Natural Engineer

Application Analysis & Modification for Windows

Version 8.4

October 2017

ADABAS & NATURAL

Manual Order Number: NEE84-023WIN

This document applies to Natural Engineer version 8.4 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

Acknowledgements

TABLE OF CONTENTS

ABOUT THIS MANUAL

Purpose of this manual

This manual contains the Application Analysis and Modification for Natural Engineer.

It describes the Analysis and Modification processes available to interrogate and maintain your Natural applications within Natural Engineer.

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

- How to create multiple Impact versions of Impact search criteria for an application, which allow multiple analysis of applications by more than one user.
- How to specify the Impact Criteria, including Impact Sets, Incremental Impact, the supported Natural keywords, Cobol keywords, JCL keywords, combination and miscellaneous keywords.
- How to specify Object Builder line range criteria.
- The Impact execution process and how to review the Impact results using the Impact Element Maintenance screen and the Impact reporting options.

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

- Specifying Modification Preferences to reference Text Logic Members (TLM) during Modification execution to include in-house written processes into the modified objects.
- Reviewing and re-specifying Modification parameters and settings prior to applying the actual Modification.
- The Modification execution process and how to review the Modification results.

Target Audience

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

Typographical Conventions used in this manual

The following conventions are used throughout this manual:

UPPERCASE TIMES	Commands, statements, names of programs and utilities referred to in text paragraphs appear in normal (Times) uppercase.
UPPERCASE BOLD COURIER	In illustrations or examples of commands, items in uppercase bold courier must be typed in as they appear.
< >	Items in angled brackets are placeholders for user-supplied information. For example, if asked to enter <file number>, you must type the number of the required file.
Underlined	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 the Application Analysis and Modification options of Natural Engineer in the following chapters:

Chapter	Contents
1	Describes the various Impact options available to create Impact versions of search criteria, the search criteria, the various search keywords available, the Impact execution process and how to review the Impact results.
2	Describes the various Modification options available to review and re-specify the Modification parameters and settings, execute Modification to modify the object source code within an application and how to review the Modified objects.
3	Describes the combination search keywords that are available when specifying Impact Criteria. These keywords allow various multiple sub-criteria to be specified and are used to handle more complex analysis within objects. An example of this would be the search keyword MVSNAT22TO31 that checks for Natural 2.2 to 3.1 migration compliance of applications.

Terminology

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

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

Analysis

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

Application

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

Browser

An Internet Browser such as Microsoft Internet Explorer or Netscape.

Category

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

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

Cobol

Abbreviation of Common Business Orientated Language. A programming language.

Cobol Link

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

Consistency

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

Database Access Definition

A collective term used to identify DDMs, SQL Tables or Predict User Views.

Data Item

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

Environment

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

Exception

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

Generated Code

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

Impact

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

Iteration

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

JCL

Job Control Language.

JCL object

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

Library

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

Modification

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

Refactoring

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

Soft Link

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

TLM

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

Type

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

Variable

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

Related Literature

The complete set of Natural Engineer manuals consists of:

1 **Natural Engineer Concepts and Facilities (NEE84-006ALL)**

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 (NEE84-008ALL)**

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

3 **Natural Engineer Installation Guide for Windows (NEE84-010WIN)**
Natural Engineer Installation Guide for Mainframes(NEE84-010MFR)
Natural Engineer Installation Guide for Unix (NEE84-010UNX)

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

4 **Natural Engineer Administration Guide (NEE84-040WIN)**
Natural Engineer Administration Guide (NEE84-040MFR)

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 (NEE84-020WIN)**
Natural Engineer Application Management (NEE84-020MFR)

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

6 **Natural Engineer Application Documentation (NEE84-022WIN)**
Natural Engineer Application Documentation (NEE84-022MFR)

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.

7 **Natural Engineer Application Analysis and Modification (NEE84-023WIN)**
Natural Engineer Application Analysis and Modification (NEE84-023MFR)

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.

Natural Engineer Application Analysis & Modification

8 Natural Engineer Application Restructuring (NEE84-024WIN)
Natural Engineer Application Restructuring (NEE84-024MFR)

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

9 Natural Engineer Utilities (NEE84-080WIN)
Natural Engineer Utilities (NEE84-080MFR)

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

10 Natural Engineer Reporting (NEE84-025ALL)

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] (NEE84-026MFR)

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

12 Natural Engineer Messages and Codes (NEE84-060ALL)

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

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

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

14 Natural Engineer Advanced Services (NEE84-017WIN)

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

1

IMPACT ANALYSIS PROCESSES

Chapter Overview

The Impact Analysis processes provide all the facilities to run analytical interrogation of each application held on the Repository. The results from these interrogations provide the basis for any desired modifications to the objects within an application.

Once Impact Analysis has been executed, there are various reporting options to view the results either online or using textual reports.

All the Impact Analysis processes are available from the Analysis menu.

The topics covered in this chapter:

Impact Criteria

Impact Criteria provides a user friendly interface to specify criteria to analyze the data and objects within the Natural Engineer repository. For the analysis of database and non-database fields or literals and constants, a selection of screens are provided to guide the user through the specification of the criteria and the execution of the analysis.

Advanced Options are available to specify more comprehensive and complex impact analyses including the ability to search for Definitions, System Functions, Natural, COBOL and JCL Keywords, and to specify combination search keywords such as ADJUST, CODE IMPROVEMENT, OBJECT BUILDER, NATRPC, MULTI SEARCH, MVSNAT22TO31, PORTING or REFACTORING.

Impact Criteria available are:

1. Literals & Constants
2. Fields
3. Advanced Options

Literals & Constants

The Literals & Constants option allows for the interrogation of any literal or constant within the chosen application. The information is provided interactively. If the amount of data exceeds certain limits then the LITERAL impact analysis option from Advanced Options should be used.

The Literals & Constants screen is accessed using the following menu navigation: Analysis ➔Impact Criteria ➔Literals & Constants.

The following Figure 1-1-1 illustrates the Literals & Constants screen.

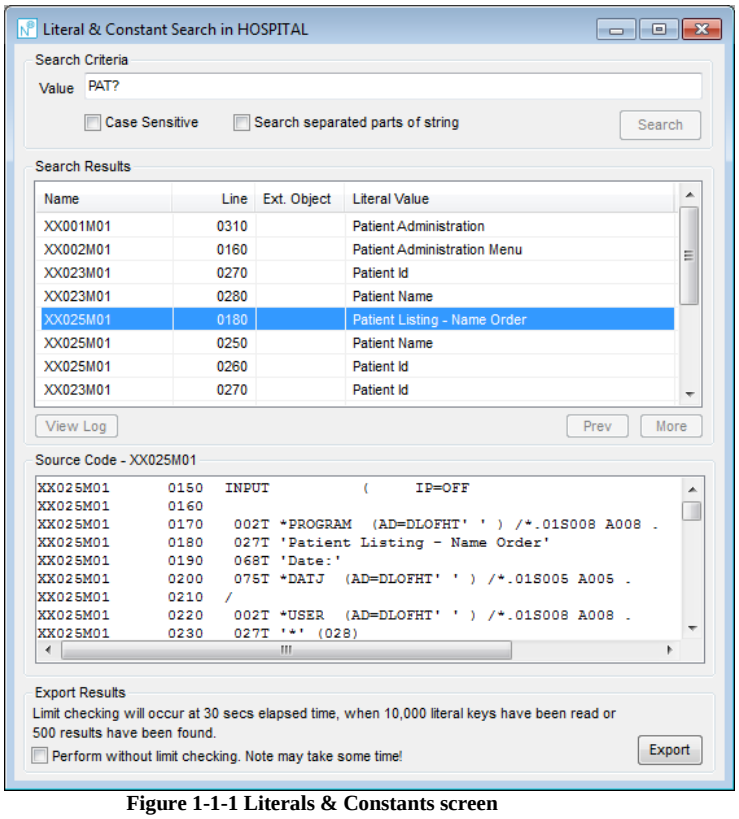

Figure 1-1-1 Literals & Constants screen

SCREEN ITEMS	DESCRIPTION
Search Criteria Group:	
Value	A literal search locates all the text or numeric constants within objects for the specified value. Values can be input using either complete values or part values using a '?' wildcard. For example: **Hello** Will search for all literal or constant strings matching Hello. **Hello?** Will search for all literal or constant strings starting with Hello. **?Hello** Will search for all literal or constant strings ending with Hello. **?Hello?** Will search for all literal or constant strings containing Hello.
Case Sensitive	The Case Sensitive option determines whether the search value entered is to be searched using the same upper and lower case format as used in literal or constant text strings within the objects. Therefore, if the search value was 'Hello' and there were two literal text strings 'Hello' and 'HELLO', if CASE was specified then only 'Hello' would be shown. If CASE was not specified then both would be shown.
Search Separated Parts of String	This treats the literal text string as individual words and the search value is validated against each word. For Example: 'ADABAS and Natural', with 'Search Separated Parts of String' specified will validate the three values as separate entries against the search value. Therefore, if the search value was ADABAS and there were two literal text strings 'ADABAS and Natural' and 'An ADABAS Database', if 'Search Separated Parts of String' was specified then the two literal text strings would be impacted. If 'Search Separated Parts of String' was not specified then neither would be impacted.
Search Results Group:	
Literal & Constant List	Lists all the Literals & Constants that match the specified search criteria. A context menu is available to navigate between the Literals & Constants screen and the Object Viewer screen or View Source by using the **right hand mouse button** on a selected literal or constant. The columns available are: **Name** The object name where the literal or constant has been found.

SCREEN ITEMS	DESCRIPTION	
	Line	The statement line number for the literal or constant.
	Ext. Object	The name of the object that contains the definition if the literal or constant is defined externally, such as in a COPYCODE.
	Literal Value	The actual value of the literal or constant.
Source Code	Displays all the statement references for the selected literal or constant including the object, the line number and the line of code.	
Export Results Group:		
Perform without limit checking	If 'Perform without limit checking is not set then limit checking occurs after 30 seconds elapsed time, when 10,000 keys have been read or 500 results have been found. A message will then be displayed allowing the user to continue the analysis or stop. If 'Perform without limit checking' is selected then the analysis will continue until completed.	

BUTTON NAME	DESCRIPTION
Search Criteria Group:	
Search	Will search the application for the chosen Search Criteria.
Search Results Group:	
View Log	Invokes the analysis log which contains any processing messages from the search.
Prev	Scrolls the results list to previous page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
More	Scrolls the results list forward one page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
Export Results Group:	
Export	Exports the search results to the default spreadsheet.

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

Fields

The Fields Impact Criteria provides a sequence of screens where criteria may be specified to search for database fields, non-database fields or both. The screens are shown sequentially guiding the user through the specification of the impact criteria.

The Fields screen is accessed using the following menu navigation: Analysis ➔Impact Criteria ➔Fields.

The following Figure 1-2-1 illustrates the initial Fields screen.

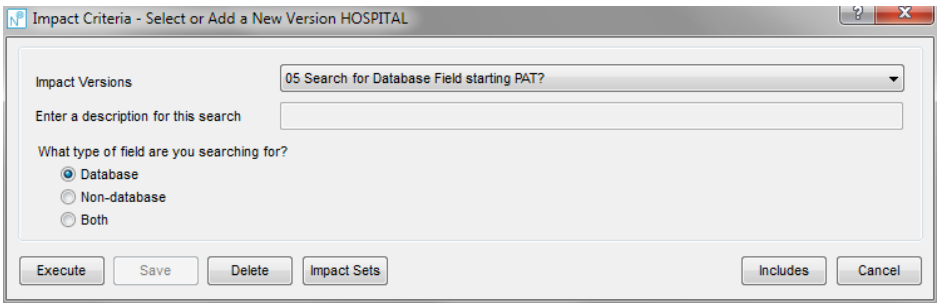

Figure 1-2-1 Initial Fields screen

SCREEN ITEMS	DESCRIPTION	
Impact Versions	Change the Impact version to review alternate Impact criteria for the application or add a new one	
Search Description	Will allow for the inclusion of a suitable comment to assist in identifying each version. Maximum of 80 characters allowed.	
	NB: This is only available when adding a new version.	
Type of Search	The type of search to be performed.	
	Options are:	
	Database	If selected only database fields will be searched for.
	Non-database	If selected only non-database fields will be searched for.
	Both	If selected both database and non-database fields will be searched for.

BUTTON NAME	DESCRIPTION
Execute	Will run the Impact Analysis process for the specified criteria. NB: This is only active if impact criteria have already been added.
Save	Will Save any Impact Criteria that have been added.
Delete	Will Delete any Impact Criteria that have been added.
Impact Sets	Invoke the Impact Sets process. Note: For more information on the Impact Sets screen refer to the section Impact Sets.
Cancel	Cancels the Fields Impact Criteria specification process and returns to the main Natural Engineer screen.
Includes	Invokes the Include Patterns fields screen.

Include Patterns Fields Screen

The Include Patterns Fields screen allow for the specification of Impact Criteria to be included in the search.

The following Figure 1-2-2 illustrates the Include Patterns Fields screen.

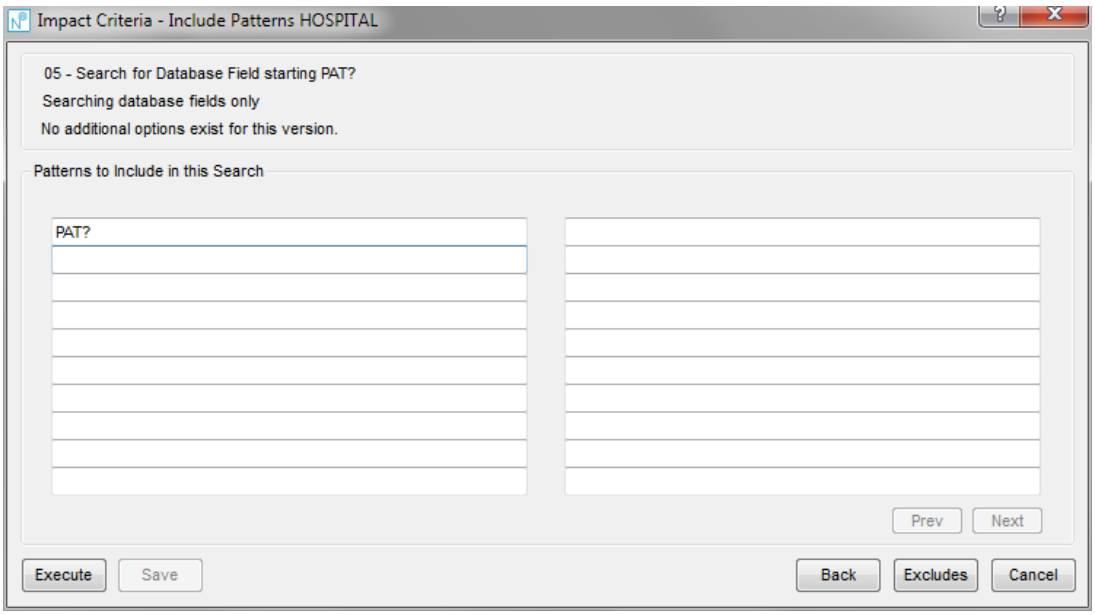

Figure 1-2-2 Include Patterns Fields screen

SCREEN ITEMS	DESCRIPTION	
Patterns	Allows for the specification of include patterns for the analysis.	
	PATIENT-ID	Will search for all items matching PATIENT-ID.
	PATIENT-ID?	Will search for all items starting with PATIENT-ID.
	? PATIENT-ID	Will search for all items ending with PATIENT-ID.
	? PATIENT-ID?	Will search for all items containing PATIENT-ID.
	NB: It is also possible to limit the search criteria by adding a format and length to the criteria value. This can be specified as an absolute format or a range.	
	For example:	
	'?PATIENT-ID? (A10)' will include any field with a name that includes FRED and with a format and length of A10.	
	'PATIENT-ID (N2-N9)' will include any field with a name of #PATIENT-ID and a format and length that falls between the range N2 to N9.	

BUTTON NAME	DESCRIPTION
Execute	Will run the Impact Analysis process for the specified criteria. *NB: This is only active if impact criteria have already been added.*
Save	Will Save any Impact Criteria present.
Back	Will return to the previous Initial Fields screen.
Cancel	Cancels the Include Patterns Fields screen and returns to the main Natural Engineer screen.
Excludes	Invokes the Exclude Patterns fields screen if you wish to specify any exclusions to the include criteria already specified.

