Natural Engineer

Utilities for Unix

Version 9.1

October 2018

ADABAS & NATURAL

TABLE OF CONTENTS

ABOUT THIS MANUAL

Purpose of this manual

This manual contains the Utilities for Natural Engineer.

It describes the various utility options available within Natural Engineer, which include:

- Automating the execution of Natural Engineer processes using the Task Scheduler option.

- Convert Natural Reporting mode objects into Natural Structured mode objects using the Mode Conversion option.

- Review maintenance changes within objects using the Change Management Tracking option.

- Apply Global or Application specific Coding Standards to be applied to Natural Objects.

- Extract and Load SQL Tables into your Natural Engineer Repository.

Target Audience

The target audience for this manual is intended to be any User of Natural Engineer at any level of experience.

Typographical Conventions used in this manual

The following conventions are used throughout this manual:

UPPERCASE TIMES	Commands, statements, names of programs and utilities referred to in text paragraphs appear in normal (Times) uppercase.
`UPPERCASE BOLD COURIER`	In illustrations or examples of commands, items in uppercase bold courier must be typed in as they appear.
< >	Items in angled brackets are placeholders for user-supplied information. For example, if asked to enter <file number>, you must type the number of the required file.
<u>Underlined</u>	Underlined parts of text are hyperlinks to other parts within the online source manual. This manual was written in MS-Word 97 using the "hyperlink" feature.

The following symbols are used for instructions:

$\Rightarrow$	Marks the beginning of an instruction set.
□	Indicates that the instruction set consists of a single step.
1.	Indicates the first of a number of steps.

How this manual is organized

This manual is organized to reflect all the Utilities options of Natural Engineer in the following chapters:

Chapter	Contents
1	Describes the Task Scheduler option, which provides the facility to automate the execution of the various Environment, Analysis and Modification options. This provides the facility to execute long-running batch tasks in an unattended mode to minimize the impact on machine and human resources.
2	Describes the Mode Conversion option, which provides the facility to convert Natural Reporting mode objects into Natural Structured mode objects.
3	Describes the Change Management Tracking (CMT) option, which provides the facility to review audit trail records of changes applied to individual objects within Natural Engineer.
4	Describes the Architectural Governance option, which provides the facility to set Coding Standards Criteria to be applied to Natural Objects.
5	Describes the Bulk Extract & Load option, which provides the facility to extract and load multiple applications into your Natural Engineer repository.
6	Describes the Bulk Import Object Documentation option, which provides the facility to import documentation from multiple applications into your Natural Engineer repository.
7	Describes the SQL Table Maintenance option, which provides the facility to extract and load SQL Tables into your Natural Engineer Repository.

Terminology

This section offers some of the terms that are specific to the Natural Engineer product.

Note: Familiarity is assumed with the general terminology of Natural, Adabas, Microsoft and Mainframe operating systems.

Analysis

The Analysis process of Natural Engineer searches application data within the Natural Engineer Repository, according to specified Search Criteria and generates reports on the search results.

Application

An Application is a library or group of related libraries, which define a complete Application. In Natural Engineer, the Application can have a one-to-one relationship with a single library of the same name, or a library of a different name, as well as related steplibs. The Application refers to all the source code from these libraries, which Natural Engineer loads into the Repository.

Browser

An Internet Browser such as Microsoft Internet Explorer or Netscape.

Category

Categories in Natural Engineer specify whether and how a Modification is applied to the Natural code. Valid categories are: Automatic change, Manual change, Reject the default Modification, No change to the data item, and the data item is in Generated Code.

A category is further broken down according to type of change (for example: Keyword, Literal, Data Item, Database Access, Definition).

Cobol

Abbreviation of Common Business Orientated Language. A programming language.

Cobol Link

A Cobol Link is the link between the individual Cobol modules and the executable Cobol program referenced in the JCL object.

Consistency

An option in the Analysis process that causes Natural Engineer to trace an Impact through the code, using left and right argument resolution to identify further code impacted by the code found.

Data Item

A collective term used for any data fields within a programming object. These can be user-defined variables, DDM fields or System Variables. It is inter-changeable with the term 'variable'.

Environment

The Environment process is the means by which Natural Engineer generates a structured view of the application code in the Natural Engineer Repository. This provides application analysis reports and inventory information on the application and is used as the basis for Impact Analysis.

Exception

An Exception is an Item identified as impacted that does not require a Modification. Where there are a few similar Exception Items, they can be treated as Exceptions, and rejected in the Modification review process. Where there are many similar (therefore not Exceptions), consideration should be given to changing the Search Criteria so they are not identified as impacted in the first place.

Generated Code

This is code which has been generated by a Natural code generator, such as Construct, and which is not normally modified directly in the Natural editor.

Impact

An Impact is an instance of a Natural code Item; e.g., data item or statement (a "hit" scored by the Analysis process) that matches the defined Search Criteria used in the Analysis process.

Iteration

An Iteration is one examination cycle of a field identified according to the specified Search Criteria. For example, one Iteration is reading the field right to left. Multiple Iterations are performed when the option of 'Consistency' or Multi Search is requested for Analysis, and Natural Engineer performs as many Iterations as necessary to exhaust all possibilities of expressing and tracing the field, and can be limited by a setting in the NATENG.INI file.

JCL

Job Control Language.

JCL object

A JCL object is a collection of Job Control statements in the order which they are to be executed in a mainframe batch environment. Commonly referred to as JCL.

Library

A single library of source code, which exists in the Natural system file.

Modification

A Modification is a change suggested or made to an object or data item resulting in the required compliance of that object or data item. Modifications in Natural Engineer are classified according to Category and Type.

Natural Engineer Utilities

Refactoring

Improving a computer program by reorganizing its internal structure without altering its external behavior.

Soft Link

A Soft Link is where a link between two objects has been defined using an alphanumeric variable rather than a literal constant.

TLM

Text Logic Members are used to contain the code required to support inclusion of common code into the application. An example of this is the code to include into an application before updating a database.

Type

The Type of Modification available, for example: Data Item, Keyword and Literal.

Variable

A collective term used for any data fields within a programming object. These can be user-defined variables, DDM fields or System Variables. It is inter-changeable with the term 'data item'.

Related Literature

The complete set of Natural Engineer manuals consists of:

1   Natural Engineer Concepts and Facilities (NEE91-006ALL)

The Concepts and Facilities manual describes the many application systems problems and solutions offered by Natural Engineer, providing some guidelines and usage that can be applied to Natural applications.

2   Natural Engineer Release Notes (NEE91-008ALL)

The Release Notes describe all the information relating to the new features, upgrades to existing functions and documentation updates that have been applied to Natural Engineer.

3   Natural Engineer Installation Guide for Windows (NEE91-010WIN)
Natural Engineer Installation Guide for Mainframes(NEE91-010MFR)
Natural Engineer Installation Guide for Unix (NEE91-010UNX)

The Installation Guide provides information on how to install Natural Engineer on PC, Unix and mainframe platforms.

4   Natural Engineer Administration Guide (NEE91-040WIN)
Natural Engineer Administration Guide (NEE91-040MFR)
Natural Engineer Administration Guide (NEE91-040UNX)

The Administration Guide provides information on all the various control settings available to control the usage of the different functions within Natural Engineer.

5   Natural Engineer Application Management (NEE91-020WIN)
Natural Engineer Application Management (NEE91-020MFR)
Natural Engineer Application Management (NEE91-020UNX)

The Application Management manual describes all the functions required to add Natural applications into the Repository.

6   Natural Engineer Application Documentation (NEE91-022WIN)
Natural Engineer Application Documentation (NEE91-022MFR)
Natural Engineer Application Documentation (NEE91-022UNX)

The Application Documentation manual describes all the available functions to document a Natural application within the Repository. These functions will help enhance / supplement any existing systems documentation such as BSD / CSD / Specifications etc.

Natural Engineer Utilities

7   **Natural Engineer Application Analysis and Modification (NEE91-023WIN)**
Natural Engineer Application Analysis and Modification (NEE91-023MFR)
Natural Engineer Application Analysis and Modification (NEE91-023UNX)

The Application Analysis and Modification manual describes all the available functions to carry out analysis of Natural applications; including basic keyword searches. The modification process is described and detailed to show how it can be applied to modify single selected objects within a Natural application, or the entire Natural application in one single execution.

8   **Natural Engineer Application Restructuring (NEE91-024WIN)**
Natural Engineer Application Restructuring (NEE91-024MFR)
Natural Engineer Application Restructuring (NEE91-024UNX)

The Application Restructuring manual describes the analysis and modification functionality required to carryout some of the more sophisticated functions such as Object Builder.

9   **Natural Engineer Utilities (NEE91-080WIN)**
Natural Engineer Utilities (NEE91-080MFR)
Natural Engineer Utilities (NEE91-080UNX)

The Utilities manual describes all the available utilities found within Natural Engineer and, when and how they should be used.

10   **Natural Engineer Reporting (NEE91-025ALL)**

The Reporting manual describes each of the reports available in detail, providing report layouts, how to trigger the report and when the report data becomes available. The various report-producing mediums within Natural Engineer are also described.

11   **Natural Engineer Batch Processing [Mainframes] (NEE91-026MFR)**
Natural Engineer Batch Processing [Unix] (NEE91-026UNX)

The Batch Processing manual describes the various batch jobs (JCL/Scripts) and their functionality.

12   **Natural Engineer Messages and Codes (NEE91-060ALL)**

The Messages and Codes manual describes the various messages and codes produced by Natural Engineer.

13   **Natural Engineer Web Interface Installation and Configuration Guide(NEA84-010ALL)**

The Web Interface Installation and Configuration Guide provides information on how to install and configure the Natural Engineer Web Interface.

14   **Natural Engineer Advanced Services (NEE91-017WIN)**
Natural Engineer Advanced Services (NEE91-017MFR)
Natural Engineer Advanced Services (NEE91-017UNX)

The Advanced Services manual describes various advanced options such as the Refactoring of Natural application source code with Natural Engineer, conversion of applications for Natural for Ajax, Business Rule processing and Data Masking.

1

TASK SCHEDULER

Chapter Overview

This chapter describes the Task Scheduler option within Natural Engineer, which is used to manage and execute automatically Natural Engineer Environment, Impact and Modification tasks. These tasks will execute in unattended batch mode.

The topics covered in this chapter are:

1. Task Scheduler Overview
2. Task Scheduler Summary Screen
3. Task Details Screen
4. Task Scheduler Initiator Batch Job

Task Scheduler Overview

The Task Scheduler option allows you to specify tasks to be executed at a specified date and time. These tasks may be scheduled to run with a frequency of once, daily, weekly or monthly and may also have dependencies on other defined tasks to allow a series of tasks to run in a controlled sequence.

Once the task or tasks have been specified, they are ready for execution and will be triggered by the Task Scheduler Initiator batch job. This must be invoked in order for the tasks to be released for execution (if their individual specifications have been attained).

Each task execution will have a history log entry available showing the execution events for that task. Any tasks that have experienced problems during execution will be highlighted with a status of 'Error' in the Task Scheduler Summary screen.

Using the Task Scheduler

The Task Scheduler allows long executing tasks for individual applications to be scheduled to run overnight or at weekends, in order that the normal working day is less disrupted waiting on these tasks to complete.

For example: If an extract and load of a very large application is required, where the extract and load processes are likely to take several hours each to complete, then a task can be scheduled to start executing at the end of the normal working day (i.e., overnight). The task can then be reviewed the next working day.

Using the Frequency Task option

A frequency can be set for any tasks that are to be repeated many times. This means a task need only be specified the once, but with the appropriate frequency set, it will execute each time it is initiated via the Task Scheduler.

For example: Natural Engineer is being used to maintain an application where various modifications are being applied using the modification options of Natural Engineer. The application has been specified within Natural Engineer to apply all the modifications to the base application library. To maintain integrity of the modified application on the Repository, the Extract Source Code function with Synchronize Source Code option activated requires running once a week. A task would be set up with the correct details specified to run Extract Source Code with a frequency of 'Weekly' set.

Using the Dependency Task option

Task dependencies can be specified, so that tasks are only initiated provided the previous task has completed.

For example: Task 0001 is specified to run an Extract Source Code function for an application. A second task, 0002, is specified to run the Load Repository function for the same application with a dependency of task 0001 set. When the initiator is run, task 0002 will not execute until task 0001 has completed.

Task Scheduler Summary Screen

The Task Scheduler Summary screen is accessed by selecting option 'T' (Task Scheduler) from the Utilities Menu screen.

The Task Scheduler Summary screen utilizes left and right screen scrolling to show the complete information for each task. When the screen is positioned to the left, all the task execution information is displayed. When the screen is scrolled to the right, all the task function information is displayed.

The following Figure 1-1 illustrates the Task Scheduler Summary screen displaying the task execution information.

```
                    - Task Scheduler Summary -

  Task User       Date       Time  Freq Depn Status     Application
_ 0040 XGSLXX     09/09/2001 14:29           Complete   HOSPITAL
_ 0041 XGSLXX     09/09/2001 14:35           Pending    HOSPITAL
_ 0042 XGSLXX     10/09/2001 22:00 Week                 PAYROLL










Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help            Exit  Add               Prev  Next        W<    W>    Main
```

Figure 1-1 Task Scheduler Summary screen displaying task execution information

The following Figure 1-2 illustrates the Task Scheduler Summary screen displaying the task function information.

```
                        - Task Scheduler Summary -

  Task User      Date        Application Function Grp Function Name        Ver
_ 0001 XGSLXX    09/09/2001  NEEDOCN     Environment   Extract Source Code
_ 0002 XGSLXX    09/09/2001  NEEDOCN     Environment   Load Repository
_ 0003 XGSLXX    10/09/2001  PAYROLL     Analysis      Impact Execution      01











Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
     Help           Exit Add              Prev Next        W<    W>    Main
```

Figure 1-2 Task Scheduler Summary screen displaying task function information

SCREEN ITEMS	DESCRIPTION
Select	Select task to perform various task administration options. Valid selections are:
	'V' View task details.
	'U' Update task details.
	'R' Reset task.
	'D' Delete task.
	'L' View task log. Invokes the Task Log screen.
	'S' Suspend task.
Task	The task id that has been assigned for the task. This is an internally generated sequential number starting from 0001. Deleted task numbers are reused.
User	The user id of the person that added the task.
Date	The task execution start date specified in the Task Details screen.

