

Natural Engineer

Version 4.3.1

Application Management

for Windows

**Manual Order Number: NEE431-020WIN**

This document applies to Natural Engineer version 4.3.1 and to all subsequent releases.

Specifications contained herein are subject to change, and these changes will be reported in subsequent revisions or editions.

Readers' comments are welcomed. Comments may be addressed to the Documentation Department at the address on the back cover. Internet users may send comments to the following e-mail address:

[document@gensystems.com](mailto:document@gensystems.com)

Copyright © September 2001, Generation Systems Ltd., East Grinstead, UK.

The documentation often refers to numerous hardware and software products by their trade names. In most, if not all cases, these designations are claimed as trademarks or registered trademarks by their respective companies.

# TABLE OF CONTENTS

<b>ABOUT THIS MANUAL</b> .....	<b>1</b>
Purpose of this manual .....	1
Target Audience .....	1
Typographical Conventions used in this manual .....	2
How this manual is organized .....	3
Terminology .....	4
Related Literature .....	7
<b>1. MANAGING APPLICATIONS</b> .....	<b>9</b>
Chapter Overview.....	9
Open an Application.....	10
Delete an Application .....	11
Delete Object.....	12
Application Preferences.....	15
<b>2. BUILDING A REPOSITORY OF YOUR APPLICATIONS</b> .....	<b>21</b>
Chapter Overview.....	21
Extract Selection Criteria.....	22
Pre-Parser .....	28
Extract Source Code .....	34
Load Repository .....	35
Extract and Load .....	36
Extract, Load and Impact.....	36
Extract Missing Objects .....	37
<b>3. APPLICATION MANAGEMENT</b> .....	<b>39</b>
Chapter Overview.....	39
Soft Links .....	40
User Documentation.....	56

**Natural Engineer Application Management**

<b>4. APPLICATION INVENTORY</b> .....	<b>78</b>
Chapter Overview.....	78
Quality Logs.....	79
Application Reports.....	81
<b>INDEX</b> .....	<b>83</b>

# ABOUT THIS MANUAL

## Purpose of this manual

---

This manual covers the Application Management for Natural Engineer version 4.3.1.

It describes the various processes available that enable you to create and manage Natural applications within Natural Engineer.

The topics cover the Application options found under the Applications menu, which include:

- How to create new applications, open existing applications and delete applications.
- Customizing the control of the application by setting preferences.

Also covered, are all the options to Extract and Load your Natural applications from the Natural application library into the Repository. An overview of the reporting available for these functions is also provided.

## Target Audience

---

The target audience for this manual is intended to be any User of Natural Engineer 4.3.1 at any level of experience.

## Typographical Conventions used in this manual

---

The following conventions are used throughout this manual:

<b>UPPERCASE TIMES</b>	Commands, statements, names of programs and utilities referred to in text paragraphs appear in normal (Times) uppercase.
<b>UPPERCASE BOLD COURIER</b>	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:

⇒	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 Application Management options of Natural Engineer version 4.3.1 in the following chapters:

<b>Chapter</b>	<b>Contents</b>
1	Describes the various options available to select and delete applications, and set up preferences that define the application environments.
2	Describes the various extract and load processes available within Natural Engineer that allow you to load your applications into the Repository.
3	Describes the Soft Links option to specify object link information.  Describes the User Documentation option which allows you to specify comments for each object within an application on the Repository. This complements the object source code information already stored in the Repository.
4	Provides an overview of the reporting options available for the topics covered in this manual.

## Natural Engineer Application Management

# Terminology

---

It is assumed that you are familiar with general Natural and mainframe terminology, as well as the terms and concepts relating to MS-Windows environments. This section explains some terms that are specific to the Natural Engineer product.

## 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).

## 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.

## **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.

## **Library**

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

## **Natural Engineer Application Management**

### **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.

### **Presentation Split Process**

The Presentation Split Process is a sub-function of the Object Builder function that removes screen I/O statements from current application objects and places them in generated subprograms.

### **Soft Link**

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

### **Technical Split Process**

The Technical Split Process is a sub-function of the Object Builder function that results in the encapsulation of each database access within the application, into a sub-program so that the application is separated into 'presentation and logic' and 'database access'.

### **Type**

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

### **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.

## Related Literature

---

The complete set of Natural Engineer manuals consists of:

**1. Natural Engineer Concepts and Facilities (NEE431-006ALL)**

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 (NEE431-008ALL)**

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 4.3.1.

**3. Natural Engineer Installation Guide (NEE431-010ALL)**

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

**4. Natural Engineer Administration Guide (NEE431-040WIN)**

**Natural Engineer Administration Guide (NEE431-040MFR)**

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 (NEE431-020WIN)**

**Natural Engineer Application Management (NEE431-020MFR)**

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

**6. Natural Engineer Application Documentation (NEE431-022WIN)**

**Natural Engineer Application Documentation (NEE431-022MFR)**

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 Application Management**

### **7. Natural Engineer Application Analysis and Modification (NEE431-023WIN)**

#### **Natural Engineer Application Analysis and Modification (NEE431-023MFR)**

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 (NEE431-024WIN)**

#### **Natural Engineer Application Restructuring (NEE431-024MFR)**

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 (NEE431-080WIN)**

#### **Natural Engineer Utilities (NEE431-080MFR)**

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

### **10. Natural Engineer Reporting (NEE431-025ALL)**

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] (NEE431-026MFR)**

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

# MANAGING APPLICATIONS

## Chapter Overview

---

This chapter describes how to select and manage applications for processing in Natural Engineer, and how to set up preferences for the applications.

In the context of Natural Engineer, ‘Application’ is as described in the Terminology section and is the name Natural Engineer uses to describe a library or set of related libraries.

An Application is a library or group of related libraries that 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.

The following options are available from the Application menu and are described in this chapter:

1. **Open**

Open an existing Application or define a new Application.

2. **Delete**

Delete an Application from Natural Engineer.

3. **Delete Object**

Delete an object from the Natural Engineer Repository.

4. **Preferences**

Define Application preferences and defaults.

# 1

## Natural Engineer Application Management

### Open an Application

---

Use this option either to select a previously processed application from the list, or to specify a new application.

This is accessed by using the following menu navigation Application→Open.

This will open the Open an Application selection screen, where an existing application can be selected or a new application name typed in.

*Note: For a new application, if the application name you want to use in Natural Engineer is different from the library name in the FUSER, you will have to set the Natural Library parameter in the Preferences screen to the FUSER library name. For more information see the [Application Preferences](#) section in this chapter.*

The following Figure 1-1 illustrates the Open an Application screen.

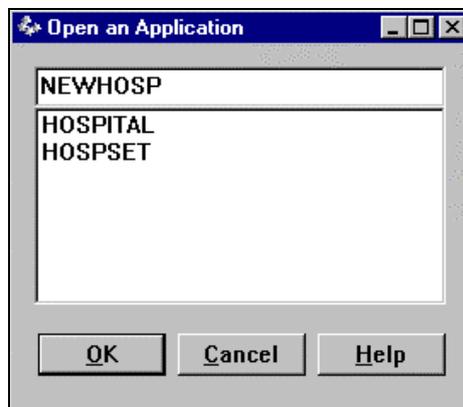


Figure 1-1 Open an Application screen

SCREEN ITEMS	DESCRIPTION
<b>Selected Application</b>	The name of the application to be opened.
<b>Application List</b>	List of all the available applications on the Repository.

BUTTON NAME	DESCRIPTION
<b>OK</b>	Accepts the selected application.
<b>Cancel</b>	Cancels the open process and returns back to the main Natural Engineer screen.
<b>Help</b>	Invokes the Open an Application help.

## Delete an Application

This option allows you to delete an Application that exists in Natural Engineer. You can select the Application from the list.

All application data will be removed from the Natural Engineer Repository. This deletes all Application, Analysis and Modification information.

This is accessed by using the following menu navigation Application→Delete.

This will open the Delete Application selection screen, where an existing application can be selected.

The following Figure 1-2 illustrates the Delete Application screen.

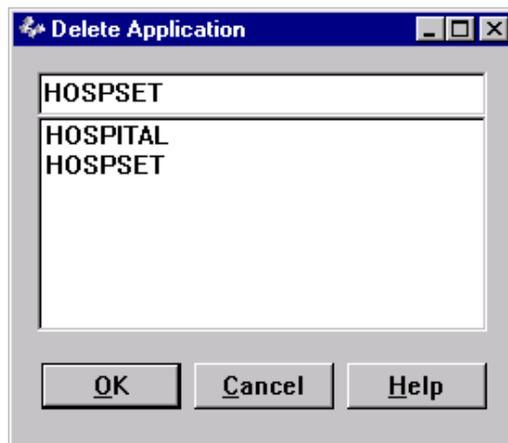


Figure 1-2 Delete Application screen

# 1

## Natural Engineer Application Management

SCREEN ITEMS	DESCRIPTION
--------------	-------------

<b>Selected Application</b>	The name of the application to be deleted.
<b>Application List</b>	List of all the available applications on the Repository.

BUTTON NAME	DESCRIPTION
-------------	-------------

<b>OK</b>	Accepts the selected application and invokes the delete application process.
<b>Cancel</b>	Cancels the delete process and returns back to the main Natural Engineer screen.
<b>Help</b>	Invokes the Delete an Application help.

## Delete Object

This allows you to select an object for deletion from the Natural Engineer Repository, with confirmation.

Use this option to remove object data from the Natural Engineer Repository. This deletes Application, Analysis and Modification information for the object.

*Note: If the object is to be deleted permanently from the application Repository, you must also delete it from the source library, so that it is not re-extracted by error in the future.*

This is accessed by using the following menu navigation Application → Delete Object.

This will open the Delete Object selection screen, where all the objects loaded for the currently selected application are listed. From here objects can be selected for deletion.

The following Figure 1-3 illustrates the Delete Object screen.

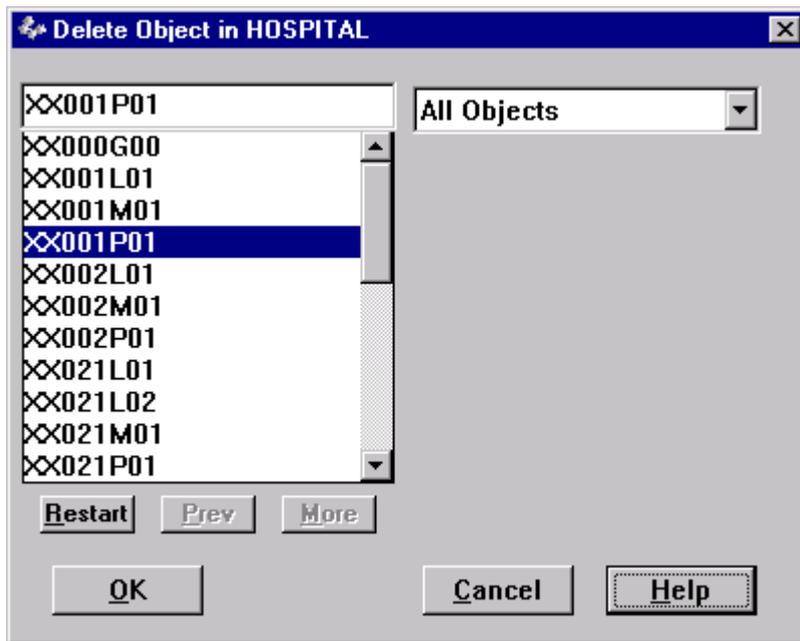


Figure 1-3 Delete Object screen

SCREEN ITEMS	DESCRIPTION
<b>Selected Object</b>	The name of the object to be deleted.
<b>Object List</b>	List of all the available objects for the currently selected application. <i>Note: The name of the currently selected application is shown in the title bar.</i>
<b>Object Types</b>	This controls the list of objects available in the objects list. Available selections are: <ul style="list-style-type: none"> <li>● All Objects</li> <li>● Programs</li> <li>● Maps</li> <li>● Data Defn. Modules</li> <li>● Parameter Data Areas</li> <li>● Local Data Areas</li> <li>● Copycodes</li> <li>● Subprograms</li> <li>● Subroutines</li> <li>● Help routines</li> </ul>

# 1

## Natural Engineer Application Management

SCREEN ITEMS	DESCRIPTION
<b>Object List</b>	<ul style="list-style-type: none"><li>● <b>Global Data Areas</b></li><li>● <b>Dialogs</b></li></ul> Scrollable list of all the available objects for the currently selected application. <i>Note: The name of the currently selected application is shown in the title bar.</i>

BUTTON NAME	DESCRIPTION
<b>Restart</b>	Allows the Object List to be restarted from a particular object name.
<b>Prev</b>	Scrolls the object list to previous page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
<b>More</b>	Scrolls the object list forward one page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
<b>OK</b>	Accepts the selected object and invokes the delete object process.
<b>Cancel</b>	Cancels the delete process and returns back to the main Natural Engineer screen.
<b>Help</b>	Invokes the Delete an Object help.