Exclude Patterns Fields Screen

The Exclude Patterns Fields screen allow for the specification of Impact Criteria to be excluded from the search.

The following Figure 1-2-3 illustrates the Exclude Patterns Fields screen.

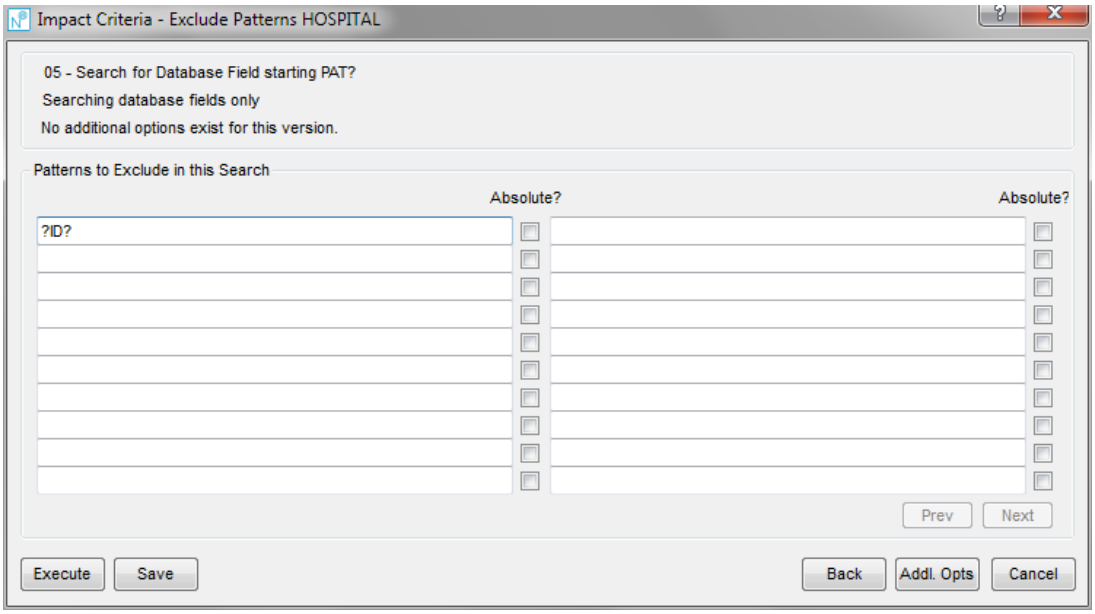

Figure 1-2-3 Exclude Patterns Fields screen

SCREEN ITEMS	DESCRIPTION	
Patterns	Allows for the specification of patterns to exclude	
	PATIENT-ID	Will exclude all items matching PATIENT-ID.
	PATIENT-ID?	Will exclude all items starting with PATIENT-ID.
	? PATIENT-ID	Will exclude all items ending with PATIENT-ID.
	? PATIENT-ID?	Will exclude all items containing PATIENT-ID.

SCREEN ITEMS	DESCRIPTION
Absolute Exclude	Setting this on will identify this as an Absolute Exclude. These criteria types will result in the Impact process excluding any fields that are an exact match for the value specified. These fields will be permanently excluded from the Impact process. If this is set off then these criteria types will result in the Impact process excluding parts of a field name, while the remainder of the field name is still searched for inclusion. These criteria types make use of the 'Byte-Lockout' process.

BUTTON NAME	DESCRIPTION
Execute	Will run the Impact Analysis process for the specified criteria. NB: This is only active if impact criteria have already been added.
Save	Will Save any Impact Criteria present.
Back	Will return to the previous Fields screen.
Cancel	Cancels the Exclude Patterns Fields screen and returns to the main Natural Engineer screen.
Addl. Opts	Invokes the Additional Options fields screen.

Additional Options Fields Screen

The Additional Options screen allow for the specification of additional information to control the Impact Criteria.

The following Figure 1-2-4 illustrates the Additional Options Fields screen.

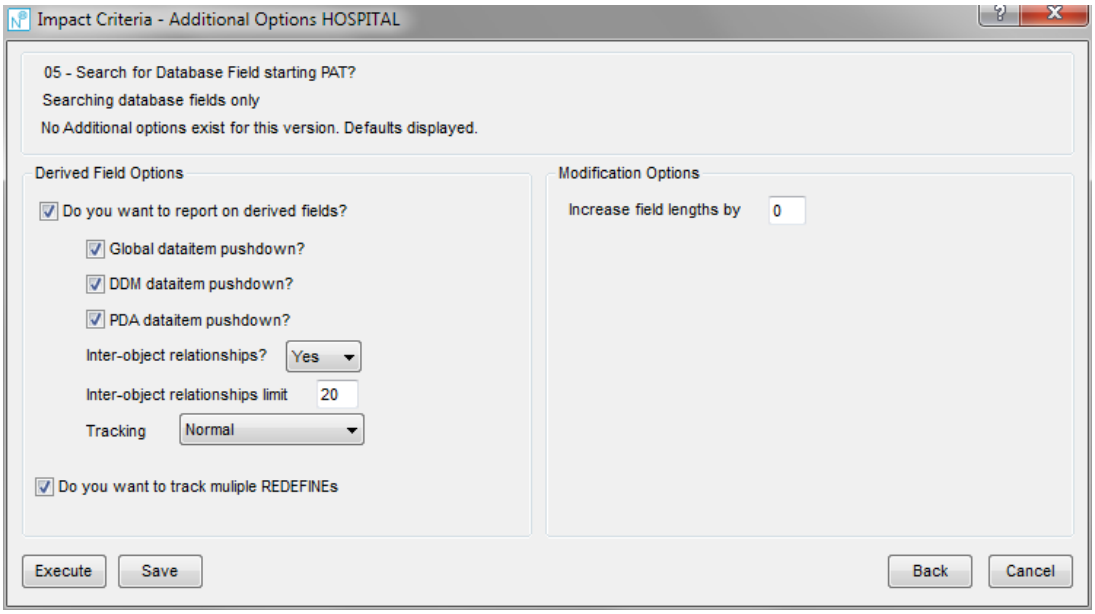

Figure 1-2-4 Additional Options Fields screen

SCREEN ITEMS	DESCRIPTION

Derived Field Options Group

Report on Derived Fields ?

Causes the Analysis to trace code identified, for further impact on other code. For example:

MOVE #A TO #B

Using the include pattern #A the above statement will be impacted as follows:

With Report on Derived Fields set off:

#A will be impacted as it is the specified item.

With Report on Derived Fields set on:

#A will be impacted as the specified item and #B will be impacted as it is being propagated by #A.

Note: Report on Derived Fields is synonymous with the Consistency function available for DBFILE, DBFIELD, DEFINITION and MULTI-SEARCH options of Advanced Options Impact Criteria.

Global dataitem pushdown?

Used when impacts have been made to Global Data Areas. If selected, Natural Engineer will track these fields, and derivations of these fields, until all possible impacts have been identified.

DDM dataitem pushdown?

Used when impacts have been made to DDMs. If selected, Natural Engineer will track these fields, and derivations of these fields, until all possible impacts have been identified.

PDA dataitem pushdown?

Used when impacts have been made to Parameter Data Areas. If selected, Natural Engineer will track these fields, and derivations of these fields, until all possible impacts have been identified.

Inter-Object Relationships ?

If selected data elements are tracked across object boundaries following the impact process.

Inter-Object Relationships Limit

This is the number of iterations that the Inter-Object Relationship process will track objects across object boundaries.

SCREEN ITEMS	DESCRIPTION	
	Tracking	Controls the tracking direction for a variable. If set to Forwards, a variable is tracked in a forward direction, showing all the derivatives being populated from the variable. If set to Backwards, a variable is tracked in a backward direction, showing where the variable and derivatives have been populated. If set to Normal, both the forward and backward directions will be shown. *For more information refer to section Forward/Backward Tracking.*
Track Multiple REDEFINEs ?	If selected, multiple redefines are tracked.	
Modification Options Group		
Increase field length by	Will increase the default length of an impacted field when Modification is executed. For example: Field #A is defined as (N2) and Increase field length is set to 2. After Modification, #A will have its length increased to (N4).	

BUTTON NAME	DESCRIPTION
Execute	Will run the Impact Analysis process for the specified criteria. NB: This is only active if impact criteria have already been added.
Save	Will Save any Impact Criteria present.
Back	Will return to the previous Fields Analysis – Exclude Patterns screen.
Cancel	Cancels the Additional Options Fields screen and returns to the main Natural Engineer screen.

Advanced Options

The Advanced Options allow for the specification of Impact Criteria which are used to identify instances of specified keywords and/or keyword values within object source code. The Impact Criteria can also be specified to hold replacement values, which are used by the Modification process.

Impact Criteria can be added, deleted, updated and reviewed. Each set of criteria will apply to a single Impact version, allowing for multiple sets of criteria to be specified for a single application.

The Impact Criteria can be saved to a PC text file, allowing them to be re-used across applications. These files will have a file extension of '.IRE'. By default these files will be saved to the data folder where Natural Engineer is installed.

The Advanced Options option uses a multi-purpose 'tabbed' screen to control all the Impact Version and Impact Criteria selection processes. Selecting the required tab will result in the display of the appropriate screen content for the process required.

There are three tab options available:

1. Impact Version

Provides a list of available Impact Versions for a selected application. From here, Impact Versions can be added, deleted, updated and reviewed.

2. Criteria Summary

Provides a summary list of the Impact Criteria for each Impact Version. From here, Impact Criteria can be added, deleted, updated and reviewed.

3. Criteria Detail

Provides the data entry facility to specify the required Impact Criteria. The data entry facility is controlled by the selection of Impact Types and Search Keywords. Display of the required data entry inputs will be relevant to the selection made.

The Advanced Options screen is accessed using the following menu navigation: Analysis ➔Impact Criteria ➔Advanced Options.

Each of the tab selections is described in the following sections.

Impact Version Tab Screen

The Impact Version tab screen will display the Impact Version content within the Impact Criteria screen. This allows you to save multiple sets of Impact Search Criteria for an application and provides a management facility to administer these search criteria for all options found in the Analysis and Modification menus.

The Impact Version tab screen allows you to add new versions, delete or modify existing versions and create Impact sets.

When deleting an Impact Version, then all the associated Impact search criteria and any Impact data that may have been generated during any previous Impact execution for this version, will all be deleted.

The Impact Version tab screen supports up to a maximum of 99 versions per application and is accessed using the Impact Version tab from the Impact Criteria screen.

The following Figure 1-3-2 illustrates the Impact Version tab screen.

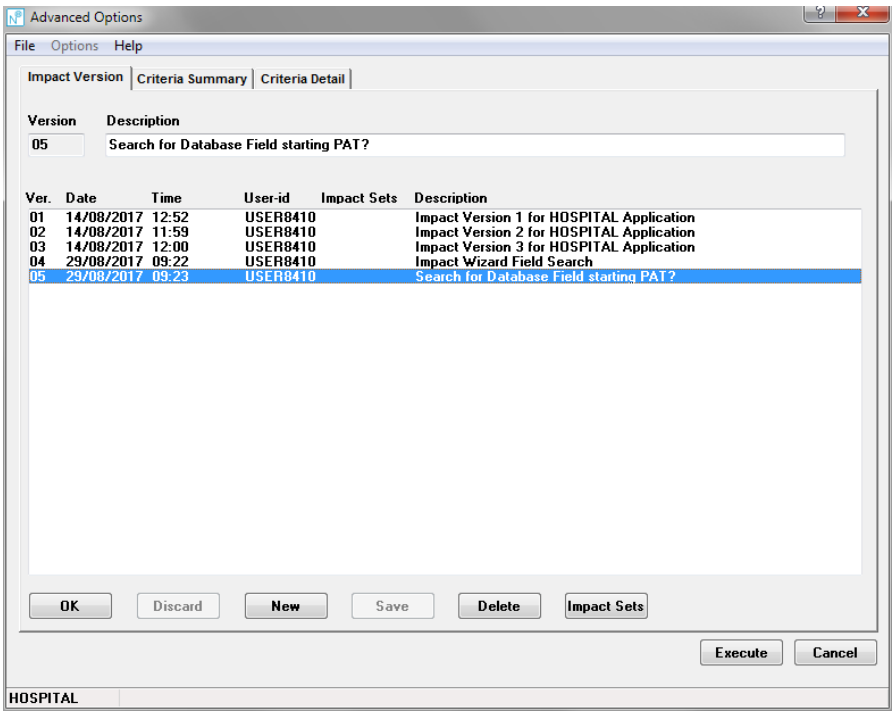

Figure 1-3-2 Impact Version tab screen

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Open…**	Not available for Impact Version.
	Save As…	Not available for Impact Version.
	Exit	Exit the Advanced Options screen and return back to the main Natural Engineer screen.
Options	Not available for Impact Version.	
Help	Invoke the Impact Version help.	

SCREEN ITEMS	DESCRIPTION
Version	The currently selected Impact Version number. If adding a new Impact Version, the next available version number.. This value is internally generated and cannot be modified. Any deleted version numbers will be reused. Up to 99 Impact Versions can be specified for each application.
Description	The description for the currently selected Impact Version number. A suitable comment can be added to assist in identifying each version. Maximum of 80 characters allowed. Existing descriptions can be modified by over typing here and then using the 'Save' button.
Ver.	The Impact Version number for the application.
Date	The date the Impact Version was modified.
Time	The time the Impact Version was modified.
User-id	The user identifier of the person who modified the Impact Version.
Impact Sets	Indicates if any Impact Sets have been specified for the Impact Version. Possible values are: **' ' (blank)** No Impact Sets. **Y** Impact Sets specified.
Description	Comment to describe the purpose of the Impact Version.

BUTTON NAME	DESCRIPTION
Impact Version tab screen:	
OK	Has dual functionality: **1. If selecting a version from the Advanced Options screen** This will accept the selected version making it current for any further Impact or Modification processes and open the Criteria Summary tab screen. **2. Changing the version from other Natural Engineer screens** This will accept the selected version making it current for any further Impact or Modification processes and return back to the calling Natural Engineer screen.
Discard	Reset any modified description changes. If the previous action was to add new Impact Version, then the Impact Version and description will be reset.
New	Assign the next available version number and reset the description for the new details required.
Save	Save the new or modified Impact Version details.
Delete	Delete the selected Impact Version details. The deleted version number will be reused the next time a new version is added. *Note: When a version is deleted, the criteria set and any impacted data from previous impact executions will also be removed.*
Impact Sets	Invoke the Impact Sets process. *Note: For more information on the Impact Sets screen refer to the section Impact Sets.*
Impact Criteria screen:	
Cancel	Has dual functionality: **1. If selecting a version from the Advanced Options screen** Cancel any selection made and return back to the main Natural Engineer screen. **2. Changing the version from other Natural Engineer screens** Cancel any selection made and return back to the calling Natural Engineer screen.

STATUS BAR ITEM	DESCRIPTION
The Advanced Options status bar is divided into 2 individual panes.	
Pane 1	Name of the selected application.
Pane 4	Any Advanced Options processing messages.

Impact Sets

Impact Sets are a means of creating a sub-set of objects within an application for a given Impact Version, which will allow impact to be executed against the specified Impact Set only. This means that impact can be executed against sets of objects within an application without the need for creating individual applications containing the sub-sets of objects required.

Objects are selected from the 'Select from' list (on left-hand side of dialog) and are transferred to the 'Selected' list (on right-hand side of dialog).

Once all selections have been made, the objects now form the Impact set, which will be used during any subsequent Impact Executions for this version. Only objects within the Impact Set will be impacted. All other objects in the application, but outside the Impact Set, will not be impacted.

If no Impact Set exists for an Impact version within an application, then all the objects in the application will be impacted.

Impact Sets Window

The Impact Sets screen is accessed using the 'Impact Sets' button from the Impact Version tab screen.

The following Figure 1-3-3 illustrates the Impact Sets screen.