SCREEN ITEMS	DESCRIPTION
Time	The task execution start time specified in the Task Details screen.
Freq	The task frequency. This controls the amount of executions for the task. The following frequencies are available:

'**O**'   Task will execute once only.

'**D**'   Task will execute daily at the same time based on the original date and time specified.

'**W**'   Task will execute weekly at the same time based on the original date and time specified.

'**M**'   Task will execute monthly at the same time based on the original date and time specified.

Depn	The task's dependency on other scheduled tasks. This will contain the task id of the task that needs to complete before this task will execute.
Status	The task status. The following are available:

' '   Task is waiting for the specified start date and time, i.e., a date and time have been set in the future.

Pending   Task has missed the specified date and time and is ready for execution the next time the Initiator is invoked.

Error   Task has experienced an error during execution.

Complete   Task has completed successfully.

In Progress   Task is currently executing.

Suspended   Task has been suspended and will not be released for execution until the Release option is used from the context menu.

Application	The name of the application being used by the task.
Function Grp	The main function area being used by the task. The function groups available are:

- **Environment**
- **Analysis**
- **Modification**

SCREEN ITEMS	DESCRIPTION
Function Name	The name of the sub-function within the main function. The sub-functions available are: For the function group Environment: ▪ **Extract Source Code** ▪ **Load Repository** ▪ **Extract & Load** ▪ **Extract, Load & Impact** ▪ **Environment Bulk Reports** For the function group Analysis: ▪ **Impact Execution** ▪ **Impact Bulk Reports** For the function group Modification: ▪ **Modify All** ▪ **Modification Bulk Reports**
Ver	The impact version number being used by the task. This is only applicable to function names: Impact Execution and Modify All. All other tasks will show '00'.

PFKEYS	DESCRIPTION
PF1	Activates the help function.
PF3	Exit from the current function and return to previous screen.
PF4	Add new Task. Invokes the Task Details screen.
PF7	Displays previous page.
PF8	Displays next page.
PF10	Scrolls screen display to the left.
PF11	Scrolls screen display to the right.
PF12	Returns to the Natural Engineer Main Menu.

Task Scheduler Task Log Screen

The Task Log screen is displayed when a task is selected using option '**L**' from the Task Scheduler Summary screen.

The Task Log screen will show history details of activity for the selected task.

The following Figure 1-3 illustrates the Task Scheduler Task Log screen.

```
                     - Task Scheduling - Task Log

                          Task Id: 0001

 09/09/2001 17:01:12  - Extract HOSPITAL - Started
 09/09/2001 17:01:14  - Extract HOSPITAL - Completed

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
     Help        Exit                    Prev  Next        W<    W>    Main
```

Figure 1-3 Task Scheduler Task log screen

SCREEN ITEMS	DESCRIPTION
Task Log Entries	Each line details a task event.

PFKEYS	DESCRIPTION
PF1	Activates the help function.
PF3	Exit from the current function and return to previous screen.
PF7	Displays previous page.
PF8	Displays next page.
PF10	Scrolls screen display to the left.
PF11	Scrolls screen display to the right.
PF12	Returns to the Natural Engineer Main Menu.

Task Details Screen

The Task Details screen is where each task can be specified and added to the Task Scheduler. This screen is also used to update or view the details for a task.

The Task Details screen can be invoked by using '**PF4**' (Add) from the Task Scheduler Summary screen, by selecting a task using selection options '**U**' (to update task) or '**V**' (to view task) from the Task Scheduler Summary screen.

The following Figure 1-4 illustrates the Task Scheduler Task Details screen.

```
                        - Task Details -

                 Task Id: 0004

             Start Date: 09 / 09 / 2001 (DD/MM/YYYY)
                    Time: 14 : 55         (HH:MM)

                  Job Id: XGSLXX04 Job Class: _
             Dependency: ____       Frequency: _

               Application:
            Function Group:
                  Function:

     Comments:
     _____



Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help         Exit        Save  Appl  Prop  Func  Vers  ECrt  ICrt  Main
```

Figure 1-4 Task Scheduler Task Details screen

SCREEN ITEMS	DESCRIPTION
Start Date	The date the task is to execute using date DD/MM/YYYY format. The default date will be the current date when the Task Details screen was invoked.
Start Time	The time the task is to execute using time HH:MM format. The default time will be the current time when the Task Details screen was invoked.
Job Id	The Job name. This defaults to the first 4 characters of the logon id plus the task id.
Job Class	The Job Class to be used during execution.
Dependency	Select a task id on which this task is to be dependent. Task ids available will be the current tasks available in the Task Scheduler Summary screen.
Frequency	The task frequency. This controls the amount of executions for the task. The following frequencies are available:

'**O**'   Task will execute once only.

'**D**'   Task will execute daily at the same time based on the original date and time specified.

'**W**'   Task will execute weekly at the same time based on the original date and time specified.

'**M**'   Task will execute monthly at the same time based on the original date and time specified.

Application	The name of the application that this task is to reference.
Function Group	The main function area being used by the task. The function groups available are:

- **Environment**
- **Analysis**
- **Modification**

SCREEN ITEMS	DESCRIPTION
Function Name	The name of the sub-function within the main function. The sub-functions available are:
	For the function group Environment:
	▪ **Extract Source Code**
	▪ **Load Repository**
	▪ **Extract & Load**
	▪ **Extract, Load & Impact**
	▪ **Environment Bulk Reports**
	For the function group Analysis:
	▪ **Impact Execution**
	▪ **Impact Bulk Reports**
	For the function group Modification:
	▪ **Modify All**
	▪ **Modification Bulk Reports**
Comments	Up to 72 characters of text can be entered to serve as a comment for the task. These are treated as information only.

PFKEYS	DESCRIPTION
PF1	Activates the help function.
PF3	Exit from the current function and return to previous screen.
PF5	Saves the Task details specified and return back to the Task Scheduler Summary screen.
PF6	Invokes the Select an Application screen. An existing application can be selected or a new application name input.
PF7	Invokes the Application Properties screen. Application Properties can be specified and saved for the selected application.
PF8	Invokes a series of pop-up windows to select first the Function Group and then the Function Name to be used for the task.

The Function Groups available are:
- **Environment**
- **Analysis**
- **Modification**

The Function Names available are:

For the function group Environment:
- **Extract Source Code**
- **Load Repository**
- **Extract & Load**
- **Extract, Load & Impact**
- **Environment Bulk Reports**

For the function group Analysis:
- **Impact Execution**
- **Impact Bulk Reports**

For the function group Modification:
- **Modify All**
- **Modification Bulk Reports**

PFKEYS	DESCRIPTION
PF9	Invokes the Impact Version screen. A new version can be created or an existing version selected. This option is only available for the following function names:
	▪ **Extract, Load & Impact**
	▪ **Impact Execution**
	▪ **Impact Bulk Reports**
	▪ **Modify All**
	▪ **Modification Bulk Reports**
	Note: The impact versions available are the same as for the application when using the Impact Version option from the Impact Analysis Menu screen.
PF10	Invokes the Extract and Load Selection Criteria screen. Extract selection criteria can be specified and saved and default Load settings may be overwritten.
	Note: The criteria specified here are only relevant to the task for which they have been defined. The task criteria are independent of the main application extract and load selection criteria defined using the Extract & Load Selection Criteria option from the Environment Menu screen.
PF11	Invokes the Impact Search Criteria Summary screen. Search criteria can be specified here and saved for the current version.
	Note: The impact search criteria available are the same as for the application when using the Impact Search Criteria option from the Impact Analysis Menu screen.
PF12	Returns to the Natural Engineer Main Menu.

Task Scheduler Initiator Batch Job

The Task Scheduler tasks are initiated using the Task Scheduler Initiator Batch Job: TASKSCH. This needs to be invoked before any scheduled tasks will be executed.

The Task Scheduler Initiator Batch Job makes use of program: MFUAEP01. Once invoked, MFUAEP01 will interrogate the Repository looking for any tasks that are ready for execution, i.e., any tasks that have a date and time stamp which is less than or equal to the date and time when MFUAEP01 is executing.

For each scheduled task that has all its criteria satisfied, the relevant batch job for the scheduled task will be submitted.

Note: The Task Scheduler Initiator Batch Job can be located as a member TASKSCH within dataset NEEvrs.JOBS supplied on the mainframe Natural Engineer installation tape. This contains sample JCL, which needs to be modified to meet local site standards.

In order to allow the scheduled tasks to execute, it is recommended that the TASKSCH job is entered into the sites automated operations scheduler to be executed at pre-determined times.

For more information on the Natural Engineer installation tape refer to Chapter 2 in the Natural Engineer Installation Guide.

MODE CONVERSION

Chapter Overview

This chapter describes the Mode Conversion option available from the Utilities menu. The Mode Conversion option provides the facility to convert Natural Reporting mode objects into Natural Structured mode objects.

The following topics are covered:

1. Mode Conversion Overview

2. Mode Conversion Techniques

3. Mode Conversion GDA Processing

4. Mode Conversion Object Processing

Mode Conversion Overview

Applications that utilize Natural Reporting mode can become more difficult to maintain, as they become more complex through continual expansion and development.

Key issues with Natural Reporting mode are:

- User defined variables can be defined anywhere, making it difficult to identify what variables are in use.

- Database fields do not need to be defined anywhere, making it difficult to identify the database data that is being referenced.

- Loop constructs are closed using the LOOP statement, making it difficult to identify where loop processing ends. For example READ, FIND, REPEAT and FOR blocks.

- Logical constructs are bound using the DO/DOEND statements, making it difficult to identify what conditional statements will be executed within complex logical blocks (nested IF statements).

Applications written in Natural Structured mode provide clear and well-defined program structure for complex application solutions.

Key benefits with Natural Structured mode are:

- All user defined variables are contained in the DEFINE DATA statement at the top of each object, making it easier to identify what variables are in use.

- All database fields must be defined using a view of the database file and are contained in the DEFINE DATA statement at the top of each object, making it easier to identify the database data being referenced.

- Loop constructs must be explicitly closed using corresponding 'END' constructs, making it easier to identify where each individual loop construct ends. For example END-READ, END-FIND, END-REPEAT and END-FOR.

- Logical constructs must be bound by using END-IF statements, making it easier to identify what conditional statements will be executed within complex logical blocks.

The Mode Conversion option provides the facility to convert Natural Reporting mode objects into Natural Structured mode objects.

Applications that are to be converted are extracted and loaded into the Repository using the Extract and Load processes. The modification library (where the converted objects will reside) is controlled within the Application Properties screen. The Mode Conversion option can be used once the application has been loaded into the Repository.

Note: For more information on the Application Properties, Extract and Load processes refer to the Application Management for Windows manual.

The Mode Conversion process is split into two sub-processes:

1. Global Data

The Global data requirements within an application are specified first and if required, a new GDA object can be generated. Once generated, it will contain all the Global data referenced within the application.

2. Object Conversion

Objects to be converted can be selected individually, in groups, or all objects within the application. The GDA option specified will be taken into account and the correct references added to the converted objects.

Mode Conversion can only be applied to the following Natural object types:

- Copycodes
- Helproutines
- Programs
- Subprograms
- Subroutines

After the Mode Conversion process has completed, the conversion details are displayed in the batch job output file.

Mode Conversion Techniques

This section describes the main conversion techniques applied during the Mode Conversion process.

Global Data Definitions

There are 2 conversion techniques available for Global data definitions:

1. Generate a new GDA.

- Will rationalize all existing GDA objects and/or in-line Global data definitions into one single new GDA object.

- Existing DEFINE DATA GLOBAL USING statements will be modified to reference the new GDA object name.

- A DEFINE DATA GLOBAL USING statement will be added to any objects that reference in-line Global data definitions only.

- Format and length attributes will be removed from in-line Global data definitions.

Example:

Before Conversion:

```
GDA1                                    GDA2
0010 DEFINE DATA GLOBAL                  0010 DEFINE DATA GLOBAL
0020 1 +ALPHA(A5)                        0020 1 +NUMBER(N3)
0030 END-DEFINE                          0030 END-DEFINE

PROGRAM1                                 PROGRAM2
0010 DEFINE DATA                         0010 DEFINE DATA
0020 GLOBAL USING GDA1                   0020 GLOBAL USING GDA2
0030 END-DEFINE                          0030 END-DEFINE
0040 MOVE 'ABCDE' TO +ALPHA              0040 MOVE 123 TO +NUMBER
0050 FETCH 'PROGRAM2'                    0050 MOVE 'HELLO WORLD'
0060 END                                 0060   TO +TEXT-GLOBAL(A20)
                                         0070 END
```

After Conversion:

<u>NEWGDA</u>
```
0010 DEFINE DATA GLOBAL
0020 1 +ALPHA(A5)
0030 1 +NUMBER(N3)
0040 1 +TEXT-GLOBAL(A20)
0050 END-DEFINE
```

<u>PROGRAM1</u>
```
::::
0050 DEFINE DATA
0060 GLOBAL USING NEWGDA
0070 END-DEFINE
0080 MOVE 'ABCDE' TO +ALPHA
0090 FETCH 'PROGRAM2'
0100 END
```

<u>PROGRAM2</u>
```
::::
0050 DEFINE DATA
0060 GLOBAL USING NEWGDA
0070 END-DEFINE
0080 MOVE 123 TO +NUMBER
0090 MOVE 'HELLO WORLD'
0100 TO +TEXT-GLOBAL
0110 END
```

2. Use existing GDA object structure.

- No new GDA objects will be generated.

