotion packages whose target environment matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

   Prom Pkg Tgt Stage

   Display promotion packages whose target stage matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

   A package selection list opens that matches the selection criteria specified on the Endevor Package ID Entry screen.

4. Type a X to the left of the package you want to execute. Then press Enter.

   The confirmation screen opens.

5. Select one of the following choices:

   a. Enter E to execute the package immediately.

      Note: PF12 (Info key) lets you view Action requests and API messages.

   b. Enter S to submit the package for batch execution.

      A pop-up dialog permits entry of the appropriate job card information for the batch job. The batch JCL skeleton is contained in the Natural text object NDVPROCP

Commit a Package

The commit a package action lets you commit a package. You can commit a package which has a status of EXECUTED.

To commit a package

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select the Package Processing option, on the CA Endevor Integration for the Natural Environment Main Menu. Then press Enter.

   The Endevor Package ID Entry screen opens.

3. Type in a package ID. Name masking is allowed. Change the filter criteria for the package information as necessary to limit the package selection list. Then press Enter.

   Valid values for the package status criteria are Y for yes or N for no.

   Valid values for the other criteria are as follows:

   Enterprise Pkg

   A - Display all packages - Default

   E - Display enterprise packages only

   X - Exclude display of enterprise packages

   Promotion Pkg

   A - Display all packages - Default

   P - Display promotion packages only

   X - Exclude display of promotion packages

Prom Pkg Tgt Env

Display promotion packages whose target environment matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

Prom Pkg Tgt Stage

Display promotion packages whose target stage matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

A package selection list opens that matches the selection criteria specified on the Endevor Package ID Entry screen.

4. Type a C to the left of the package you want to commit. Then press Enter.

   The confirmation screen opens.

5. Click Y to commit.

   The package is committed.

Delete a Package

The delete package action lets you delete a package. Packages to be deleted can be in any status.

To delete a package

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select the Package Processing option, on the CA Endevor Integration for the Natural Environment Main Menu. Then press Enter.

   The Endevor Package ID Entry screen opens.

3. Type in a package ID. Name masking is allowed. Change the filter criteria for the package information as necessary to limit the package selection list. Then press Enter.

   Valid values for the package status criteria are Y for yes or N for no.

   Valid values for the other criteria are as follows:

   Enterprise Pkg

   A - Display all packages - Default

   E - Display enterprise packages only

   X - Exclude display of enterprise packages

Promotion Pkg

A - Display all packages - Default

P - Display promotion packages only

X - Exclude display of promotion packages

Prom Pkg Tgt Env

Display promotion packages whose target environment matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

Prom Pkg Tgt Stage

Display promotion packages whose target stage matches the value specified in this field. No filter is applied if this field is left blank or promotion packages are not displayed.

A package selection list opens that matches the selection criteria specified on the Endevor Package ID Entry screen.

4. Type a # to the left of the package you want to delete. Then press Enter.

   The confirmation screen opens.

5. Enter Y.

   The package gets deleted.

Display Package Approvers

To review the approver groups associated with a package, you can display a list of the approver groups required to approve a package. You can also view information about the approvers in each group.

To display package approvers

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select the Package Processing option from the CA Endevor Integration for the Natural Environment Main Menu. Then, press Enter.

   The Endevor Package ID Entry panel opens.

3. Type a package ID. Change the filter criteria for the package information as necessary to limit the package selection list. Then, press Enter.

   Note: Name masking is allowed.

   A package selection list opens that matches your selection criteria.

4. Type an A to the left of the package for which you want to display the approver groups. Then press Enter.

   The Package Approver Groups panel for the package you specified opens. It lists all the approver groups associated with this package.

5. Type any character to the left of the approver group you want to view and press Enter.

   The Approver Group Display panel opens for the approver group you selected. This panel displays information about individual approvers.

6. Press the End key.

   The Package Approver Groups panel reopens. On this panel you can do one of the following:

   a. Select another approver group for display.

   b. Press the End key again to return to the Package Processing panel.

Display Cast Report

You can display the cast report for a specified package.

To display a cast report

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select the Package Processing option from the CA Endevor Integration for the Natural Environment Main Menu. Then, press Enter.

   The Endevor Package ID Entry screen opens.

3. Type a package ID and then press Enter. Change the filter criteria for the package information as necessary to limit the package selection list.

   Note: Name masking is allowed.

   A package selection list opens that matches the selection criteria specified on the Endevor Package ID Entry screen.

4. Type an R to the left of the package for which you want to display the cast report. Then, press Enter.

   The Cast Report for Package panel for the package you specified opens.

   Note: On the Cast Report for Package panel, the messages that follow the action statements provide a trace of the cast activities, and note errors that occur. When errors occur in component validation, informational messages identify the component that failed validation, and the associated footprint information.

Display SCL

You can display the SCL for a specified package.

To display SCL

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. Select the Package Processing option, on the CA Endevor Integration for the Natural Environment Main Menu. Then, press Enter.

   The Endevor Package ID Entry screen opens.

3. Type in a package ID and then press Enter. Name masking is allowed. Change the filter criteria for the package information as necessary to limit the package selection list.

   A package selection list opens that matches the selection criteria specified on the Endevor Package ID Entry screen.

4. Type an S to the left of the package whose SCL you want to view. Then, press Enter.

   The SCL for Package panel opens for the package you specified.

Submit Actions in Batch

The user interface for batch submission of actions is very similar to the interface used for foreground submission.

To submit batch requests do the following

1. Access the CA Endevor Integration for the Natural Environment Main Menu.

   Note: For more information, see Start a User Session (see page 15).