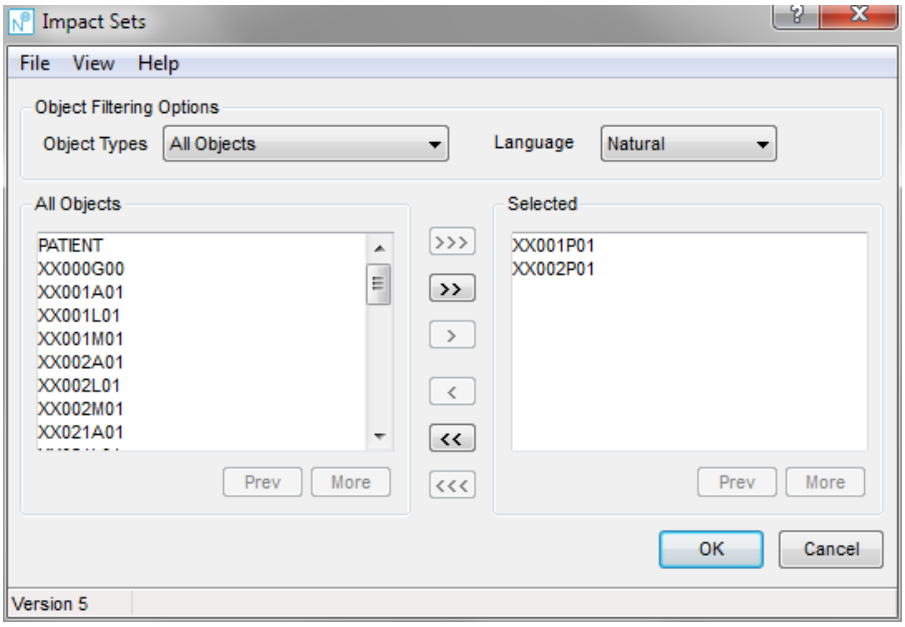

Figure 1-3-3 Impact Sets screen

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Exit**	Exit the Impact Sets screen and return back to the Impact Version tab screen.
View	**Change Start Position of Object List…**	Reposition the list of objects to start from a particular object name. The reposition value can be input using either a complete name or part name using an '*' (asterisk) wildcard. The reposition value is appended to the object list title to highlight the type of repositioning being applied.

MENU ITEMS	OPTIONS	DESCRIPTION
		Possible reposition values are:

Value	Result
' ' (blank)	Reposition to the top of the object list.
*****	Reposition to the top of the object list.
ABC*	Only show objects that are prefixed by 'ABC'.
XYZ	Reposition to the first object that either matches or is greater than 'XYZ' and then continue the object list from that point.

Help Invoke the Impact Sets help.

SCREEN ITEMS	DESCRIPTION

Object Filtering Options group:

Object Types Allows you to select the types of object to be listed. The types available are tailored to the objects present within your application.

Language Allows you to select the programming language of the objects to be listed.

Available selections are:

- **All**
- **Cobol**
- **JCL**
- **Natural**

Impact Sets screen:

Object List List of all the objects used by the currently selected application.

The list of objects can be tailored to your requirements using the options available in the Object Types and Language menus. Further refinement can be made using the option 'Change Start Position of Object List…' from the View menu.

The Object List title reflects the Object Types being listed and will append any reposition values that may have been specified.

Objects can be selected by using a double click with the **left hand mouse button**.

MENU ITEMS	OPTIONS	DESCRIPTION
Selected		Lists all the objects that have been selected for the current function. Objects can be de-selected by using a double click with the **left hand mouse button**.
	Type	This is a 2-byte value, which denotes the type of Impact. *Note: For more information on type of impact refer to the section Impact Types.*
	Line	The statement line number for the impact item within the selected object.
	Attribute	The format and length of the impact item if the item is a data item from a data definition area within the object.
	External Object	The name of the object that contains the impact item if the item is in an external object, for example GDA, LDA, PDA or Copycode.
	Name	The name of the impact items impacted within the selected object.

BUTTON NAME	DESCRIPTION
Object List group:	
Prev	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.
More	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.
Selection / De-selection buttons:	
>>>	Select all objects in the object list (when more than one page is available, as set by the LISTBOXMAX parameter in the NATENG.INI file).
>>	Select all objects on the current page in the object list.
>	Select all selected objects in the object list.
<	De-select all selected objects in the selected list.
<<	De-select all objects on the current page in the selected list.

BUTTON NAME	DESCRIPTION
<<<	De-select all objects in the selected list (when more than one page is available, as set by the LISTBOXMAX parameter in the NATENG.INI file).
Selected group:	
Prev	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.
More	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.
Impact Sets screen:	
OK	Save the Impact Sets settings.
Cancel	Cancel the Impact Sets process and return back to the Impact Version tab 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.

STATUS BAR ITEM	DESCRIPTION
The Impact Sets status bar is divided into 2 individual panes.	
Pane 1	The currently selected Impact Version number.
Pane 2	Any Impact Sets processing messages.

Criteria Summary Tab Screen

The Criteria Summary tab screen shows a summary of the criteria that have been specified for an Impact Version within the Advanced Options screen.

The Criteria Summary tab screen is accessed using the Criteria Summary tab from the Advanced Options screen. It can also be invoked by selecting an Impact Version from the Impact Version tab screen by using a double-click with the left mouse button.

The following Figure 1-3-4 illustrates the Criteria Summary tab screen.

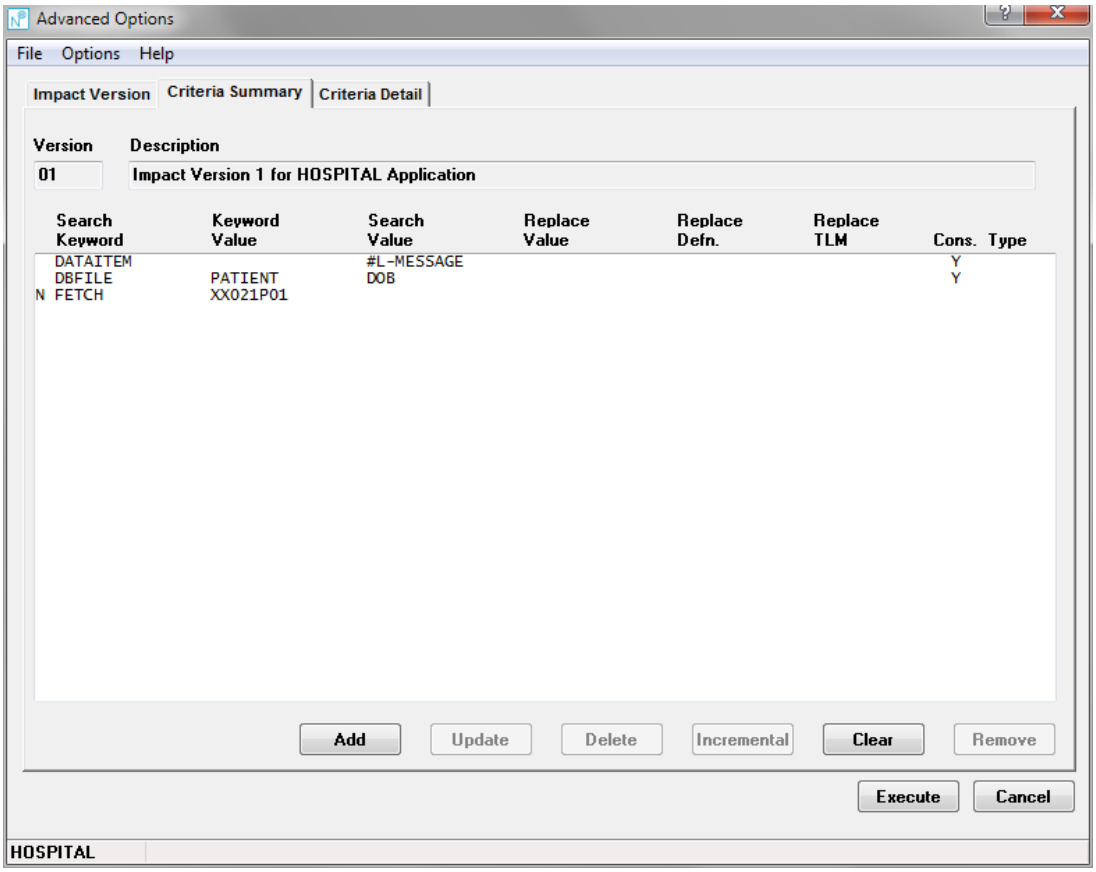

Figure 1-3-4 Criteria Summary tab screen

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Open…**	Open an Impact Criteria PC file.
		These files will have a file extension of '.IRE'. By default these files are located in the data folder where Natural Engineer is installed.
	Save As…	Save the defined Impact Criteria for the currently selected version to a PC file.
		These files will have a file extension of '.IRE'. By default these files will be saved to the data folder where Natural Engineer is installed.
	Exit	Exit the Advanced Options screen and return back to the main Natural Engineer screen.
Options	**Select All**	Not available for Criteria Summary.
	Deselect All	Not available for Criteria Summary.
	Impact Results…	Invoke the Impact Results screen to specify the maximum iteration number that is to be kept for the completed Impact execution. All other data at a higher iteration will be deleted.
Help	Invoke the Criteria Summary help.	

SCREEN ITEMS	DESCRIPTION
Version	The currently selected Impact Version number.
Description	The description for the currently selected Impact Version number.
Language Indicator	The programming language indicator for the Search Keyword. Possible values are: **' ' (blank)** All languages. **C** Cobol. **N** Natural.
Search Keyword	The keyword to be searched for.
Keyword Value	The associated value with the keyword, such as an object name.
Search Value	The value specified for searching, such as the text in a literal.
Replace Value	The value to replace the value found.
Replace Defn.	The format and length to replace the definition found e.g. replace N8 with A8, where N8 is the Search Value (Definition) and A8 is the Replace Definition.
Replace TLM	The TLM to be inserted into the code.
Cons.	Consistency - Causes the Analysis to trace impacts through the code. There are 3 types of Consistency available: **Y** Standard Consistency when using search keywords ADJUST, DATAITEM, DBFILE and DEFINITION. **S** Single Iteration when using Multi Search criteria without consistency. **M** Multiple Iteration when using Multi Search criteria with consistency.
Type	This shows the type of search criteria applicable. Valid types are : **Blank** Standard Impact criteria. **IN1** Incremental, “Apply to whole application”. **IN2** Incremental, “Only previously impacted objects”. **IN3** Incremental, “Only previously impacted statements”. **OEM** Object Builder or manual source code update. *Note: For more information on the Incremental types refer to the section Incremental Impact Criteria Preferences.* *Note: For more information on the Object Builder types refer to the section Object Builder Processing.*

BUTTON NAME	DESCRIPTION
Criteria Summary tab screen:	
Add	Invoke the Criteria Detail tab screen where the new criteria details can be specified. *Note: For more information on the Impact Criteria screen refer to the section Criteria Detail tab screen.*
Update	Invoke the Criteria Detail tab screen for the selected criteria. Only single criteria can be selected for this option. *Note: For more information on the Impact Criteria screen refer to the section Criteria Detail tab screen.*
Delete	Delete the selected criteria. Multiple criteria can be selected for this option. *Note: The delete process will only remove the criteria, any previously impacted data will still be available until the next impact execution.*
Incremental	Invoke the Criteria Detail tab screen, where the criteria can be specified and then the Incremental Impact Criteria Preferences set. This option is only available when impact data exists (created by one or more criteria). *Note: For more information on the Impact Criteria screen refer to the section Criteria Detail tab screen.* *Note: For more information on the Incremental Impact Criteria Preferences refer to the section Incremental Impact Criteria Preferences.*
Clear	Delete all the criteria for the current selected version. *Note: The clear process will only remove the criteria, any previously impacted data will still be available until the next impact execution.*
Remove	Delete all the criteria and associated impact data for the selected criteria. Only single criteria can be selected for this option.
Impact Criteria screen:	
Cancel	Cancel the Impact Criteria process and return back to the main Natural Engineer screen.

Note: Criteria can be selected from the Impact Search Criteria Summary screen by using the left mouse button with a single click. Holding down the shift or control keys allows multiple criteria to be selected.

STATUS BAR ITEM	DESCRIPTION
The Impact Criteria status bar is divided into 2 individual panes.	
Pane 1	Name of the selected application.
Pane 2	Any Impact Criteria processing messages.

Incremental Impact Criteria Preferences

Incremental Impact Criteria Preferences allow you to perform an Analysis over the results of a previous Analysis, i.e., only newly added search criteria will be used during the impact execution. The previous search criteria must not be modified; only new entries should be added.

This is called Incremental Impact.

The Incremental Impact Criteria Preferences can be set when adding Impact search criteria using the '**Incremental**' button on the Criteria Summary tab screen.

This will allow you to add new criteria, specifying the details in the same way as standard criteria. The difference is that when confirming the criteria using the '**Add**' button, the Incremental Impact Criteria Preferences screen is presented.

The following Figure 1-3-5 illustrates the Incremental Impact Criteria Preferences screen.

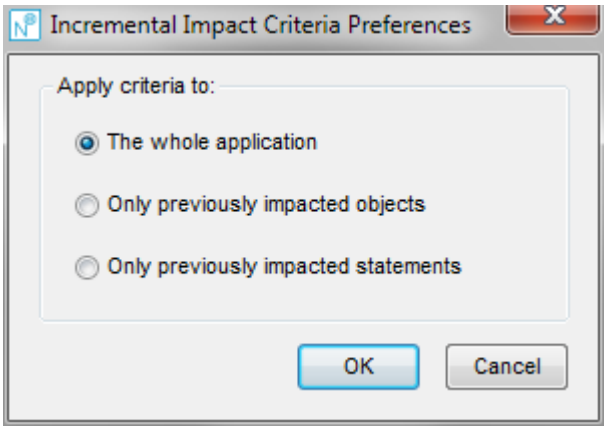

Figure 1-3-5 Incremental Impact Criteria Preferences screen

The Incremental Impact Criteria Preferences screen allows the specification of how the Incremental Criteria are to be applied.

- "The Whole Application"

  Impact Execution will apply the incremental criteria against all the objects in the current application, adding the incremental criteria impacts to the existing criteria.

This can prove to be useful if the application contains a large number of objects and re-running all the impact criteria in addition to the newly added criteria may take a long time to complete.

These incremental criteria will have a criteria option set to “IN1”.

- “Only Previously Impacted Objects”

  Impact execution will apply the incremental criteria only against the objects that have been previously impacted during the last impact execution. All objects within the application that have not been previously impacted will not be impacted for the new incremental criteria.

  This is a useful way of refining the impact results to help identify more specifically impact data of interest.

  These incremental criteria will have a criteria option set to “IN2”.

- “Only Previously Impacted Statements”

  Impact execution will apply the incremental criteria only against the statements that have been previously impacted during the last impact execution. All previously non-impacted statements in all objects within an application will not be impacted for the new incremental criteria.

  This is another useful way of refining the impact results to help identify more specifically impact data of interest.

  These incremental criteria will have a criteria option set to “IN3”.

  After incremental impact has been executed, there are three options that can be taken:

  1. Add a new non- incremental criteria to the impact criteria in the current version

  This will result in the next impact execution disregarding any previous impact results and will apply ALL the criteria in the current version against all the objects in the application.

  2. Add a new Incremental criteria.

  This will restart the Incremental impact process depending on the criteria option selected for the incremental criteria. When adding new incremental criteria it is not possible to go backwards in the incremental chain, i.e., it is only possible to add an incremental criteria at the same level as the previous incremental or the next level down.

Examples:

(1) If the previous impact execution was for an incremental “IN1”, then new incremental criteria can be added as IN1, IN2 or IN3.

(2) If the previous impact execution was for an incremental “IN2”, then new incremental criteria can be added as IN2 or IN3 only.

(3) If the previous impact execution was for an incremental “IN3”, then new incremental criteria can be added as IN3 only.

3. Update the Incremental criteria

As there has been extensive post-processing to the results of all the criteria, it is not possible to update any criteria previous to the last incremental one. Therefore, the only criteria you can update is the last incremental one. The incremental criteria can only be changed as to how the incremental criteria is to be applied during impact execution (IN1; IN2; IN3).

Criteria Detail Tab Screen

The Criteria Detail tab screen is used to add or update Impact Criteria for a selected Impact Version within the Advanced Options screen.