*Note: For more information on the NATENG.INI file parameter LISTBOXMAX refer to Chapter 1 in the Natural Engineer Administration Guide for Windows manual.*

## Application Preferences

---

The Application Preferences screen allows you to define the preferences that are to be applied during the extract and modification processes within Natural Engineer.

For example, applications that make use of steplibs.

If an application makes use of steplibs, then these can be specified on the Application Preferences screen. This maybe applicable where applications make use of standard routines which are held on a separate natural library rather than including them within the application natural library.

*Note: The steplibs would then need to be extracted and loaded into the Repository as applications in their own right, in order that Natural Engineer can build the necessary cross-reference information and maintain integrity with the applications using the steplib.*

This is accessed by using the following menu navigation Application → Preferences.

# 1

## Natural Engineer Application Management

The following Figure 1-4 illustrates the Application Preferences screen.

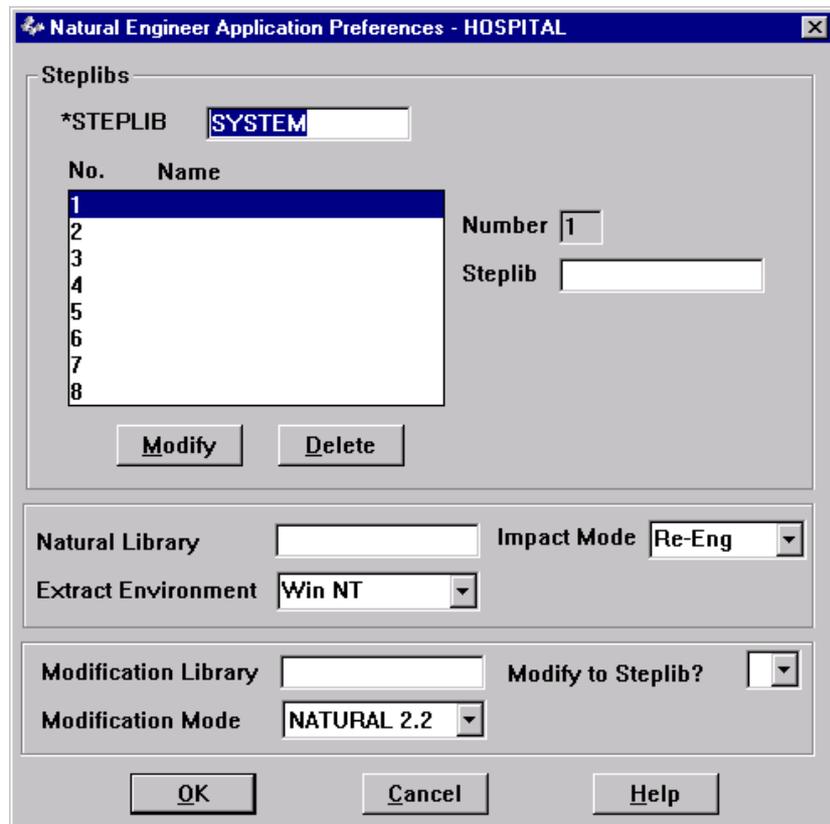


Figure 1-4 Application Preferences screen

SCREEN ITEMS	DESCRIPTION
<b>*STEPLIB</b>	The master steplib library name assigned to the Natural system variable *STEPLIB. This is normally set to SYSTEM.
<b>Steplib</b>	<p>This option allows you to define multiple Steplib libraries from which Natural Engineer can retrieve objects referenced from the primary Natural library.</p> <p>You can define up to 8 steplib Natural library names that Natural Engineer will search for the application code. Natural Engineer will search the steplib chain for the following items if they do not exist on the base library:</p> <p><i>Note: A steplib library name cannot be the same as the Application name or the base Natural library name.</i></p> <hr/> <p><b>Data Areas</b> (LDAs, PDAs GDAs)</p> <p><b>Copycode</b></p> <p><b>Subprograms</b> (invoked via CALLNAT)</p> <p><b>Programs</b> (invoked via FETCH/FETCH RETURN/FETCH REPEAT)</p> <p><b>Maps</b> (invoked by INPUT USING MAP/WRITE USING FORM)</p> <p><b>Helproutines</b> (invoked via HE=)</p> <p><b>Objects</b> (invoked via STACK TOP COMMAND/STACK COMMAND)</p> <p><b>DDMs</b></p> <p><i>Note: For the STACK command, Natural Engineer will attempt to determine if the command that is being stacked is an actual object or not. It does this by interrogating an exclude table that lists common commands that are not objects e.g. STOW, EDIT.</i></p> <hr/>
<b>Natural Library</b>	If the Natural Engineer Application name is to be different from the library name in the FUSER, specify the actual FUSER library name in the Natural library field. If a Natural Library is not specified, Natural Engineer assumes that the application name is the same as the FUSER library name.

# 1

## Natural Engineer Application Management

SCREEN ITEMS	DESCRIPTION
<b>Modification Library</b>	<p>Use the Modification Library field if you wish to specify a library that the modified code will be written to.</p> <p>If a Modification Library is not specified Natural Engineer places all modified code in a library name with an 'X' as the last character of the application name. If the name is already 8 characters long, the last character is removed and replaced with the 'X'.</p> <p>The Modification Library name can be the same as the base Natural Library name. This allows any modified objects from the modification process to be applied to the base Natural Library. If this is set, a warning message is produced to highlight that the base Natural Library will be updated.</p>
<b>Modify to Steplib?</b>	<p>This option allows you to specify to where objects that are on steplib libraries are to be modified, either to the Steplib library or the Application Modification library.</p> <hr/> <p><b>N</b>    Modify all objects to the application Modification library.</p> <p><b>Y</b>    Modify steplib objects in the application to the steplib Modification library.</p>
<b>Impact Mode</b>	<p>The Impact Mode identifies the type of Analysis that will be performed by Natural Engineer. Valid options are:</p> <p><b>Re-Eng</b>    Use this option to execute Analysis using general Reengineering functionality. The Default Mode.</p>
<b>Modification Mode</b>	<p>This option is used to determine which Natural version code options are to be used for Modification. Selections available are</p> <p><b>Natural 2.2</b>    This will use Natural 2.2 code options.</p> <p><b>Natural 2.3</b>    This will use Natural 2.3 code options.</p>
<b>Extract Environment</b>	<p>This is a documentation facility used by Natural Engineer's Reengineering functions, that specifies the application environment from which the Natural source code comes. Selections available are:</p> <p><b>MVS</b>  <b>LightStorm</b>  <b>Win NT</b>  <b>Unix</b>  <b>VMS</b>  <b>BS2000</b>  <b>VSE</b></p>

<b>BUTTON NAME</b>	<b>DESCRIPTION</b>
<b>Modify</b>	To add a new Steplib or change an existing Steplib, either select an empty numbered Steplib line and enter the name in the box, or select the Steplib to be changed, from the list, overwrite the name in the Steplib box and then select the Modify button.
<b>Delete</b>	To delete an existing Steplib, select that Steplib in the numbered Steplib list, and select the Delete button.
<b>OK</b>	Accepts the application preferences that have been specified and returns back to the main Natural Engineer screen.
<b>Cancel</b>	Cancels the Application Preferences process and returns back to the main Natural Engineer screen.
<b>Help</b>	Invokes the Application Preferences help.



# BUILDING A REPOSITORY OF YOUR APPLICATIONS

## Chapter Overview

---

The first stage in using Natural Engineer is to build a comprehensive Repository of information on your Natural applications, creating a structured view of the application code and providing application Analysis and inventory information. The following options are available from the Environment menu and are described in this chapter:

### 1. **Extract Selection Criteria**

Select one or more objects from the Natural System File for extraction.

### 2. **Pre-Parser**

The Pre-parser is an optional pre-Extract process, which provides basic application information, identifies missing objects and checks for incomplete syntax.

### 3. **Extract Source Code**

Extracts the Natural source codes and creates an output file of neutral records.

### 4. **Load Repository**

Loads the Extract output file information into the Natural Engineer Repository.

### 5. **Extract & Load**

Allows you to execute both the Extract and the Load processes in one single step.

### 6. **Extract, Load & Impact**

Allows you to execute all of the Extract, Load and Impact Analysis processes in one step, using previously defined Search Criteria.

### 7. **Extract Missing Objects**

Extracts the objects identified as missing on the Missing Objects Report.

## Extract Selection Criteria

You use this option to select one or more objects from the Natural System File for extraction. This function is optional; it allows you to include selected objects that may have changed as a result of maintenance.

You can specify individual objects, select several objects using wildcards, and ranges of objects using the Extract Selection Criteria screen box illustrated below:

The default for Extract Selection Criteria is to extract all objects from the Natural application library specified in application preferences.

*Note: Refer to the section [Application Preferences](#) in Chapter 1 for more information on application preferences.*

### Extract Selection Criteria Window

The Extract Selection Criteria window is accessed using the following menu navigation: Environment → Extract Selection Criteria.

The following Figure 2-1 illustrates the Extract Selection Criteria screen.

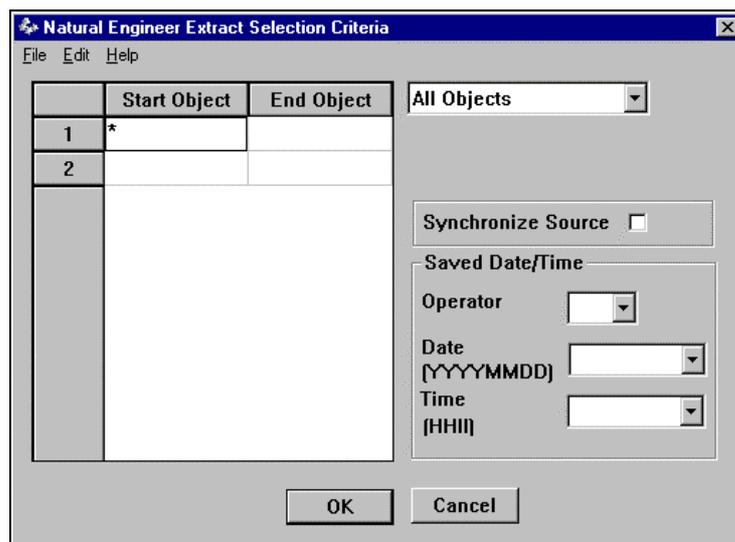


Figure 2-1 Extract Selection Criteria

MENU ITEMS	OPTIONS	DESCRIPTION
<b>File</b>	<b>Valid Missing Objects</b>	List objects that are validly missing from the application, such as System and Construct objects. <i>Note: Refer to the section <a href="#">Valid Missing Objects</a> for more information on this option.</i>
	<b>Exit</b>	Exits the Extract Selection Criteria screen.
<b>Edit</b>	<b>Insert Row</b>	Add a new row into the object list box.
	<b>Delete Row</b>	Delete an existing row from the object list box.
<b>Help</b>		Invokes the Extract Selection Criteria help.