- Existing DEFINE DATA GLOBAL USING statements will not be modified.

Example:

Before Conversion:

<u>OLDGDA</u>
```
0010 DEFINE DATA GLOBAL
0020 1 +LAST-PGM(A8)
0030 1 +NEXT-PGM(A8)
0040 END-DEFINE
```

<u>PROGRAM1</u>
```
0010 DEFINE DATA
0020 GLOBAL USING OLDGDA
0030 END-DEFINE
0040 MOVE 'PROGRAM1' TO +LAST-PGM
0050 MOVE 'PROGRAM2' TO +NEXT-PGM
0060 FETCH +NEXT-PGM
0070 END
```

After Conversion:

<u>OLDGDA</u>
```
0010 DEFINE DATA GLOBAL
0020 1 +LAST-PGM(A8)
0030 1 +NEXT-PGM(A8)
0040 END-DEFINE
```

<u>PROGRAM1</u>
```
::::
0050 DEFINE DATA
0060 GLOBAL USING OLDGDA
0070 END-DEFINE
0080 MOVE 'PROGRAM1' TO +LAST-PGM
0090 MOVE 'PROGRAM2' TO +NEXT-PGM
0100 FETCH +NEXT-PGM
0110 END
```

User Defined Variables

- A DEFINE DATA LOCAL statement is added to the object and all user defined variables used within the object are defined here.

- Format and length attributes will be removed from the in-line user defined variables.

- RESET statements are added immediately after the END-DEFINE statement for any in-line user defined variables that have been defined using the RESET statement. For example: RESET #FIELD(A10).

Example:

Before Conversion:

```
PROGRAM1
0010 RESET #NAME(A20) #ADDRESS(A50)
0020 REDEFINE #ADDRESS(#ADDRESS-1(A25) #ADDRESS-2(A25))
0030 MOVE 'SMITH' TO #NAME
0040 MOVE '101 THE AVENUE' TO #ADDRESS-1
0050 MOVE 'UNITED KINGDOM' TO #ADDRESS-2
0060 MOVE 1234567890        TO #TELEPHONE-NUMBER(N10)
0070 END
```

After Conversion:

```
PROGRAM1
::::
0050 DEFINE DATA LOCAL
0060 1 #NAME (A020)
0070 1 #ADDRESS (A050)
0080 1 REDEFINE #ADDRESS
0090 2 #ADDRESS-1 (A025)
0100 2 #ADDRESS-2 (A025)
0110 1 #TELEPHONE-NUMBER (N010)
0120 END-DEFINE
0130 RESET #NAME #ADDRESS
0140 MOVE 'SMITH' TO #NAME
0150 MOVE '101 THE AVENUE' TO #ADDRESS-1
0160 MOVE 'UNITED KINGDOM' TO #ADDRESS-2
0170 MOVE 1234567890 TO #TELEPHONE-NUMBER
0180 END
```

Database Fields

- A DEFINE DATA LOCAL statement is added to the object and view definitions are added here.

- A separate view definition will be created for each database access statement referencing the same database file.

- The view definitions will contain the definitions found for each database field referenced within the object.

Example:

Before Conversion:

```
PROGRAM1                              PROGRAM2
0010 READ EMPLOYEES                   0010 FIND VEHICLES WITH MAKE = 'FORD'
0020 DISPLAY PERSONNEL-ID NAME        0020   DISPLAY MAKE MODEL
0030 END                             0030 LOOP
                                     0040 FIND VEHICLES WITH MAKE = 'TVR'
                                     0050   DISPLAY MAKE MODEL
                                     0060 LOOP
                                     0070 END
```

After Conversion:

```
PROGRAM1                              PROGRAM2
::::                                 ::::
0050 DEFINE DATA LOCAL                0050 DEFINE DATA LOCAL
0060 1 EMPLOYEES VIEW OF EMPLOYEES    0060 1 VEHICLES VIEW OF VEHICLES
0070 2 PERSONNEL-ID                   0070 2 CAR-DETAILS
0080 2 FULL-NAME                      0080 3 MAKE
0090 3 NAME                           0090 3 MODEL
0100 /*                              0100 /*
0110 END-DEFINE                       0110 1 VEHICLES-1 VIEW OF VEHICLES
0120 READ-0010.                      0120 2 CAR-DETAILS
0130 READ EMPLOYEES                   0130 3 MAKE
0140 DISPLAY PERSONNEL-ID NAME        0140 3 MODEL
0150 END-READ                        0150 /*
0160 END                             0160 END-DEFINE
                                     0170 FIND-0010.
                                     0180 FIND VEHICLES WITH MAKE = 'FORD'
                                     0190 DISPLAY MAKE MODEL
                                     0200 END-FIND
                                     0210 FIND-0040.
                                     0220 FIND VEHICLES-1 WITH MAKE = 'TVR'
                                     0230 DISPLAY MAKE MODEL
                                     0240 END-FIND
                                     0250 END
```

Loop Constructs

- LOOP processing statements are replaced with corresponding END constructs.

- Applies to the following Natural statements:

Statement	Corresponding END Construct
CALL FILE	END-FILE
CALL LOOP	END-LOOP
FIND	END-FIND
FOR	END-FOR
HISTOGRAM	END-HISTOGRAM
READ	END-READ
READ WORK FILE	END-WORK
REPEAT	END-REPEAT
SORT	END-SORT

Example:

Before Conversion:

```
PROGRAM1
0010 READ EMPLOYEES
0020 DISPLAY PERSONNEL-ID NAME
0030 LOOP
::::
```

After Conversion:

```
PROGRAM1
::::
0120 READ-0010.
0130 READ EMPLOYEES
0140 DISPLAY PERSONNEL-ID NAME
0150 END-READ
::::
```

Conditional Logic Blocks

- Conditional statements will have the corresponding END constructs added.

- Any DO/DOEND statements will be removed.

- Applies to the following Natural statements:

Statement	Corresponding END Construct
IF	END-IF
IF NO RECORDS FOUND	END-NOREC
AT BREAK	END-BREAK
AT START OF DATA	END-START
AT END OF DATA	END-ENDDDATA
AT TOP OF PAGE	END-TOPPAGE
AT END OF PAGE	END-ENDPAGE
ON ERROR	END-ERROR

Example:

Before Conversion:

```
PROGRAM1
::::
0030 REPEAT
0040 ADD 1 TO #LOOP-CONTROL
0050 IF #LOOP-CONTROL LE 10
0060 DO
0070 WRITE 'STILL SOME PROCESSING TO DO'
0080 ESCAPE TOP
0090 DOEND
0100 ELSE
0110 DO
0120 WRITE 'PROCESSING COMPLETED'
0130 ESCAPE
0140 DOEND
0150 LOOP
::::
```

After Conversion:

<u>**PROGRAM1**</u>
```
::::
0100 REPEAT-0030.
0110 REPEAT
0120 ADD 1 TO #LOOP-CONTROL
0130 IF #LOOP-CONTROL LE 10
0140 WRITE 'STILL SOME PROCESSING TO DO'
0150 ESCAPE TOP
0160 ELSE
0170 WRITE 'PROCESSING COMPLETED'
0180 ESCAPE BOTTOM
0190 END-IF
0200 END-REPEAT
::::
```

Subroutine Blocks

- Internal and external subroutines will have the RETURN statement replaced by the corresponding END-SUBROUTINE construct.

Example:

<u>**Before Conversion:**</u>

<u>**PROGRAM1**</u>
```
::::
0060 PERFORM ##CALCULATE-COST
0070 /*
0080 WRITE 'TOTAL COST WITH TAX =' #TOTAL-COST
0090 /*
0100 DEFINE SUBROUTINE ##CALCULATE-COST
0110 RESET #TEMP-COST #TOTAL-COST
0120 COMPUTE #TEMP-COST = #COST * #TAX
0130 #TOTAL-COST = #COST + #TEMP-COST
0140 RETURN
0150 END
```

<u>**After Conversion:**</u>

PROGRAM1

```
::::
0160 PERFORM ##CALCULATE-COST
0170 /*
0180 WRITE 'TOTAL COST WITH TAX =' #TOTAL-COST
0190 /*
0200 DEFINE SUBROUTINE ##CALCULATE-COST
0210 RESET #TEMP-COST #TOTAL-COST
0220 COMPUTE #TEMP-COST = #COST * #TAX
0230 COMPUTE #TOTAL-COST := #COST + #TEMP-COST
0240 END-SUBROUTINE
0250 END
```

Open-ended ESCAPE Statements

- Any ESCAPE statements that do not specify a destination will have a destination of
  BOTTOM added.

Example:

<u>**Before Conversion:**</u>

<u>**PROGRAM1**</u>

```
::::
0030 REPEAT
0040 ADD 1 TO #LOOP-CONTROL
0050 IF #LOOP-CONTROL GE 10
0060 DO
0070 WRITE 'LOOP WILL NOW STOP'
0080 ESCAPE
0090 DOEND
0100 LOOP
::::
```

<u>**After Conversion:**</u>

<u>**PROGRAM1**</u>

```
::::
0100 REPEAT-0030.
0110 REPEAT
0120 ADD +1 TO #LOOP-CONTROL
0130 IF #LOOP-CONTROL GE 10
0140 WRITE 'LOOP WILL NOW STOP'
0150 ESCAPE BOTTOM
0160 END-IF
0170 END-REPEAT
::::
```

Short Form ASSIGN and COMPUTE Statements

- Short form ASSIGN and COMPUTE statements (i.e. statements with the ASSIGN/COMPUTE keyword omitted) will have the ASSIGN/COMPUTE keyword added.

Example:

Before Conversion:

```
PROGRAM1
::::
0030 #TAX = 0.0010
0040 ASSIGN #COST = 15000
::::
0120 COMPUTE #TEMP-COST = #COST * #TAX
0130 #TOTAL-COST = #COST + #TEMP-COST
::::
```

After Conversion:

```
PROGRAM1
::::
0130 ASSIGN #TAX = 0.0010
0140 ASSIGN #COST = 15000
::::
0220 COMPUTE #TEMP-COST = #COST * #TAX
0230 COMPUTE #TOTAL-COST = #COST + #TEMP-COST
::::
```

Multiple Statements per Statement Line

- Any single statement line containing more than one statement will be split so that each statement is on a separate line.

Example:

Before Conversion:

```
PROGRAM1
::::
0140 READ (10) PERSONNEL
0150 IF NAME EQ 'ADKINSON' DO WRITE PERSONNEL-NUMBER NAME DOEND
0160 LOOP
::::
```

After Conversion:

```
PROGRAM1
::::
0430 READ-0140.
0440 READ (10) PERSONNEL
0450 IF NAME EQ 'ADKINSON'
0460 WRITE PERSONNEL-NUMBER NAME
0470 END-IF
0480 END-READ
::::
```

MOVE INDEXED Statements

- MOVE INDEXED statements are replaced with standard MOVE statements.

- Suitable array definitions will be added for any MOVE INDEXED source variables that are not defined as arrays.

Example:

Before Conversion:

```
PROGRAM1
::::
0010 RESET #BLOCK-MULTIPLE(A26) #FIRST(A1) #LAST(A1)
0020 REDEFINE #BLOCK-MULTIPLE(#BLOCK-SINGLE(A1))
0030 RESET #CURRENCY-CODES(A3/4) #INDEX(I02)
0040 /*
0050 FORMAT LS=250
0060 /*
0070 MOVE 'ABCDEFGHIJKLMNOPQRSTUVWXYZ' TO #BLOCK-MULTIPLE
0080 MOVE INDEXED #BLOCK-SINGLE <1> TO #FIRST
0090 MOVE INDEXED #BLOCK-SINGLE <26> TO #LAST
0100 WRITE '=' #FIRST '=' #LAST
0110 /*
0120 FORMAT LS=250
0130 MOVE 3 TO #ISN
0140 FIND EMPLOYEES WITH PERSONNEL-ID EQ '11100106'
0150   OBTAIN CURR-CODE(1-4)
0160   WRITE '=' PERSONNEL-ID / '=' NAME
0170   FOR #INDEX EQ 1 TO 4
0180     MOVE INDEXED CURR-CODE<#INDEX> TO #CURRENCY-CODES(#INDEX)
0190     WRITE #INDEX '=' #CURRENCY-CODES(#INDEX)
0200   LOOP
0210 LOOP
0220 /*
0230 END
```

After Conversion:

PROGRAM1

```
::::
0050 DEFINE DATA LOCAL
0060 1 #BLOCK-MULTIPLE (A026)
0070 1 REDEFINE #BLOCK-MULTIPLE
0080 2 #BLOCK-SINGLE (A001)
0090 1 REDEFINE #BLOCK-MULTIPLE
0100 2 #NEE@MI-#BLOCK-SINGLE (A001/1:26)
0110 1 #FIRST (A001)
0120 1 #LAST (A001)
0130 1 #CURRENCY-CODES (A003/1:4)
0140 1 #INDEX (I002)
0150 1 #ISN (N007)
0160 1 EMPLOYEES VIEW OF EMPLOYEES
0170 2 PERSONNEL-ID
0180 2 FULL-NAME
0190 3 NAME
0200 2 INCOME
0210 3 CURR-CODE(0000001:0000004)
0220 /*
0230 END-DEFINE
0240 RESET #BLOCK-MULTIPLE #FIRST #LAST
0250 RESET #CURRENCY-CODES (1:4) #INDEX
0260 /*
0270 FORMAT LS=250
0280 /*
0290 MOVE 'ABCDEFGHIJKLMNOPQRSTUVWXYZ' TO #BLOCK-MULTIPLE
0300 MOVE #NEE@MI-#BLOCK-SINGLE (1) TO #FIRST
0310 MOVE #NEE@MI-#BLOCK-SINGLE (26) TO #LAST
0320 WRITE '=' #FIRST '=' #LAST
0330 /*
0340 FORMAT LS=250
0350 MOVE 3 TO #ISN
0360 FIND-0140.
0370 FIND EMPLOYEES WITH PERSONNEL-ID EQ '11100106'
0380 WRITE '=' PERSONNEL-ID / '=' NAME
0390 FOR-0170.
0400 FOR #INDEX EQ 1 TO 4
0410 MOVE CURR-CODE (#INDEX) TO #CURRENCY-CODES (#INDEX)
0420 WRITE #INDEX '=' #CURRENCY-CODES (#INDEX)
0430 END-FOR
0440 END-FIND
0450 /*
0460 END
```

Statement Reference Notation

- Statement labels are added to all loop processing statements and also to GET and STORE statements. The format used is:

  KEYWORD-NNNN

  where:

  KEYWORD   is the Natural keyword used in the statement. For example READ or FIND.