The Criteria Detail tab screen is accessed using either the Criteria Detail tab from the Advanced Options screen, or by using the '**Add**' or '**Update**' buttons on the Criteria Summary tab screen. It can also be invoked by selecting any existing criteria from the Criteria Summary tab screen by using a double-click with the left mouse button.

The Criteria Detail tab screen content is controlled by the selection of Impact Types and Search Keywords. Display of the required content will be relevant to the selection made.

The following Figure 1-3-6 illustrates the Criteria Detail tab screen for Impact Type DATAITEM.

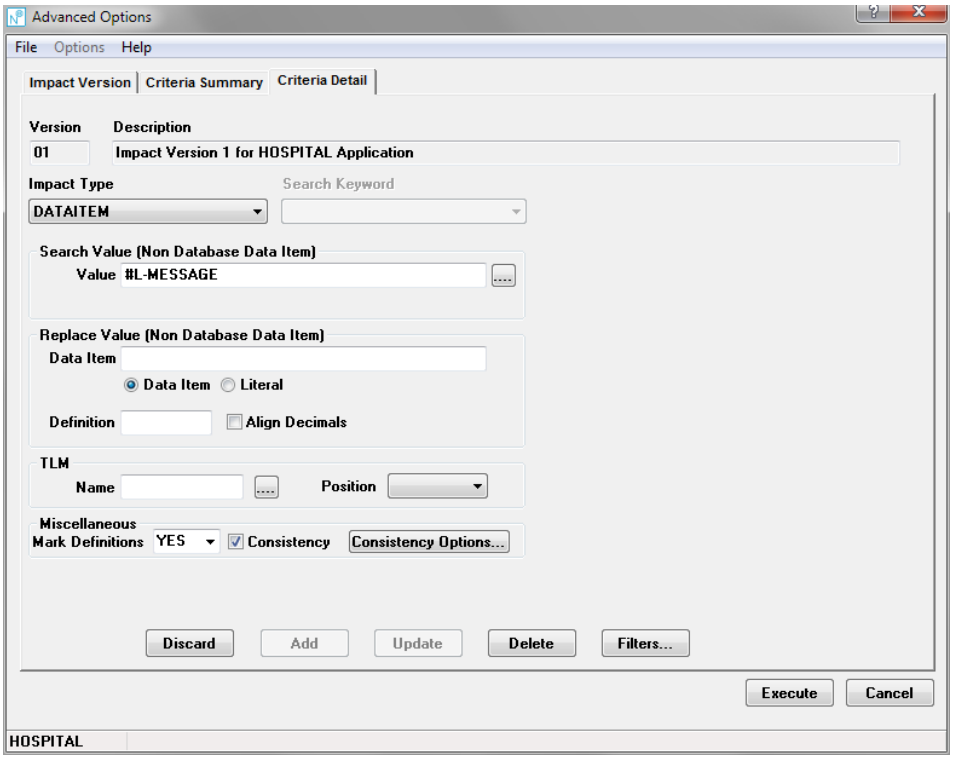

Figure 1-3-6 Criteria Detail tab screen for Impact Type DATAITEM

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Open…**	Not available for Criteria Detail.
	Save As…	Not available for Criteria Detail.
	Exit	Exit the Impact Criteria screen and return back to the main Natural Engineer screen.
Options	**Select All**	Select all the available options being displayed.
	Deselect All	De-select all the available options being displayed.
	Note: This 'Select All' and 'Deselect All' options are only available for screen contents showing check box selections, for example MVSNAT22TO31.	
	Impact Results…	Not available for Criteria Detail.
Help	Invoke the Criteria Detail help.	

SCREEN ITEMS	DESCRIPTION
Impact Types List	List of all the types of Impact that are to be applied. The list includes all the Combination Keywords, programming language specific triggers, 'NATURAL KEYWORDS', 'COBOL KEYWORDS' and 'JCL KEYWORDS', which control the Search Keyword List. *Note: For more information on the combination keywords refer to the section Combination Keywords.*
Search Keyword List	List of all the programming language keywords. The list is dependent on the Impact Type selections: **NATURAL KEYWORDS** Natural keywords. **COBOL KEYWORDS** Cobol keywords. **JCL KEYWORDS** JCL keywords. *Note: For more information on the available search keywords refer to the section Search Keywords.*

SCREEN ITEMS	DESCRIPTION
Keyword Value group:	
Keyword Value	The name of an object directly associated with the keyword. This option is only available for Impact Types and Search Keywords that reference other objects. For example: **FETCH** has an associated program object name. **CALLNAT** has an associated subprogram object name. **INPUT MAP** has an associated map object name. **HISTOGRAM** has an associated DDM name. The Keyword Value can be specified using a complete name or part name using a '?' wildcard. For example: **XX?** Will search for all call names starting with XX. **?XX** Will search for all call names ending with XX. **?XX?** Will search for all call names containing XX. The Keyword Value can also be specified using a mask value for the call name. This can only be used for non-DDM related keywords, for example: CALLNAT, FETCH and INPUT MAP. *Note: For more information on using call name masks refer to the section Keyword Options.*
Search Value group:	
Value / Data Item / Literal / Definition	The value to be searched for. This can be either text within a literal string, a data item, a definition attribute (format and length) or number of parameters being used within calls to other objects. Certain Impact Types and Search Keywords will show radio buttons that help to further qualify the search value being used. Possible radio button selections are: **Data Item** A data item search locates all data items within objects for the specified value. Data item names can be input using either a complete name or part name using a '?' wildcard. For example: **#ABC?** Will search for all data items starting with #ABC. **?#ABC** Will search for all data items ending with #ABC. **?#ABC?** Will search for all data items containing #ABC. **Literal** A literal search locates all the text, numeric constants or edit mask definitions within objects for the specified value. Values can be input using either complete values or part values using a '?' wildcard. For example:

SCREEN ITEMS	DESCRIPTION
	Hello? Will search for all literal strings starting with Hello. **?Hello** Will search for all literal strings ending with Hello. **?Hello?** Will search for all literal strings containing Hello. **Definition** A definition search locates all the data items within objects for the specified value. Definitions can be input using either a single format and length value or a range of format and lengths values using a '-' (hyphen) to separate the range. For example: **A001** Will locate all one-byte alphanumeric data items. **A001-A010** Will locate all alphanumeric data items with a length greater than or equal to one and less than or equal to ten. *Note: Further refinement options are available using the Definition Options. For more information refer to the section Definition Options.* **Parameter** A parameter search locates all the statements that pass data items within objects for the specified value. For example CALLNAT, FETCH. Parameter values are specified as a number of parameters. For example: Specifying a value of 3 with the CALLNAT keyword will locate all CALLNATs that have three data items specified for the parameter reference.
Replace Value group:	
Data Item / Literal	The value to replace the value found by the Search Value. This value is used during the modification process. This can be either text within a literal string, a data item or an object. Certain Impact Types and Search Keywords will show radio buttons that help to further qualify the search value being used. Possible radio button selections are: **Data item** Identifies that the Replace Value is to be used as a data item. For example, if you are searching for a literal and want to replace it with a data item this option is required to replace the literal correctly. **Literal** Identifies that the Replace Value is to be used as a literal. For example, if you are searching for a data item and want to replace it with a literal then this option is required to replace the data item correctly.

SCREEN ITEMS	DESCRIPTION
	Mask This option provides the facility to modify applications to support language code processing. This option is used to identify the Replace Value as a mask value, which will be used against the search value. This option is only available for Search Keywords utilizing language code processing for example: CALLNAT, FETCH and INPUT MAP. The Replace Value must be specified using the following convention: **.** (period) Indicates a single position that is not to be checked. ***** (asterisk) Wildcard used to indicate that the replacement applies to the last character position. **&** (ampersand) The replacement character. Examples:

Old Name	Mask Value	New Name
MAP001M	…..&	MAP00&M
MAP001	…..&	MAP00&
SUBPGM01	*&	SUBPGM0&
SPGM1	*&	SPGM&

DDM	The value to replace the name of a DDM. This option is only available for Impact Type DBFILE where only a Keyword Value has been specified, i.e. the name of a DDM.
DDM Field	The value to replace the name of a DDM field. This option is only available for Impact Type DBFILE where a Keyword Value and Search Value have been specified, i.e. the name of a DDM and DDM field.
Definition	Replaces the data item definition with the new format and length.
Align Decimals	Specify whether decimal places (DP) are to be aligned. This applies to any derived fields, i.e., any fields that are found to be propagated by the specified field. This option is only available for Impact Types DBFILE, DATAITEM and DEFINITION, and Consistency is set on. If unchecked, any derived field DP will have the replace definition DP added to them. If checked, and the derived field DP are greater than the replace definition DP, then no change is made to the derived field DP.

SCREEN ITEMS	DESCRIPTION

	If checked, and the derived field DP are less than the replace definition DP, then the derived field DP are set to the replace definition DP value.
	
	For example, for the following source code, an Impact Criteria of DATAITEM #A and replace value of N5.2 is specified.
	`0010 #A (N7)`
	`0020 #B (N7.1)`
	`0030 #C (N7.3)`
	`0040 #D (N9)`
	`0050 MOVE #A TO #B #C #D`
	
	If Align Decimals is unchecked, the following modifications will be made:
	`0010 #A (N5.2)`
	`0020 #B (N5.3)`
	`0030 #C (N5.5)`
	`0040 #D (N7.2)`
	
	If Align Decimals is checked, the following modifications will be made:
	`0010 #A (N5.2)`
	`0020 #B (N5.2)`
	`0030 #C (N5.3)`
	`0040 #D (N7.2)`

Keyword	This option allows you to select a keyword that will replace the Search Keyword. This option is only available for keywords that execute other objects, for example CALLNAT and you can replace this with, for example a CALL or FETCH statement. This option is only available for the following Search Keywords: • **CALL** • **CALL FILE** • **CALL INTERFACE4** • **CALL LOOP** • **CALLNAT** • **ESCAPE ROUTINE** • **FETCH** • **FETCH REPEAT** • **FETCH RETURN** • **INVESTIGATE** • **PERFORM** • **RUN** • **RUN REPEAT** • **RUN RETURN** • **STOP** • **TERMINATE**
TLM group:	
Name	The name of the Text Logic Member (TLM) to be inserted in the object source code.
Position	Determines the actual placement of the Replace TLM. Available options are: • **REPLACE** • **AFTER** • **BEFORE**

SCREEN ITEMS	DESCRIPTION
Miscellaneous group:	
Mark Definitions	Only used for search keywords DATAITEM and DBFILE. May be set to Yes, No or Only. Default value is Yes. **Yes** – Causes the analysis to mark both definitions and processing code if impacted. **No** – Causes the analysis to only mark processing code if impacted. **Only** - Causes the analysis to mark definitions only, not processing code. This setting is ignored for Consistency.
Consistency	Causes the Analysis to trace code identified, for further impact on other code. For example: MOVE #A TO #B. Using the search keyword DATAITEM and search value #A the above statement will be impacted as follows: **With Consistency set off:** #A will be impacted as it is the specified item. **With Consistency set on:** #A will be impacted as the specified item and #B will be impacted as it is being propagated by #A. To set specific consistency options please select the Consistency Options… button. *Note: This option is only available for Search Keywords ADJUST, DATAITEM, DBFILE and DEFINITION.*

Note: The Keyword Value, Search Value, Replace Value, TLM and Miscellaneous groups are not available for Cobol and JCL Keywords.

BUTTON NAME	DESCRIPTION
Keyword Value group:	
Keyword Value Selection [….]	Invokes the General Selection screen, listing either Objects or DDMs loaded in the Repository for an application. The items listed are dependent on the Search Keyword being used. *Note: For more information on the General Selection screen refer to Chapter 2 in the Concepts and Facilities manual.*
Search Value group:	
Search Value Selection [….]	Invokes the General Selection screen, listing either data items or DDM Fields loaded in the Repository for an application. The items listed are dependent on the Search Keyword being used. *Note: For more information on the General Selection screen refer to Chapter 2 in the Concepts and Facilities manual.*

BUTTON NAME	DESCRIPTION
TLM group:	
Replace TLM Selection [….]	Invokes the General Selection screen, listing any Objects with object type 'Text' either from the Modification library or the Natural library SYSTEM. *Note: For more information on the General Selection screen refer to Chapter 2 in the Concepts and Facilities manual.*
Miscellaneous Group:	
Consistency Options…	Invokes the Consistency Options Screen to allow the user to override the default Consistency options.
Criteria Detail tab screen:	
Discard	Reset any inputs made to the currently selected Criteria Details and return back to the Criteria Summary tab screen.
Add	Save the currently selected Criteria Details.
Update	Update the currently selected Criteria Details. *Note: This button is only available when applying changes to previously saved Criteria Details.*
Delete	Delete the currently selected Criteria Details.
Filters…	Invoke the Filters screen to specify object refinements for the currently selected Criteria Details. If a Criteria Detail has Filters specified, the Filters button will have an '*' (asterisk) showing to the left of the button text. *Note For more information on Filters refer to the Filters Window section.*
Advanced Options screen:	
Cancel	Cancel the Advanced Options process and return back to the main Natural Engineer screen.

STATUS BAR ITEM	DESCRIPTION
The Advanced Options status bar is divided into 2 individual panes.	
Pane 1	Name of the selected application.
Pane 2	Any Advanced Options processing messages.

Filters Window

The Filters screen provides refinement options to limit the objects to be included during the Analysis process. Refinements can be made by object type, object name and programming language. The filters are applied to each Criteria Detail being specified, i.e., each Criteria Detail can have its own tailored filter.

The Filters screen is accessed using the 'Filters' button from the Criteria Detail tab screen.

The following Figure 1-3-7 illustrates the Filters screen.

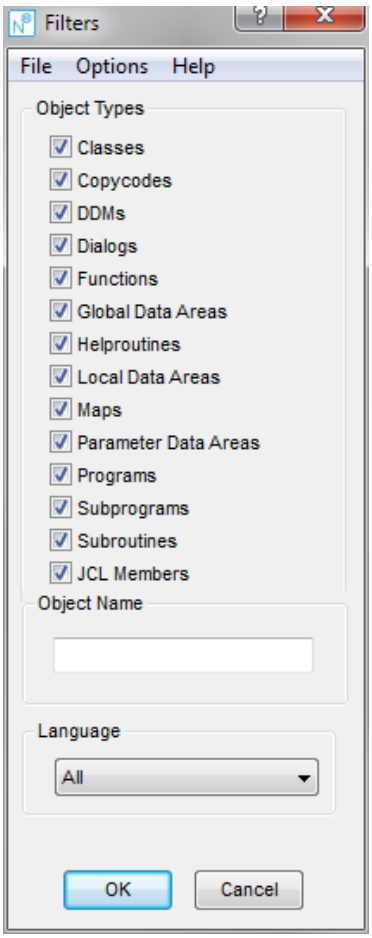

Figure 1-3-7 Filters screen

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Exit**	Exit the Filters screen and return back to the Criteria Detail tab screen.
Options	**Select All**	Select all the Object Types.
	Deselect All	De-select all the Object Types.
Help	Invoke the Filters help.	

SCREEN ITEMS	DESCRIPTION
Object Types	Allows you to select the types of object to be included. Available selections are: ▪ **Classes** ▪ **Copycodes** ▪ **DDMs** ▪ **Dialogs** ▪ **Functions** ▪ **Global Data Areas** ▪ **Helproutines** ▪ **Local Data Areas** ▪ **Maps** ▪ **Parameter Data Areas** ▪ **Programs** ▪ **Subprograms** ▪ **Subroutines** ▪ **JCL Members** *Note: Object Types are not available if programming language Cobol is selected.*
Object Name	The name of the object to be included. The object name can be entered in full. For example 'XX021P01' will include object XX021P01 only. A group of objects can be selected by typing in a part name using an '*' (asterisk) wildcard. For example 'XX001*' will include all objects that are prefixed with 'XX001'. All Objects can be selected by typing in a single '*' (asterisk).

SCREEN ITEMS	DESCRIPTION
Language	Allows you to select the programming language of the objects to be included. Available selections are: ▪ **All** ▪ **Cobol** ▪ **JCL** ▪ **Natural**

BUTTON NAME	DESCRIPTION
OK	Save the Filters settings.
Cancel	Cancel the Filters process and return back to the Criteria Detail tab screen.