2. On the Main Menu, enter B on the command line and press Enter.

   The Batch Processing Menu opens.

3. On the Batch Processing Menu, enter the appropriate codes in the Object Code and Action Code fields and press Enter. If you want to exit this panel, enter a period (.) in either field and press Enter. The following options are available:

   Code Object

   N-Natural Programming Objects

   E-Error Message Texts (SYSERR)

   D-DDMs

   P-PREDICT Objects

   Code Action

   D-Display

   R-Retrieve

   A-Add/Update

   M-Move

   #-Delete

   G-Generate

   S-Sign-in

   T-Transfer

   Note: PF11(SCL) can be used to build SCL from scratch.

4. Select element and actions for processing. The batch processing screens are almost identical in appearance and behavior to the foreground version of the screens, except they are subtitled Batch Mode indicating they are for batch processing. However, the batch selection screens indicate your selections by displaying "Written" next to each one you select.

   For example, if you select Natural Programming Objects and Retrieve on the Batch Processing Menu, after you press Enter, the Retrieve Natural Programming Objects screen opens. After you enter your selection criteria, the Retrieve Selection screen opens. On the Retrieve Selection screen, the selection of an element for processing results in the message “Written” displayed next to it, allowing for downward and upward scrolling to select other elements, prior to the your exiting this screen to optionally select other elements and actions for processing.

5. Display the summary list of your selections. After elements are selected on any batch selection interface, you can display the summary list of all elements and actions that you selected, by pressing the PF10 key (BReqs). You can delete any items from the summary list that were selected in error.

6. Submit the batch requests. Use PF3 to return to the Retrieve Natural Programming Objects screen or to the Batch Processing Menu. The total number of your batch action requests is noted at the top of the screen.

   a. Use PF5 (Submit) to submit the requests for batch processing. The Batch Job Submittal window displays.

   b. Use PF4 (Save) to save the generated SCL to a Natural text member in the library EINESCL. The SCL can then be edited using the Edit SCL functions or by logging on to the EINESCL library and using the standard Natural editor.

      Note: Add and Retrieve actions cannot be processed by the Saved SCL submit function. These actions will be removed from the SCL before it is saved.

7. On the Batch Job Submittal window, you can edit the STOPRC value for generated SCL and customize the job card, which will be submitted at the top of the generated JCL, if job card modifications have not been restricted by the EINE Administrator. Press Enter to submit the batch job for processing. If you want to cancel the submission, press PF3.

   The submitted JCL consists of the following:

   - Unload steps— one for each Natural/Predict element selected for ADD processing (to unload the element from Natural/Predict)
   - Retrieve check steps— one for each Natural/Predict element selected for Retrieve processing (determines if the element exists in the Natural/Predict location)
   - A single step for CA Endevor SCM processing of generated SCL (for all selected actions)
   - Load steps— one for each Natural DDM/SYSERR/Predict element selected for Retrieve processing (to load the element into the correct Natural/Predict location).
   - Load steps— three for each Natural programming element selected for Retrieve processing (to load the element into the correct Natural location).
     - The first load step is a SYSOBJH LOAD with Transfer option. If the Natural element is stored in CA Endevor SCM in a SYSOBJH Internal format, this load step will fail with a condition code 50. A condition code 50 is an acceptable condition code during a batch Retrieve.
     - The next step is a CATALL SAVE of the loaded object. If the SYSOBJH LOAD with Transfer completes successfully, this step is executed. If the SYSOBJH LOAD with Transfer step receives a condition code of 50, the CATALL SAVE step is flushed.
     - The final step is a SYSOBJH LOAD with Internal format option.This step is executed if the load with Transfer option step receives a condition code of 50. Otherwise, this step is flushed if the load with Transfer option completed successfully.

How Load Steps for a Retrieve Are Processed

The following table shows how the Load steps for Retrieve processing are processed depending on the format in which the Natural element is stored in CA Endevor SCM:

	SYSOBJH Internal Format— If the Natural element stored in CA Endevor SCM is in *SYSOBJH Internal Format*, the Load steps will execute as follows:	**SYSOBJH Transfer Format—** If the Natural element stored in CA Endevor SCM is in *SYSOBJH Transfer Format*, the Load steps will execute as follows:
RETRIEVE Load step 1: SYSOBJH with Transfer option	This step receives a condition code of 50.	This step receives a condition code of 0.
RETRIEVE Load step 2: CATALL SAVE	This step is flushed.	This step is executed. Expect a condition code of 0.
RETRIEVE Load step 3: SYSOBJH Internal format option	This step is executed. Expect a condition code of 0.	This step is flushed.

Note: The exact number and sequence of the Natural programming load steps performed can be modified by the CA Endevor Integration for the Natural Environment administrator. If the Natural load steps do not match this sequence, contact your local administrator.

Glossary

actions

An *action* is a command to manage elements.

Add Elements Action

The *Add Elements Action* places members from an external data set under control of CA Endevor SCM.

alternate ID

An *alternate ID* is a facility that protects the CA Endevor SCM-controlled data sets (such as the Master Control Files, package data sets, and base and delta libraries) from access by users, while still allowing CA Endevor SCM to have access. Information for the alternate ID is provided by three parameters in the CA Endevor SCM Defaults Table: RACFGRP, RACFPWD, and RACFUID.

approval

An *approval* is the electronic signoff mechanism for packages. An approval may be required for a package before execution.

approved

The *approved* status is assigned after the required approvers have reviewed, signed off, and approved a package.

approver

An *approver* is a person authorized to sign off on a package, prior to execution. Signing off on a package involves reviewing the information in the package and approving or denying the package.

approver group

An *approver group* is a collection of one or more approvers. An approver group is defined within each environment and can be associated with inventory areas.

approver group relationship

An *approver group relationship* is established between an approver group and one or more inventory areas, authorizing members of the approver group to review (and then approve or deny) packages related to the inventory areas.

approver type

An *approver type* is the specification of the kinds of packages that an approver group can review. When the approver type is Standard, the approver group can only review standard packages related to its authorized inventory areas. When the approver type is Emergency, the approver group can only review emergency packages related to its authorized inventory areas.