SCREEN ITEMS	DESCRIPTION
Start Object	The name of the first object to be extracted. This can be a single object name or part of a range of objects if End Object has been specified.  Note: Refer to the section <a href="#">Specifying Object Names</a> below for more information on how to specify object names and ranges.
End Object	The name of the last object to be extracted. This is only valid if a Start Object has been specified.  Note: Refer to the section <a href="#">Specifying Object Names</a> below for more information on how to specify object names and ranges.
Object Type	You can use the drop-down box to limit the objects selected to one or all objects types. Valid object types are: <ul style="list-style-type: none"> <li>● <b>All Objects</b></li> <li>● <b>Programs</b></li> <li>● <b>Maps</b></li> <li>● <b>Parameter Data Areas</b></li> <li>● <b>Global Data Areas</b></li> <li>● <b>Local Data Areas</b></li> <li>● <b>Copycodes</b></li> <li>● <b>Subprograms</b></li> <li>● <b>Subroutines</b></li> <li>● <b>Help routines</b></li> <li>● <b>Dialogs</b></li> <li>● <b>Classes</b></li> </ul>

## 2

### Natural Engineer Application Management

SCREEN ITEMS	DESCRIPTION												
<b>Synchronize Source</b>	When this option is selected, Natural Engineer compares each object's saved date in the Repository with the saved date of the object in the Natural System File. If the object in the System File has been saved more recently than that in the Repository, Natural Engineer will re-extract that object into the Repository and overwrite the existing object.												
<b>Saved Date/Time</b>	This option allows you to specify the Saved Date and/or Time of the object in the Natural System File. If specified, the Extract process will only extract objects that meet the specified range.												
<b>Operator</b>	The Operator drop-down box allows you to select the operator to be used with the date/time specified. Options are:												
	<table border="1"> <thead> <tr> <th>Operator</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><b>Blank</b></td> <td>Function not applicable.</td> </tr> <tr> <td><b>EQ</b></td> <td>Objects with Saved Dates equal to the date/time entered.</td> </tr> <tr> <td><b>NE</b></td> <td>Objects with Saved Dates not equal to the date/time entered.</td> </tr> <tr> <td><b>LT</b></td> <td>Objects with Saved Dates less than the date/time entered.</td> </tr> <tr> <td><b>GT</b></td> <td>Objects with Saved Dates greater than the date/time entered.</td> </tr> </tbody> </table>	Operator	Description	<b>Blank</b>	Function not applicable.	<b>EQ</b>	Objects with Saved Dates equal to the date/time entered.	<b>NE</b>	Objects with Saved Dates not equal to the date/time entered.	<b>LT</b>	Objects with Saved Dates less than the date/time entered.	<b>GT</b>	Objects with Saved Dates greater than the date/time entered.
Operator	Description												
<b>Blank</b>	Function not applicable.												
<b>EQ</b>	Objects with Saved Dates equal to the date/time entered.												
<b>NE</b>	Objects with Saved Dates not equal to the date/time entered.												
<b>LT</b>	Objects with Saved Dates less than the date/time entered.												
<b>GT</b>	Objects with Saved Dates greater than the date/time entered.												
<b>Date</b>	The date must be entered in format: YYYYMMDD.												
<b>Time</b>	The time must be entered in format: HHII.												

BUTTON NAME	DESCRIPTION
<b>OK</b>	As appropriate, accepts entries on the screen and may leave the screen, or starts a process.
<b>Cancel</b>	Cancels the current function, does not accept any screen entries and leaves the current screen.

## Specifying Object Names

The object names specified in the Start Object and End Object columns on the Extract Selection Criteria screen use the following standard conventions:

### Single Object Name

Enter full object names in Start Object list.

### Multiple Object Group

Enter partial object name in Start Object list, with an asterisk (\*). This will allow you to Extract all objects starting with the values before the asterisk.

### Multiple Object Range

Enter a Start Object name and an End Object name in the same row. This will Extract all objects in alphanumeric order starting from the Start Object and ending with the End Object.

### Combination Selection Types

You can enter multiple rows with different criteria, including multiple single objects, groups and ranges.

### Examples:

Start Object	End Object	Result
*		Extracts all objects.
ABC*		Extracts all objects with names prefixed with ABC.
AC*	ND*	Extracts all objects in the alphabetic range starting from ACxxxxxx to NDxxxxxx.

## 2

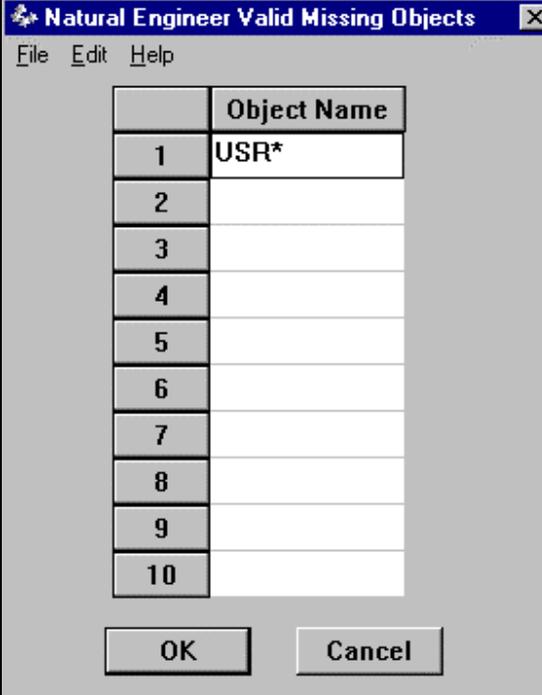
### Natural Engineer Application Management

#### Valid Missing Objects

It is possible to enter a list of object names with wildcards, to a maximum of 10 that will not be marked as missing during the Extract process.

The Valid Missing Objects option is accessed from the Extract Selection Criteria screen using the menu navigation: File → Valid Missing Objects.

The following Figure 2-2 illustrates the Valid Missing Objects screen.



	Object Name
1	USR*
2	
3	
4	
5	
6	
7	
8	
9	
10	

Figure 2-2 Valid Missing Objects

MENU ITEMS	OPTIONS	DESCRIPTION
<b>File</b>	<b>Exit</b>	Exit the Valid Missing Objects screen and return back to the Extract Selection Criteria screen.
<b>Edit</b>	<b>Delete Row</b>	Delete the object name from the selected row.
<b>Help</b>		Invoke the Valid Missing Objects help.

SCREEN ITEMS	DESCRIPTION
<b>Object Name</b>	The object name to be marked as a valid missing object. This can be a exact name or a part name with wildcard. For Example: <b>XX003P01</b> Object XX003P01 would be marked as a valid missing object. <b>XX001*</b> Any objects prefixed with XX001 would be marked as valid missing objects.

BUTTON NAME	DESCRIPTION
<b>OK</b>	As appropriate, accepts entries on the screen and may leave the screen, or starts a process.
<b>Cancel</b>	Cancels the current function, does not accept any screen entries and leaves the current screen.

## Pre-Parser

---

This is an optional process, which provides you sooner, with information about the application you want to Load into the Repository, without actually extracting it. The Pre-Parser checks the Natural code for objects that are referenced but missing from the Natural application.

The Pre-parser can perform the following functions

- Examine the Natural code for objects that are missing and report on them.
- Identify the size of the application in terms of lines of code and number of objects.
- Provide Load balancing information for multiple Extract, Load and Analysis executions, in parallel.

If the Pre-parser has been executed previously, you are given the option of re-executing it, or viewing the data generated by the last execution.

### Source Code Pre-Parser Window

The Source Code Pre-Parser window is accessed using the following menu navigation: Environment → Pre-Parser.

The window will not be opened until the pre-parser process has executed. This process is similar to running the Extract Source Code option, where the Pre-Parser will run through the objects that have been selected for extraction.

The following Figure 2-3 illustrates the Source Code Pre-Parser screen.

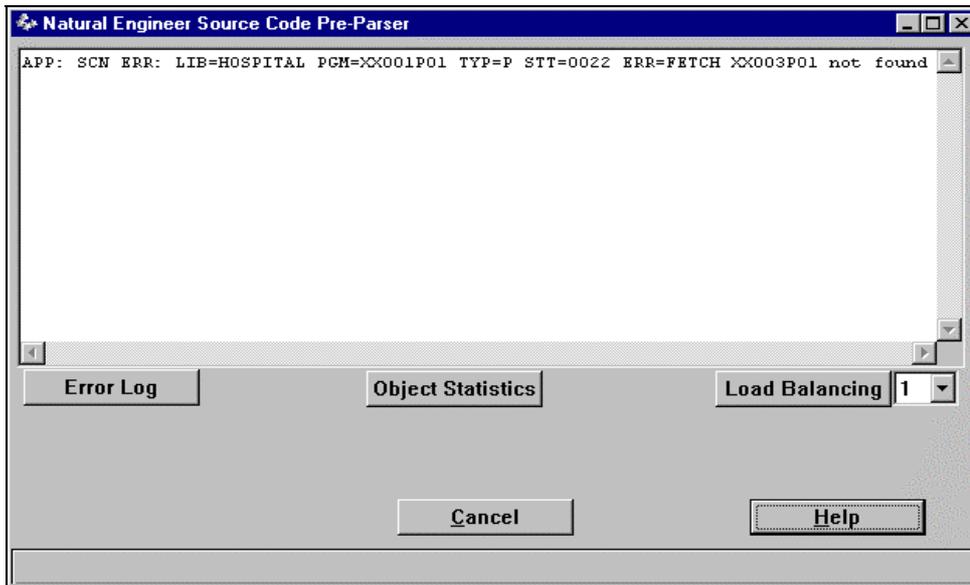


Figure 2-3 Source Code Pre-Parser

SCREEN ITEMS	DESCRIPTION
<b>Pre Parser Results</b>	<p>This shows the Pre Parser results. There are three different types of details that can be displayed:</p> <ol style="list-style-type: none"> <li><b>1. Error Log details</b> Shows all the missing object details in log format. <i>Note: Refer to the section <a href="#">Error Log</a> for more information.</i></li> <li><b>2. Object Statistics</b> These are shown if screen display mode selected after using the Object Statistics button. <i>Note: Refer to the section <a href="#">Object Statistics</a> for more information.</i></li> <li><b>3. Load Balancing</b> These are shown if screen display mode selected after using the Load Balance button. <i>Note: Refer to the section <a href="#">Load Balancing</a> for more information.</i></li> </ol>

BUTTON NAME	DESCRIPTION
<b>Error Log</b>	Use this option to examine the source code for missing Natural objects. Natural Engineer will check for all programming objects being available and also DDMs.
<b>Object Statistics</b>	Use this option to identify the size of the application. Natural Engineer will identify the lines of code size of the application as well as the expanded. The expanded size incorporates any included objects. You can also see a list of objects that are used by each Natural object. The statistics show values both including and excluding comments.
<b>Load Balancing</b>	Use this option to identify the split of the application in terms of lines of code required so that Extract and Load can be executed in parallel. Natural Engineer performs the calculation based on the expanded lines of code in the application.
<b>Cancel</b>	Cancels the current function, does not accept any screen entries and leaves the current screen.
<b>Help</b>	Invokes the Pre-Parser help.

## Error Log

The Error Log displays any errors found by the Pre-parser, including missing objects and incomplete syntax.

The Error log is the default display when the Pre-Parser process completes and opens the Source Code Pre-Parser screen. It is also possible to display the Error Log details by using the Error Log button on the Source Code Pre-Parser screen.

*Note: Refer to Figure 2-3 for an example of the error log details and how they are presented.*

The Error Log details consist of report lines containing the following information:

<b>REPORT ITEM</b>	<b>DESCRIPTION</b>
<b>LIB=</b>	Identifies the name of the application being processed.
<b>PGM=</b>	The name of the Object being extracted.
<b>TYP=</b>	The type of Natural object, i.e. Map, Program Local Data Area.
<b>STT=</b>	The line number of the external object being referenced.
<b>ERR=</b>	Details the missing object.

## Object Statistics

The Object Statistics displays application information including: Application name, total lines with and without comments, with and without expanded code and a list of objects, with their number of lines including and excluding comments and expanded code.

The Object Statistics can be accessed using the Object Statistics button on the Source Code Pre-Parser screen.

When invoking this option, the Report Confirmation window will be displayed. From here, a selection for the display mode can be made so that the Object Statistics can be viewed in either Reporter, Screen (this will place the report within the Pre-Parser results on the Source Code Pre-Parser window) or Excel.

*Note: For more information on the Report Confirmation window refer to Chapter 1 in the Natural Engineer Reporting manual.*

The following Figure 2-4 illustrates the Object Statistics report displayed on the Source Code Pre-Parser screen.