  NNNN   is the line number for the start of the statement in the original object.

- Any existing statement reference notation using source-code line numbers will be modified to use the new statement labels.

- Any existing statement labels will remain unchanged.

- Any database access view names will be modified to reflect the new view definitions.

- Any view name qualifiers on database fields will be modified to reflect the new view definitions.

Example:

Before Conversion:

PROGRAM1
```
0010 LIMIT 5
0020 READ EMPLOYEES
0030   DISPLAY PERSONNEL-ID NAME
0040 LOOP
0050 /*
0060 R2.
0070 READ EMPLOYEES
0080   DISPLAY PERSONNEL-ID(R2.) NAME(R2.)
0090 LOOP(R2.)
0100 /*
0110 READ EMPLOYEES
0120   DISPLAY PERSONNEL-ID(0110) NAME(0110)
0130 LOOP(0110)
0140 /*
0150 READ EMPLOYEES
0160   DISPLAY EMPLOYEES.PERSONNEL-ID EMPLOYEES.NAME
0170 LOOP
0180 /*
0190 END
```

After Conversion:

<u>**PROGRAM1**</u>

```
::::
0050 DEFINE DATA LOCAL
0060 1 EMPLOYEES VIEW OF EMPLOYEES
0070 2 PERSONNEL-ID
0080 2 FULL-NAME
0090 3 NAME
0100 /*
0110 1 EMPLOYEES-1 VIEW OF EMPLOYEES
0120 2 PERSONNEL-ID
0130 2 FULL-NAME
0140 3 NAME
0150 /*
0160 1 EMPLOYEES-2 VIEW OF EMPLOYEES
0170 2 PERSONNEL-ID
0180 2 FULL-NAME
0190 3 NAME
0200 /*
0210 1 EMPLOYEES-3 VIEW OF EMPLOYEES
0220 2 PERSONNEL-ID
0230 2 FULL-NAME
0240 3 NAME
0250 /*
0260 END-DEFINE
0270 LIMIT 5
0280 READ-0020.
0290 READ EMPLOYEES
0300 DISPLAY PERSONNEL-ID NAME
0310 END-READ
0320 /*
0330 R2.
0340 READ EMPLOYEES-1
0350 DISPLAY PERSONNEL-ID (R2.) NAME (R2.)
0360 END-READ
0370 /*
0380 READ-0110.
0390 READ EMPLOYEES-2
0400 DISPLAY PERSONNEL-ID(READ-0110.) NAME(READ-0110.)
0410 END-READ
0420 /*
0430 READ-0150.
0440 READ EMPLOYEES-3
0450 DISPLAY EMPLOYEES-3.PERSONNEL-ID EMPLOYEES-3.NAME
0460 END-READ
0470 /*
0480 END
```

OBTAIN Statements

- OBTAIN statements are removed and are replaced with index values specified in the view definition for each database field referenced in the old OBTAIN statements.

Example:

Before Conversion:

```
PROGRAM1
0010 FIND EMPLOYEES WITH PERSONNEL-ID EQ '88888888'
0020   OBTAIN CURR-CODE(1:4)
0030   OBTAIN BONUS(1:4,1:4)
0040   DISPLAY PERSONNEL-ID BONUS(*,*) CURR-CODE(*)
0050 LOOP
0060 END
```

After Conversion:

```
PROGRAM1
::::
0050 DEFINE DATA LOCAL
0060 1 EMPLOYEES VIEW OF EMPLOYEES
0070 2 PERSONNEL-ID
0080 2 INCOME
0090 3 CURR-CODE(0000001:0000004)
0100 3 BONUS(0000001:0000004,0000001:0000004)
0110 /*
0120 END-DEFINE
0130 FIND-0010.
0140 FIND EMPLOYEES WITH PERSONNEL-ID EQ '88888888'
0150 DISPLAY PERSONNEL-ID BONUS (*,*) CURR-CODE (*)
0160 END-FIND
0170 END
```

SORT Statements

- The END-ALL statement will be inserted prior to the SORT statement.

- A statement notation label will be added using the format SORT-NNNN, where NNNN is the statement line number for the SORT statement in the original object.

- Corresponding END-SORT added.

Example:

Before Conversion:

```
PROGRAM1
::::
0190 FIND EMPLOYEES WITH CITY = 'DERBY'
0200   OBTAIN SALARY(1:2)
0210 /*
0220   COMPUTE #TOTAL-SALARY (P11) = SALARY (1) + SALARY (2)
0230   ACCEPT IF #TOTAL-SALARY GT 0
0240 /*
0250   SORT BY PERSONNEL-ID USING #TOTAL-SALARY SALARY(*) CURR-CODE
0260      GIVE AVER(#TOTAL-SALARY)
0270 /*
0280    AT START OF DATA
0290    DO
0300     WRITE NOTITLE '*' (40)
0310       'AVERAGE CUMULATIVE SALARY:' *AVER(#TOTAL-SALARY)
0320     MOVE *AVER (#TOTAL-SALARY) TO #AVERAGE (P11)
0330    DOEND
0340 /*
0350    COMPUTE #AVERAGE-PERCENT (N3.2) = #TOTAL-SALARY / #AVERAGE * 100
0360    ADD #TOTAL-SALARY TO #TOTAL-TOTAL (P11)
0370 /*
0380    DISPLAY NOTITLE PERSONNEL-ID SALARY (1) SALARY (2)
0390      #TOTAL-SALARY CURR-CODE (1)
0400      'PERCENT/OF/AVER' #AVERAGE-PERCENT
0410 /*
0420    AT END OF DATA
0430    WRITE / '*' (40) 'TOTAL SALARIES PAID: ' #TOTAL-TOTAL
0440 /*
0450 END
```

After Conversion:

<u>**PROGRAM1**</u>

```
::::
0370 FIND-0190.
0380 FIND EMPLOYEES WITH CITY = 'DERBY'
0390 /*
0400 COMPUTE #TOTAL-SALARY = SALARY (1) + SALARY (2)
0410 ACCEPT
0420 IF #TOTAL-SALARY GT 0
0430 /*
0440 END-ALL
0450 SORT-0250.
0460 SORT BY PERSONNEL-ID USING #TOTAL-SALARY SALARY (*) CURR-CODE (1:1)
0470 GIVE AVER (#TOTAL-SALARY)
0480 /*
0490 AT START OF DATA
0500 WRITE NOTITLE '*' (40)
0510 'AVERAGE CUMULATIVE SALARY:' *AVER (#TOTAL-SALARY)
0520 MOVE *AVER (#TOTAL-SALARY) TO #AVERAGE
0530 END-START
0540 /*
0550 COMPUTE #AVERAGE-PERCENT = #TOTAL-SALARY / #AVERAGE * 100
0560 ADD #TOTAL-SALARY TO #TOTAL-TOTAL
0570 /*
0580 DISPLAY NOTITLE PERSONNEL-ID SALARY (1) SALARY (2)
0590 #TOTAL-SALARY CURR-CODE (1)
0600 'PERCENT/OF/AVER' #AVERAGE-PERCENT
0610 /*
0620 AT END OF DATA
0630 WRITE / '*' (40) 'TOTAL SALARIES PAID: ' #TOTAL-TOTAL
0640 /*
0650 END-ENDDATA
0660 END-SORT
0670 END
```

FIND FIRST/NUMBER/UNIQUE Statements

- FIND FIRST statements are converted to FIND (1) statements and a GET statement is added to reference any data. FIND FIRST is not valid in Structured mode.

- FIND NUMBER statements using the WHERE clause, are converted to FIND (1) statements. The WHERE clause for FIND NUMBER is not valid in Structured mode. If the WHERE clause is not present, then the FIND NUMBER statement is retained.

- FIND UNIQUE statements are converted to FIND (1) statements and a GET statement is added to reference any data. FIND UNIQUE is not valid in Structured mode.

Example:

Before Conversion:

```
PROGRAM1
::::
0040 FIND FIRST EMPLOYEES WITH NAME = 'ADKINSON'
0050   WHERE SEX EQ 'F'
0060 WRITE '=' (70)
0070 WRITE 'THE NUMBER OF ADKINSON''S ON FILE =' *NUMBER
0080 WRITE 'THE NUMBER OF FEMALE ADKINSON''S  =' *COUNTER
0090 WRITE 'THE FIRST RECORD LOCATED IS:'
0100 / 5T'P-id:' PERSONNEL-ID (CD=RE)
0110 / 5T 'Name:' NAME (CD=RE)
0120 / 5T 'Sex :' SEX (CD=RE)
::::
0160 FIND NUMBER VEHICLES WITH MAKE = 'BMW'
0170   WHERE COLOR EQ 'BLACK'
0180 WRITE 'THE NUMBER OF BMW''S FOUND =' *NUMBER
0190 WRITE 'THE NUMBER OF BLACK BMW''S =' *COUNTER
::::
0230 FIND NUMBER VEHICLES WITH MAKE = 'FORD'
0240 WRITE 'THE NUMBER OF FORD''S FOUND =' *NUMBER
::::
0280 FIND UNIQUE PERSONNEL WITH NAME = 'BAYER'
0290 WRITE 'PERSONNEL INFORMATION FOUND :'
0300 / 'Last  Name:' NAME (CD=GR)
0310 / 'First Name:' FIRST-NAME (CD=GR)
0320 / 'Job Title :' JOB (CD=GR)
::::
```

After Conversion:

PROGRAM1

```
::::
0280 FIND-0040.
0290 FIND (1) EMPLOYEES WITH NAME = 'ADKINSON'
0300 WHERE SEX EQ 'F'
0310 END-FIND
0320 GET-FIND-0040.
0330 GET EMPLOYEES *ISN ( FIND-0040. )
0340 WRITE '=' (70)
0350 WRITE 'THE NUMBER OF ADKINSON''S ON FILE =' *NUMBER (FIND-0040.)
0360 WRITE 'THE NUMBER OF FEMALE ADKINSON''S  =' *COUNTER (FIND-0040.)
0370 WRITE 'THE FIRST RECORD LOCATED IS:'
0380 / 5T 'P-id:' PERSONNEL-ID (CD=RE)
0390 / 5T 'Name:' NAME (CD=RE)
0400 / 5T 'Sex :' SEX (CD=RE)
::::
0440 FIND-0160.
0450 FIND (1) VEHICLES WITH MAKE = 'BMW'
0460 WHERE COLOR EQ 'BLACK'
0470 END-FIND
0480 WRITE 'THE NUMBER OF BMW''S FOUND =' *NUMBER (FIND-0160.)
0490 WRITE 'THE NUMBER OF BLACK BMW''S =' *COUNTER (FIND-0160.)
::::
0530 FIND-0230.
0540 FIND NUMBER VEHICLES-1 WITH MAKE = 'FORD'
0550 WRITE 'THE NUMBER OF FORD''S FOUND =' *NUMBER (FIND-0230.)
::::
0590 FIND-0280.
0600 FIND (1) PERSONNEL WITH NAME = 'BAYER'
0610 END-FIND
0620 GET-FIND-0280.
0630 GET PERSONNEL *ISN (FIND-0280.)
0640 WRITE 'PERSONNEL INFORMATION FOUND :'
0650 / 'Last  Name:' NAME (CD=GR)
0660 / 'First Name:' FIRST-NAME (CD=GR)
0670 / 'Job Title :' JOB (CD=GR)
::::
```

Mode Conversion GDA Processing

The first stage of the Mode Conversion process is to address the Global data requirements within the application.

GDA Options Overview

When the Mode Conversion option is invoked, the Mode Conversion GDA Information window is displayed showing the GDA usage for the current application and the default GDA Options that will be set on the Mode Conversion GDA Options screen.

For each application being converted, there are three possible GDA options available:

1. Use new GDA

This will generate a new GDA object which will contain all the Global data used within the application. This may include all Global data from one or more existing GDA objects already used by the application and/or any in-line Global data definitions found within individual objects. The Mode Conversion process will use the new GDA object name as part of the 'DEFINE DATA GLOBAL USING' statement within each newly converted object that references Global data.

It is recommended that this option is used to ensure all Global data used by the application, is encapsulated within one GDA object. For applications using multiple GDA objects, this option will rationalize the GDA objects into one single GDA object.

2. Use converted GDA

This will use a previously generated GDA object, from a previous Mode Conversion execution using option 1 above. The Mode Conversion process will use the previously converted GDA object name as part of the 'DEFINE DATA GLOBAL USING' statement within each newly converted object that references Global data.

This option may be used if an application is being converted in phases using object ranges, or when additional single objects are being converted.

Note: This option is only available if the Mode Conversion process has been previously executed using the 'Use new GDA' option and the modification library contains the GDA object specified.

3. Use existing GDA

This will use the existing application GDA objects and the Mode Conversion process will not change the 'DEFINE DATA GLOBAL USING' statements within each newly converted object that references the GDA data.

If the application uses any in-line Global data definitions in addition to GDA objects, then the Mode Conversion In-line Global Variables option is invoked. This option allows you to specify the GDA object name, which will be generated to encapsulate all in-line Global data definitions used by the application. The Mode Conversion process will then add the 'DEFINE DATA GLOBAL USING' statements within each newly converted object that only references in-line Global data definitions.