Archive Elements Action

The *Archive Elements Action* writes an element and all related CA Endevor SCM information to a sequential data set.

authorization

An *authorization* is the ability to perform certain privileged functions within the IBM z/OS environment.

Automated Configuration Manager (ACM)

The *Automated Configuration Manager (ACM)* is an optional facility that enables you to monitor selected libraries and data sets and maintain a component list for each element in the monitored areas. The component list provides an audit trail of program-component information at the time of each compile.

backin

The *backin* command restores the executable members of a package to the state they were in before the package was backed out. A backin will also reverse the backout process.

backout

The *backout* command returns the executable members of a package to the state they were in prior to package execution.

base level

The *base level* is the lowest level of an element within a particular stage when storing elements with forward deltas. This level represents the source for the element in that stage. If an element exists in both stages, there is a base level in each stage. When storing elements with reverse deltas, the base level is the current level.

base library

The *base library* is a partitioned data set (PDS), a CA-Panvalet, a CA-Librarian, or ELIB file that stores the base members for elements defined to CA Endevor SCM. A base library is defined for each element type, but can be shared across types.

base member

The *base member* is a located in a base library. Each base member corresponds to an element, and contains the source for the base level of the element. Base member names are generated internally by CA Endevor SCM, and do not correspond to the element name unless you use reverse delta with non-encryption.

base regression

The *base regression* is the percent by which the statements stored in a new level of an element change the statements stored in the base level.

batch

A *batch* is an IBM term referring to an environment in which non-interactive programs are executed. In CA Endevor SCM, batch refers to the execution of actions and reports in a non-interactive region, as opposed to execution in foreground.

browse

The *browse* command lets you view the content of a data set, without being able to change the contents.

Bulk Data Transfer (BDT)

The *Bulk Data Transfer (BDT)* is an IBM transmission utility supported by the CA Endevor SCM package shipment utility. BDT is abbreviated as BDT2 for BDT version 2 and BDTN for BDT with NJE/NJI.

C1DEFLTS Table

The *C1DEFLTS Table* is the CA Endevor SCM Defaults Table.

cast a package

The *cast a package* command freezes the actions included in the package. After a package is cast, it cannot be edited and only approvers can work with it.

CCID definition data set

The *CCID definition data set* identifies the CCIDs to be used within CA Endevor SCM. The definition file must be a card-image data set (80-byte, fixed-format records).

CCID validation

The *CCID validation* checks a CCID specified on an action against the CCIDs defined in the CCID definition data set.

Change Control ID (CCID)

A *Change Control ID (CCID)* is a logical grouping mechanism by which user-specified portions of the CA Endevor SCM inventory can be tagged, and then viewed, tracked, and manipulated. CCIDs are optional, but may be required on a system-by-system basis. The same is true of comments.

change regression

The *change regression* is the percent by which the statements stored in a new level change the revisions made by the previous level.

checksum

The *checksum* is an internally calculated value within a package. CA Endevor SCM uses the checksum to determine if a package has been changed.

command field

The *command field* appears on the second line of panels in which you can specify a TSO command. You can enter any appropriate TSO command in the command field.

comment

A *comment* is a user-defined remark (up to 40 characters) associated with an action or package. A comment is used in conjunction with CCIDs, and generally describes the reason for the action or purpose of the package.

commit a package

The *commit a package* command records all events related to a package, and removes all backin and backout information. After a package is committed, the package can no longer be backed out or backed in.

committed

The *committed* status appears after a package has been committed.

complementary data sets

Complementary data sets can be shipped along with package shipments. The complementary data sets for a shipment contain a backout of the shipment.

component

A *component* (primarily for CA Endevor SCM ACM) is the output produced or the input read in by a generate or move processor. The components of a generated element include the input components that were included to produce an output of the generate processor, the element itself, all outputs created by the generate or move processor, and user-defined data. Components are referenced by element or member names.

component list

A *component list* is a list of all components created or read by a generate or move processor. The component list can be viewed using the Print Elements action, or the Display Element/Component panel (CA Endevor SCM ACM). The component list provides an audit trail of program-component information at the time of each compile.

component monitoring

Component monitoring is a feature of CA Endevor SCM ACM that lets you check selected data sets for component relationships.

component validation

Component validation occurs when a package is cast. CA Endevor SCM then validates that all dependent components are in the package and that those components have not changed since they were last used.

configuration management

Configuration management is the capture and storage of program-component relationships and the tracking of these relationships over time.

CONNECT:Direct

CONNECT:Direct is a network transmission utility provided by Sterling Commerce and supported by the CA Endevor SCM package shipment utility. It is abbreviated as NWD when establishing destinations. CONNECT:Direct was formerly known as Network DataMover.

consolidation

A *consolidation* is a CA Endevor SCM facility that lets you specify a number of delta levels to retain when a member reaches the consolidation level specified for its type. If you do not specify the number of levels to retain, CA Endevor SCM consolidates all levels.

control section (CSECT)

The *control section (CSECT)* is an IBM term for the part of a program that is a relocatable unit and for which all components are loaded into adjoining main storage locations.