STATUS BAR ITEM	DESCRIPTION
The Impact Sets status bar is divided into 2 individual panes.	
Pane 1	The currently selected Impact Version number.
Pane 2	Any Impact Sets processing messages.

Consistency Options Window

The Consistency Options screen provides the ability to override the default Consistency Options that are set in the initialization file.

The Consistency Options screen is accessed using the 'Consistency Options…' button from the Criteria Detail tab screen.

The following Figure 1-3-8 illustrates the Consistency Options screen.

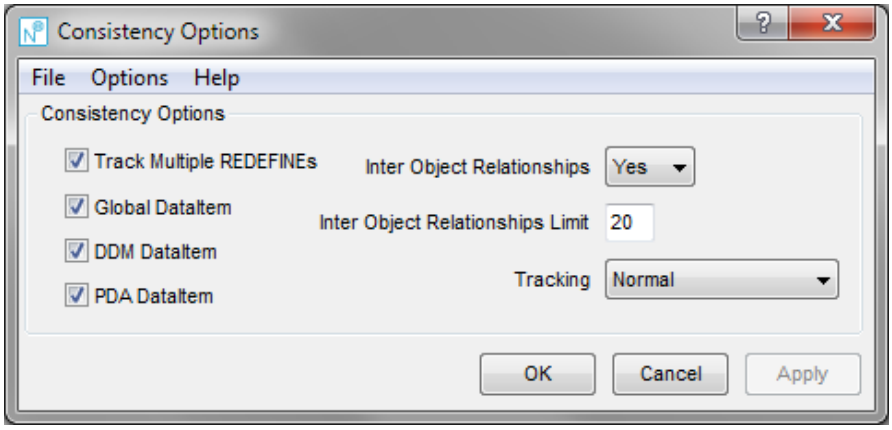

Figure 1-3-8 Consistency Options screen

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Exit**	Exit the Consistency Options screen and return back to the Criteria Detail tab screen.
Options	**Select All**	Select all the Track Multiple REDEFINEs, Global Dataitem, DDM Dataitem and PDA Dataitem options.
	Deselect All	De-select all the Track Multiple REDEFINEs, Global Dataitem, DDM Dataitem and PDA Dataitem options.
Help	Invoke the Consistency Options help.	

SCREEN ITEMS	DESCRIPTION
Inter Object Relationships	If set to Y, data elements are tracked across object boundaries following the impact process. Setting the value to ONLY will ensure that Natural Engineer only performs inter object relationships. Please note that this setting should only be used following a successful multiple impact or to restart a previously failed Inter Object Relationships (IOR) process. Possible values Y,N, ONLY
Inter Object Relationships Limit	This is the number of Iterations that IOR will track objects across object boundaries.
Tracking	Controls the tracking direction for a variable. If set to Forward by Value, a variable is tracked in a forward direction, showing all the derivatives being populated from the variable. If set to Backward by Value, a variable is tracked in a backward direction, showing where the variable and derivatives have been populated. If set to Forward by Usage, a variable is tracked in a forward direction, showing all the derivatives being used. If set to backward by Usage, a variable is tracked in a backward direction, showing where the variable and derivatives have been used. If set to Normal, both the forward and backward directions will be shown
Track Multiple REDEFINEs	If selected, multiple redefines are tracked.
Global Dataitem	Used when impacts have been made to Global Data Areas. If selected, Natural Engineer will track these fields, and derivations of these fields, until all possible impacts have been identified.
DDM Dataitem	Used when impacts have been made to DDMs. If selected, Natural Engineer will track these fields, and derivations of these fields, until all possible impacts have been identified.
PDA Dataitem	Used when impacts have been made to Parameter Data Areas. If selected, Natural Engineer will track these fields, and derivations of these fields, until all possible impacts have been identified.

BUTTON NAME	DESCRIPTION
OK	Will save the Consistency Options settings and return back to the Criteria Detail tab screen.
Cancel	Cancel the Consistency Options process and return back to the Criteria Detail tab screen.
Apply	Will save the Consistency Options settings.

Search Keywords

The Analysis process allows the selection of a Search Keyword in the Criteria. These may be Natural Keywords, Cobol Keywords, JCL Keywords, Combination Keywords or Miscellaneous Keywords.

Natural Keywords

The following table lists the Natural Keywords available to Impact Criteria.

ACCEPT	END-ENDPAGE	MOVE ROUNDED
ADD	END-ERROR	MULTIPLY
ASSIGN	END-FILE	NEWPAGE
AT BREAK OF	END-FIND	NEWPAGE TITLE
AT END OF DATA	END-FOR	OBTAIN
AT END OF FILE	END-HISTOGRAM	ON ERROR
AT END OF PAGE	END-IF	OPEN CONVERSATION
AT START OF DATA	END-INTERFACE	OPEN DIALOG
AT TOP OF PAGE	END-LOOP	OPTIMIZE
BACKOUT	END-METHOD	OPTIONS
BEFORE BREAK	END-NOREC	ORDER BY
BROWSE	END-PROCESS	PASSW
BROWSE BY	END-PROPERTY	PERFORM
BROWSE WHERE	END-REPEAT	PLOT
BROWSE WITH	END-READ	PROCESS
CALL	END-SELECT	PRINT
CALL FILE	END-SORT	PROCESS COMMAND
CALL INTERFACE4	END-START	PROCESS GUI ACTION
CALL LOOP	END-TOPPAGE	PROCESS SQL

CALLNAT	END-SUBROUTINE	PROCESS REPORT
CLOSE CONVERSATION	END-WORK	PROCESS REPORTER
CLOSE DIALOG	ENTER	PROPERTY
CLOSE PC	ESCAPE	READ
CLOSE PRINTER	ESCAPE BOTTOM	READ BY
CLOSE WORK	ESCAPE ROUTINE	READ PC
COMMIT	EXAMINE	READ WITH
COMPOSE	EXAMINE TRANSLATE	READ WORK
COMPOSE ASSIGNING	EXPAND	REDEFINE
COMPOSE EXTRACTING	FETCH	REDUCE
COMPOSE FORMATTING	FETCH REPEAT	REINPUT
COMPOSE MOVING	FETCH RETURN	REINPUT MARK
COMPOSE RESETTING	FIND	REJECT
COMPRESS	FIND WITH	RELEASE SETS
COMPUTE	FIND WHERE	RELEASE STACK
CREATE OBJECT	ESCAPE TOP	RELEASE VARIABLES
DECIDE	FIND COUPLED	READ WHERE
DECIDE VALUE	FIND RETAIN	REPEAT
DECIDE VALUE ALL	FIND FIRST	RETRY
DECIDE VALUE ANY	FIND NUMBER	REPEAT UNTIL
DECIDE VALUE NONE	FIND UNIQUE	REPEAT WHILE
DECIDE WHEN	FORMAT	RESET
DECIDE WHEN ALL	FIND SORTED	RUN
DECIDE WHEN ANY	FOR	RETURN
DECIDE WHEN NONE	HISTOGRAM VALUE	ROLLBACK
DEFINE CLASS	GET	RULEVAR
DEFINE DATA CONTEXT	GET SAME	SEPARATE

DEFINE DATA GLOBAL	GET TRANSACTION	RUN REPEAT
DEFINE DATA INDEPENDENT	GROUP BY	RUN RETURN
DEFINE DATA LOCAL	HAVING	SELECT
DEFINE DATA OBJECT	HISTOGRAM	SEND
DEFINE DATA PARAMETER	HISTOGRAM WHERE	SEND EVENT
DEFINE INITIAL	IF	SEND METHOD
DEFINE PRINTER	IF NO RECORDS	SET CONTROL
DEFINE REPORT	IF SELECTION	SET GLOBALS
DEFINE SUBROUTINE	IGNORE	SET KEY
DEFINE WINDOW	IMPORT	SETTIME
DEFINE WORK	INCLUDE	SET WINDOW
DELETE	INPUT	SETTIME
DELETE FROM	INPUT MARK	SHOW
DISPLAY	INPUT MAP	SKIP
DIVIDE	INPUT WINDOW	SORT
DLOGOFF	INPUT TEXT	STACK
DLOGON	INSERT	STACK DATA
DO	INTERFACE	STACK COMMAND
DOEND	INVESTIGATE	STOP
DOWNLOAD	LABEL	STORE
DRAW	LIMIT	SUBTRACT
EJECT	LOOP	SUSPEND
ELSE	MAP	TERMINATE
END	METHOD	TRANSFER
END TRANSACTION	MOVE	UPDATE
END-ALL	MOVE ALL	UPLOAD
END-BEFORE	MOVE BY NAME	WHILE

END-BREAK	MOVE BY POSITION	WRITE
END-BROWSE	MOVE EDITED	WRITE PC
END-CLASS	MOVE ENCODED	WRITE TITLE
END-DECIDE	MOVE INDEXED	WRITE TRAILER
END-DEFINE	MOVE LEFT	WRITE WORK
END-ENDDATA	MOVE NORMALIZED	
END-ENDFILE	MOVE RIGHT	

There are some generic level Search Keywords available that can be specified, which allow you to include related keywords. This allows you to specify a range of related keywords as one single criteria and Impact Analysis will then identify all related keywords.

These Search Keywords can be identified by a '?' at the end of the keyword. For example:

1. A Search Keyword of ‘DEFINE ?’ will identify the following keywords:

- DEFINE DATA CONTEXT
- DEFINE DATA GLOBAL
- DEFINE DATA INDEPENDENT
- DEFINE DATA LOCAL
- DEFINE DATA OBJECT
- DEFINE DATA PARAMETER
- DEFINE CLASS
- DEFINE INITIAL
- DEFINE PRINTER
- DEFINE REPORT
- DEFINE SUBROUTINE
- DEFINE WINDOW
- DEFINE WORK

2. A Search Keyword of ‘DEFINE DATA ?’ will identify the following keywords:

- DEFINE DATA CONTEXT
- DEFINE DATA GLOBAL
- DEFINE DATA INDEPENDENT
- DEFINE DATA LOCAL
- DEFINE DATA OBJECT
- DEFINE DATA PARAMETER

3. A Search Keyword of DEFINE DATA GLOBAL will only identify DEFINE DATA GLOBAL keywords.

Special Search Value for Search Keyword IF

For the Search Keyword '**IF**', a special Search Value of 'IS(format)' can be specified. This search value will impact any usage of the IS option within an IF statement.

Note: The IS option can be used to check whether the content of an alphanumeric field can be converted to a specific other format. For example, the IS option can be used to check the content of a field before the mathematical function VAL (extract numeric value from an alphanumeric field) is used to ensure that it will not result in a runtime error.

The search value IS(format) is specified in the search value field on the Impact Criteria screen and is only valid with the search keyword IF, where '(format)' is the desired format and length. Examples of possible search values are:

IS(N7)
IS(I002)
IS(D)

For Example:

```
::::
0090 DEFINE DATA LOCAL
0100 01 #ALPHA              (A7)
0110 01 #NUMERIC            (N7)
0120 END-DEFINE
::::
0250 IF #ALPHA IS(N5)
```

```
0260   COMPUTE #NUMERIC := VAL(#ALPHA) * 1
0270   WRITE #NUMERIC
0280 END-IF
::::
```

Using Impact criteria of search keyword 'IF' and search value 'IS(N5)' would provide Impact results for statement line number 0250.

Cobol Keywords

The following table lists the Cobol Keywords available to Impact Criteria.

++INCLUDE	END-READ	OBJECT SECTION
ACCEPT	END-RETURN	ON
ADD	END-REWRITE	OPEN
ALTER	END-SEARCH	PERFORM
BASED-STORAGE SECTION	END-START	PROCEDURE DIVISION
BASIS	END-STRING	PROCESS
BOSS	END-SUBSTRACT	READ
CALL	END-UNSTRING	READY
CANCEL	END-WRITE	RECEIVE
CBL	END-XML	RELEASE
CLOSE	ENTER	REPLACE
COMPUTE	ENTRY	REPORT SECTION
CONFIGURATION SECTION	ENVIRONMENT DIVISION	RESET
CONTINUE	EVALUATE	RESET TRACE
CONSTANT SECTION	EXAMINE	RETURN
COPY	EXEC ADABAS	REWRITE
DATA DIVISION	EXEC CICS	SCREEN SECTION
DECLARATIVES	EXEC DLI	SEARCH
DECIMAL-POINT	EXEC SQL	SEEK
DELETE	EXHIBIT	SELECT
DISABLE	EXIT	SEND
DISPLAY	EXIT METHOD	SERVICE
DIVIDE	EXIT PERFORM	SERVICE LABEL
EJECT	EXIT PROGRAM	SERVICE RELOAD

ENABLE	FILE SECTION	SET
ELSE	GENERATE	SKIP1
END CLASS	GOBACK	SKIP2
END DECLARATIVES	GO	SKIP3
END METHOD	ID DIVISION	SORT
END OBJECT	IDENTIFICATION DIVISION	START
END PROGRAM	IF	STOP
END-ADD	INITIALIZE	STRING
END-CALL	INPUT-OUTPUT SECTION	SUBTRACT
END-COMPUTE	INSERT	TITLE
END-DELETE	INSPECT	TRANSFORM
END-DISPLAY	INVOKE	UNSTRING
END-DIVIDE	LINKAGE SECTION	UNLOCK
END-EVALUATE	LOCAL-STORAGE SECTION	USE
END-EXEC	MERGE	WHEN
END-IF	METHOD-ID	WORKING-STORAGE SECTION
END-INVOKE	MOVE	WRITE
END-MULTIPLY	MULTIPLY	XML PARSE
END-PERFORM	NEXT SENTENCE	

JCL Keywords

The following table lists the JCL Keywords available to Impact Criteria.

<IN	JOB	SET
DD	PEND	
EXEC	PROC	

The JCL keyword '<IN' is used to identify in-stream data used by the ddname 'CMSYNIN' or 'CMOBJIN'. This is normally associated with Natural batch program execution. For example:

```
0001 //CTRLEXT1 EXEC PGM=NATBAT41,
0002 //            PARM='PROFILE=NEE84N42',
0003 //            REGION=8M,COND=(0,NE)
0004 //STEPLIB  DD  DSN=PPEX.NATURAL.LOAD,DISP=SHR
0005 //          DD  DSN=RZDBA.DB177.NEWLOAD,DISP=SHR
0006 //          DD  DSN=RZDBA.DB177.LOAD,DISP=SHR
0007 //DDKARTE  DD  DUMMY
0008 //DDCARD   DD  *
0009 ADARUN PROG=USER,DB=11177,MODE=MULTI,SVC=249,DEVICE=3390
0010 //DDPRINT  DD  SYSOUT=*
0011 //SYSPUNCH DD  SYSOUT=*
0012 //SYSUDUMP DD  SYSOUT=*
0013 //SYSOUT   DD  DUMMY
0014 //CMPRINT  DD  SYSOUT=*
0015 //CMPRT01  DD  SYSOUT=*
0016 //CMSYNIN  DD  *
0017 LOGON MFNALL
0018 NEESETUP
0019 MFRJEP02
0020 NBBBBBBBBEXTRDEL 00TTTT
0021 FIN
0022 /*
```

The JCL search keyword '<IN' would Impact statement line numbers 0017, 0018, 0019, 0020 and 0021.

Combination Keywords

These are special keywords that perform predefined functions. They are not available for Cobol objects.

The combination keywords are:

- ADJUST
- CODE IMPROVEMENT
- OBJECT BUILDER
- NATRPC
- MULTI SEARCH
- MVSNAT22TO31
- PORTING
- REFACTORING
- SYSTEM FUNCTIONS

Note: For more information on each of the combination keywords refer to Chapter 3 Combination Search Keywords.

Miscellaneous Keywords

These are additional Search Keywords that will identify all instances of non database data items (fields), Database data items (fields), DDMs, data item definitions (format and length) and literal strings within objects.

The miscellaneous keywords are:

1. DATAITEM
2. DBFILE
3. DEFINITION
4. LITERAL

DATAITEM

This allows the user to search for any non-database field. The full value or a partial value, of the data item to be searched for, is specified in the Search Value field. Partial values can be input using wild card '?'. For example:

#ABC	Will search for data items named #ABC
#ABC?	Will search for all data items starting with #ABC
?#ABC?	Will search for all data items containing #ABC
?#ABC	Will search for all data items ending with #ABC

DBFILE

This allows the user to search for any DDM and database field. The full value or a partial value, of the DDM to be searched for is specified in the Keyword Value field. Partial values can be input using wild card '?'. For example:

FILE1	Will search for all DDMs named FILE1
FILE?	Will search for all DDMs starting with FILE
?FILE?	Will search for all DDMs containing FILE
?FILE	Will search for all DDMs ending with FILE

Note: Specifying '?' on its own in the Keyword Value field, will result in all DDMs being searched for.