Note: The Mode Conversion In-line Global Variables option is mandatory for any application that uses a mix of GDA objects and in-line Global data definitions when using the 'Use existing GDA' option. This is to ensure that the application Global data usage integrity is maintained after conversion.

GDA Processing Considerations

1. Duplicate Global data definitions.

Any Global data definitions that have the same name, format and length will be rationalized as one variable in the generated GDA.

Any Global data definitions that have the same name but different format and/or length will not be added to the generated GDA object. These will be reported in the Mode Conversion Log. For example:

```
Duplicate Global variable name with different format detected in Object: G02
  - Global variable: +DUPLICATE-NAME-DIFF-FORMAT (A020)
  - Global variable already included from Object: G01
```

2. Global data definitions used by subprograms.

Any Global data definitions found within GDA objects only used by subprograms will not be added to the generated GDA object. These will be reported in the Mode Conversion Log. For example:

```
GDA object only used by Subprograms detected - Object: SPGMGDA1
  - This GDA is not included in the new GDA
```

Any in-line Global data definitions found within subprograms will not be added to the generated GDA object. These will be reported in the Mode Conversion Log. For example:

```
In-line Global variable only used by Subprograms detected - Object: SUBPGM1
  - Global variable: +SUBPGM-INLINE-GLOBAL
  - This Global variable is not included in the new GDA
```

Note: Any subprograms containing in-line Global data definitions will not be converted. These will have to be manually rationalized before conversion is allowed. Subprograms using GDA objects will be converted.

3. Applications using multiple GDA objects.

Applications using multiple GDA objects can be addressed in any one of three ways:

1. Manually rationalize the multiple GDA objects into a new single GDA object. Any in-line Global data definitions would need to be identified and added to the new single GDA object. The application objects referencing Global data need to be changed to reference the new single GDA. Once this has been completed, the application needs to be extracted and loaded into the Repository, then the Mode Conversion process executed using the 'Use new GDA' option.

Note: This is the recommended method of dealing with multiple GDA objects as it will ensure the stability of the Global data usage within the application.

2. Automatically rationalize the multiple GDA objects into a new single GDA object using the Mode Conversion 'Use new GDA' option. This will generate a new single GDA object based on the GDA objects and any in-line Global data definitions found within the application.

3. Use the existing GDA objects. The existing GDA object usage within the application will be retained and any in-line Global data definitions will be encapsulated within a new single GDA object.

Mode Conversion GDA Information Screen

The Mode Conversion GDA Information screen displays the Global data usage information for the application along with the default options that will be displayed on the Mode Conversion GDA Options screen.

The Mode Conversion GDA Information screen will be displayed after selecting option '**C**' (Mode Conversion) from the Utilities Menu screen.

The following Figure 3-1 illustrates an example of the Mode Conversion GDA Information screen.

```
                      - Utilities Menu -        Application: MODECONV

                            Information

     Your application has no Global Data Area in use




     Your application uses in-line global variables

     Default GDA name set to NEERSGDA
     Default option 'Use new GDA' has been selected

    Press ENTER to continue



Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
     Help      Exit                                                  Main
```

Figure 3-1 Example of the Mode Conversion GDA Information screen

The information displayed will be based on the following cases:

1. Application uses no Global data.

The application uses no Global data, either as GDA objects or in-line Global variable definitions.

SCREEN ITEMS	DESCRIPTION
GDA Information	For all GDA process executions:
	"Your application has no Global variables"
Default Options	Not applicable for this case.
ENTER key	Invoke the Mode Conversion Object Selection screen.

2. Application uses in-line Global variables only.

The application uses Global variables that have been defined in-line. The application does not use any GDA objects.

SCREEN ITEMS	DESCRIPTION
GDA Information	For all GDA process executions:
	"Your application has no Global Data Area in use"
	"Your application uses in-line global variables"
Default Options	For first time GDA process execution:
	"Default GDA name set to NEERSGDA"
	"Default option 'Use new GDA' has been selected"
	For subsequent GDA process execution:
	"Default option 'Use converted GDA' has been selected"
ENTER key	Invoke the Mode Conversion GDA Options screen.

3. Application uses a single GDA object only.

The application uses Global variables that have been defined in a single GDA object. The application does not use any Global variables that have been defined in-line.

SCREEN ITEMS	DESCRIPTION
GDA Information	For all GDA process executions:
	"Your application has one Global Data Area in use"
	- 'GDA-name'
Default Options	For all GDA process executions:
	"Default option 'Use existing GDA' has been selected"
ENTER key	Invoke the Mode Conversion GDA Options screen.

Note: 'GDA-name' will be the GDA object name within the application.

4. Application uses a single GDA object and in-line Global variables.

The application uses Global variables that have been defined using a single GDA object and in-line definitions.

SCREEN ITEMS	DESCRIPTION
GDA Information	For all GDA process executions:
	"Your application has one Global Data Area in use"
	- 'GDA-name'
	"Your application uses in-line global variables"
Default Options	For first time GDA process execution:
	"Default GDA name set to 'GDA-name'"
	"Default option 'Use new GDA' has been selected"
	For subsequent GDA process execution:
	"Default option 'Use converted GDA' has been selected"
ENTER key	Invoke the Mode Conversion GDA Options screen.

Note: 'GDA-name' will be the GDA object name within the application.

5. Application uses multiple GDA objects and no in-line Global variables.

The application uses Global variables that have been defined in more than one GDA object. The application does not use any Global variables that have been defined in-line.

SCREEN ITEMS	DESCRIPTION
GDA Information	For all GDA process executions:
	"Your application has more than one Global Data Area in use"
	- 'GDA-name' - 'GDA-name'
	Note: A maximum of four GDA object names will be displayed. If there are more than four GDA objects are used, then the text '<MORE>' will be shown. For example:
	- 'GDA-name' *- 'GDA-name'* *- 'GDA-name'* *- 'GDA-name'* *- <MORE>*
Default Options	For all GDA process executions:
	"Default option 'Use existing GDAs' has been selected"
ENTER key	Invoke the Mode Conversion GDA Options screen.

Note: 'GDA-name' will be the GDA object name within the application.

6. Application uses a mix of multiple GDA objects and in-line Global variables.

The application uses Global variables that have been defined using both GDA objects and in-line definitions.

SCREEN ITEMS	DESCRIPTION
GDA Information	For all GDA process executions:
	"Your application has more than one Global Data Area in use"
	- 'GDA-name' - 'GDA-name'
	"Your application uses in-line global variables"
	Note: A maximum of four GDA object names will be displayed. If there are more than four GDA objects are used, then the text '<MORE>' will be shown. For example:
	- 'GDA-name' *- 'GDA-name'* *- 'GDA-name'* *- 'GDA-name'*
	- <MORE>
Default Options	For all GDA process executions:
	"Default option 'Use existing GDAs' has been selected"
ENTER key	Invoke the Mode Conversion GDA Options screen.

Note: 'GDA-name' will be the GDA object name within the application.

Mode Conversion GDA Options Screen

The Mode Conversion GDA Options screen allows you to specify what GDA processing is to be applied to the converted application.

Note: If the application uses no Global data (either as GDA objects or in-line Global variable definitions) then the Mode Conversion GDA Options screen will not be displayed.

The Mode Conversion GDA Options screen is invoked by using the '**ENTER**' key on the Mode Conversion GDA Information screen.

The following Figure 3-2 illustrates the Mode Conversion GDA Options screen.

```
                      - Utilities Menu -         Application: MODECONV

                         Mode Conversion - GDA Options


      Y  Use new GDA             _____
      _  Use converted GDA     (NEERSGDA)
      _  Use existing GDA




      Press ENTER to continue
            PF3   to exit



 Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit                                              Main
```

Figure 3-2 Mode Conversion GDA Options screen

SCREEN ITEMS	DESCRIPTION
Use new GDA	Specify this option to create a new GDA object to be used by the converted application.
GDA Name	Specify the GDA object name to be used. The GDA name is mandatory if the 'Use new GDA' option has been selected.
	The GDA name will normally appear 'blank' except under the following conditions:
	1. Will be set to NEERSGDA if the application only uses in-line Global definitions and has no GDA objects.
	2. Will be set to the 'GDA-name' used by the application if the application uses in-line Global definitions and a single GDA object.
	Note: These default values may be overwritten as required.
Use converted GDA	Specify this option if you wish to use a previously converted GDA object.
	The name of the previously converted GDA object will appear enclosed in brackets. For example [NEERSGDA].
	Note: This option is only available if the Mode Conversion process has been previously executed using the 'Use new GDA' option and the modification library contains the GDA object specified.
Use existing GDA	Specify this option if you wish to use the application's existing GDA objects.
	Note: If the application uses any in-line Global definitions, then the Mode Conversion In-line Global Variables screen will be displayed.

PFKEYS	DESCRIPTION
ENTER	Validates the selection made and invokes one of the following processes depending on the option selected:
	1. 'Use new GDA'
	Mode Conversion Object Selection screen displayed.
	2. 'Use converted GDA'
	Mode Conversion Object Selection screen displayed.
	3. 'Use existing GDA'
	If the application uses no in-line Global definitions then the Mode Conversion Object Selection screen will be displayed.
	If the application does use in-line Global definitions, then the Mode Conversion In-line Global Variables screen will be displayed.
PF3	Exit from the current function and return to previous screen.

Mode Conversion In-line Global Variables Screen

The Mode Conversion In-line Global Variables screen allows you to specify the GDA object name to be used for any in-line Global variable definitions found within the application.

This option is mandatory if the application uses both GDA objects and in-line Global variable definitions.

Note: The Mode Conversion In-line Global Variables screen is only displayed if the 'Use existing GDA' option has been selected on the Mode Conversion GDA Options screen and the application contains in-line Global variable definitions.

The Mode Conversion In-line Global Variables screen is invoked by selecting option 'Use existing GDA' on the Mode Conversion GDA Options screen and using the '**ENTER**' key.

The following Figure 3-3 illustrates the Mode Conversion In-line Global Variables screen.

```
                    - Utilities Menu -        Application: MODECONV

            Mode Conversion - In-line Global Variables

     You have selected to use existing GDA(s) but the application
     also uses in-line Global variables. These need to be
     rationalized into a new single GDA object as the conversion
     process will remove any in-line format and length attributes.

     The new GDA will be referenced by objects using in-line
     Global variables only.

                  GDA name      NEERSGDA


      Press ENTER to continue
            PF3    to exit


 Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Exit                                                    Main
```

Figure 3-3 Mode Conversion In-line Global Variables screen

SCREEN ITEMS	DESCRIPTION
Explanation text	Provides an explanation for the Mode Conversion In-line Global Variables option.
GDA name	Specify the GDA object name to be used. The GDA name is mandatory and the GDA object name must not already exist on the modification library. The default GDA name will be set to 'NEERSGDA'. This may be overwritten as required.

PFKEYS	DESCRIPTION
ENTER	Validates the selection made and invokes the Mode Conversion Object Selection screen. *Note: If the GDA name specified already exists on the modification library an error message will be displayed. The GDA name will need to be changed before you can continue.*
PF3	Exit from the current function and return to previous screen.

Mode Conversion Object Processing

The second stage of the Mode Conversion process is to select the objects to be converted and then invoke the Mode Conversion Object processing.

Mode Conversion Object Selection Screen

The Mode Conversion Object Selection screen allows you to select the objects that are to be converted for the application.

The Mode Conversion Object Selection screen is invoked by using the '**ENTER**' key on the Mode Conversion GDA Options screen.

The following Figure 3-4 illustrates the Mode Conversion Object Selection screen.

```
                        - Mode Conversion -       Application: MODECONV
                        - Object Selection -
                         Select     Object Name
                                      MCONVN07
                           _          MCONVP01
                           _          MCONVP02
                           _          MCONVP03
                           _          MCONVP04
                           _          MCONVP05
                           _          MCONVP06
                           _          MCONVP07
                           _          MCONVP08
                           _          MCONVP09
                           _          MCONVP10
                           _          MCONVP11
                           _          MCONVP12
                           _          MCONVP13
                           _
                          Reposition ->  _____
Object Types: CHPNS   Object Selection -> *_____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Types Exit        Sub        Prev  Next                     Main
```

Figure 3-4 Mode Conversion Object Selection screen

SCREEN ITEMS	DESCRIPTION
Select	This is the selection column where individual objects can be selected. Valid selections are:
	'S' Select object.
Object Name	List of Reporting mode objects loaded in the Repository for the current application. No Structured mode objects will be listed. This list can be tailored to your requirements using '**PF2**' (Types) option.
Reposition	Reposition the list of objects starting from the new value entered. The reposition value can be input using either a complete object name or part name using an '*****' (asterisk) wildcard. For example:
	* Will reposition at the start of the object list.
	MCONV* Will reposition at the first object name that matches the mask MCONV or is greater than the mask input.
	MCONVP01 Will reposition at the first object name that matches the mask exactly or is greater than the object name input.
Object Selection	The selected object or range of objects to be converted.
	Note: Object Selection is mandatory except when using the 'Use new GDA' option, where the Object Selection can be 'blank' and will result in the Mode Conversion process generating a new GDA object only.
	There are three valid cases allowed:
	* Will apply the Mode Conversion process for ALL objects in the current application.
	Note: This is the default setting.
	MCONV* Will apply the Mode Conversion process for ALL objects that have a name prefixed with MCONV.
	For example: convert objects MCONVP01 and MCONVN01 but not MCONXP01.
	MCONVP01 Will apply the Mode Conversion process for object MCONVP01 only.
Object Types	Displays the object types to be referenced. Object types can be selected/de-selected by using '**PF2**' (Types). Available selections are:
	'C' Copycodes
	'H' Helproutines
	'P' Programs
	'N' Subprograms
	'S' Subroutines

PFKEYS	DESCRIPTION
PF1	Activates the help function.
PF2	Invoke the Object Type Selection pop-up window.
	Available selections are:
	Object Types
	'C' Copycode
	'H' Helproutines
	'P' Programs
	'N' Subprograms
	'S' Subroutines
PF3	Exit from the current function and return to previous screen.
PF5	Invoke the Mode Conversion Information window.
PF7	Displays previous page.
PF8	Displays next page.
PF12	Returns to the Natural Engineer Main Menu.