Copy Elements Action

The *Copy Elements Action* copies an element from an archive data set to a data set external to CA Endevor SCM.

copyback

A *copyback* (also known as fetch) searches for an element along the map (beginning at a designated stage), finds the element, and then copies it back to the initial stage. Copyback is available as an explicit option with the Generate Elements action. Copyback is also used when adding, transferring, and moving elements.

create a package

Create a package is performed to build the SCL for a package, and then associate the SCL with other package-related information such as a package ID, an execution window, and so forth.

current level

The *current level* is the most recent source for an element. When using forward deltas, the current level of an element comprises the base level plus all subsequent change levels. When using reverse deltas, the current level of an element is the current source.

data set name (DSN)

The *data set name (DSN)* is the data set name for the SCL.

data set validation

Data set validation is an optional feature to verify that retrieved elements are added or updated from the data set to which they were last retrieved. This ensures that the same copy of the element (revised as appropriate) is placed back in CA Endevor SCM. Data set validation can be specified separately by system. If this facility is in effect for a system, you can override it to add the element back from a different data set, provided you have proper authority.

default

A *default* value is the value that CA Endevor SCM assumes to be in a field or statement if you do not provide an alternative value. On foreground panels, fields usually display default values.

defaults table

The *defaults table* is a reference to C1DEFLTS, also known as the CA Endevor SCM Defaults Table.

Delete Elements Action

The *Delete Elements Action* erases base and delta forms of an element and removes element information from a Master Control File (MCF) or a component list.

delete processor

A *delete processor* is a processor that is run when an element is deleted from a stage. Typically, the delete processor deletes the output created by the corresponding generate processor.

delta level

A *delta level* is a record of a change to the base level of an element. Each change to an element creates a delta level. CA Endevor SCM compares the current level to the new source and builds a delta level containing the changes to the source.

delta library

A *delta library* is a partitioned data set (PDS), a CA-Panvalet, a CA-Librarian, or a ELIB file that stores the delta members for elements defined to CA Endevor SCM. A delta library is defined for each element type.

delta member

A *delta member* is a member in a delta library. Each delta member corresponds to an element, and contains all the levels for that element subsequent to the base level. Delta member names are generated internally by CA Endevor SCM and do not correspond to the element name, unless reverse deltas are used.

denied

A *denied* status of a package appears when the package has been reviewed, but denied, by an approver.

deny a package

The *deny a package* command is an option for a package approver. If one approver denies a package, it cannot be executed.

destination

A *destination* is where package outputs are shipped. A destination record contains the information needed by CA Endevor SCM to ship package outputs to the destination.

Display Elements Action

The *Display Elements Action* is used to view environment definitions, element information, and footprint-related data.

DSN mapping rule

A *DSN mapping rule* is a user-defined correspondence between host and remote data set names. DSN mapping rules are used when shipping package outputs.

element

An *element* is a partitioned data set (PDS) member, a CA-Panvalet, CA-Librarian, or sequential data set that has been placed under control of CA Endevor SCM. The default element name is the member name.

element change

The *element change* is a view of element information that shows the current level of an element, annotated to indicate the level at which each line was added to the source.

element component

The *element component* is the part of a component list referred to as element information. This information includes the footprint of the CA Endevor SCM source.

element history

The *element history* is a view of element information that shows all lines that have been present in a piece of source code, annotated to show the level at which the line was added, deleted, or both from source.

element master

The *element master* is a view of Master Control File (MCF) information about an element.

element name

The *element name* is the name assigned to an element, used to identify the element within CA Endevor SCM. It is recommended that any outputs created by output management be assigned the name of the corresponding element. Element names must be unique within each system, subsystem, and type combination.

element summary of levels

The *element summary of levels* is a summary view of activity against an element at all levels. Information provided includes the number of statements at each level, the number of lines added, and the number of lines deleted.

emergency approval

Emergency approval is assigned to emergency packages. An approver group must be assigned the authority to approve emergency packages.

emergency package

An *emergency package* is one of two types of packages (either standard or emergency when created). Emergency packages require approval from emergency approver groups.

enable backout

The *enable backout* option appears when creating a package. You can decide whether or not to allow the package to be backed out.

Endevor classification

Endevor classification is the system, subsystem, and type associated with an element.

Endevor Defaults Table

The *Endevor Defaults Table* (also known as C1DEFLTS) is a table of site-specific information necessary for CA Endevor SCM operation. The Endevor Defaults Table includes environment and stage definitions, installed options, and site-specific hardware settings. There is only one Endevor Defaults Table for each site.

Endevor LIB

An *Endevor LIB* (ELIB) is a high performance alternative to OS partitioned data sets under CA Endevor SCM. These data sets can reorganize member space automatically as members are rewritten or deleted, exploit 31-bit storage for VSAM-organized data sets, expand directories and data sets automatically, provide improved directory processing, and maintain additional statistical information about member size.

Endevor Link

Endevor Link is a CA product that enables communication between CA Endevor SCM Workstation and CA Endevor SCM. This is also known as CA Endevor SCM Change Manager Link Option.

Endevor listing libraries

Endevor listing libraries are libraries used to store compressed compiler listings produced by processors.

Endevor location

The *Endevor location* refers to the stage and environment where an element resides.

Endevor Parallel Development Manager (PDM)

Endevor Parallel Development Manager (PDM) is a CA product that automatically compares and integrates three versions of source code, allowing you to resolve conflicts resulting from concurrent development or from applying vendor updates to applications that have been customized in-house. Also known as the CA Endevor SCM Change Manager Parallel Development Option.