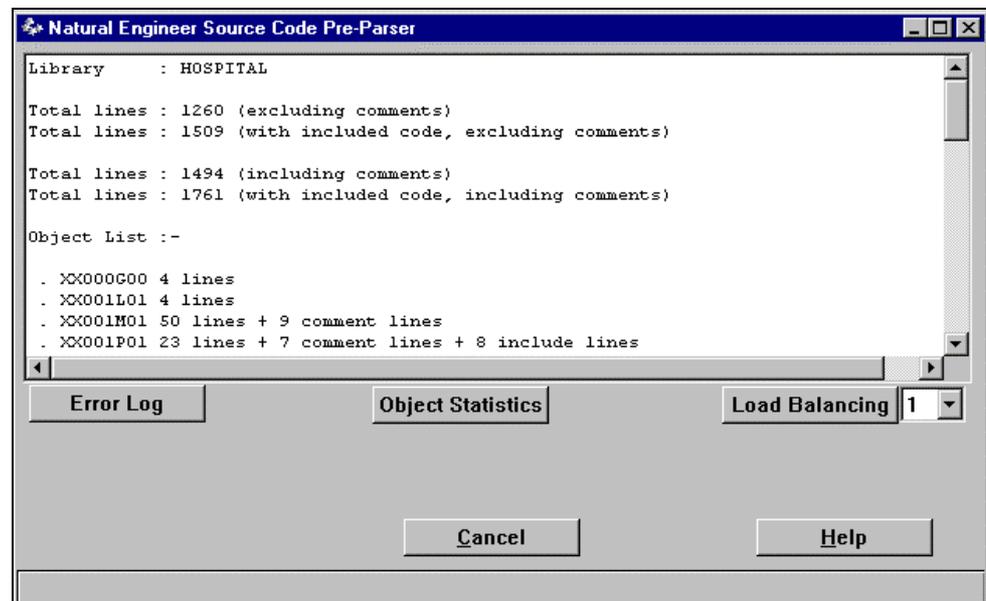


Figure 2-4 Object Statistics

## Load Balancing

The Load Balancing function allows you to select the number of concurrent executions required and then select the Load Balancing button. You will be provided with object ranges to enter into the Extract Selection Criteria for each of the separate executions. Object range values can be set from 1 to 50 and are found to the right hand side of the Load Balancing button.

The Load Balancing can be accessed using the Load Balancing button on the Source Code Pre-Parser screen.

The following Figure 2-5 illustrates the Load Balancing report displayed on the Source Code Pre-Parser screen.

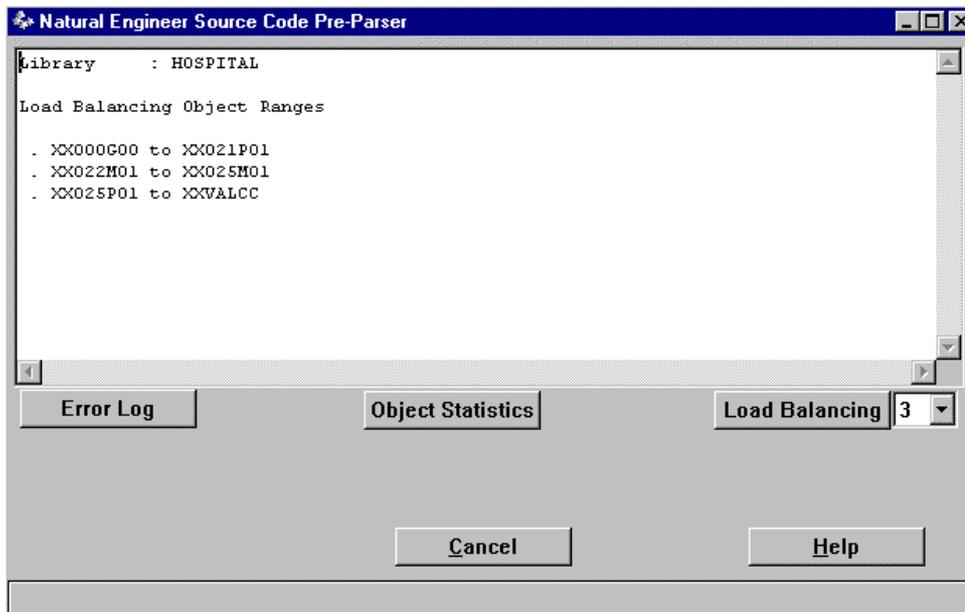


Figure 2-5 Load Balancing

## Extract Source Code

---

This is the first real processing step in creating the Natural Engineer Repository and extracts the Natural source code for the defined application.

Natural Engineer reads the application code and creates a "neutral" view of the code, that is, irrespective of the Natural version or the mode used (structured or reporting).

The Extract process writes out files that contain the neutral application records, as well as an error file. If you are executing multiple Extract processes it is advisable to make a copy the current error file so that is always available. The structure of the file name in the X:\PROGRAM FILES\SOFTWARE AG\NEE\V431\DATA directory (where X: is the drive on which Natural Engineer has been installed), is '*application-name*.EEX'. Save this file as another name.

The Extract function also performs a basic quality check, which identifies any invalid statements or syntax within the objects. If any errors are found, they are logged and displayed on completion of the Extract process.

### Related Processes

The Extract process allows you to correct errors before building the Repository. It also allows you to include identified missing objects, such as DDMs, Data Areas and Copycode before further processing. Missing objects can be seen in the Missing Natural Objects report, which can be accessed using the following menu navigation: Environment → Quality Logs → Missing Natural Objects.

*Note: For more information on the Missing Natural Objects Report refer to Chapter 3 in the Natural Engineer Reporting manual.*

After Extract has executed, any errors can be seen in the Extract Source Code option using the following menu navigation: Environment → Quality Logs → Extract Source Code.

*Note: For more information on the Extract Source Code error log refer to Chapter 3 in the Natural Engineer Reporting manual.*

Objects that are no longer required in the Repository are best removed from the source library. If the Repository has been loaded, these can also 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 this manual for more information.*

## Load Repository

---

The next step after Extracting the source code is to Load Natural Engineer's Repository from the output file of the Extract process. All information, cross-references, and relationships are built into the Repository for interrogation, reporting, diagramming and further processing.

### Related Processes

After loading of the Repository, check the Missing Objects Report option and selectively Extract and Load those objects after adding them to the application library. Alternatively you can execute the Extract Missing Object option after adding them to the appropriate library.

The Missing Natural Objects report is accessed via the Quality Logs option. You may add new or changed objects to the loaded Repository using the Load Repository option.

## Extract and Load

---

The Extract & Load option allows you to perform both the Extract and Load operations in a single step.

Any missing objects or objects with errors identified during the Extract process will be reported, and can be added selectively to the Repository after completion of the Load process.

## Extract, Load and Impact

---

This option allows you to perform the Extract, Load and Impact Analysis processes in a single step.

Any missing objects or objects with errors identified during the Extract process will be reported, and can be added selectively to the Repository after completion of the Load process.

## Extract Missing Objects

---

The Extract Missing Objects option extracts the missing called objects as identified in the Missing Natural Objects report. If an object is a data area or copycode, it will also extract the calling object (unless it has been extracted in the same execution).

*Note:*

- *A maximum of 200 missing objects will be extracted using this process.*
- *Missing DDMs will not be extracted using this process.*

The process to follow for extracting the most complete application is:

1. Extract and Load an application into Natural Engineer.
2. Check the Missing Natural Objects report to identify any missing objects.
3. Copy those missing objects to the Natural library or Steplib library defined to Natural Engineer.
4. Run the Extract Missing objects option.

If an object is displayed on the Missing Natural Objects report but not copied as part of step 3, then the Extract process will identify this as an error with the message:

- NO SUCH OBJECT EXISTS IN DIRECTORY

If a DDM is identified as missing then the DDM should be located and copied to the application library, a defined steplib or the SYSTEM library. All objects that reference the missing DDM need to be re-extracted. Either identify the missing object by running the DDMs referenced by Object report and then selectively extracting and loading those objects, or re-execute the Extract and Load processes for all objects in the application.



# APPLICATION MANAGEMENT

## Chapter Overview

---

This chapter describes the Application Management options available from the Environment menu.

Application Management provides the facility to manage supplementary information on the objects within applications held on the Repository.

The Application Management option is accessed using the following menu navigation: Environment → Application Management.

The Application Management option provides a sub menu of facilities for the user to add, remove and update additional information for individual objects.

The topics covered in this chapter:

1. [Soft Links](#)
2. [User Documentation](#)

## Soft Links

---

The Soft Links option allows you to manually update the Repository with information regarding the linking between objects. A Soft Link is one where a link between two objects has been defined using an alphanumeric variable rather than a literal constant.

For example;

1) A link to a subprogram using a literal constant: -

**0090 CALLNAT 'SUBPROG1' #PARAMETER-GROUP**

2) A link to a subprogram using an alphanumeric variable: -

**0250 MOVE 'SUBPROG1' TO #CALL-NAME**

**0260 CALLNAT #CALL-NAME #PARAMETER-GROUP**

This is what Natural Engineer recognizes as a Soft Link.

Natural Engineer will provide a list of objects within an application that contain Soft Links statements. Only objects containing Soft Links will be available for selection from the Soft Link Maintenance screen. That is to say, any objects that use literal constants only will not be shown in the object list on this screen.

A single object may contain one or more Soft Link statements. Each statement can be selected to specify the object name reference for the Soft Link. Up to a maximum of 1008 object names may be specified per statement.

Once all the Soft Links have been specified, they will provide the cross-reference information into the inter-object tracing function within the Analysis process.

Soft Links can be saved / opened using a PC text format file enabling common Soft Links to be applied to a complete application as one single operation. Soft Links can be easily removed either from the complete application, a range of objects, one single object or from a single statement line.

A Soft Links report is available to view instantly all the details of which Soft Links have been specified, for each statement line within each object, within the application. This report can be viewed in Reporter, on the screen or via an Excel spreadsheet.

## Objects Soft Link Maintenance Window

The Soft Links option is accessed using the following menu navigation: Environment → Application Management → Soft Links. When this option is selected, the Objects Soft Link Maintenance screen is displayed.

The following Figure 3-1 illustrates the Objects Soft Link Maintenance screen.

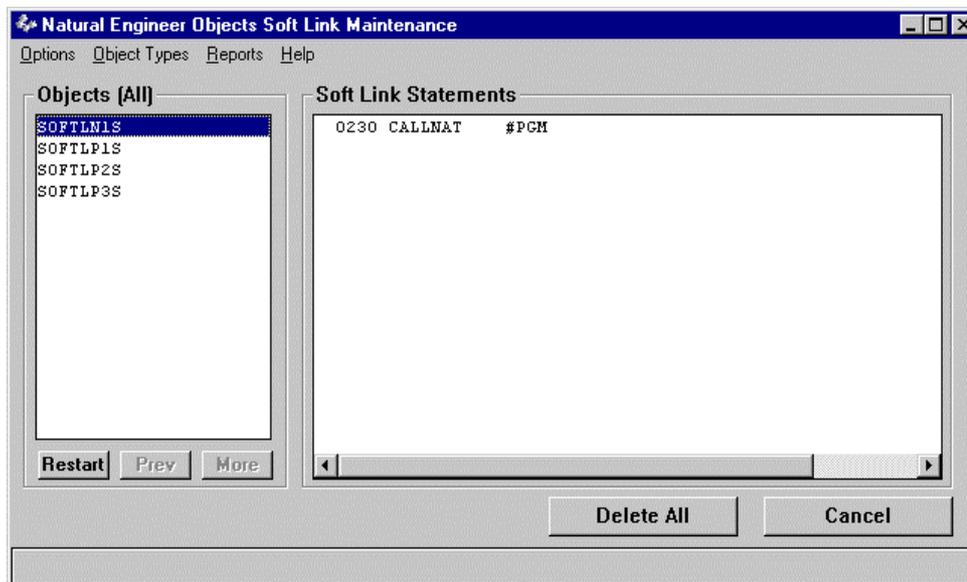


Figure 3-1 Objects Soft Link Maintenance screen

MENU ITEMS	OPTIONS	DESCRIPTION
<b>Options</b>	Provides various Soft Links handling options.	
	<b>Open Soft Link File</b>	Open and read in an existing Soft Link file. These files will have file extension of '.SFT' and will contain previously saved/modified Soft Link records. When opened, they will apply the contents to the current application.