Mode Conversion Information Screen

The Mode Conversion Information screen summarizes the GDA options to be applied during the conversion process. It is possible at this stage to cancel the conversion process and make new object selection and/or change the GDA options to be used.

Note: If the application uses no Global data (either as GDA objects or in-line Global variable definitions) then the Mode Conversion Information screen will not be displayed.

The following Figure 3-5 illustrates an example of the Mode Conversion Information screen.

```
                          - Mode Conversion -        Application: MODECONV
                          - Object Selection -
                                Information

       You are about to execute Mode Conversion with the following
       GDA option:

            - Use new GDA (NEERSGDA)




       Do you wish to continue ?  Y  (Y/N)


Object Types: CHPNS   Object Selection -> *_____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Types Exit        Sub         Prev  Next                    Main
```

Figure 3-5 Example of the Mode Conversion Information screen

SCREEN ITEMS	DESCRIPTION
You are about to execute Mode Conversion with the following GDA option:	List the GDA option that will be used during the conversion process. The GDA options available are: **- Use new GDA ['GDA-name']** Conversion will use the new GDA 'GDA-name' when converting objects. **- Use converted GDA ['GDA-name']** Conversion will use the previously converted GDA 'GDA-name' when converting objects. **- Use existing GDA** Conversion will use the existing application GDA when converting objects. *Note: 'GDA-name' will contain the name of the GDA that has been specified on the Mode Conversion GDA Options screen.*
Do you wish to continue ?	**'Y'** Invoke NATRJE Job Submission screen. **'N'** Cancel the Mode Conversion process and return back to the Mode Conversion Object Selection screen.

The following Figure 3-6 illustrates the NATRJE submission screen for the Mode Conversion option.

```
                        - Job Submission -       Application: MODECONV

           Job Selection details
           --------------------
               Job Selected  : (RMRMOD) MODE CONVERSION

           Job Card details
           ----------------
                   Job Name  : XGSLXX__
                   Job Class : _

           Job Control Record details
           ------------------------
           Control Status  :
           Last Job Submitted
              Job Name :
                  Opid :
                  Step :
           Return Code :

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Sub   Ref                 Rel               Main
```

Figure 3-6 NATRJE Job Submission screen

Note: For more information on the NATRJE Job Submission screen refer to the Natural Engineer Batch Processing (Unix) manual.

Mode Conversion Log

The Mode Conversion Log is displayed in the batch job output file. The Mode conversion log will contain a list of entries showing the conversion process activity.

The following Figure 3-7 illustrates the Mode Conversion Log showing conversion details.

```
**************************************************************************
*
*                Mode Conversion Log                    11 Jun 2003 11:57
*
*     Library Information
*       Application        : MODECONV
*       Modification library : MODECNVX
*
*     GDA Options
*       Create new GDA     : Yes
*       GDA name           : NEERSGDA
*
*     Object Selection
*       Convert objects    : Yes
*       Object selection   : *            (All Objects)
*       Types              : CHPNS
*
**************************************************************************
*
GDA NEERSGDA created in MODECNVX
MCONVN07 converted to Structured mode
MCONVP01 converted to Structured mode
MCONVP02 converted to Structured mode
MCONVP03 converted to Structured mode
MCONVP04 converted to Structured mode
MCONVP05 converted to Structured mode
MCONVP06 converted to Structured mode
MCONVP07 converted to Structured mode
MCONVP08 converted to Structured mode
MCONVP09 converted to Structured mode
MCONVP10 converted to Structured mode
MCONVP11 converted to Structured mode
MCONVP12 converted to Structured mode
MCONVP13 converted to Structured mode
```

Figure 3-7 Mode Conversion Log showing conversion details

REPORT ITEM	DESCRIPTION
Application	The name of the application being processed.
Modification library	The name of the modification library where the converted objects will reside.
Create new GDA	Indicates if a new GDA will be generated. Possible values are:
	Yes A new GDA object will be generated.
	No No new GDA object will be generated.
GDA name	The name of the generated GDA object.
	Note: This will be blank if not using option 'Use new GDA'.
Convert objects	Indicates if any objects have been selected for conversion. Possible values are:
	Yes Objects have been selected for conversion.
	No No objects have been selected for conversion. Only a new GDA object will be generated.
	Note: This situation is only available for the 'Use new GDA' option.
Object selection	Indicates the object selection that has been used for the conversion. Possible values are:

'blank'	**(No Objects)**	No objects have been selected for conversion.
*	**(All Objects)**	All objects in the application will be converted.
MCONV*	**(Object Range)**	A range of objects have been selected for conversion. For example all objects prefixed with MCONV will be converted.
MCONVP01	**(Single Object)**	Only one object selected for conversion. For example MCONVP01.

REPORT ITEM	DESCRIPTION
Object Types	Indicates the object types that have been selected for conversion. Possible values are:
	'C' Copycodes 'H' Helproutines 'P' Programs 'N' Subprograms 'S' Subroutines
Details	A separate entry for each object and/or process action will be displayed here. Example entries: **GDA NEERSGDA created in MODECNVX** **PGM1 converted to Structured mode** **SUBPGM1 cannot be converted. SubProgram contains in-line globals** *Note: Entries for duplicate Global data definitions and Global data definitions used by subprograms are also displayed. For examples of these refer to section GDA Processing Considerations.*

3

CHANGE MANAGEMENT TRACKING (CMT)

Chapter Overview

This chapter describes the Change Management Tracking (CMT) option available from the Utilities menu. The CMT option provides the facility to track changes that have been applied to objects within an application using Natural Engineer.

The following topics are covered:

1. Change Management Tracking Overview

2. Change Management Tracking Reports Selection Screen

3. Change Management Tracking Example

Change Management Tracking Overview

The CMT option provides audit trail data per object within an application that has used Natural Engineer to administer maintenance changes.

Any updates applied by Natural Engineer to an application's objects generate audit trail records for the application on the Repository. These audit trail records contain the before and after images of updated code, a date and time stamp of when the updates were made and the User Id of the person making the update.

The audit trail records are accumulated per object, and are only deleted when the application is deleted from the Repository. This allows for complete tracking of updates for an application during its maintenance life cycle within Natural Engineer.

The audit trail records can be viewed as hardcopy reports from the batch job output file.

CMT Reports Selection Screen

The CMT Reports Selection screen allows you to review audit trail records for individual or a range of objects within an application using hardcopy reports by submitting a batch job.

The CMT Reports Selection screen is accessed by selecting option '**M**' (Change Management Tracking) from the Utilities Menu screen.

The following Figure 2-1 illustrates the CMT Reports Selection screen.

```
                  Change Management Tracking   Application: HOSPITAL
                     Reports Selection              Version: 01


  Object:          _____ to _____
  or All Objects: N

  All Versions:   N

  Date Operator:  __  Date: __ __ ____ (dd mm yyyy)
                        To: __ __ ____ (dd mm yyyy)

  User Id:          _____

  Sort Order:     1  Timestamp - Asc

  Object Lang.:   * All          (PF2 to set)

  Object Types:   4C3GHLMAPNS    (PF2 to set)

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
     Help  Types Exit        Sub   Vers                               Main
```

Figure 2-1 CMT Reports Selection screen

SCREEN ITEMS	DESCRIPTION
Object	Allows you to select an individual or range of objects to report on. For Example: if XX021P01 is entered, then only audit trail records for XX021P01 will be reported.
All Objects	If selected, then all the objects with audit trail records within the current application will be reported. Valid selections are: '**N**' Single object or range of objects. '**Y**' All objects
All Versions	If selected, then all the Impact Versions of audit trail records for the selected object will be reported. Valid selections are: '**N**' Reports on current selected Impact Version only. '**Y**' Reports on all Impact Versions.
Date	Date ranges, to limit the audit trail records reported to the selected date range values.

	Date Operator	The operator used to qualify the date range specified. Valid operators are: **GT – greater than.** **LT – less than.** **EQ – equal to**.
	From Date	Start from date using format DDMMYYYY.
	To Date	End at date using format DDMMYYYY.

SCREEN ITEMS	DESCRIPTION
User Id	User Id of the person responsible for the changes, to limit the audit trail records reported to the selected user. This input is case dependent. For example: 'xx209' will only report a User Id of 'xx209' and not 'XX209'.
Sort Order	The Sort order that the audit trail records will be displayed in the report. Available selections are: '**1**' Timestamp – Ascending. '**2**' Timestamp – Descending. '**3**' User Id – Ascending. '**4**' User Id – Descending. '**5**' Line Number – Ascending. '**6**' Line Number – Descending. *Note: Field level help is available and is invoked by typing '**?**' in the field. A pop-up window is displayed with all the available Sort Orders.*

SCREEN ITEMS	DESCRIPTION
Object Lang.	Allows you to select the programming language of the objects, to limit the audit trail records reported to the selected programming language. Selection is made by using '**PF2**' (Types). Available selections are:

'∗' All

'**C**' Cobol

'**N**' Natural

Object Types	Displays the object types to be referenced. Object types can be selected/de-selected by using '**PF2**' (Types). Available selections are:

'**4**' Classes

'**C**' Copycode

'**3**' Dialogs

'**G**' Global Data Areas

'**H**' Helproutines

'**L**' Local Data Areas

'**M**' Maps

'**A**' Parameter Data Areas

'**P**' Programs

'**N**' Subprograms

'**S**' Subroutines

PFKEYS	DESCRIPTION
PF1	Activates the help function.
PF2	Invoke the Object Type Selection pop-up window.
	Available selections are:
	Object Language
	'C' COBOL
	'N' Natural
	Object Types
	'4' Classes
	'C' Copycode
	'3' Dialogs
	'G' Global Data Areas
	'H' Helproutines
	'L' Local Data Areas
	'M' Maps
	'A' Parameter Data Areas
	'P' Programs
	'N' Subprograms
	'S' Subroutines
PF3	Exit from the current function and return to previous screen.
PF5	Invoke NATRJE Job Submission screen.
PF6	Invoke the Impact Version screen.
PF12	Returns to the Natural Engineer Main Menu.

After all the CMT report criteria have been specified, use '**PF5**' (Sub) to submit the batch job via the NATRJE Job Submission screen.

The following Figure 2-2 illustrates the NATRJE submission screen for the CMT option.

```
                     - Job Submission -        Application: HOSPITAL

          Job Selection details
          --------------------
               Job Selected  : (CMTREP) CMTS REPORT

          Job Card details
          ----------------
                  Job Name  :  XGSLXX__
                  Job Class :  _

          Job Control Record details
          ------------------------
          Control Status  :
          Last Job Submitted
            Job Name :
                Opid :
                Step :
          Return Code :


Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Sub  Ref                   Rel               Main
```

Figure 2-2 NATRJE Job Submission screen

Note: For more information on the NATRJE Job Submission screen refer to the Natural Engineer Batch Processing (Unix) manual.

CMT Example

To illustrate the Change Management Tracking process, an example is shown using the sample Natural application HOSPITAL.

This example will demonstrate the CMT Reports Selection option to produce a CMT report showing a simple set of audit trail records for objects that have been modified using the sample application HOSPITAL.

All the objects from the HOSPITAL application have been extracted and loaded into the Repository and the steps in this example start from the Impact Analysis process.

Step 1 Version 1 impact search criteria have been specified to search for keyword DATAITEM with search values of #G-MESSAGE, #L-MESSAGE and #M-MESSAGE. Replace values for each of these criteria have been specified as #G-MSG, #L-MSG and #M-MSG respectively.

The following Figure 2-3 illustrates the Impact Criteria Summary screen.

```
                      Impact Criteria Summary   Application: HOSPITAL
                                                     Version: 01
     Search           Keyword          Search        Replace         Replace
     Keyword          Value            Value          Value           TLM
  _  DATAITEM                          #G-MESSAGE     #G-MSG
  _  DATAITEM                          #L-MESSAGE     #L-MSG
  _  DATAITEM                          #M-MESSAGE     #M-MSG




















Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  DeleC Exit  GetSa SaveA Add   Prev  Next        W<    W>    Main
```

Figure 2-3 Impact Criteria Summary screen

Step 2 After Impact Analysis has been executed; modification has been applied to all the impacted objects.

The following Figure 2-4 illustrates the Modification Element Categorization screen after all the objects have been modified. Object XX000G00 has been selected and also data item #G-MESSAGE, just to show the modification parameters that have been applied.

```
                     - Element Categorization -    Application: HOSPITAL
                                                       Version: 01
Object: XX000G00                                         Page:   1
Field : #G-MESSAGE
Attr  : A070   External Object Name:
Category     : A   Automatic         Replace Defn: _____  TLM: _____
Type         : I_  Data Item                              Pos: _____
Replace Value: #G-MSG_____
User Comment : _____
TLM Data     : _____
Reason : Data item can be automatically changed
User ID:          Last Update    :
                  Execution Date : 28 Sep 2001 at 17:00:39
Stmt Source Line
0020  1 #G-MESSAGE                      A   70




Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
     Help         Exit        Updt  SCrit Prev  Next +Parm Ctxt        Main
```

Figure 2-4 Modification Element Maintenance screen after all objects have been modified

Step 3 The CMT option is selected using '**M**' (Change Management Tracking) from the Utilities Menu screen. On the CMT Reports Selection screen; module XX001P01 has been selected and all the other options have been left as per default settings.

This will result in the CMT report displaying all the audit trail information for object XX001P01 in timestamp ascending order.