ENDEVOR return code (NDVR RC)

The *ENDEVOR return code (NDVR RC)* results from action processing. Values can be 00, 04, 08, 12, and 16.

environment

An *environment* is the top level of the logical structure used to classify elements in CA Endevor SCM. Environments usually correspond to functional levels in an organization (for example, development, quality assurance, and production). Each environment has two stages. There is no limit to the number of environments you can use.

environment name

The *environment name* is a name to identify each environment within CA Endevor SCM. The name can be up to eight characters, and include any of the following: A-Z, 0-9, @, #, and $.

environment title

The *environment title* is a descriptive title assigned to each environment that is used in various displays and reports (up to 40 characters).

execute a package

The *execute a package* command runs a package. Packages that execute successfully can be backed out or committed.

Execution Report

The *Execution Report* is output when you run CA Endevor SCM actions. The Execution Report documents the actions requested and the processing that occurred. The report can be viewed online by browsing data sets userid.C1TEMPR1.MSGS or userid.C1TEMPR2.MSGS. CA Endevor SCM prints the report as member C1MSGS1 on the batch SYSOUT.

execution window

The *execution window* is a timeframe (start and end date) within which a package must be executed.

exit

An *exit* is part of the CA Endevor SCM exit interface designed for use with exits written in either assembler or in high-level languages such as COBOL.

export a package

The *export a package* command copies package SCL into an external data set.

External Security Interface (ESI)

The *External Security Interface (ESI)* is an optional interface used to implement external security at your site. When installed, this interface replaces the native security facility supplied with the installation files (and implemented through the security tables), with calls to RACF, eTrust CA-ACF2 Security, or eTrust CA-Top Secret Security. It is also known as the Endevor Change Manager Interface for External Security.

footprint

A *footprint* is encrypted data added by processors to individual source, object, or load modules, to identify the CA Endevor SCM element associated with the module. CA Endevor SCM uses the data to display and process information for the element. A footprint includes the site ID, environment name, stage number, system name, subsystem name, element name, element type, element version/level, and the date and time the footprint was assigned, all in encrypted format.

foreground

The *foreground* is an IBM term referring to an environment in which interactive programs are executed. In CA Endevor SCM, you run actions in foreground by requesting those actions through the foreground Options Menu.

forward delta

Forward delta is a method for recording changes that stores a base version of code, and then builds current versions by applying changes made to the base.

forward recovery

Forward recovery is the process of taking an old level of an element and making it the current (new level). As a result, any changes made by the levels between are backed out. To perform forward recovery, you first retrieve the older (to-be-recovered) level, and then add or update the element using the retrieved source to create the new level.

Generate Elements Action

The *Generate Elements Action* translates source into executables, and then populate output libraries with the executables by executing the generate processor for an element.

generate processor

A *generate processor* translates source into executables, and then populates output libraries with these executables.

Global Type Sequencing

Global Type Sequencing is an optional method of defining the order in which element actions are processed. When global type sequencing is enabled, element actions are processed by type sequence regardless of the inventory location of each action. API element actions (if wildcarded) and SCL element actions are executed in the type sequence order defined at the site level in the Type Sequence member created by the CA Endevor SCM administrator.

group name

A *group name* is a name associated with a security configuration that applies for multiple users, found within the definition of the Access and User Security Tables. The name is then associated with any number of specific user IDs, to associate the IDs with the group-level security. This is a convenient way to assign security to several users having identical levels of access to the CA Endevor SCM environment.

identify record (IDR)

An *identify record (IDR)* is an IBM term for a record in a load module that contains user-defined data. An IDR is created by the linkage editor when it encounters an IDENTIFY statement in the object deck. Within CA Endevor SCM, IDRs are used to store the footprints associated with load modules.

image

An *image* is the current level of an element, when the element is stored in reverse delta format, using non-encryption.

image library

An *image library* is a library that contains elements stored in reverse delta format, using non-encryption.

import a package

The *import a package* command is used to create a package by copying SCL from an external data set.

INCLUDE library

An *INCLUDE library* is a CA-Panvalet library, CA-Librarian library, or partitioned data set (PDS) that contains INCLUDE members referenced within CA Endevor SCM elements. This library is optional and can be defined for each element type. The INCLUDE library is used by Retrieve Elements actions if you specify that you want to expand INCLUDEs at the time the element is retrieved. It is also used by the CONWRITE utility, if you specify that you want to expand INCLUDEs during CONWRITE processing.

input component

An *input component* is an included component to produce an output when executing the generate or move processor when using CA Endevor SCM ACM. For example, a copybook is the input component when compiling a COBOL program.

inventory

An *inventory* contains the software components that make up your application software systems.

Inventory Analyzer

The *Inventory Analyzer* is a CA product that lets you analyze your software inventory, classifying it according to CA Endevor SCM types. This product is used when implementing CA Endevor SCM.

inventory area

The *inventory area* is a subset of a software inventory, defined by its CA Endevor SCM location (environment and stage) and classification (system, subsystem, and type).

jump

A *jump* moves an element from stage 2 in one environment to a stage in another environment on a map route, when a version of the element exists at an intermediate stage that is not part of the map route.

last action

The *last action* is the most recent action executed for an element. Once executed, each action is recorded as the last action except Archive, Delete, Display, List, and Print.

last action CCID

The *last action CCID* is the CCID specified for the last action executed against an element.

level

A *level* is the source for an element at a particular time. When an element is first added to a stage, there is one level (known as the base level). Each time CA Endevor SCM actions change the source thereafter, a new delta level is created.

level number

The *level number* is an identifier for a specific level of an element. CA Endevor SCM assigns each set of changes a level number that is one higher than the number assigned to the preceding level.

library management

Library management is the classification, control, and storage of the physical components of a software inventory.