The full value or a partial value of the database field to be searched for is specified in the Search Value field. Partial values can be input using wild card '?'. For example:

For example:

NAME	Will search for all database fields named NAME
NAME?	Will search for all database fields starting with NAME
?NAME?	Will search for all database fields containing NAME
?NAME	Will search for all database fields ending with NAME

Note: Specifying '?' on its own in the Search Value field, will result in all database fields being searched for.

Note: The DBFILE process does not identify any database fields that are used on a Map. This is because they are actually defined as Parameter Data within the map itself. These can be identified by using the DATAITEM search keyword.

The following Figure 1-3-9 illustrates the database fields defined within a map.

```
0010 * MAP2: PROTOTYPE              --- CREATED BY WNT 4.1.0 ---
0020 * INPUT USING MAP 'XXXXXXXX'
0030 *     #C-ADDRESS #C-ARRIVED #C-DOB #C-DUE-FOR-SURGERY #C-FIRST-NAME
0040 *     #C-PATIENT-ID #C-RELEASED #C-SURNAME #G-MESSAGE #M-MAP-HEADING
0050 *     #M-UNDERLINE PATIENT.ADDRESS(*) PATIENT.ARRIVED PATIENT.DOB
0060 *     PATIENT.DUE-FOR-SURGERY PATIENT.FIRST-NAME PATIENT.PATIENT-ID
0070 *     PATIENT.RELEASED PATIENT.SURNAME
0080 DEFINE DATA PARAMETER
0090 1 #C-ADDRESS (C)
0100 1 #C-ARRIVED (C)
0110 1 #C-DOB (C)
0120 1 #C-DUE-FOR-SURGERY (C)
0130 1 #C-FIRST-NAME (C)
0140 1 #C-PATIENT-ID (C)
0150 1 #C-RELEASED (C)
0160 1 #C-SURNAME (C)
0170 1 #G-MESSAGE (A070)
0180 1 #M-MAP-HEADING (A040)
0190 1 #M-UNDERLINE (A040)
0200 1 PATIENT.ADDRESS (A030/00001:00004)
0210 1 PATIENT.ARRIVED (A020)
0220 1 PATIENT.DOB (N06.0)
0230 1 PATIENT.DUE-FOR-SURGERY (A006)
0240 1 PATIENT.FIRST-NAME (A020)
0250 1 PATIENT.PATIENT-ID (N07.0)
0260 1 PATIENT.RELEASED (D)
0270 1 PATIENT.SURNAME (A020)
0280 END-DEFINE
```

Figure 1-3-9 Database fields defined within a map

DEFINITION

This option allows for the searching of a format and length or a range of format and lengths within the objects. The data is entered as a format type and length and with a range this is repeated with a '-' (hyphen) between the values.

For example:

A001 Will locate all one-byte alphanumeric data items.

A001-A010 Will locate all alphanumeric data items with a length greater than or equal to one and less than or equal to ten.

Note: When this Search Keyword is selected, further refinement options become available by using the 'Definition Options...' button from the Criteria Detail tab screen. For more information on these refinement options refer to the section Definition Options.

LITERAL

The literal search locates all text and numeric constants in objects, as well as edit mask definitions. The literal can be specified in full or a partial value can be used. Partial values can be input using wildcards '?'.

For example:

Hello? Will search for all literal strings starting with Hello

?Hello Will search for all literal strings ending with Hello

?Hello? Will search for all literal strings containing Hello

Hello Will search for all literal strings named Hello

Note: When this Search Keyword is selected, further refinement options become available by using the Literal Options group. For more information on these refinement options refer to the section Literal Options.

Forward/Backward Tracking

Forward/Backward tracking can be used to control the tracking direction for a variable, when using search keyword MULTISEARCH or the search keywords DATAITEM, DBFILE or DEFINITION (which make use of the consistency option).

Forward/Backward tracking is controlled by the TRACKING parameter in the NATENG.INI file.

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

Forward by Value tracking will track the forward direction of a variable showing all the derivatives being populated from the variable.

Backward by Value tracking will track the backward direction of a variable showing all the derivatives that have populated the variable.

Forward/Backward by Value tracking affects any statements using the following Natural Keywords:

- MOVE
- ASSIGN
- MOVE BY NAME
- ASSIGN ROUNDED
- MOVE LEFT
- MOVE RIGHT
- MOVE ROUNDED

Forward/Backward tracking by Usage will track the direction of the variable using all syntactical relationships.

Example of Forward/Backward Tracking

The following example uses a search keyword of DATAITEM, with a search value of #A and Consistency switched on. Results for each of the tracking options is then described.

Sample Source Code:

```
::::
0110 MOVE #A TO #B
0120 IF #A > #C
0130   WRITE 'HELLO'
0140 END-IF
0150 MOVE #D TO #A
0160 END
```

1. Forward by Value tracking will result in:

- #A at statement lines 0110, 0120 and 0150 being marked as 'Specified' matches.
- #B at statement line 0110 would be marked as 'Derived'.

2. Backward by Value tracking will result in:

- #A at statement lines 0110, 0120 and 0150 being marked as 'Specified' matches.
- #D at statement line 0150 would be marked as 'Derived'.

3. Forward by Usage tracking will result in:

- #A at statement lines 0110, 0120 and 0150 being marked as 'Specified' matches.
- #B at statement line 0110 would be marked as 'Derived'.
- #C at statement line 0120 would be marked as 'Derived'.

4. Backward by Usage tracking will result in:

- #A at statement lines 0110, 0120 and 0150 being marked as 'Specified' matches.
- #D at statement line 0150 would be marked as 'Derived'.

5. Normal tracking will result in:

- #A at statement lines 0110, 0120 and 0150 being marked as 'Specified' matches.
- #B at statement line 0110 would be marked as 'Derived'.

- #C at statement line 0120 would be marked as 'Derived'.
- #D at statement line 0150 would be marked as 'Derived'.

Keyword Options

The Keyword Options are displayed on the Criteria Details tab screen when any of the Search Keywords described below are selected. It provides the facility to specify further refinement options for either sub-keywords or to allow mask values to be used for call names.

There are two main levels of refinement options available:

1. Specify Sub Keywords.

Sub Keywords represent the optional clauses that can be used with a Natural Keyword. The following Natural Keywords are supported:

COMPOSE	FIND	SEND
STACK	WRITE	

2. Allow the use of call name mask values specified in Keyword Value.

Call names can be specified as a Keyword Value using a full or partial call name, or, as a mask value. If a mask value is used then the refinement option needs to be specified to indicate that the call name is a mask value.

Note: Impact will only match call names that are coded as literal strings when using mask values.

The following Search Keywords are supported:

CALL ?	CALL	CALL FILE
CALL INTERFACE4	CALL LOOP	CALLNAT
FETCH ?	FETCH	FETCH REPEAT
FETCH RETURN	INPUT MAP	INVESTIGATE

OPEN DIALOG	RUN ?	RUN
RUN REPEAT	RUN RETURN	

The Keyword Options group is controlled by the Search Keyword selected. Only the relevant options for the selected Search Keyword will be displayed. For example, the Search Keyword WRITE allows sub keywords FORM and MAP to be selected but not the use of call name mask values. Therefore, only the aforementioned sub keywords will be displayed.

The following Figure 1-3-10 illustrates the Keyword Options group display for call name mask values.

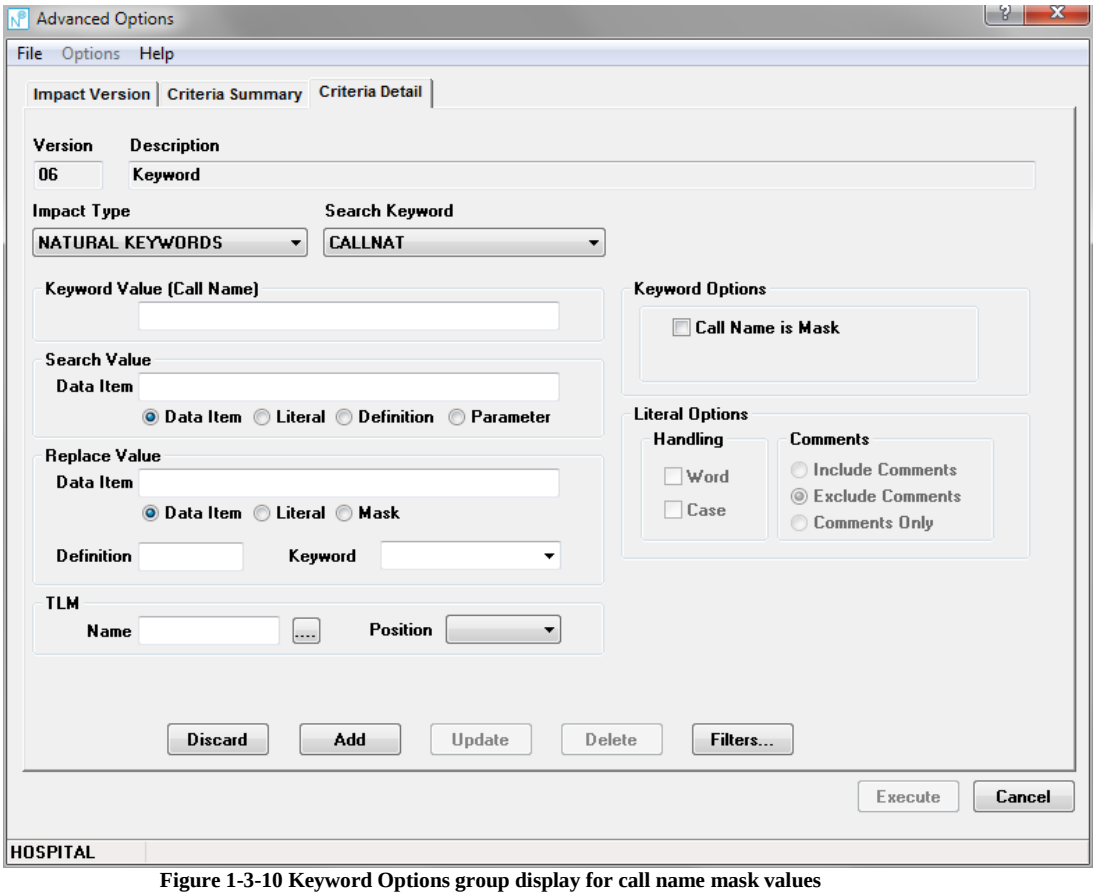

Figure 1-3-10 Keyword Options group display for call name mask values

The following Figure 1-3-11 illustrates the Keyword Options group display for COMPOSE sub keywords.

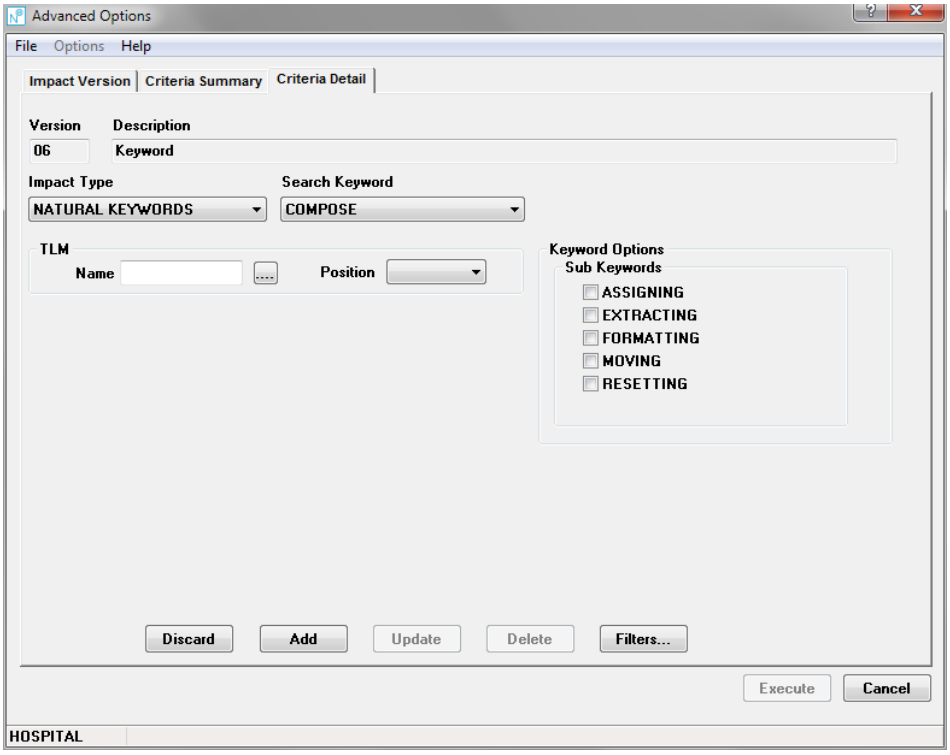

Figure 1-3-11 Keyword Options group display for COMPOSE sub keywords

The following describes the Keyword Options group only. All other screen items are described in the Criteria Detail tab screen section.

SCREEN ITEMS	DESCRIPTION
Call Name is Mask	A tick in the check box indicates selection. If selected, this option identifies the call name used for the Keyword Value as a mask value. This option is only available for any Search Keywords that utilize call names, for example CALLNAT, FETCH and INPUT MAP. The mask value used must be specified using the following convention: **.** (period) Indicates a single position that is not to be checked.

SCREEN ITEMS	DESCRIPTION
***** (asterisk)	Wildcard used to indicate that the last character position is to be checked by the following mask character. For example: *N will check the last character in a call name for a numeric digit.
N	The position is to be checked for a numeric digit.
A	The position is to be checked for an alphabetical character (upper or lower case).
C	The position is to be checked for an alphabetical character (upper or lower case), numeric digit or a blank.
'c'	One or more positions to be checked for the characters bound by apostrophes. For example: 'ABC' will check the call name to contain 'ABC'.

Note: The maximum length for a mask value is 8 bytes.

Examples:

For the following code statements:

0220 CALLNAT 'XX001P01' #PARM

0350 CALLNAT 'XXN01' #FIELD-A #FIELD-B #RESPONSE

1000 CALLNAT 'XXP01A'

1550 CALNAT 'XXABCP1A'

1600 CALLNAT 'ABCMAP&'

A Search Keyword of CALLNAT is used with the following mask values used as Keyword Values:

Mask Value	Statement	Description
AANNNANN	0220	Checks for alphabetic characters in positions 1, 2 and 6. Checks for numeric digits in positions 3, 4, 5, 7 and 8.
…….N	0220	Ignore positions 1 to 7 and check for numeric digit in position 8.
***N**	0220 0350	Check last position for numeric digit.
…….A	1550	Ignore positions 1 to 7 and check for alphabetic character in position 8.

SCREEN ITEMS	DESCRIPTION		
	***A**	1000 1550	Check last position for alphabetic character.
	..'ABC'…	1550	Check for the characters 'ABC' in positions 3, 4 and 5.
	..'ABC'	1550	Check for the characters 'ABC' in positions 3, 4 and 5.
	***'&'**	1600	Check last position for '&'.

Sub Keywords

This will list the appropriate optional clauses for the Natural Keyword selected. The optional clauses available for each supported Natural Keyword are:

Natural Keyword	Optional Clauses
COMPOSE	ASSIGNING EXTRACTING FORMATTING MOVING RESETTING
FIND	FIRST NUMBER UNIQUE RETAIN SORTED
SEND	EVENT METHOD
STACK	COMMAND DATA
WRITE	FORM MAP

Literal Options

The Literal Options are displayed on the Criteria Details tab screen when either the Impact Type LITERAL is selected or if the Search Value group 'Literal' radio button is selected.

It provides the facility to specify further refinement options for literal string search values. This includes case dependencies, searching for exact word values and whether to include comment lines.

The following Figure 1-3-12 illustrates the Literal Options group display.