# 3

## Natural Engineer Application Management

MENU ITEMS	OPTIONS	DESCRIPTION
	<b>Save Soft Link File</b>	<p>Save the current set of Soft Links specified for the current application. The file is saved with a file extension of '.SFT'. By default, this file will be saved to the data folder where Natural Engineer is installed.</p> <p>For Example: C:\PROGRAM FILES\SOFTWARE AG\NEE\V431\DATA</p>
	<b>Delete All Soft Links</b>	<p>Deletes all Soft Links for the objects available in the Objects list box. The list of objects is controlled by the Object Types menu option.</p> <p>For Example: If objects of type 'Program' are listed, then all the Soft Links for program objects only within the application will be deleted. Any maps, copycodes, subprograms, subroutines or help routines will not have their Soft Links deleted.</p>
	<b>Exit</b>	Will close the Objects Soft Link Maintenance screen and return you back to the main Natural Engineer screen.
<b>Object Types</b>		Allows you to select the Object Types to be listed.
<b>Reports</b>		Allows selection of the Soft Link reports.
	<b>Soft Links Report</b>	<p>This report will show the Soft Link details for all the objects containing Soft Links.</p> <p><i>Note: For more information on the Soft Links report refer to the Natural Engineer Reporting manual.</i></p>
	<b>Review Soft Link Error Log</b>	This reports any errors that may have occurred when using the Open Soft Link File option. The details shown will be for the last use of this option for the application.
<b>Help</b>		Invokes the Soft Links help.

SCREEN ITEMS	DESCRIPTION
--------------	-------------

<b>Objects</b>	Lists all the objects in the application that have been loaded into the Repository, where Natural Engineer has recognized that they contain Soft Links. This list can be tailored to your requirements using the options in the Object Types menu.  Objects that have had Soft Links specified for them will be indicated with an '*' (asterisk) to the right hand side of the Object name.
<b>Soft Link Statements</b>	Displays all the statements within the selected object, for which Soft Links exist.

Column	Description
<b>1</b>	Statement Soft Link indicator showing whether a Soft Link has been specified for this statement line. Possible values: ' ' (blank)                               No Soft Links specified. '*' (asterisk)                         Soft Links specified.
<b>3-6</b>	The statement line number.
<b>8-17</b>	The Natural call type. For Example: CALLNAT.
<b>19-50</b>	The name of the alphanumeric variable used to make the call. For Example: #CALL-PGM-NAME.
<b>52-59</b>	The external object name. If the link statement is held in another physical object, for example: Copycode, then the object name of the Copycode is shown here.
<b>67-72</b>	Sequence number used by Natural Engineer.

BUTTON NAME	DESCRIPTION
-------------	-------------

<b>Restart</b>	Allows the Object List to be restarted from a particular object name.
<b>Prev</b>	Scrolls the object list to previous page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
<b>More</b>	Scrolls the object list forward one page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
<b>Delete All</b>	Deletes all the Soft Links within the current selected object.

### 3

#### Natural Engineer Application Management

<b>BUTTON NAME</b>	<b>DESCRIPTION</b>
<b>Cancel</b>	Exit the Objects Soft Links Maintenance screen and return you back to the main Natural Engineer screen.

*Note: For more information on the NATENG.INI file parameter LISTBOXMAX refer to Chapter 1 in the Natural Engineer Administration Guide for Windows manual.*

#### Add Soft Links Window

To specify the Soft Links for a statement line within an object simply select the statement line from the Objects Soft Link Maintenance screen and the Add Soft Links screen will be presented. On this screen it is possible to specify up to 1008 individual Soft Link names per statement line.

The following Figure 3-2 illustrates the Add Soft Links screen.

**Natural Engineer Add Soft Links**

File Help

**Call Details**

Application : SOFTLINK  
Object Name : SOFTLN1S  
Statement No. : 230  
Call Name : #PGM

**Soft Links**


Number of Links : 1 View: 1 to 48

OK Top Prev Next Delete All Cancel

Figure 3-2 Add Soft Links screen

MENU ITEMS	OPTIONS	DESCRIPTION
<b>File</b>		Provides various Soft Links handling options.
	<b>Search</b>	Allows you to search for a specific Soft Link name for the currently selected statement line. This function will search for exact names only.
	<b>Exit</b>	Will close the Add Soft Links screen and return you back to the main Natural Engineer screen.
<b>Help</b>		Invokes the Soft Links help.

SCREEN ITEMS	DESCRIPTION
<b>Call Details</b>	<b>Application</b> The name of the Application.
	<b>Object Name</b> The name of the object currently selected.
	<b>Statement No.</b> The statement line number.
	<b>Call Name</b> The name of the alphanumeric variable used to make the call. For Example: #CALL-PGM-NAME.
<b>Soft Links</b>	This section of the screen allows for the specification of the Soft Links names to be used for a statement line.
	<b>Soft Links</b> Soft Link Names can be specified as required. Each page view caters for up to 48 entries, with a maximum of 1008 entries per statement allowed.
	<b>Number of Links</b> Shows the number of Soft Link names specified for the current statement line.
	<b>View</b> Shows the range of Soft Link names being displayed. For Example: View: 1 to 48 indicates that you are looking at Soft Links 1 to 48.

BUTTON NAME	DESCRIPTION
<b>OK</b>	Saves the Soft Links that have been specified. A sort will be performed, sorting the Soft Link names into ascending alpha sequence. Duplicate entries will be removed.
<b>Top</b>	Scrolls the Soft Links list to start from the first entry.
<b>Prev</b>	Scrolls the Soft Link list back one-page view.
<b>Next</b>	Scrolls the Soft Link list forward one page view.
<b>Delete All</b>	Deletes all the Soft Links for the current Statement line number.
<b>Cancel</b>	Exit the Add Soft Links screen and return you back to the Objects Soft Link Maintenance screen.

## Soft Link File

It is possible to re-use a set of Soft Links across more than one application. This level of flexibility allows quick and easy Soft Links specification, if 'call' processing is common across several applications.

In order to achieve this, Soft Links can be saved to a PC file with a file extension of '.SFT'. By default this file will be saved to the data folder where Natural Engineer is installed. For Example: **C:\PROGRAM FILES\SOFTWARE AG\NEE\V431\DATA.**

Once a Soft Link file has been saved it can be modified outside of Natural Engineer and re-used as desired. This can be done using a common text editor such as Notepad.

Each record in the Soft Link PC file is a total of 70 bytes in length. A record is written for each Soft Link name entered per statement number per object selected. To illustrate this, lets assume an application has 2 Soft Link call statements at lines 0230 and 0990 in one single object. If you set up 10 Soft Link objects per statement and then save the Soft Link PC file, you will have 20 records (1 object \* 2 statement lines \* 10 Soft Link names).

The record format used in the Soft Links PC file:

Field Name	Format / Length	Description
Record Option	A1	Option dictates whether the Soft Link Data for this record is to be added or deleted. Valid values are: <b>A</b> – Add <b>D</b> – Delete.
Object Name	A8	Name of the object containing the Soft Link call(s). Valid values are: <b>Full object name up to 8 bytes.</b> If the object name is less than 8 bytes then the outstanding bytes must be padded out with spaces to maintain position within the record. <b>Wildcard '*' (asterisk)</b> Indicate that this Soft Link record can be applied to all objects. This must be padded out with 7 spaces to maintain position within the record.

# 3

## Natural Engineer Application Management

Field Name	Format / Length	Description
Object Type	A1	<p>The object type of the calling object. Valid values are:</p> <p><b>P</b> Programs</p> <p><b>M</b> Maps</p> <p><b>C</b> Copycodes</p> <p><b>N</b> Subprograms</p> <p><b>S</b> Subroutines</p> <p><b>H</b> Help routines</p> <p>* Wildcard for ALL object types.</p>
External Object name	A8	<p>The external object name. If the Soft Link call statement is held in another physical object. For Example: Copycode, then the object name of the Copycode is shown here.</p> <p>Valid values are:</p> <p><b>Full external object name up to 8 bytes.</b></p> <p>If the external object name is less than 8 bytes then the outstanding bytes must be padded out with spaces to maintain position within the record.</p> <p><b>Wildcard '** (asterisk)</b></p> <p>Indicate that this Soft Link Record can be applied to all external objects. This must be padded out with 7 spaces to maintain position within the record.</p>
Statement Number	A4	<p>The statement line number using zeroes to pad the number out to 4 bytes.</p> <p>Valid values are:</p> <p><b>0010, 0130, 2345 etc.</b></p> <p>This will apply to the exact statement number specified.</p> <p><b>Wildcard '** (asterisk)</b></p> <p>Indicate that this Soft Link Record can be applied to all statement numbers. This must be padded out with 3 spaces to maintain position within the record.</p>

Field Name	Format / Length	Description
Sequence Number	A8	<p>Sequence number used by Natural Engineer.</p> <p>Valid values are:  <b>00000002, 00000022 etc.</b></p> <p>This will apply to data matching the sequence number within the Repository. If specified it must be padded with zeroes to the full 8 byte length.</p> <p><b>Wildcard '*' (asterisk)</b></p> <p>Indicate that this Soft Link Record can be applied to all sequence numbers. This must be padded out with 7 spaces to maintain position within the record.</p>
Call Variable name	A32	<p>The name of the alphanumeric variable used to make the call. For Example: #CALL-PGM-NAME.</p> <p>Valid values are:  <b>#CALL-PGM-NAME, #PROGRAM etc</b></p> <p>If the call variable name is less than 32 bytes then the outstanding bytes must be padded out with spaces to maintain position within the record.</p> <p><b>Wildcard '*' (asterisk)</b></p> <p>Indicate that this Soft Link Record can be applied to all call variable names. This must be padded out with 31 spaces to maintain position within the record.</p>
Soft Link name	A8	<p>The Soft Link name of the called object. This must be the actual object name up to 8 bytes long.</p> <p><b>Wildcard '*' (asterisk)</b></p> <p>This can be only be used for a record option of 'D' (delete) and will delete all Soft Link object names.</p>

### Examples of modified Soft Links PC files and what actions they will perform.

The following examples illustrate the contents of a single Soft Link PC file record required to carryout each requirement.

1. Add the Soft Link object name SOFTLN2S to object SOFTLN1S, which has an object type of subprogram, a Soft Link call at statement line number 0230, sequence number of 21 and uses the alphanumeric variable #PGM in the Soft Link call.

Soft Link File record:

ASOFTLN1SN	02300000021#PGM	SOFTLN2S
------------	-----------------	----------

2. Add the Soft Link object name SOFTLN2S to all objects, which have an object type of subprogram, a Soft Link call at statement line number 0230, any sequence number and uses the alphanumeric variable #PGM in the Soft Link call.

Soft Link File record:

A*	N	0230*	#PGM	SOFTLN2S
----	---	-------	------	----------

3. Add the Soft Link object name SOFTX to all objects, all object types, Soft Link calls at any statement line number, any sequence number and any alphanumeric variable in the Soft Link call. Basically, add Soft Link object name to all identified Soft Link objects within the application.

Soft Link File record:

A*	**	*	*	*	SOFTX
----	----	---	---	---	-------

4. Delete the Soft Link object name SOFTLN2S from object SOFTLN1S, which has an object type of subprogram, a Soft Link call at statement line number 0230, sequence number of 21 and uses the alphanumeric variable #CALL-PROGRAM-NAME in the Soft Link call.

Soft Link File record:

DSOFTLN1SN	02300000021#CALL-PROGRAM-NAME	SOFTLN2S
------------	-------------------------------	----------

5. Delete all Soft Links from the application.

Soft Link File record:

D*	**	*	*	*	*
----	----	---	---	---	---

## Soft Link Error Log

This is available when using the Open Soft Link File option from the Options menu on the Objects Soft Link Maintenance screen. When the Open Soft Link PC file is used, it reads in the Soft Link records and applies the Soft Link data to the application. This may be adding Soft Links, deleting Soft Links, or a combination of both.

If the Soft Link process experiences any anomalies within the Soft Link records, it writes out entries to the Soft Link Error Log for each erroneous Soft Link record, and continues with the next one until it gets to the end of the file.

The error log details are written to a work file saved in the data folder where Natural Engineer is installed (E.g., C:\PROGRAM FILES\SOFTWARE AG\NEE\V431\DATA). The file name is aaaaaaa.ESL, where 'aaaaaaa' is the application name.

Upon completion a pop-up window shows the status of the Soft Link PC file processing that has just been completed.