The following Figure 2-5 illustrates the CMT Reports Selection screen showing the specified reporting options.

```
                    Change Management Tracking   Application: HOSPITAL
                         Reports Selection              Version: 01


  Object:        XX001P01 to _____
  or All Objects: N

  All Versions:   N

  Date Operator:  __  Date: __ __ ____ (dd mm yyyy)
                        To: __ __ ____ (dd mm yyyy)

  User Id:         _____

  Sort Order:    1  Timestamp - Asc

  Object Lang.:  * All          (PF2 to set)

  Object Types:  4C3GHLMAPNS    (PF2 to set)

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
     Help  Types Exit        Sub  Vers                               Main
```

Figure 2-5 CMTS Reports Selection screen showing the specified reporting options

Step 4   By using '**PF5**' (Sub) on the CMT Reports Selection screen; the NATRJE Job Submission screen is displayed. After the correct Job Name and Job Class have been specified, the CMT Report job is submitted using '**PF5**' (Sub).

The following Figure 2-6 illustrates the NATRJE Job Submission screen after the CMT Report job has been submitted.

```
                       - Job Submission -       Application: HOSPITAL

           Job Selection details
           --------------------
               Job Selected  :  (CMTREP) CMTS REPORT

           Job Card details
           ----------------
                   Job Name  :  XGSLXX01
                   Job Class :  X

           Job Control Record details
           ------------------------
            Control Status  :
            Last Job Submitted
              Job Name :
                  Opid :
                  Step :
            Return Code :

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
     Help        Exit       Sub  Ref             Rel              Main
Job : XGSLXX01 Submitted Successfully
```

Figure 2-6 NATRJE Job Submission screen after submitting job

Step 5   Once the job has completed, the CMT Report can be viewed from the job output file.

The following Figure 2-7 illustrates the CMT audit trail records for object XX001P01.

```
              Change Management Tracking System Report       28/09/2001
Application:   HOSPITAL
Criteria Order: Sorted by Timestamp - Ascending
Criteria Range: Object: XX001P01 (Object Types: DGLASNHMP3C4)
Criteria Dates: For all Timestamps
Criteria Vers:  1
Object Name: XX001P01
------------------------------------------------------------------------------
  Version:   1 Timestamp: 28/09/2001 17:00:41.3 User: XGSLXX01
    Before: 0120 RESET #L-MESSAGE
    After:  0120 RESET #L-MSG /* NEE MODIFIED AUTO
  Version:   1 Timestamp: 28/09/2001 17:00:41.3 User: XGSLXX01
    Before: 0250    MOVE "INVALID OPTION SELECTED" TO #L-MESSAGE
    After:  0250    MOVE "INVALID OPTION SELECTED" TO #L-MSG /* NEE MODIFIED
  Version:   1 Timestamp: 28/09/2001 17:00:41.4 User: XGSLXX01
    Before: 0270    MOVE "INVALID PF KEY PRESSED" TO #L-MESSAGE
    After:  0270    MOVE "INVALID PF KEY PRESSED" TO #L-MSG /* NEE MODIFIED
```

Figure 2-7 CMT audit trail records for object XX001P01

ARCHITECTURAL GOVERNANCE

Chapter Overview

This chapter describes the Architectural Governance option available from the Utilities menu. The Architectural Governance option provides the facility to apply global or application specific coding standards to Natural Objects.

The Architectural Governance screen is accessed by selecting option '**A**' (Architectural Governance) from the Utilities Menu screen.

The topics covered are:

1. Global Coding Standards
2. Application Coding Standards
3. Coding Standards Execution
4. Impact Element Maintenance
5. Release Job Lock

Global Coding Standards

The Global Coding Standards option provides the facility for the specification of coding standards that are to be used as the default settings within Natural Engineer.

Using Global Coding Standards, it is possible to specify the Natural application coding standards employed at your site, or if none exist, specify the standards that you wish to apply and adhere to. Then by loading each of your applications into the Repository and running the Architectural Governance impact analysis it is possible to report on each application's compliance.

If an Application has individual coding standards than the global definitions may be overridden by using the Application Coding Standards options.

The Global Coding Standards screen is accessed by selecting option '**G**' (Global Coding Standards) from the Architectural Governance Menu screen.

The following Figure 4-1 illustrates the Global Coding Standards screen.

```
                      - Global Coding Standards -

 Data Item Specification and Usage          Miscellaneous
  24  Data Item Name Length                 _  Allow Numeric Back Refs
  @   Local Data Item Start Character        _  Check for Unused Variables
  Y   Allow Dynamic Variables               _  Check for Redundant Code
  _   Allow AIVs
  _   Allow Use Reserved Words as Data Items
  _   Allow Use of Hexadecimal

 Database Access
  _   Allow Database Access

 Object Complexity
  McCabe                          _____
  Halstead Difficulty            _____
  Maximum Nested Levels           ___
  Max Nbr for Conditional Stmts   ___


 Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
       Help        Exit        Save                     PKeyw              Main
```

Figure 4-1 Global Coding Standards screen

SCREEN ITEMS	DESCRIPTION
Data Item Specification and Usage	Identifies standards specific to Data Items. Data Item Specification and Usage options are
	Data Item Name Length
	A number from 1 to 32 can be selected. Natural Engineer will identify data items that exceed this number.
	Local Data Item Start Character
	The first character required for locally defined data items.
	Allow Dynamic Variables
	Dynamic variables, such as &variables are permitted by standards.
	Allow AIVs
	Application Independent Variables are permitted by the standards.
	Allow Reserved Words as Data Items
	If selected, data item names may also be reserved words.
	Allow use of Hexadecimal
	If selected, hexadecimal code is permitted by standards.
Database Access	Identifies usage of Database Access statements e.g., READ, FIND or HISTOGRAM
	Allow Database Access
	If selected, Database Access statements are permitted by standards.
Object Complexity	Allows the specification of industry standard complexity limits.
	McCabe
	If defined any objects that exceed the specified McCabe number will be identified.
	Halsted Difficulty
	If defined any objects that exceed the specified Halsted difficulty will be identified.
	Maximum Nested Levels
	The number of nested levels that are allowed.
	Max Nbr for Conditional Stmts
	The number of conditional statements that are allowed.
Miscellaneous	Identifies standards for miscellaneous options. Miscellaneous options are:
	Allow Numeric Back References
	If selected numeric back references are allowed by standards.

SCREEN ITEMS	DESCRIPTION
	Check for Unused Variables
	Impact will look for any unused variables. These can be user-defined variables or logical view variables.
	Check for Redundant Code
	Impact will look for any unused source code lines within programming objects, across a whole application.
	Any source code that is driven by event rather than position is ignored. For example AT BREAK, AT END OF PAGE, WRITE TITLE.
	Unused source code within internal subroutines is included, but for external subroutines the Unused Objects report should be referenced.
	Note: Only Structured Mode objects will be impacted. For Reporting Mode objects, it is recommended that they are converted to Structured Mode first using the Mode Conversion function, and then impacted.

PFKEYS	DESCRIPTION
PF1	Activates the help function.
PF3	Exit from the current function and return to previous screen.
PF5	Save the Global Coding Standards Criteria.
PF9	Invoke the Prohibited Keywords screen where keywords can be selected that are not allowed to be used in the Natural Objects.
PF12	Returns to the Natural Engineer Main Menu.

Application Coding Standards

The Application Coding Standards option provides the facility for the specification of coding standards that are specific to a particular application within Natural Engineer. If set, these will override the Global Coding Standards.

The Application Coding Standards screen is accessed by selecting option '**A**' (Application Coding Standards) from the Architectural Governance Menu screen.

The Application Coding Standards option provides the facility for the specification of coding standards that are specific to a particular application within Natural Engineer. If set, these will override the Global Coding Standards.

Selecting this option will allow an application to be chosen that the Application Coding Standards criteria should apply to.

The following Figure 4-2 illustrates the Select Application screen.

```
                  - Select Application -

                 Application:  HOSPITAL

                    Sel  Applications
                     _      TST13900
                     _      TST22100
                     _      TST22300
                     _      TST23100
                     _      TS170101
                     _      TS170103
                     _      TS170104
                     _      TS170106
                     _      TS170401
                     _      TS170501
                     _      TS170601
                     _      TS170603

               Reposition ->  _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
     Help      Exit                      Prev Next                      Main
```

Figure 1-1 Select Application screen

The following Figure 4-3 illustrates the Application Coding Standards screen.

```
                      - Coding Standards -      Application: HOSPITAL

Data Item Specification and Usage          Miscellaneous
  24  Data Item Name Length                 _  Allow Numeric Back Refs
  @   Local Data Item Start Character       _  Check for Unused Variables
  Y   Allow Dynamic Variables              _  Check for Redundant Code
  _   Allow AIVs
  _   Allow Use Reserved Words as Data Items
  _   Allow Use of Hexadecimal

 Database Access
   _   Allow Database Access

 Object Complexity
  McCabe                          _____
  Halstead Difficulty            _____
  Maximum Nested Levels          ___
  Max Nbr for Conditional Stmts  ___


Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
     Help        Exit      Save                    PKeyw            Main
```

Figure 4-3 Application Coding Standards screen

SCREEN ITEMS	DESCRIPTION
Data Item Specification and Usage	Identifies standards specific to Data Items. Data Item Specification and Usage options are
	Data Item Name Length
	A number from 1 to 32 can be selected. Natural Engineer will identify data items that exceed this number.
	Local Data Item Start Character
	The first character required for locally defined data items.
	Allow Dynamic Variables
	Dynamic variables, such as &variables are permitted by standards.
	Allow AIVs
	Application Independent Variables are permitted by the standards.

SCREEN ITEMS	DESCRIPTION
	Allow Reserved Words as Data Items
	If selected, data item names may also be reserved words.
	Allow use of Hexadecimal
	If selected, hexadecimal code is permitted by standards.
Database Access	Identifies usage of Database Access statements e.g., READ, FIND or HISTOGRAM
	Allow Database Access
	If selected, Database Access statements are permitted by standards.
Object Complexity	Allows the specification of industry standard complexity limits.
	McCabe
	If defined any objects that exceed the specified McCabe number will be identified.
	Halsted Difficulty
	If defined any objects that exceed the specified Halsted difficulty will be identified.
	Maximum Nested Levels
	The number of nested levels that are allowed.
	Max Nbr for Conditional Stmts
	The number of conditional statements that are allowed.
Miscellaneous	Identifies standards for miscellaneous options. Miscellaneous options are:
	Allow Numeric Back References
	If selected numeric back references are allowed by standards.
	Check for Unused Variables
	Impact will look for any unused variables. These can be user-defined variables or logical view variables.

SCREEN ITEMS	DESCRIPTION
	Check for Redundant Code
	Impact will look for any unused source code lines within programming objects, across a whole application.
	Any source code that is driven by event rather than position is ignored. For example AT BREAK, AT END OF PAGE, WRITE TITLE.
	Unused source code within internal subroutines is included, but for external subroutines the Unused Objects report should be referenced.
	Note: Only Structured Mode objects will be impacted. For Reporting Mode objects, it is recommended that they are converted to Structured Mode first using the Mode Conversion function, and then impacted.

PFKEYS	DESCRIPTION
PF1	Activates the help function.
PF3	Exit from the current function and return to previous screen.
PF5	Save the Global Coding Standards Criteria.
PF9	Invoke the Prohibited Keywords screen where keywords can be selected that are not allowed to be used in the Natural Objects.
PF12	Returns to the Natural Engineer Main Menu.

Coding Standards Execution

Coding Standards Execution option provides the facility to select one or more Applications to run the Coding Standards criteria against. If an Application has specific Application Coding Standards criteria set then those individual settings will be checked for during the impact execution otherwise the Global Coding Standards criteria will be checked for.

The Coding Standards Execution screen is accessed by selecting option '**E**' (Coding Standards Execution) from the Architectural Governance Menu screen.

The following Figure 4-4 illustrates the Coding Standards Execution Select Applications screen.
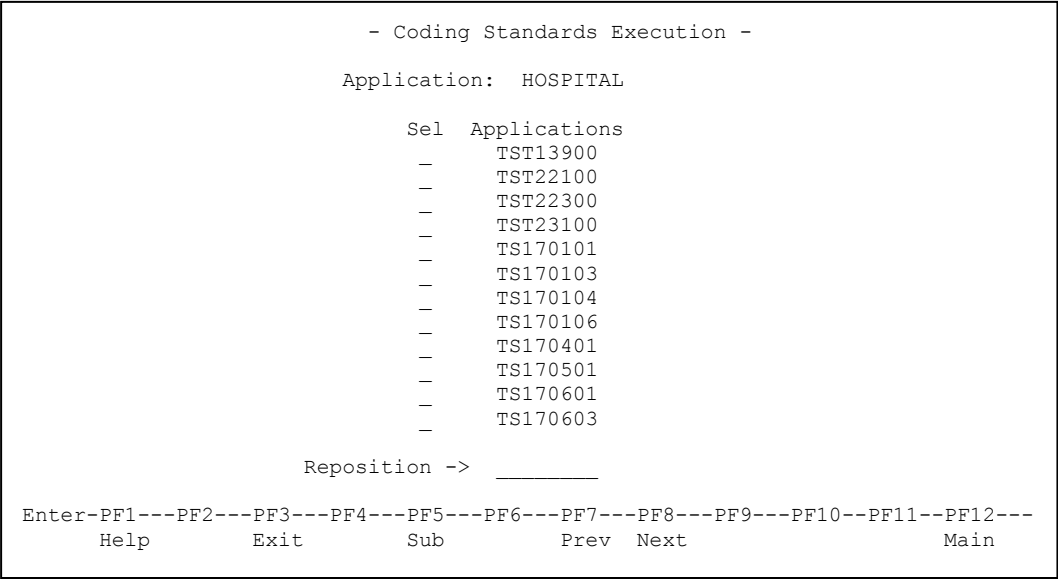
```
                    - Coding Standards Execution -

                 Application:  HOSPITAL

                    Sel  Applications
                     _      TST13900
                     _      TST22100
                     _      TST22300
                     _      TST23100
                     _      TS170101
                     _      TS170103
                     _      TS170104
                     _      TS170106
                     _      TS170401
                     _      TS170501
                     _      TS170601
                     _      TS170603

                 Reposition ->  _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Sub         Prev Next                   Main
```

Figure 4-4 Coding Standards Execution Select Applications screen

The Coding Standards may be executed by selecting '**PF5**' (Sub) to submit the batch job via the NATRJE Job Submission screen.