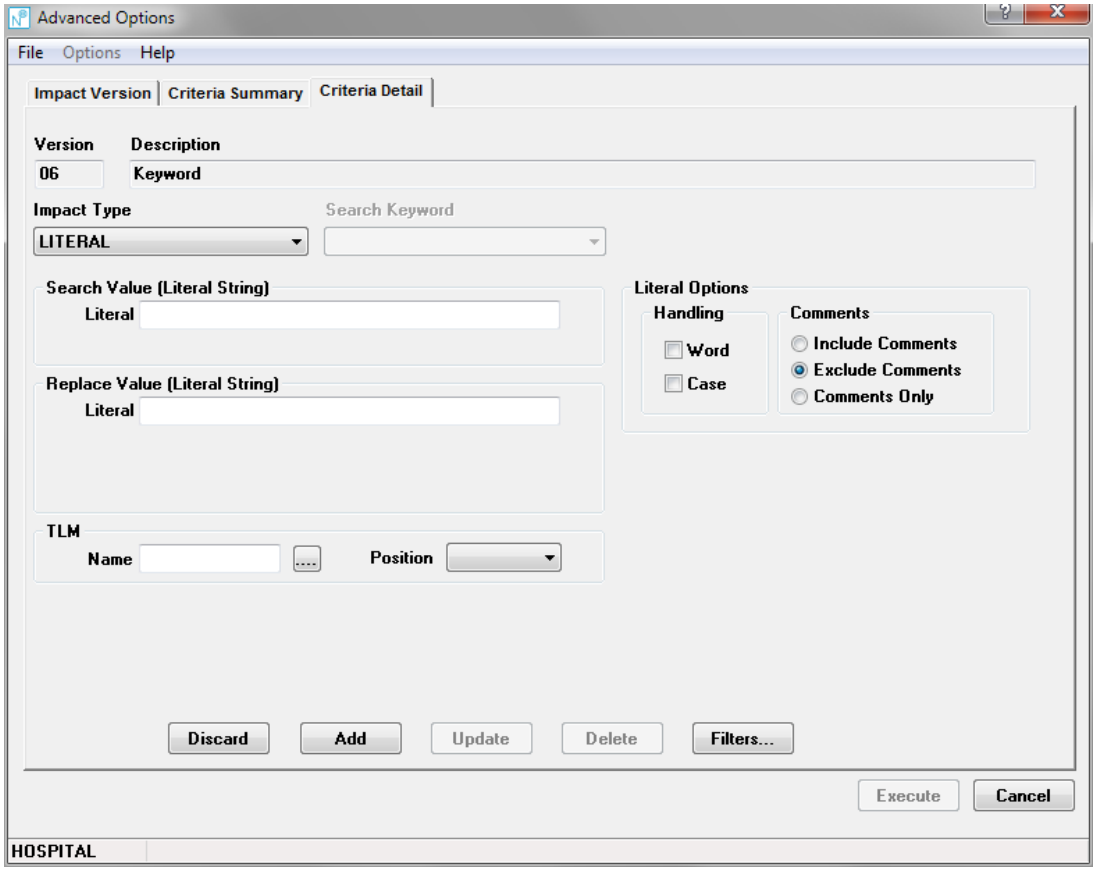

Figure 1-3-12 Literal Options group display

The following describes the Literal Options group only. All other screen items are described in the Criteria Detail tab screen section.

SCREEN ITEMS	DESCRIPTION
Handling	These options will specify the handling characteristics to be used for the literal search value that has been specified. Available selections are: **Word** This treats the literal text string as individual words and the search value is validated against each word. For Example: 'ADABAS and Natural', with 'Word' specified will validate the three values as separate entries against the search value. Therefore, if the search value was ADABAS and there were two literal text strings 'ADABAS and Natural' and 'An ADABAS Database', if WORD was specified then the two literal text strings would be impacted. If WORD was not specified then neither would be impacted. **Case** The Case option determines whether the search value entered is to be searched using the same upper and lower case format as used in literal text strings within the objects. Therefore, if the search value was 'Hello' and there were two literal text strings 'Hello' and 'HELLO', if CASE was specified then only 'Hello' would be impacted. If CASE was not specified then both would be impacted.
Comments	These options provide the handling options for literal strings within comment lines. Available selections are: **Include Comments** Will report any impacts found within comment lines. **Exclude Comments** Will ignore any impacts found within comment lines. **Comments Only** Will only report impacts found within comment lines.

Definition Options

The Definition Options are accessed from the Criteria Detail tab screen by using the 'Definition Options…' button.

The Definitions Options are available when the Impact Type DEFINITION is selected.

It provides the facility to specify further refinement options for data definition search values. For parent data items this includes fixed and variable arrays, constant and initial values, and dynamic fields. For redefinition data items this includes data types and array identification.

The following Figure 1-3-13 illustrates the Criteria Detail tab screen for Impact Type DEFINITION.

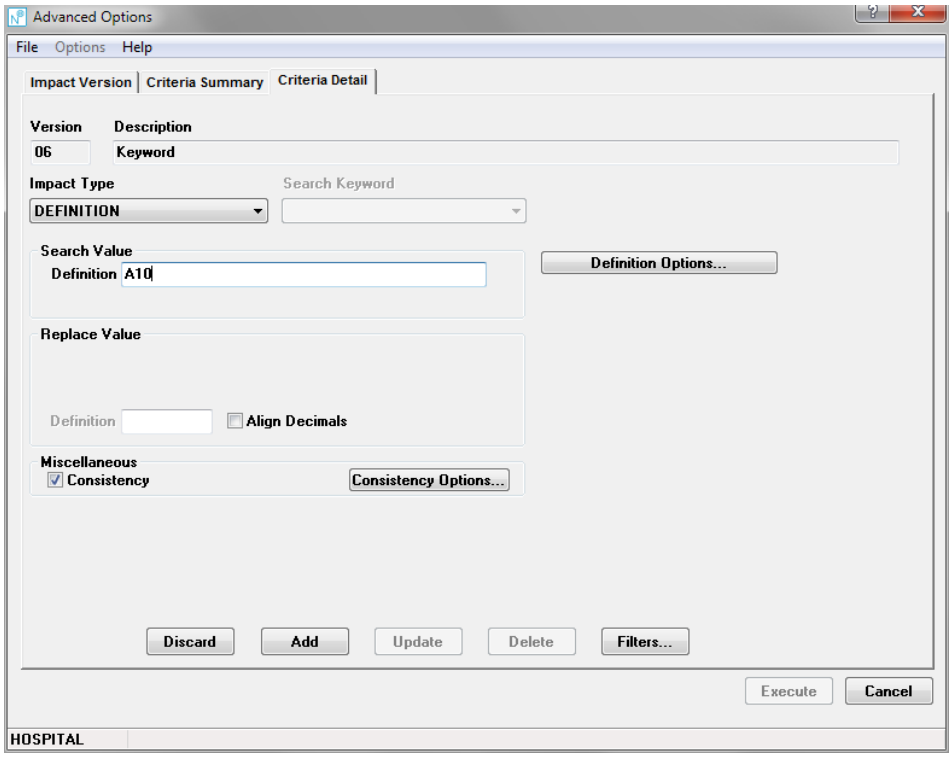

Figure 1-3-13 Criteria Detail tab screen for Impact Type DEFINITION

Definition Options Window

The Definition Options screen uses a multi-purpose 'tabbed' screen to control all the refinement selections. Selecting the required tab will result in the display of the appropriate screen content for the process required.

There are two tab options available:

1. General Tab Screen

Provides a list of general refinement options for parent data items as well as search logic to include either both parent and redefinition options.

The following Figure 1-3-14 illustrates the Definition Options screen with the General tab selected.

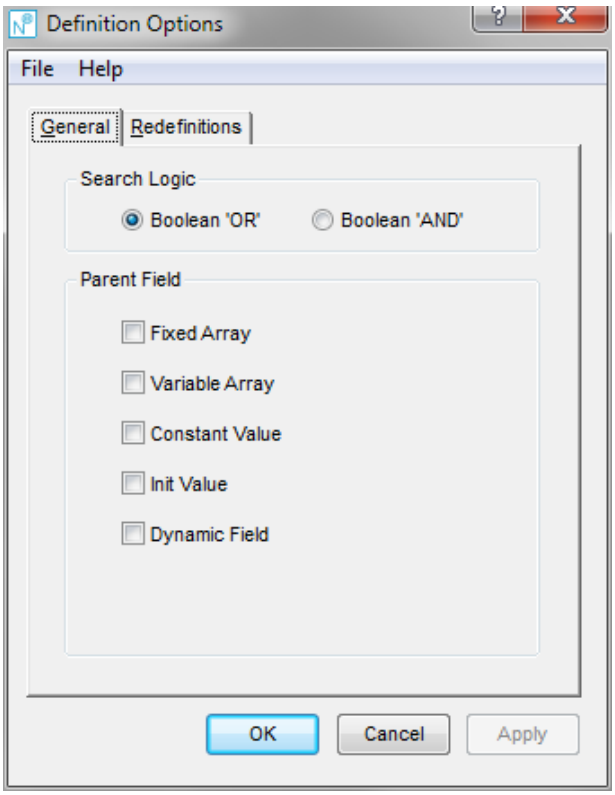

Figure 1-3-14 Definition Options screen with the General tab selected

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Exit**	Exit the Definition Options screen and return back to the Criteria Detail tab screen.
Help	Invoke the Definition Options help.	

SCREEN ITEMS	DESCRIPTION
Search Logic group:	
Search Logic	Determines the relationship between the Parent Field and Redefinition options to be used during the Analysis process.
	Boolean 'OR' Analysis will apply 'OR' logic for the Parent Field and Redefinition options.
	Boolean 'AND Analysis will apply 'AND' logic for the Parent Field and Redefinition options.
	For example:
	Specifying Fixed Array from the Parent Field options and Packed Numeric from the Redefinitions options and running Analysis against the following source code

```
0010 01 #FIXED-ARRAY-1           (A10/1:10)
0020 01 #FIXED-ARRAY-2           (A10/1:10)
0030 01 REDEFINE #FIXED-ARRAY-2
0040   02 #REDEF-ARRAY           (P10/1:10)
```

For Boolean 'OR', the impact result will show

```
0010 01 #FIXED-ARRAY-1           (A10/1:10)
0020 01 #FIXED-ARRAY-2           (A10/1:10)
0030 01 REDEFINE #FIXED-ARRAY-2
```

For Boolean 'AND', the impact result will show

```
0020 01 #FIXED-ARRAY-2           (A10/1:10)
0030 01 REDEFINE #FIXED-ARRAY-2
```

Parent Field group:	
Fixed Array	Search for any data item that is defined as a fixed array.
	For example:

```
01 #FIXED-ARRAY (A10/1:5)
```

SCREEN ITEMS	DESCRIPTION
Variable Array	Search for any data item that is defined as a variable array. For example: `01 #VARIABLE-ARRAY (A10/1:#INDEX)`
Constant Value	Search for any data item that is defined using a CONSTANT value. For example: `01 #CONSTANT-FIELD (A5) CONSTANT <'ABCDE'>`
Init Value	Search for any data item that is defined with an initial value. For example: `01 #INIT-FIELD (A5) INIT <'VWXYZ'>`
Dynamic Field	Search for any data item that is defined using the DYNAMIC clause. For example: `01 #DYNAMIC-FIELD (A) DYNAMIC`

BUTTON NAME	DESCRIPTION
Definition Options screen:	
Ok	Save the Definition Options settings and close the current screen.
Cancel	Cancel the Definition Options process and return back to the Criteria Detail tab screen.
Apply	Save the Definition Options settings and retain the current screen. *Note: This button is only enabled if any changes have been made.*

2. Redefinitions Tab Screen

Provides a list of the refinement options to be applied to redefinition data items.

The following Figure 1-3-15 illustrates the Definition Options screen with the Redefinitions tab selected.

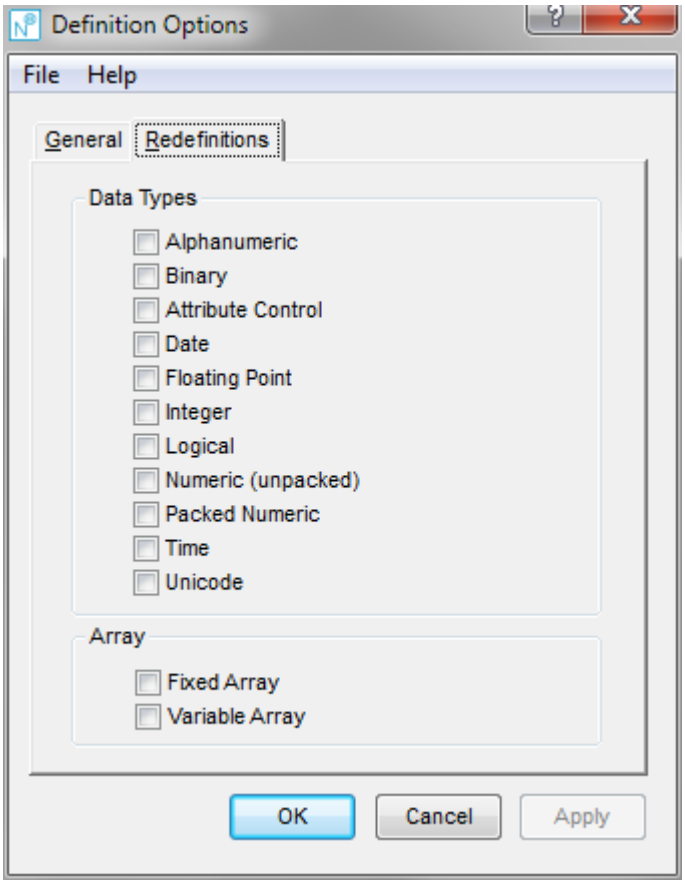

Figure 1-3-15 Definition Options screen with the Redefinitions tab selected

The following describes the Redefinitions tab screen only. All other screen items are described in General tab screen section.

SCREEN ITEMS	DESCRIPTION
Data Types group:	
Alphanumeric	Search for any data item that is defined using format A.
Binary	Search for any data item that is defined using format B.
Attribute Control	Search for any data item that is defined using format C.
Date	Search for any data item that is defined using format D.
Floating Point	Search for any data item that is defined using format F.
Integer	Search for any data item that is defined using format I.
Logical	Search for any data item that is defined using format L.
Numeric [unpacked]	Search for any data item that is defined using format N.
Packed Numeric	Search for any data item that is defined using format P.
Time	Search for any data item that is defined using format T.
Unicode	Search for any data item that is defined using format U.
Array group:	

Fixed Array	Search for any data item that is redefined as a fixed array.
	For example:
	`01 #ALPHA-FIELD (A50)`
	`01 REDEFINE #ALPHA-FIELD`
	`  02 #FIXED-ARRAY (A10/1:5)`

Variable Array	Search for any data item that is redefined as a variable array.
	For example:
	`01 #ALPHA-FIELD (A50)`
	`01 REDEFINE #ALPHA-FIELD`
	`  02 #VARIABLE-ARRAY (A10/1:#INDEX)`

Impact Execution

The Impact Execution option invokes the Impact process, which will execute the Impact Criteria against the application code held in the Repository.

Each criteria is checked against every element of the Repository. If 'Consistency' has been selected for the criterion, every impact found is then re-processed against the code in order to find the impacts of the impacts, using every left-right argument. This tracing process continues through the code until no further impacts or related impacts are found.

The number of times these Iterations occur can be limited by using the Maximum Iterations setting found in the Criteria Summary tab screen.

Note: For more information on the number of iterations refer to the section Maximum Iteration.

Impact Execution is accessed by using the following menu navigation: Analysis➔Impact Execution from the main Natural Engineer screen. A confirmation window is displayed showing the Impact Version to be used.

The following Figure 1-3-16 illustrates the Impact Execution confirmation window.