The following Figure 3-3 illustrates the Soft Link PC file information pop-up screen.

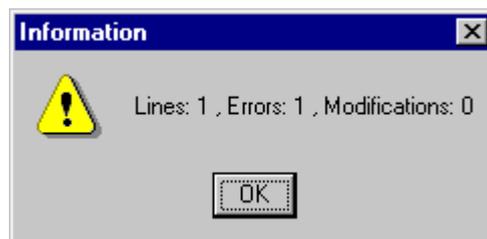


Figure 3-3 Soft Link PC file information pop-up screen

SCREEN ITEMS	DESCRIPTION
<b>Processing information message</b>	<p>Shows the processing that has been carried out using the Open Soft Link File option.</p> <p>It consists of three components:</p> <p><b>Lines:</b> Shows the number of Soft Link records processed from the opened Soft Link PC file. For Example:</p> <p>If the Soft Link contains 20 records then 'Lines: 20' would be displayed.</p> <p><b>Errors:</b> Shows the number of Soft Link records that are in error. These records will not apply any processing to the application data during the process.</p> <p><b>Modifications:</b> Shows the number of Soft Link records that have successfully been applied to the application.</p> <p>The number of Lines = number of Errors + number of Modifications.</p>

BUTTON NAME	DESCRIPTION
<b>OK</b>	<p>This button is used to escape the Soft Link PC file processing information screen and will result in one of two actions:</p> <p>If No Errors have occurred, i.e., Errors: 0, then the Objects Soft Link Maintenance screen is displayed with the Object List refreshed.</p> <p>If Errors have occurred, then the Quality Log screen will be displayed showing the details for each error.</p>

The Soft Link Error Log file can be viewed using the Quality Log screen accessed from the Objects Soft Link Maintenance screen using menu option Report → Review Soft Link Error Log.

This screen is automatically presented immediately after an Open Soft Link file run has been executed if there were any errors present, i.e., when 'OK' button is used from the Soft Link PC file information pop-up screen.

The following Figure 3-4 illustrates the Quality Log screen showing the Soft Link PC File errors.

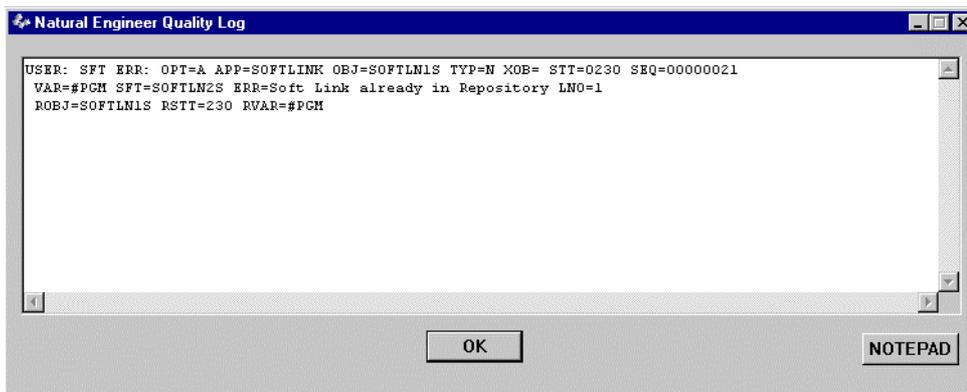


Figure 3-4 Quality Log screen showing the Soft Link PC File errors

SCREEN ITEMS	DESCRIPTION
--------------	-------------

<b>Error details</b>	For each record found to be in error, a three-line entry will be shown in the Quality Log error details.
----------------------	--

BUTTON NAME	DESCRIPTION
-------------	-------------

<b>OK</b>	Exit the Quality Log screen and return back to the Objects Soft Link Maintenance screen.
<b>NOTEPAD</b>	Opens the text editor Notepad using the Soft Link error file aaaaaaaa.ELS, where 'aaaaaaa' is the application name. From here it is possible to print the error log.

## Soft Link Error Types

There are 5 types of error that can occur during the Open Soft Link File option:

### 1. Undefined option

This error is produced when an invalid Soft Link record option has been specified. Only 'A' (add) or 'D' (delete) are allowed.

Example: A Soft Link record has been detected where the record option is set to 'Y' (OPT=Y).

```
USER: SFT ERR: OPT=Y APP=SOFTLINK OBJ=SOFTLP1S TYP=P XOB= STT=0190 SEQ=00000014
VAR=#PROGRAM-NAME SFT=SOFTLP3S ERR=Undefined option LNO=3
ROBJ=SOFTLP1S RSTT=120 RVAR=#PROGRAM-NAME
```

### 2. Cannot add \* as a Softlink

This error is produced when a wildcard '\*' (asterisk) has been specified for the Soft Link name, when the Soft Link record option is set to 'A' (add).

Example: A Soft Link record has been detected where the record option is set to 'A' (OPT=A) and the Soft Link name is set to '\*' (SFT=\*).

```
USER: SFT ERR: OPT=A APP=SOFTLINK OBJ=SOFTLP1S TYP=P XOB= STT=0120 SEQ=00000007
VAR=#PROGRAM-NAME SFT=* ERR=Cannot add * as a Softlink LNO=2
ROBJ=SOFTLP1S RSTT=120 RVAR=#PROGRAM-NAME
```

### 3. Soft Link already in Repository

This error is produced when a Soft Link name is being added when it already exists on the Repository.

Example: A Soft Link record has been detected where the Soft Link name is set to 'SOFTLN2S' (SFT=SOFTLN2S), for object SOFTLP1S (OBJ=SOFTLP1S), an object type of 'N' (TYPE=N), at statement line number 0230 (STT=0230) and with sequence number 00000021 (SEQ=00000021). On the Repository the Soft Link name SOFTLN2S already exists for this criteria.

```
USER: SFT ERR: OPT=A APP=SOFTLINK OBJ=SOFTLN1S TYP=N XOB= STT=0230 SEQ=00000021
VAR=#PGM SFT=SOFTLN2S ERR=Soft Link already in Repository LNO=1
ROBJ=SOFTLN1S RSTT=230 RVAR=#PGM
```

#### 4. Soft Link Not in Repository

This error is produced when a Soft Link name is being deleted, and the Soft Link object is not found in the Repository. This indicates that either no Soft Link name has been specified for the object, object type, statement number, Soft Link variable name in the Soft Link record, or, a Soft Link name has been found but does not match the one on the Soft Link record.

Example: A Soft Link record has been detected to delete a Soft Link, where the Soft Link name is set to 'SOFTLN1S' (SFT=SOFTLN1S), for object SOFTLP1S (OBJ=SOFTLP1S), an object type of 'P' (TYP=P), at statement line number 0300 (STT=0300) and with sequence number 00000027 (SEQ=00000027). On the Repository the Soft Link name SOFTLN1S does not exist for this criteria.

```
USER: SFT ERR: OPT=D APP=SOFTLINK OBJ=SOFTLP1S TYP=P XOB= STT=0300 SEQ=00000027
VAR=#PROGRAM-NAME SFT=SOFTLN1S ERR=Soft Link Not in Repository LNO=4
ROBJ=SOFTLP1S RSTT=300 RVAR=#PROGRAM-NAME
```

#### 5. Not Found

This error is produced when an object name has been specified for the Soft Link, which does not exist in the Repository for the specified application.

Example: A Soft Link record has been detected where the object 'SOFTLXXX' (OBJ=SOFTLXXX) has been specified to add a Soft Link name of 'SOFTLN2S' (SFT=SOFTLN2S). The application name used is 'SOFTLINK' (APP=SOFTLINK). Object SOFTLXXX does not exist in this application.

```
USER: SFT ERR: OPT=A APP=SOFTLINK OBJ=SOFTLXXX TYP=P XOB= STT=0100 SEQ=00000006
VAR=#PROGRAM-NAME SFT=SOFTLN2S ERR=Not found LNO=5
ROBJ=XX001P01 RSTT=10 RVAR=XX000G00
```

## User Documentation

---

The User Documentation option allows users to specify and save information on each object within an Application. This provides useful system documentation within Natural Engineer, complementing the source code information stored on the Repository. The information that can be stored for each object falls into two main categories:

### 1. Object Title

An object title can be specified. The default is the object name and in the case of CONSTRUCT or PREDICT generated objects, an indication of the code generator. The input is in free format style allowing a maximum of 70 characters to be input. Standard PC Copy, Paste and Cut functions are available.

### 2. Comments

These comments can be specified as required to provide detailed information on the object, such as: the function or functions performed, any database or flat file access, runtime considerations etc. The input is in free format style allowing a maximum of 50 lines of 70 characters per line. It is possible to import the 'real' object comments found at the top of an object (i.e., before the first Natural statement). Standard PC Copy, Paste and Cut functions are available.

The User Documentation for each object within an Application is stored on the Repository. It is also possible to save the User Documentation as a PC text format file, enabling common User Documentation to be applied across many Applications using the same objects, or re-applied in the event of a re-extract and re-load of an Application.

User Documentation can be removed from the complete Application, a range of objects or a single object.

## User Documentation Window

All the available User Documentation functions are controlled from the User Documentation screen.

The User Documentation option is accessed using the following menu navigation: Environment → Application Management → User Documentation. When this option is selected the User Documentation screen is displayed.

The following Figure 3-5 illustrates the User Documentation screen.

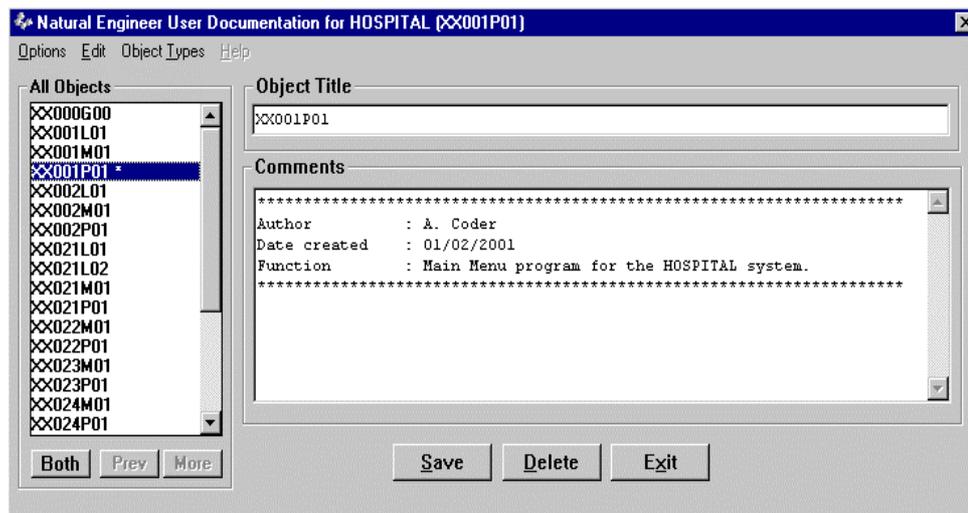


Figure 3-5 User Documentation screen

MENU ITEMS	OPTIONS	DESCRIPTION
<b>Options</b>		Provides various User Documentation handling options.
	<b>Change Start Position of Object List</b>	Allows the Object List to be restarted from a particular object.