Note: For more information on the NATRJE Job Submission screen refer to the Natural Engineer Batch Processing (Unix) manual.

Impact Element Maintenance

The Coding Standards Impact Element Maintenance option provides the facility to review the results of the last executed Impact Analysis for the Architectural Governance Option. All impacted objects within the chosen applications are available for selection. Once selected the impacted items within the object are listed.

The impacted items can be selected to reveal the source code context within the object and the impact match reason showing why the item has been impacted. The context of the data item within the data definitions of the selected object are also shown.

The Coding Standards Impact Element Maintenance screen is accessed by selecting option 'I' (Impact Element Maintenance) from the Architectural Governance Menu screen.

Note: For further information on the Impact Element Maintenance screen refer to the Impact Element Maintenance section in the Natural Engineer Application Analysis & Modification for Unix manual.

Release Job Lock

If an execution of Architectural Governance has had a serious failure, then the control record will be locked preventing any further job batch job submission. The lock can be released using option 'J' from the Architectural Governance screen. This will result in a pop-up window being displayed asking for a password.

The password is set to '**GENRJE01**' by default. The password may be changed by modifying the Natural Engineer User Exit, NEEUEX1 which is located in the SYSNEE library.

After typing in the password use the '**ENTER**' key. This will give you the Application Batch Release screen.

The following Figure 4-5 illustrates the Application Batch Release screen.

```
                     - Application Batch Release -


   Job Control Record details for Arc. Gov. Coding Standards
   ---------------------------------------------------------

                         Control Status  :  Y

          Last Job Submitted - Job Name     :  XGSLXX28
                             - Job Class    :  K
                             - Opid         :  XGSLXX
                             - Step         :  ARCEXRUN
                             - Return Code  :  00







Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
              Exit        Upd                                          Main
```

Figure 4-5 Application Batch Release screen

SCREEN ITEMS	DESCRIPTION
Control Status	This will be set to '**Y**' (locked). After releasing the application this will update to show '**N**' (unlocked).
Job Name	The Job Name used for the last submitted job.
Job Class	The Job Class used for the last submitted job.
Opid	The opid used for the last submitted job.
Step	The last completed job step.
Return Code	Always set to '**00**'.

PFKEYS	DESCRIPTION
PF3	Exit from the current function and return to previous screen.
PF5	Releases the Architectural Governance lock. The control status will be updated to '**N**' (unlocked). Architectural Governance jobs can now be submitted again.
PF12	Returns to the Natural Engineer Main Menu.

BULK EXTRACT & LOAD

Chapter Overview

This chapter describes the Bulk Extract & Load option available from the Utilities menu. The Bulk Extract & Load option provides the facility to extract and load multiple applications into your Natural Engineer repository.

The Bulk Extract & Load screen is accessed by selecting option '**B**' (Bulk Extract & Load) from the Utilities Menu screen.

The following topics are covered:

1. Bulk Extract & Load Overview

2. Bulk Extract & Load Selection Screen

Bulk Extract & Load Overview

The Bulk Extract & Load facility allows the selection of multiple applications to be extracted and loaded into the Natural Engineer repository. One combined extract & load job will be submitted per application selected.

If the extract portion of the job fails with a return code 255 this indicates that it has had a critical error and the load portion will not be run. If the extract reports only non-critical errors e.g., missing objects then it will give a return code 254. In this scenario the load portion of the job will run for the selected application.

Bulk Extract & Load Selection Screen

The Bulk Extract & Load Selection screen allows you to select the applications that are to be processed.

The Bulk Extract & Load screen is accessed by selecting option '**B**' (Bulk Extract & Load) from the Utilities Menu screen.

The following Figure 5-1 illustrates the Bulk Extract & Load Selection screen.

```
                          - Bulk Extract & Load -

                        Sel  Applications
                          _    COBJCLNT
                          _    HOSPITAL
                          _    HOSPSET
                          _    SAG-TOURS












                    Reposition ->  _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
       Help        Exit        Sub        Prev  Next                   Main
```

Figure 5-1 Bulk Extract & Load Selection screen

SCREEN ITEMS	DESCRIPTION
Sel	This is the selection column where individual applications can be selected. Valid selections are:
	'**S**' Select object.
Applications	List of Applications available for extracting and loading..
Reposition	Reposition the list of applications starting from the new value entered. The reposition value can be input using either a complete application name or part name using an '*****' (asterisk) wildcard. For example:

*		Will reposition at the start of the application list.
HOSP*		Will reposition at the first application name that matches the mask HOSP or is greater than the mask input.
HOSPITAL		Will reposition at the first application name that matches the mask exactly or is greater than the application name input.

PFKEYS	DESCRIPTION
PF1	Activates the help function.
PF3	Exit from the current function and return to previous screen.
PF5	Invoke the NATRJE Job Submission screen.
PF7	Displays previous page.
PF8	Displays next page.
PF12	Returns to the Natural Engineer Main Menu.

The following Figure 5-2 illustrates the NATRJE submission screen for the Bulk Extract & Load option.

```
                     - Job Submission -

         Job Selection details
         --------------------
             Job Selected  : (BULKEL) BULK EXTRACT&LOAD

         Job Card details
         ----------------
               Job Name  :  XGSLX     (nnn)
               Job Class :  _

         Job Control Record details
         ------------------------
         Control Status  :
         Last Job Submitted
            Job Name :
                Opid :
                Step :
         Return Code :

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Sub   Ref                   Rel               Main
```

Figure 5-2 NATRJE Job Submission screen

The process will submit one combined extract and load job per application submitted. A three digit incremental number will be appended to the job name to distinguish between the jobs.

Note: For more information on the NATRJE Job Submission screen refer to the Natural Engineer Batch Processing (Unix) manual.

6

BULK IMPORT OBJECT DOCUMENTATION

Chapter Overview

This chapter describes the Bulk Import Object Documentation option available from the Utilities menu. The Bulk Import Object Documentation option provides the facility to import documentation from multiple sources into your Natural Engineer repository.

The Bulk Import Object Documentation screen is accessed by selecting option '**O**' (Bulk Import Object Documentation) from the Utilities Menu screen.

The following topics are covered:

1. Bulk Import Object Documentation Overview

2. Bulk Import Object Documentation Selection Screen

Bulk Import Object Documention Overview

The Bulk Object Documentation facility allows the documentation from multiple applications to be imported into the Natural Engineer repository. One combined import job will be submitted per execution.

The object comments imported are any comments found before the first Natural statement within an object.

.

Bulk Import Object Documentation Screen

The Bulk Import Object Documentation Selection screen allows you to select the applications that are to be processed.

The Bulk Import Object Documentation screen is accessed by selecting option '**O**' (Bulk Import Object Documentation) from the Utilities Menu screen.

The following Figure 6-1 illustrates the Bulk Import Object Documentation Selection screen.

```
                      - Bulk Import Object Documentation -

                         Sel  Applications
                          _      COBJCLNT
                          _      HOSPITAL
                          _      HOSPSET
                          _      SAG-TOURS




                         Reposition ->  _____

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
        Help        Exit        Sub   Save  Prev  Next              Sel*  Main
```

Figure 6-1 Bulk Import Object Documentation Selection screen

SCREEN ITEMS	DESCRIPTION
Sel	This is the selection column where individual applications can be selected. Valid selections are:
	'S' Select object.
Applications	List of Applications available for importing comments.
Reposition	Reposition the list of applications starting from the new value entered. The reposition value can be input using either a complete application name or part name using an '*' (asterisk) wildcard. For example:
	* Will reposition at the start of the application list.
	HOSP* Will reposition at the first application name that matches the mask HOSP or is greater than the mask input.
	HOSPITAL Will reposition at the first application name that matches the mask exactly or is greater than the application name input.

PFKEYS	DESCRIPTION
PF1	Activates the help function.
PF3	Exit from the current function and return to previous screen.
PF5	Saves the list of Applications selected.
PF6	Invoke the NATRJE Job Submission screen.
PF7	Displays previous page.
PF8	Displays next page.
PF11	Will select all Applications.
PF12	Returns to the Natural Engineer Main Menu.

The following Figure 6-2 illustrates the NATRJE submission screen for the Bulk Import
Object Documentation option.

```
                    - Job Submission -

        Job Selection details
        --------------------
             Job Selected  : (BULKDC) BULK DOC IMPORT

        Job Card details
        ----------------
                Job Name  :  XGSLX     (nnn)
                Job Class :  _

        Job Control Record details
        --------------------------
        Control Status  :
        Last Job Submitted
           Job Name :
               Opid :
               Step :
        Return Code :

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Sub   Ref                 Rel               Main
```

Figure 6-2 NATRJE Job Submission screen

The process will submit one combined import job.

*Note: For more information on the NATRJE Job Submission screen refer to the Natural
Engineer Batch Processing (Unix) manual.*

7

SQL TABLE MAINTENANCE

Chapter Overview

This chapter describes the SQL Table Maintenance option available from the Utilities menu. The SQL Table Maintenance provides the facility to specify SQL Table Names and DDL Locations for subsequent extraction and load into your Natural Engineer repository.

The SQL Table Maintenance screen is accessed by selecting option '**Q**' (SQL Table Maintenance) from the Utilities menu.

The following topics are covered:

1.   Extract Selection Criteria

2.   Extract

3.   Load

Extract Selection Criteria

The Extract Selection Criteria for SQL Tables allows for the specification of SQL Table Names and the location of the corresponding DDL file for subsequent extract and load into the Natural Engineer Repository.

Extract Selection Criteria Screen

The Extract Selection Criteria screen is accessed by selecting option '**S**' (Extract Selection Criteria) from the SQL Table Maintenance screen.

The following Figure 7-1 illustrates the Extract Selection Criteria screen for SQL Tables.

```
                        - Extract Selection Criteria -


  Opt Start Table                        End Table
    _   DB2SAG.TABLE1_____   DB2SAG.TABLE9_____








     DDL Location:  SAG.SQL.DDL2.SOURCE_____


Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit        Save                                       Main
```

Figure 7-1 Extract Selection Criteria screen for SQL Tables

SCREEN ITEMS	DESCRIPTION
Opt	Line command options. Valid options are:
	'**I**' Insert a new line.
	'**D**' Delete the current line.
Start Table	The name of the first SQL Table to be extracted. This can be a single SQL Table name or part of a range of SQL Table Names if End Table has been specified.
	Note: Refer to the section <u>Specifying Table Names</u> below for more information on how to specify SQL Table names and ranges
End Table	The name of the last SQL Table to be extracted. This is only valid if a Start Table has been specified.
	Note: Refer to the section <u>Specifying Table Names</u> below for more information on how to specify SQL Table names and ranges.
DDL Location	The location of the DDL file containing the SQL Table definitions.

PFKEYS	DESCRIPTION
PF1	Activates the help function.
PF3	Exit from the current function and return to previous screen.
PF5	Save Extract Selection Criteria for SQL Tables.
PF12	Returns to the Natural Engineer Main Menu.

Specifying Table Names

The table names specified in the Start table and End table fields on the Extract Selection Criteria screen or SQL Tables use the following standard conventions. If the namespace is used in the definitions e.g., DB2COP.O* then that should be specified if non-standard criteria are used.

All Tables

Enter an asterisk (*) in the Start Table list (standard default).

Single Table Name

Enter full table names in Start Table list.

Multiple Table Group

Enter partial table name in Start Table list, with an asterisk (*). This will allow you to process all tables starting with the values before the asterisk.

Multiple Table Range

Enter a Start Table name and an End Table name in the same row. This will process all tables in alphanumeric order starting from the Start Table and ending with the End Table.

Combination Selection Types

You can enter multiple rows with different criteria, including multiple single tables, groups and ranges. Up to 10 sets of criteria may be added.

Examples:

Start Table	End Table	Result
*		Process all tables.
SQL-PATIENT		Process single table 'SQL-PATIENT'.
SQL-PAT*		Processes all tables with names prefixed with 'SQL-PAT'.
SQL-PATIENT	SQL-PATIENT9	Processes all objects in the alphabetic range starting from SQL-PATIENT and ending at SQL-PATIENT9.

Extract

This process will analyze SQL Data Definition Language (DDL) file and write out a XML file that contains the neutral SQL Table Name records. This is then used in the Load process.

This option is invoked by selecting option '**E**' (Extract) from the SQL Table Maintenance screen. This will open the NATRJE Job Submission screen.

Note: For more information on the NATRJE Job Submission screen refer to the Natural Engineer Batch Processing (Unix) manual.

Note: To run the Extract process for SQL Tables the Natural User Exit USR2021N needs to be available to the Natural session. It is recommended that this User Exit object is copied from SYSEXT to the FNAT library SYSTEM.

Load

This process will load previously extracted SQL Table Data into the Natural Engineer Repository.

This option is invoked by selecting option '**L**' (Load) from the SQL Table Maintenance screen. This will open the NATRJE Job Submission screen.

Note: For more information on the NATRJE Job Submission screen refer to the Natural Engineer Batch Processing (Unix) manual.

SQL Tables that are no longer required in the Repository can be selectively deleted from the Repository using the Delete Object option under the Application menu.

Note: See the section Delete Object in Chapter 1 of the Natural Engineer Application Management for Unix manual for more information.

7

INDEX

T

Task Scheduler, 9
  Initiator Batch Job, 23

Overview, 10
Task Details screen, 18
Task Log screen, 16
Task Scheduler Summary screen, 12