List Elements Action

The *List Elements Action* lists elements from a Master Control File (MCF) or archive data set in the form of action requests or members from a library. This action can also be used for text scanning. If the Automated Configuration Manager facility is installed, this action can also search a component list based on specified criteria.

List Panel

The *List Panel* is used by CA Endevor SCM to display lists of systems, subsystems, types, elements, or members, and also displays selection options for users. List panels are prepared and processed by the ISPF Table Display Facility.

load utility

The *load utility* is a CA Endevor SCM utility that is used to load members from an external data set into any stage in an environment.

map

A *map* is the promotion routes established for software inventory at a site. Environments and stages are mapped to each other in the CA Endevor SCM Defaults Table. Systems, subsystems, types, and processor groups are mapped to each other on their respective definition panels.

Master Control File (MCF)

The *Master Control File (MCF)* is a CA Endevor SCM file that contains the definitions of stages, systems, subsystems, element types, and elements. This file is accessed and updated by CA Endevor SCM to manage the element definitions, to execute processors, and for other miscellaneous functions. There are two Master Control Files (MCFs) for each environment-one per stage.

MAXRC

The *MAXRC* is a processor keyword that defines the highest acceptable return code for a processor step. If a step exceeds this return code, the CA Endevor SCM return code (NDVR RC) is set to 12. When this happens, the Element Master display shows *FAILED* in the NDVR RC field.

model transmission control statements

Model transmission control statements control the functioning of data transmission programs used by the package shipment utility.

Move Elements Action

The *Move Elements Action* moves elements between stages, within or across environments.

move processor

Move processors copy outputs, element information, and component lists from the source location to the target location of a Move or, optionally, a Transfer action.

name mask

A *name mask* enables you to use the asterisk (*) wildcard character and percent sign (%) placeholder characters when performing actions. Use the wildcard character to specify all names or all names beginning with a particular search string. Use the placeholder character to define a specific position within the search string. For example, the search string UPD% would return four character names beginning with UPD.

native security

The *native security* is an option supplied with the CA Endevor SCM installation files.

notification facility

The *notification facility* is a CA Endevor SCM facility that lets you notify users of events that require a response from them.

output component

An *output component* is created as a result of executing the generate or move processor when using CA Endevor SCM ACM. For example, an object deck is an output component when compiling a COBOL program.

output library

An *output library* is any of several libraries used during output management, including the CA Endevor SCM processor listing library, processor load library, and source output library; as well as user copy libraries, load libraries, listing libraries, macro libraries, JCL libraries, databases, and so forth.

output management

Output management is an aspect of CA Endevor SCM which handles the creation and maintenance of various outputs that relate to an element. The exact nature of these outputs varies depending on the corresponding element type, and is defined by the output management for that type.

package

A *package* is a group of CA Endevor SCM actions that requires approval before it can be executed. Creating packages lets you group specific actions so they can be maintained and tracked as a single unit, establish formal approval procedures to ensure data integrity through modifications, and centralize specific action groups so you can see them across environments and reuse them.

package data set

A *package data set* is where packages are stored. There is one package data set for an environment.

package events

The *package events* are an audit trail recording the events that have occurred involving a package, logged by user ID, date, and time. Package events relate to the various steps of the package processing procedure, and include Created, Last Updated, Cast, Approved, Executed, Backed Out, Backed In, Committed.

package exits

A *package exit* is called before, after (or both) package functions and subfunctions.

package shipment

A *package shipment* is the transmission of package outputs, and optionally their backouts, from host sites to remote sites.

package status

A *package status* indicates the status of a package at any specific time. Status levels for packages include In-edit, In-approval, Denied, Approved, In-execution, Executed, Exec-failed, and Committed.

partitioned data set (PDS)

A *partitioned data set (PDS)* is a data set in direct access storage that is divided into partitions, called members. Each member can contain a program, part of a program, or data.

Print Elements Action

The *Print Elements Action* prints element or member information.

PROC statement

A *PROC statement* is a job control statement used in catalogued or in-stream procedures. PROC statements can be used to assign default values to symbolic parameters contained in a procedure. A PROC statement is also used to mark the beginning of in-stream procedures.

processor component

A *processor component* is the part of a component list that includes processor information when using CA Endevor SCM ACM. This information includes the footprint of the processor.

processor group

A *processor group* identifies a set of processors for a specific element type, as well as the default symbolic overrides for the processors' JCL. A group can include up to three processors (one generate, one delete, and one move processor, or any combination). Processor groups are useful when elements of one type require slightly different processing. When you define a type to CA Endevor SCM, you can also identify a default processor group for that type. Using symbolics when writing the processors for the default processor group can allow you to use the same processors, by changing symbolic definitions, for other processor groups associated with this type.

processor group symbolics

Processor group symbolics are defined in PROC statements in one or more processors in a processor group. The symbolics and their default values appear on the Processor Group Symbolics panel. By modifying the default values, you can use one processor in more than one processor group.

processor listing library

The *processor listing library* is an optional library that stores the listings output from the CA-supplied processor named GPPROCSS.

processor load library

The *processor load library* is a CA Endevor SCM library that contains the load-module form of each processor. The modules from this library are executed when processors are invoked.

processor output library

The *processor output library* is a library (referred to in a processor) to which the processor writes output. Processor output libraries can be source libraries, executable libraries, or listing libraries.

processor return code (PROC RC)

The *processor return code (PROC RC)* is the highest return code from the execution of a processor. This returns code is set to *FAILED* if the return code for any step in a processor exceeds the MAXRC for the processor. This return code is set to *PROC'D?* if the element has not been generated after being restored or transferred from an archive data set or added/updated.