MENU ITEMS	OPTIONS	DESCRIPTION
	<b>Import Object Comments</b>	<p>Provides a sub-menu list allowing various imports of User Documentation comments. The sub-menu options available are:</p> <p><b>Import Single from File</b></p> <p>Any previously saved User Documentation comments held in PC text format files with file extension .CMT that can be imported for the currently selected object. This option will only perform the import for one single object and is only available when an object has been selected.</p> <p><b>Import Multiple from File</b></p> <p>Provides the same function as Import Single from File, except that it can be applied to a range of selected objects.</p> <p><b>Import Single from Object Source</b></p> <p>Provides the ability to import the object comments from the selected object source code. The object comments are any comments found before the first Natural statement within an object. This option will only perform the import for one single object and is only available when an object has been selected.</p> <p><b>Import Multiple from Object Source</b></p> <p>Provides the same function as Import Single from Object Source, except that it can be applied to a range of selected objects.</p>
	<b>Export Object Comments</b>	<p>Provides a sub-menu list for saving User Documentation comments to a PC text format file. The sub-menu options available are:</p> <p><b>Export Single to File</b></p> <p>Provides the ability to export (save) User Documentation comments to a PC text format file with a file extension of .CMT. This option will only perform the export for one single object.</p> <p><b>Export Multiple to File</b></p> <p>Provides the same function as Export Single to File, except that it can be applied to a range of selected objects.</p>

MENU ITEMS	OPTIONS	DESCRIPTION
	<b>Delete Multiple User Documentation</b>	Delete multiple User Documentation comments from within an application. Objects can be selected as required and deletion is executed in a single operation.
	<b>Exit</b>	Will close the User Documentation screen and return you back to the main Natural Engineer screen.
<b>Edit</b>	Allows you to perform standard PC editing tasks with the User Documentation comments	
	The options available are:	
	<b>Options</b>	<b>Description</b>
	<b>Cut</b>	'Cut' out selected data (delete) to the clipboard.
	<b>Copy</b>	Copy the selected data to the clipboard.
	<b>Paste</b>	Paste clipboard data to selected input position.
<b>Object Type</b>	Allows you to select the Object Types to be listed. Valid object types are:	
	<b>All Objects (default)</b>	
	<b>Programs</b>	
	<b>Maps</b>	
	<b>Parameter Data Areas</b>	
	<b>Global Data Areas</b>	
	<b>Local Data Areas</b>	
	<b>Copycodes</b>	
	<b>Subprograms</b>	
	<b>Subroutines</b>	
	<b>Helproutines</b>	
	<b>Dialogs</b>	
	<b>Classes</b>	
<b>Help</b>	Invokes the User Documentation help.	

SCREEN ITEMS	DESCRIPTION
<b>All Objects</b>	<p>Lists the objects in the application that have been loaded into the Repository. The title at the top of the objects list box is dependent on the setting for object types; the default is 'All Objects'. For example: If object type were set to programs, the title would be 'Programs'.</p> <p>It is possible to navigate between the User Documentation screen and the Object Viewer, Object Reference, or Object Documentation screens by using the <b>right hand mouse button</b> on a selected object.</p>
<b>Object Title</b>	<p>Provides the input of an object title. The default is the object name and in the case of CONSTRUCT or PREDICT generated objects, an indication of the code generator. The input is in free format style allowing a maximum of 70 characters to be input. Standard PC Copy, Paste and Cut functions are available. Some examples:</p> <p><b>XX001P01</b></p> <p><b>CON001P1 (Generated by Construct)</b></p> <p><b>PREDP01 (Generated by Predict)</b></p> <p><b>ABC00G01 – Global Data Area</b></p>
<b>Comments</b>	<p>Provides the ability to input any required object comments to document the object. The input is in free format style allowing a maximum of 50 lines, each of which can have a maximum of 70 characters to be input. Standard PC Copy, Paste and Cut functions are available. For Example:</p> <p><b>This program calculates the Tax value required for each sale based on a table of tax rates. The base rate is 12.25%.</b></p>

BUTTON NAME	DESCRIPTION
<b>Both</b>	This button enables the object list to be refined to either show all objects whether they have User Documentation saved or not. The default for the screen is 'Both'. Once selected it rotates through the next 2 options described below.
<b>N Doc</b>	The object list shows only the objects that have no User Documentation saved.
<b>Doc O</b>	The object list shows only the objects that have User Documentation saved.

BUTTON NAME	DESCRIPTION
<b>Prev</b>	Scrolls the object list to previous page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
<b>More</b>	Scrolls the object list forward one page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
<b>Save</b>	Will save the specified User Documentation comments. This button will only be available when data is present in the Object Title.
<b>Delete</b>	Will delete the User Documentation comments for the current selected object only.
<b>Exit</b>	Will close the User Documentation screen and return you back to the main Natural Engineer screen.

*Note: For more information on the NATENG.INI file parameter LISTBOXMAX refer to Chapter 1 in the Natural Engineer Administration Guide for Windows manual.*

## Multiple Object Selection Window

The Multiple Object Selection screen is a common screen that is presented whenever an option is selected for Multiple objects. It provides the facility to select the objects required for the desired function.

The options that invoke the Multiple Object Selection screen are:

- Import Multiple from File.
- Import Multiple from Object Source.
- Export Multiple to File.
- Delete Multiple User Documentation.

The only variation to this screen is the title displayed in the title bar, which will contain reference to the option that has invoked this screen. All screen functionality is identical for all options.

Each of the variations of the Multiple Object Selection screen are now illustrated. The subsequent screen description is only shown once, but equally applies to all variations.



The following Figure 3-8 illustrates the Multiple Object Selection screen for the Export Multiple to File option.

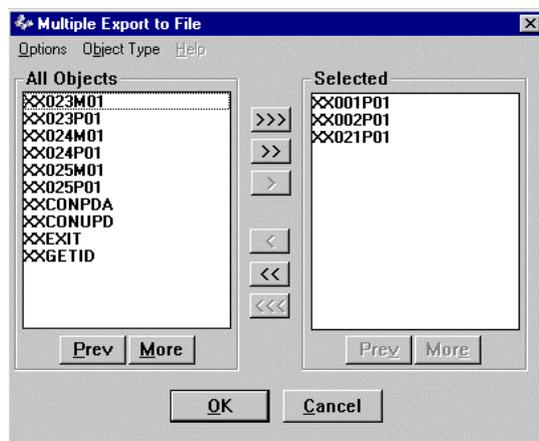


Figure 3-8 Multiple Object Selection screen for Export Multiple to File option

The following Figure 3-9 illustrates the Multiple Object Selection screen for the Delete Multiple User Documentation option.

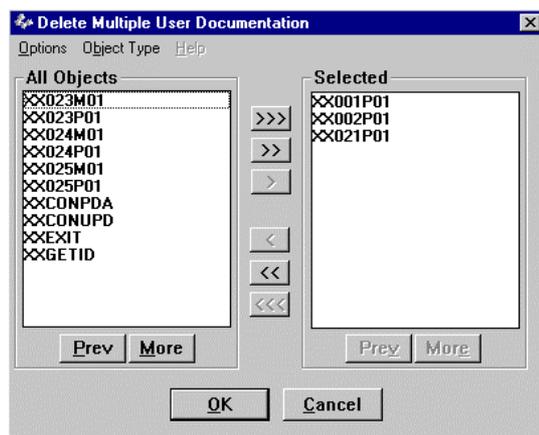


Figure 3-9 Multiple Object Selection screen for Delete Multiple User Documentation option

# 3

## Natural Engineer Application Management

MENU ITEMS	OPTIONS	DESCRIPTION
<b>Options</b>		Provides various User Documentation handling options.
	<b>Change Start Position of Object List</b>	Allows the Object List to be restarted from a particular object.
	<b>Exit</b>	Will close the Multiple Object Selection screen and return you back to the User Documentation screen.
<b>Object Types</b>		Allows you to select the Object Types to be listed. Valid object types are:
	<b>All Objects (default)</b>	
	<b>Programs</b>	
	<b>Maps</b>	
	<b>Parameter Data Areas</b>	
	<b>Global Data Areas</b>	
	<b>Local Data Areas</b>	
	<b>Copycodes</b>	
	<b>Subprograms</b>	
	<b>Subroutines</b>	
<b>Helproutines</b>		
<b>Dialogs</b>		
<b>Classes</b>		
<b>Help</b>		Invokes the Multiple Object Selection help.

SCREEN ITEMS	DESCRIPTION
<b>All Objects</b>	Lists the objects in the application that have been loaded into the Repository. The title at the top of the objects list box is dependent on the setting for object types; the default is 'All Objects'. For example: If object type were set to programs, the title would be 'Programs'.
<b>Selected</b>	Lists all the objects that have been selected for the current function.

BUTTON NAME	DESCRIPTION
<b>Prev</b>	Scrolls the object list to previous page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
<b>More</b>	Scrolls the object list forward one page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
>>>	Select all objects in the object list (when more than one page as set by the LISTBOXMAX parameter).
>>	Select all objects on this page in the object list.
>	Select all selected objects in the object list.
<	De-select all selected objects in the object list.
<<	De-select all objects on this page in the object list.
<<<	De-select all objects in 'Selected' list (when more than one page as set by the LISTBOXMAX parameter).
<i>Note: It is also possible to use the left mouse-button and double-click to select/de-select objects.</i>	
<b>OK</b>	Accept the selections made and invoke the appropriate function. Valid functions are: Import Multiple from File Import Multiple from Object Source Export Multiple to File Delete Multiple User Documentation
<b>Cancel</b>	Cancel out of the Multiple Objects Selection screen and return to the User Documentation screen.

*Note: For more information on the NATENG.INI file parameter LISTBOXMAX refer to Chapter 1 in the Natural Engineer Administration Guide for Windows manual.*

## Saving User Documentation Comments to a PC text format file

It is possible to save any User Documentation comments to a PC text format file. These files can then be used to 'import' the comments, updating the objects within an application in a single operation.

This can be used where common objects span across many applications and the User Documentation comments only need to be specified once, saved and the reused using either the Import Single from File or Import Multiple from File options from the Options drop-menu on the User Documentation screen.

Alternatively, comments can be saved for an application and then imported after any subsequent re-extract and load executions applied to the same application.

The saved files are located by default, in the Natural Engineer Data folder:

X:\PROGRAM FILES\SOFTWARE AG\NEE\V431\DATA (where X: is the drive where Natural Engineer has been installed).

The PC text format file holds records of up to 71 bytes in length for each of the Object Title and Comments detail lines present per object. The Format of each record is:

<b>Byte 1</b>	<p>One-byte control character indicating the type of record. Valid values are:</p> <p><b>H</b> Header information. This will be the Object name. There will be one header per object.</p> <p><b>T</b> The data present in the Object Title line. There will be 1 per object.</p> <p><b>C</b> The data present in the Comments lines. There can be 1-50 of these per object.</p>
<b>Bytes 2-71</b>	Contains the free format text input on the User Documentation screen.

When the file is saved, a file extension of .CMT is used. The file name defaults to one of two formats, depending on whether the data is being saved for a single object or multiple objects. The default formats are:

1. **Export Single to File.**

Default file name of **aaaaaaaa\_oooooooo.CMT** is presented, where **aaaaaaaa** is the application name, and **oooooooo** is the single object that has been selected.

For Example:

From the **HOSPITAL** application, object **XX001P01** has had the following comments specified:

Object Title = XX001P01

Comments =

```
*****  
Author           : A. Coder  
Date created    : 01/02/2001  
Function        : Main Menu Program for the HOSPITAL system.  
*****
```

These are to be saved to file.

The default file name will be **HOSPITAL\_XX001P01.CMT**.

### 3

### Natural Engineer Application Management

The following Figure 3-10 illustrates the save file screen for Export Single to File function.

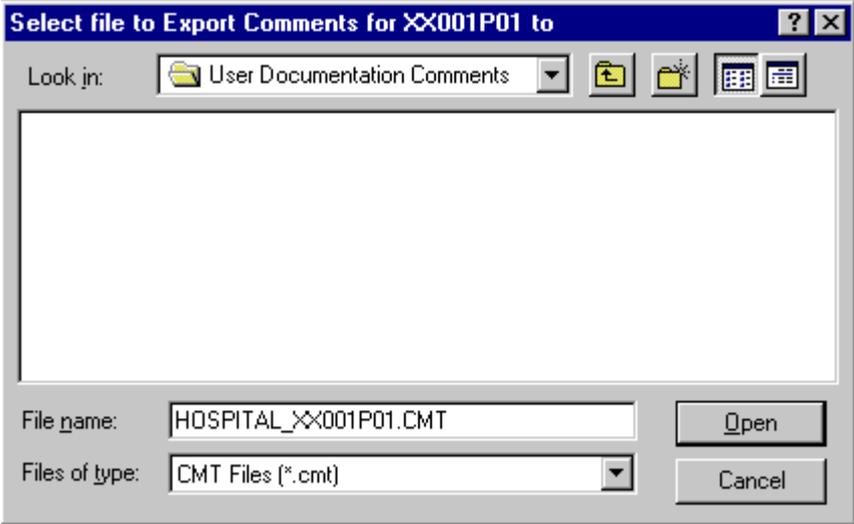


Figure 3-10 Save file screen for Export Single to File function.

The HOSPITAL\_XX001P01.CMT records are:

```
HXX001P01
TXX001P01
C*****
CAuthor      : A. Coder
CDate created : 01/02/2001
CFunction    : Main Menu Program for the HOSPITAL system.
C*****
```

## 2. Export Multiple to File.

Default file name of **aaaaaaa.CMT** is presented, where **aaaaaaa** is the application name.

For Example:

From the **HOSPITAL** application, objects **XX001L01**, **XX00M01** and **XX001P01** have had the following comments specified:

Object Title = XX001L01

Comments =

```
*****
Author           : A. Coder
Date created    : 01/02/2001
Function        : Main Menu Local Data Area for the HOSPITAL system.
*****
```

Object Title = XX001M01

Comments =

```
*****
Author           : A. Coder
Date created    : 01/02/2001
Function        : Main Menu Map for the HOSPITAL system.
*****
```

Object Title = XX001P01

Comments =

```
*****
Author           : A. Coder
Date created    : 01/02/2001
Function        : Main Menu Program for the HOSPITAL system.
*****
```

These are to be saved to file.

The default file name will be **HOSPITAL.CMT**.

### 3

#### Natural Engineer Application Management

The following Figure 3-11 illustrates the save file screen for Export Multiple to File function.

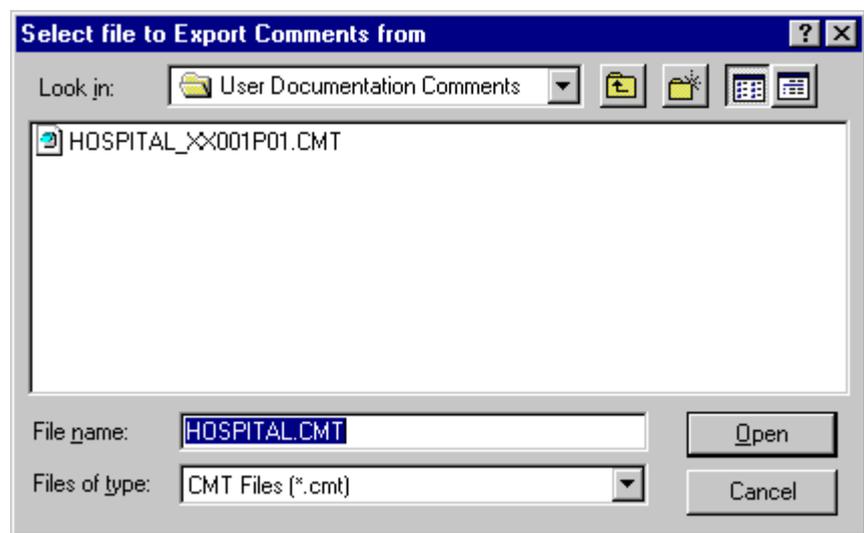


Figure 3-13 Save file screen for Export Multiple to File function.

The HOSPITAL.CMT records are:

```
HXX001P01
TXX001P01
C*****
CAuthor          : A. Coder
CDate created    : 01/02/2001
CFunction        : Main Menu Program for the HOSPITAL system.
C*****
HXX001M01
TXX001M01
C*****
CAuthor          : A. Coder
CDate created    : 01/02/2001
CFunction        : Main Menu Map for the HOSPITAL system.
C*****
HXX001L01
TXX001L01
C*****
CAuthor          : A. Coder
CDate created    : 01/02/2001
CFunction        : Main Menu Local Data Area for the HOSPITAL system.
C*****
```

In each of these two cases, the file name may be overwritten to meet individual site standards, but the file extension must not be changed from .CMT, otherwise the User Documentation import function will not locate the file.

## User Documentation Log Files

Several of the User Documentation functions provide log files detailing the processing that has been executed for that function. These Log files are available to view immediately after the function has completed.

Alternatively they may be viewed outside of Natural Engineer as they are saved in the Data folder: X:\PROGRAM FILES\SOFTWARE AG\NEE\V431\DATA (where X: is the drive where Natural Engineer has been installed).

The file is saved with a file extension of .LOG. The file name defaults to one of four formats depending on the User Documentation option used to generate them. These are described below.

1. **Import Multiple from File.**

The log file name used for this option is **aaaaaaa\_imp\_fil.LOG**, where **aaaaaaa** is the application name. This file will contain information on the import of multiple object comments for an application, from a .CMT file.

For Example:

From the **HOSPITAL** application, the Import Multiple from File option is used to import User Documentation comments using a saved file: HOSPITAL.CMT. This file contains the following records:

```

HXX001P01
TXX001P01
C*****

CAuthor          : A. Coder
CDate created   : 01/02/2001
CFunction       : Main Menu Program for the HOSPITAL system.
C*****

HXX001M01
TXX001M01
C*****

CAuthor          : A. Coder
CDate created   : 01/02/2001
CFunction       : Main Menu Map for the HOSPITAL system.
C*****

HXX001L01
TXX001L01
C*****

CAuthor          : A. Coder
CDate created   : 01/02/2001
CFunction       : Main Menu Local Data Area for the HOSPITAL system.
C*****

```

The objects **XX001L01**, **XX001M01** and **XX001P01** are selected using the Multiple Object Selection screen.

After the import has completed, the log file **HOSPITAL\_IMP\_FIL.LOG** will contain the following:

```

Mass Import from -
D:\Program Files\Software AG\NEE\v431\DATA\Hospital.cmt
01/02/2001 12:45:22 - XX001L01 imported
01/02/2001 12:45:22 - XX001M01 imported

```

### 3

#### Natural Engineer Application Management

01/02/2001 12:45:22 - XX001P01 imported

## 2. Import Multiple from Object Source

The log file name used for this option is **aaaaaaaa\_IMP\_OBJ.LOG**, where **aaaaaaaa** is the application name. This file will contain information on the import of multiple object comments from the Object source.

For Example:

From the **HOSPITAL** application, program objects **XX021P01**, **XX023P01**, **XX024P01** and **XX025P01** are selected to import multiple comments from object source. Using the menu option Options → Import Object Comments → Import Multiple form Object Source.

After the import has completed, the log file **HOSPITAL\_IMP\_OBJ.LOG** will contain the following:

```
Multiple Import from Object
01/02/2001 12:55:07 - XX021P01 saved
01/02/2001 12:55:07 - XX023P01 saved
01/02/2001 12:55:07 - XX024P01 saved
01/02/2001 12:55:07 - XX025P01 saved
```

## 3. Export Multiple to File.

The log file name used for this option is **aaaaaaaa\_EXP\_FIL.LOG**, where **aaaaaaaa** is the application name. This file will contain information on the export of multiple object comments to a .CMT file.

For Example:

From the **HOSPITAL** application, objects **XX001L01**, **XX001M01** and **XX001P01** have had the following comments specified:

Object Title = XX001L01

Comments =

```
*****
Author          : A. Coder
Date created    : 01/02/2001
Function        : Main Menu Local Data Area for the HOSPITAL system.
*****
```

# 3

## Natural Engineer Application Management

Object Title = XX001M01

Comments =

```
*****  
Author           : A. Coder  
Date created    : 01/02/2001  
Function        : Main Menu Map for the HOSPITAL system.  
*****
```

Object Title = XX001P01

Comments =

```
*****  
Author           : A. Coder  
Date created    : 01/02/2001  
Function        : Main Menu Program for the HOSPITAL system.  
*****
```

After the import has completed, the log file **HOSPITAL\_EXP\_FIL.LOG** will contain the following:

```
Mass Export to -  
D:\Program Files\Software AG\NEE\v431\DATA\Hospital.cmt  
01/02/2001 12:58:37 - XX001L01 exported  
01/02/2001 12:58:37 - XX001M01 exported  
01/02/2001 12:58:37 - XX001P01 exported
```

#### 4. Delete Multiple User Documentation

The log file name used for this option is **aaaaaaaa\_DEL.LOG**, where **aaaaaaaa** is the application name. This file will contain information on the deletion of User Documentation comments for an application.

For Example:

From the **HOSPITAL** application, objects **XX001L01**, **XX00M01** and **XX001P01** all have User Documentation comments saved for them. These comments will be deleted using Options → Delete Multiple User Documentation.

The objects **XX001L01**, **XX001M01** and **XX001P01** are selected using the Multiple Object Selection screen.

After the import has completed, the log file **HOSPITAL\_DEL.LOG** will contain the following:

```
Delete Multiple User Documentation
01/02/2001 13:05:40 - XX001L01 comments deleted
01/02/2001 13:05:40 - XX001M01 comments deleted
01/02/2001 13:05:40 - XX001P01 comments deleted
```

# APPLICATION INVENTORY

## Chapter Overview

---

This chapter provides a basic overview of the reporting available for the Application and Environment functions described in this manual.

Once an application has been defined and then extracted from the Natural application library and loaded into the Repository, Natural Engineer provides a series of reports that detail the structure and contents of the applications.

This chapter overviews the following Reporting options:

1. Quality Logs
2. Application Reports

*Note: This chapter does not describe the individual reports available in the Quality Logs and Application Reports options from the Environment menu. For more information on these reports refer to the Natural Engineer Reporting manual.*

## Quality Logs

---

The Quality Logs provides information on errors that have occurred during both the Extract and Load processes, as well as any missing or unused Natural objects from the Natural application being processed.

The Quality Logs can be accessed using the following menu navigation: Environment → Quality Logs.

## Quality Log Reports

The following list illustrates the Quality Log reports that are available:

- Extract Source Code
- Extract Source Code Summary
- Load Repository
- Missing Natural Objects
- Unused Natural Objects

The Extract Source Code and Extract Source Code Summary quality logs relate to the Extract process and are only available if there has been a problem during the Extract execution.

The following lists the Environment options that may trigger these quality logs:

- Extract Source Code
- Extract & Load
- Extract, Load & Impact
- Extract Missing Objects.

## 4

### Natural Engineer Application Management

The Load Repository quality log relates to the Load process and is only available if there has been a problem during the Load execution.

The following lists the Environment options that may trigger this quality log:

- Load Repository
- Extract & Load
- Extract, Load & Impact.

The Missing Natural Objects and Unused Natural Objects quality logs only become available after the Load process has been executed. These quality logs rely on the Load process to generate the object information required to identify missing and unused Natural objects.

The following lists the Environment options that may trigger these quality logs:

- Load Repository
- Extract & Load
- Extract, Load & Impact.

*Note: For more information on the Quality Logs refer to Chapter 3 in the Natural Engineer Reporting manual.*

## Application Reports

---

The Application Reports provide various levels of Analysis information on the application after it is loaded in the Repository (i.e., before Impact Analysis).

The Application Reports can be accessed using the following menu navigation: Environment → Application Reports.

### Application Reports

The following list illustrates the Application Reports that are available:

- Source Code Summary
- Object Summary
- Natural Keywords Summary
- Objects Referencing Objects
- Objects Referenced by Objects
- Objects Referenced by DDM fields
- External Objects Referenced by Objects
- Construct Models Referenced by Objects
- Natural Keywords Referenced
- DDMs Referenced
- DDMs Referenced by Objects
- DDMs Accessed by Objects
- Database Data Requirements
- Data Item Inventory
- Data Item Usage Inventory
- View Source Code

## 4

### **Natural Engineer Application Management**

The Application Reports option becomes available after the application has been loaded into the Repository.

*Note: For more information on the Application Reports refer to Chapter 3 in the Natural Engineer Reporting manual.*

# INDEX

## A

Application Preferences, 15  
Application Reports, 80

## D

Delete an Application, 11  
Delete Object, 12

## E

Extract and Load, 37  
Extract Missing Objects, 38  
Extract Selection Criteria, 22  
    Extract Selection Criteria window, 22  
    Specifying Object Names, 26  
    Valid Missing Objects, 27  
Extract Source Code, 35  
Extract, Load and Impact, 37

## L

Load Repository, 36

## O

Open an Application, 10

## P

Pre-Parser, 29  
    Error Log, 32  
    Load Balancing, 34  
    Object Statistics, 33  
    Source Code Pre-Parser window, 29

## Q

Quality Logs, 78

## S

Soft Links, 40  
    Add Soft Links window, 44  
    Objects Soft Link Maintenance window,  
        41  
    Soft Link Error Log, 51  
    Soft Link Error Types, 54  
    Soft Link file, 47

## U

User Documentation, 56  
    Log files, 72  
    Multiple Object Selection window, 61  
    PC text format file, 66  
    User Documentation window, 57