processors

A *processor* is a standard OS JCL job stream that manipulates elements and their outputs. There are generate, delete, and move processors. CA Endevor SCM supports both CA Endevor SCM symbolics and user-defined symbolics in processors. This capability lets you write one processor that you can use in multiple processor groups by changing the values assigned to one or more symbolics. CA Endevor SCM also provides a set of utilities for use when writing processors, and supports the use of in-stream data in processors.

production data set

The *production data set* is used to store production code. This term is used in the package shipment utility to refer to host and remote production data sets.

program pathing

Program pathing is a security option under RACF, eTrust CA-ACF2 Security, and eTrust CA-Top Secret Security, that lets you restrict the data sets available to particular users, as well as the programs and load libraries from which those data sets can be accessed. This is not a CA Endevor SCM option, but is specific to RACF, eTrust CA-ACF2 Security, and eTrust CA-Top Secret Security.

promote

The *promote* action is used to move an element from one inventory area to another inventory area.

promotion management

Promotion management is the task of coordinating and validating successive changes to the various inventory areas in a software development setting.

quorum

A *quorum* is the minimum number of approvers for which approval is required to execute a package. When a quorum size is specified, at least that many approvers must review and approve a package.

regression

Regression refers to the condition where one set of changes to element source is overwritten by a subsequent set of changes. CA Endevor SCM flags regression when the changes stored for a specific level of an element overwrite more than a predefined percentage of the element statements.

regression percent

The *regression percent* is percent of acceptable change to element source, defined for each element type. When this percent is exceeded, the result is a user specifiable CA Endevor SCM return code (0, 4, 8, or 12). Each time a new level is created for an element, CA Endevor SCM checks the changes stored in that level against this percent, both in terms of change to the statements stored for the base level (known as base regression) and change to the statements stored for the previous level (known as change regression). If the amount of change in either case exceeds the defined percent, CA Endevor SCM issues a message of user-defined severity.

Reload Utility

The *Reload Utility* lets you recover a CA Endevor SCM VSAM control file (Master Control File, package data set) or a base/delta data set that was lost as a result of a physical device failure or site disaster. The RELOAD action restores data from data sets created by the unload process.

remote footprint synchronization

Remote footprint synchronization is a procedure in which footprinted executables are shipped from a remote site to a host site, where footprint reports are run to compare the executables' footprints with host Master Control File (MCF) information.

remote nodename

A *remote nodename* is part of a package shipment destination that identifies the site to which package outputs are to be shipped. The name must be valid for the selected data transmission program.

request data set

A *request data set* contains action requests to be submitted for batch processing. You create request data sets in foreground using the SCL Generation facility.

request for data

The *request for data* is a package exit capability allowing package exit programs to make multiple, successive requests for CA Endevor SCM information on a single invocation of the exit.

reset a package

The *reset a package* action erases all package event records, returning the package to In-edit status.

Resource Security Table

The *Resource Security Table* defines the element names that are restricted to a particular system and subsystem, within a specific environment. The Resource Security Table is defined by the CA Endevor SCM administrator using the CONSDEF macro. Each CA Endevor SCM environment can contain, at most, one Resource Security Table.

Restore Elements Action

The *Restore Elements Action* restores an element to CA Endevor SCM from an archive data set.

Retrieve Elements Action

The *Retrieve Elements Action* copies any level of an element to an external data set.

reverse delta

The *reverse delta* is a method for recording changes that stores the most recent version of the code, rebuilding prior versions by backing out individual changes from the current version.

review a package

A *review a package* is performed to review the contents of the package. After reviewing a package, an approver either approves or denies the package.

route

A *route* is a series of environment and stage locations that make up the stages in a software lifecycle. Working together, all the routes at a site constitute the map for that site. See also map.

security

Security is a CA Endevor SCM feature that lets you restrict action requests and access to elements. The security system supplied with CA Endevor SCM (known as native security) is implemented using three tables.

sharable

A *sharable* package can be edited by users who did not create the package.

ship utility

The *ship utility* is a CA Endevor SCM utility that lets you ship package outputs to remote sites.

shipment confirmation

Shipment confirmation occurs in a package shipment after executing the data transmission utility and after executing the remote copy/delete job step.

shipment staging

Shipment staging is the creation and population of host staging data sets with package outputs or backout members.

Signin Elements Action

The *Signin Elements Action* removes the current signout for an element. Signin can be implicit or explicit.

signout

A *signout* is the assignment of a user ID to an element, establishing ownership of the element. Signout is automatic when adding or updating elements in or when retrieving elements from CA Endevor SCM.

site

The *site* is the location at which CA Endevor SCM is installed. The site is defined in the CA Endevor SCM Defaults Table, where it is assigned a site ID.

Software Control Language (SCL)

Software Control Language (SCL) is the CA Endevor SCM language that is used in batch processing to maintain or act against elements within CA Endevor SCM.

software distribution

Software distribution is the automated distribution and synchronization of software changes and the tracking of the implementation of the changes.

software lifecycle

The *software lifecycle* defines the stages through which software passes at a site during development and maintenance. A software lifecycle may consist of development, testing, quality assurance, and production.

software management

Software management is the process of tracking changes to software components and their interrelationships over time. This process includes configuration management, library management, software distribution, and version control.

source

The *source* is the non-executable form of an element.

source library

A *source library* is any of several libraries used during source management, including CA Endevor SCM base libraries, delta libraries, and INCLUDE libraries.

source management

Source management is the aspect of CA Endevor SCM that deals with the creation and maintenance of element source. Element source is maintained in base and delta libraries, in either an internal format or in standard IBM format (if reverse deltas and non-encryption are selected).

source output library

A *source output library* is a CA Endevor SCM library that contains the latest full source version of each element. This library is designed for use with copybooks, macros, procedures, and so forth that are copied to other places. This is an optional library. However, if used, it is specified in the definition of the corresponding element type.

stage

A *stage* is a step in the software lifecycle. There are two stages defined for each CA Endevor SCM environment.

stage ID

The *stage ID* is an identifier for the stage that is used when processing an element action to select (identify) the stage you want to process.

stage name

The *stage name* is assigned to each stage during CA Endevor SCM installation. A stage name can include any of the following characters: A-Z, 0-9, @, #, and $.

stage number

The *stage number* (either 1 or 2) is the relative number for the stage within the environment.

stage title

The *stage title* (up to 20 characters) is assigned to each stage and is used in displays and reports to describe the stage.

standard approval

A *standard approval* is one of two types of package approval. Standard packages can only be approved by standard approver groups.

standard package

A *standard package* is one of two types of packages (standard or emergency). Standard packages require approval from standard approver groups.

subsystem

A *subsystem* is a part of the CA Endevor SCM classification of an element. Subsystems are used for specific applications within a system. For example, you may have a purchase order and accounts payable subsystem within the financial system.

symbolics

A *symbolic* is used by CA Endevor SCM processors to represent a value specific to an individual run of a processor. CA Endevor SCM supports two kinds of symbolics (CA Endevor SCM symbolics and user symbolics). CA Endevor SCM symbolics (preceded by &C1) are used in CA Endevor SCM processors to represent a value specific to an individual run of the processor. Values are assigned to CA Endevor SCM symbolics when the processor is executed. User symbolics are defined by users in JCL PROC statements in processors. You can use one processor in multiple processor groups.

synchronize

A *synchronize* conflict is detected when transferring or moving an element with history, if the current level of the target does not match any level of the source. CA Endevor SCM searches for the level of the target to match a level of the source, and this level becomes the synchronization level. When there is a synchronize conflict, CA Endevor SCM does not allow the element to be transferred or moved unless the synchronization flag (SYNC option) is set to Y.

system

A *system* is a way to classify elements in CA Endevor SCM (defined to each environment where it will be used). A system typically represents the applications at a site. For example, you may have financial and manufacturing applications.

System Management Facilities (SMF)

System Management Facilities (SMF) is IBM's SMF.

System Management Facilities (SMF) interface

The *System Management Facilities (SMF) interface* is an optional interface to IBM's System Management Facilities (SMF) that lets you record historical information through SMF records. In CA Endevor SCM, they are called Action Records or Security Records. This information is used to generate historical reports. The implementation of the SMF interface is optional at each site, and the recording of historical information is optional within each environment.

System Management Facilities (SMF) records

System Management Facilities (SMF) records are written out if SMF recording is in effect, to document various CA Endevor SCM processing. An Action Record is written out at the end of (any) action processing, while a Security Record is written out for each security violation (or each error returned from the security exit, 01).

Transfer Elements Actions

The *Transfer Elements Action* transports elements from a source location to a target location. Each location can be either a CA Endevor SCM location or an archive data set.

transmission method

The *transmission method* is a part of package shipment destinations to identify the transmission utility to be used to ship packages to the destination.

transportable footprints

A *transportable footprint* is a CA Endevor SCM footprint that can be imbedded in DOS/VSE- and VM/CMS-bound object modules using a z/OS compiler, a VSE compiler, or a VM compiler.

type

A *type* is a category of source code that is used as part of the classification of a CA Endevor SCM element. For example, there are types for COBOL (COBOL code), COPYBOOK (copybooks), JCL (JCL streams), and so forth.

type processing sequence

A *type processing sequence* is a relative sequence of processing for the element types defined to each system. By defining a processing sequence, for example, you can make sure that copybooks (type COPYBOOK) are updated before any COBOL programs (type COB) that may use the copybooks.

Unload Utility

The *Unload Utility* unloads and validates the contents of the VSAM Master Control Files (base and delta files associated with the environments and systems specified on the job request). The file created by the Unload function contains a backup of all internal MCF definitions (system, subsystem, type, type sequence, data set, element master record) and base/delta data (element base, element delta, component base, component delta). Packages contained within a package data set can also be unloaded. The utility can be run for an entire environment or for selected systems within an environment. Unload may also be directed to backup an entire package data set or individual packages.

Update Elements Action

The *Update Elements Action* adds a member to CA Endevor SCM when an element with the same name is located in the target entry stage.

User Exit Table

The *User Exit Table* identifies exit programs to be called at each CA Endevor SCM exit point.

user ID

The *user ID* is used in TSO for actions that run in foreground for the session. For those actions run in batch, the user ID is the job name or the ID specified through the USER= parameter on the job card, depending on how your CA Endevor SCM Defaults Table is set up.

user menu facility

The *user menu facility* allows your CA Endevor SCM administrator to attach user-defined functions to the CA Endevor SCM TSO/ISPF front end.

User Security Table

The *User Security Table* defines the systems and subsystems to which users have access, and for each system and subsystem, the type of processing (authorization level) allowed. Each environment has one User Security Table.

validate

The *validate* function lets you ensure the integrity of one or more existing CA Endevor SCM environments and systems, and their related elements and components. These are the same checks performed as part of Unload processing, allowing this function to operate in a standalone mode.

version

The *version* is a two-digit identifier associated with an element. You cannot have two versions of an element in the same environment.

version control

Version control is the maintenance, tracking, and auditing of modifications to an application over time, allowing prior development versions to be restored.

version number

The *version number* identifies the version assigned to an element.

vvll

The vvll identifier refers to a particular version (vv) and level (ll) of element source.

Index

R

S

T

U

V

W