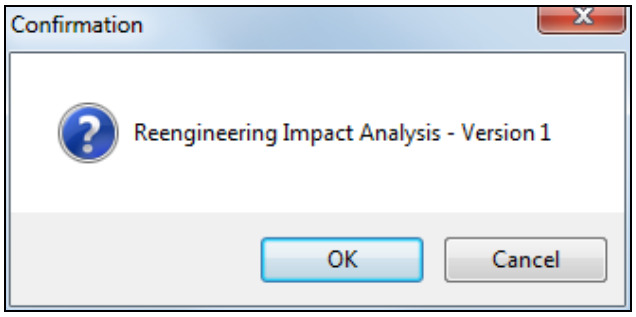

Figure 1-3-16 Impact Execution confirmation window

BUTTON NAME	DESCRIPTION
OK	Invoke the Impact Execution process for the currently selected Impact Version.
Cancel	Cancel the Impact Execution process and return back to the main Natural Engineer screen.

Impact Element Maintenance

The Impact Element Maintenance option provides the facility to review the results of the last executed Impact Analysis for the currently selected version. All impacted objects within an application are available for selection, once selected the impacted items within the object are listed.

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

The impacted items within an object can also be viewed in a Browser.

Impact Element Maintenance Window

The Impact Element Maintenance window is accessed by using the following menu navigation: Analysis➔Impact Element Maintenance from the main Natural Engineer screen.

The following Figure 1-3-17 illustrates the Impact Element Maintenance screen.

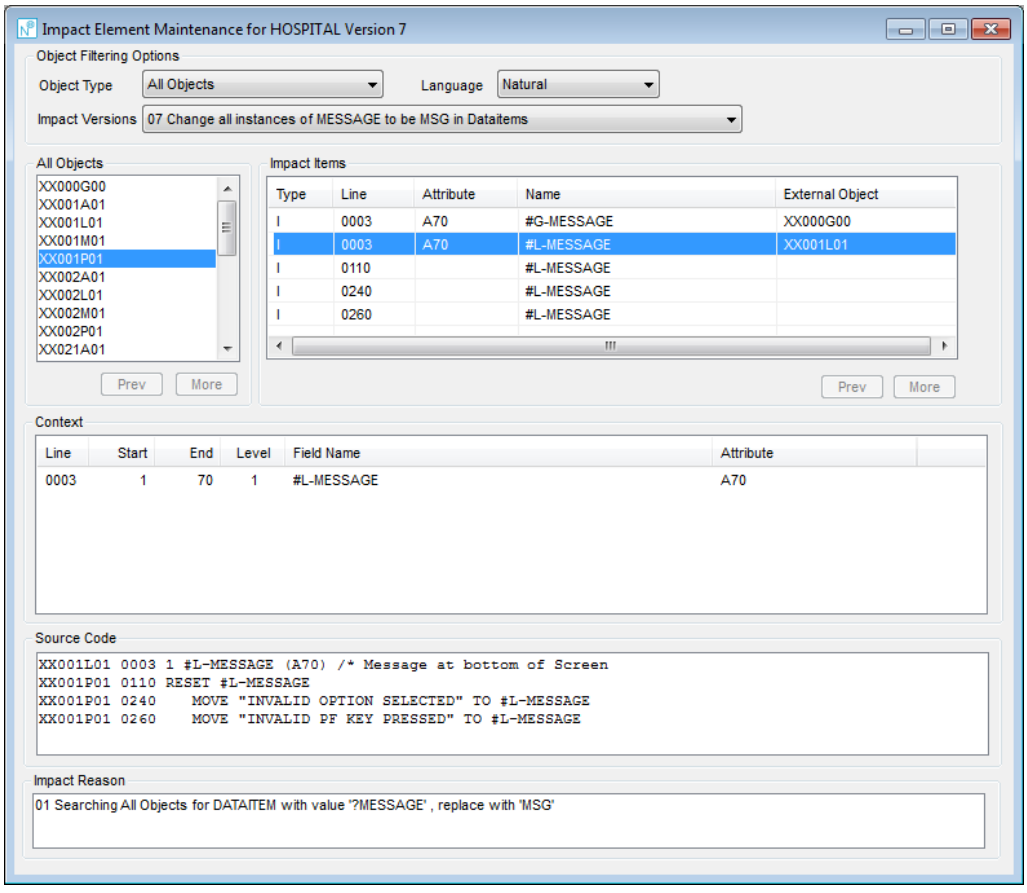

Figure 1-3-17 Impact Element Maintenance

SCREEN ITEMS	DESCRIPTION
Object Filtering group:	
Object Types	Allows you to select the types of object to be listed. The objects shown are dependent on the Objects present within your application.
Language	Allows you to select the programming language of the objects to be listed. Available selections are: ▪ **All** ▪ **Cobol** ▪ **JCL** ▪ **Natural**
Impact Version	Change the Impact version to review alternate Impact results for the application.
Object List group:	
Object List	List of all the impacted objects for the currently selected Impact Version. The list of objects can be tailored to your requirements using the options available in the Object Types and Language menus. Further refinement can be made using the option 'Change Start Position of Object List…' from the View menu. The Object List title reflects the Object Types being listed and will append any reposition values that may have been specified. A context menu is available to invoke viewing options View Structure Diagram for Search Criteria (for the selected object only) or View Impacted Code by using the **right hand mouse button** on a selected object. *Note: For more information on the Object List context menu, refer to section Impact Element Maintenance Context Menu.*
Impact Item group:	
Impact Items	List of all the impacted items for the currently selected object. A context menu is available to invoke viewing options Enter Filter Value for Impact Items by using the **right hand mouse button** on an entry in the Impact Items box to tailor the list to your requirements. The columns available are: **Type** This is a 2-byte value, which denotes the type of Impact. *Note: For more information on type of impact refer to the section Impact Types.*

SCREEN ITEMS	DESCRIPTION	
	Line	The statement line number for the impact item within the selected object.
	Attribute	The format and length of the impact item if the item is a data item from a data definition area within the object.
	External Object	The name of the object that contains the impact item if the item is in an external object, for example GDA, LDA, PDA or Copycode.
	Name	The name of the impact items impacted within the selected object.

Context List group:

Context List Displays the context of the selected data item within the data definition of the selected object and the relative offsets if applicable.

Note: Only displays details for impacted data items from data definition areas.

If the selected data item is a literal string, then no context details will be displayed.

If the selected data item is part of a view definition, group or redefinition, then the context list will show data items one level lower and one level higher in relation to the selected data item.

Some examples based on the following data definition:

```
0020 01 #GROUP
0030   02 #ALPHA(A10)
0040   02 REDEFINE #ALPHA
0050     03 #NUMERIC(N6)
```

For data item #GROUP, the context list will show:

```
0020            01 #GROUP               G
0030    1   10  02 #ALPHA               A10
```

For data item #ALPHA, the context list will show:

```
0020            01 #GROUP               G
0030    1   10  02 #ALPHA               A10
0040            02 REDEFINE #ALPHA
0050    1    6  03 #NUMERIC             N6
```

For data item #NUMERIC, the context list will show:

```
0030    1   10  02 #ALPHA               A10
0040            02 REDEFINE #ALPHA
0050    1    6  03 #NUMERIC             N6
```

SCREEN ITEMS	DESCRIPTION
Source Code group:	
Source Code	Displays all the impacted statement references for the selected impact item. The columns available are: **Object Name** The name of the object in which the Impact is identified. This will normally be the object selected, but can be an external object such as a GDA, LDA, PDA or Copycode, which is included in the selected object. **Line Number** The line number of the impacted statement code. **Statement** The statement code which is impacted.
Impact Reason group:	
Impact Reason	Displays information on the reasons for the impact based on the search criteria specified.

BUTTON NAME	DESCRIPTION
Object List group:	
Prev	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.
More	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.
Impact Item group:	
Prev	Scrolls the impact items to previous page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
More	Scrolls the impact items forward one page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.

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

Impact Types

The following table shows all of the available impact types by criteria group:

Type	Description
Standard	
I	Data Item.
ID	Data Item - Derived fields.
M	Database Access.
A	Database File and Field.
AD	Database File and Field - Derived fields.
K	Keyword.
T	Literal.
TC	Literal comments.
Q	Definition.
QD	Definition - Derived fields.
S	Source code change.
Adjust	
J	Adjust.
Code Improvement	
7A	FIND <de=val> OR <de=val>.
7B	FIND <de> GE <val> AND <de> LE <val>.
7C	FIND <de=val> AND <de=val>.
7D	FIND <de=val> WHERE <non-de>.
7E	FIND <de> AND <non-de>.
7F	Nested READ/FIND loops.
7G	Assignment stmts with different format/lengths.
7H	REPEAT UNTIL/WHILE.
7I	Comparison statements arrays vs. literals.
7J	SUBSTRING.
7K	HISTOGRAM.

Type	Description
7L	Numerical fields in calculations.
7M	Arrays within group fields.
7N	External Objects.
7O	Callnats and number of parms.
7P	System variables referenced GT 1.
7Q	PDA fields in calculations.
7R	Array assignments in non-db loops.
7S	Assignment stmts with different format/lengths.
7T	Alpha literal values and variables.
7U	DECIDE ON using system variables.
7V	Insert RECORD option for READ WORK FILE.
7W	Move Occurrence No. to each PE member.
7X	Replace MOVE INDEXED with appropriate MOVE.
7Y	FOR & REPEAT loops to use named constants.
7Z	Find unused dataitems in programs.
71	Split STACK COMMAND stmts with embedded data.
72	Find unused Global variables.
73	Find unused source code lines.
Object Builder	
G	Object Builder Line Range.
P	Object Builder Field Element.
V	Object Builder View Used.
C	Object Builder Initial/Temporary Impact.
X	Object Builder field external to line range referencing a line within the line range.
Y	Object Builder Escape Routine.

Natural RPC

4A Natural keyword FETCH prohibited.

4B Natural keyword RUN prohibited.

4C Natural keyword INPUT prohibited.

4D Natural keyword STOP works the same as ESCAPE ROUTINE.

4E Natural keyword TERMINATE works the same as ESCAPE ROUTINE.

Multi Search

1 Specified

1D Derived

1X Excluded

Natural version 2.2-3.1 Conversion

3a Redefinition of DB Arrays.

3b DEFINE WINDOW Minimum Size.

3c DIVIDE and Decimal Positions. The actual DIVIDE statement.

3h DIVIDE and Decimal Positions. The Data Definition affected.

3d Comparison Logic for MU's in FIND..WITH.

3e Empty Statement Blocks. IGNORE inserted into empty statement block.

3@ Empty Statement Blocks. Empty Statement block commented out and maximum variable value inserted.

3g No uppercase translation for *COM.

3I Results of SIN, COS and TAN functions.

3j More precise SQRT Function Results.

3k Assignments of Numbers with Decimals to Time Fields.

3l MOVE RIGHT JUSTIFIED where target field is smaller than source field.

3m Negative Values to Date Fields.

3n More precise results for Floating Point Conversions and computation of floating point exponentiation.

3o Comparison and Assignment of Variable Array Ranges.

3p	NAT1117 and NAT0924 replaced by NAT0082.
3q	Obsolete Error Messages.
3r	Changed System Variable *TPSYS under BS2000.
3s	Priority of PRINT/WORK FILE Statements.
3t	Usage of User Exit Modules Copies from SYSEXT.
3u	Internal Handling of AD=O.
3v	EJECT Statement Required Operand LESS.
3w	ESCAPE not Valid AT START OF DATA. ESCAPE TOP and ESCAPE BOTTOM not allowed in ON ERROR blocks.
3x	Decimal Digits of Constant Values.
3y	NEWPAGE Statement Required Operands LESS/TOP.
3z	PRINT Statement LS Parameter invalid.
31	BEFORE BREAK within IF condition invalid.
32	SUBSTRING clause, where the offset plus the length of the substring must not exceed the length of the field.
33	MOVE BY NAME statement where redefinition of alpha fields to numeric exist and both source and target numeric field are same length.

Porting

6A	Alpha variables with redefinition variables defined using formats Binary, Integer or Packed.
6B	Numeric variables that are moved to Alpha variables.

Refactoring

81 Screen I/O within database processing loops.

82 Missing database labels.

83 Missing non-database labels.

84 Convert numeric back references to labels.

85 Find unused source code lines.

86 Find unused dataitems in programs.

87 Duplicate fields.

88 I/O statements in copycode.

89 DB statements in copycode.

8A Similar code identifier.

System Functions

0A ABS.

0B ATN.

0C COS.

0D EXP.

0E FRAC.

0F INT.

0G LOG.

0H SGN.

0I SIN.

0J SQRT.

0K TAN.

0L VAL.

0M OLD.

0N SUM.

0O AVER.

0P TOTAL.

0Q NAVER.

0R MAX.

0S	MIN.
0T	NCOUNT.
0U	COUNT.
0V	NMIN.
0W	SORTKEY.
0X	POS.
0Y	RET.
0Z	*TRANSLATE.
01	*TRIM.

Impact Element Maintenance Context Menu

The Impact Element Maintenance context menu is invoked by placing the cursor on any of the items listed in the Object or Impact Items lists and using the right hand mouse button with a single click.

Object List Context Menu

The Object List context menu allows viewing options View Structure Diagram for Search Criteria (for the selected object only) or View Impacted Code to be invoked. It also provides the option to reposition the object list.

Note: The option View Impacted Code is not available when displaying DDM objects.

The following Figure 1-3-18 illustrates the Object List context menu.

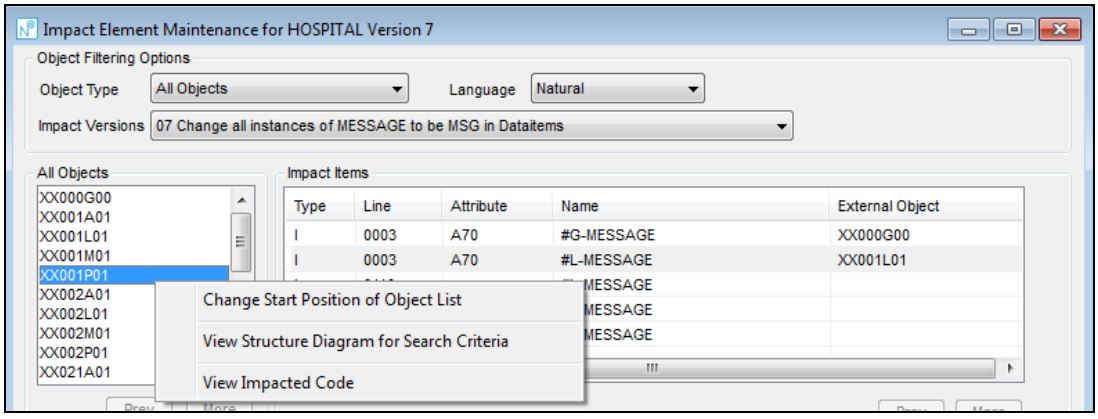

Figure 1-3-18 Object List context menu

<table>
<tr><th>CONTEXT MENU ITEM</th><th>DESCRIPTION</th></tr>
<tr><td>Change Start Position of Object List…</td><td>Reposition the list of objects to start from a particular object name.

The reposition value can be input using either a complete name or part name using an '*' (asterisk) wildcard.

The reposition value is appended to the object list title to highlight the type of repositioning being applied.

Possible reposition values are:

<table><tr><th>Value</th><th>Result</th></tr><tr><td>' ' (blank)</td><td>Reposition to the top of the object list.</td></tr><tr><td>*</td><td>Reposition to the top of the object list.</td></tr><tr><td>ABC*</td><td>Only show objects that are prefixed by 'ABC'.</td></tr><tr><td>XYZ</td><td>Reposition to the first object that either matches or is greater than 'XYZ' and then continue the object list from that point.</td></tr></table></td></tr>
<tr><td>View Structure Diagram for Search Criteria…</td><td>Invoke GenTree Structure Analyzer to display the impacts made for the specified search criteria for the selected object.

Note: For more information on GenTree refer to Chapter 2 in the Natural Engineer Reporting manual.</td></tr>
<tr><td>View Impacted Code</td><td>Display the impacts within an object using the Browser. The whole object source code will be displayed with the impacted items highlighted using the colors set by the SPECIFIED, DERIVED and EXCLUDED parameters in the NATENG.INI file.

Note: For more information on the NATENG.INI file parameters SPECIFIED, DERIVED and EXCLUDED refer to Chapter 1 in the Natural Engineer Administration Guide for Windows manual.</td></tr>
</table>

Impact Items Context Menu

The Impact Items context menu provides the option to filter the Impact Items list.

The following Figure 1-3-19 illustrates the Impact Items context menu.

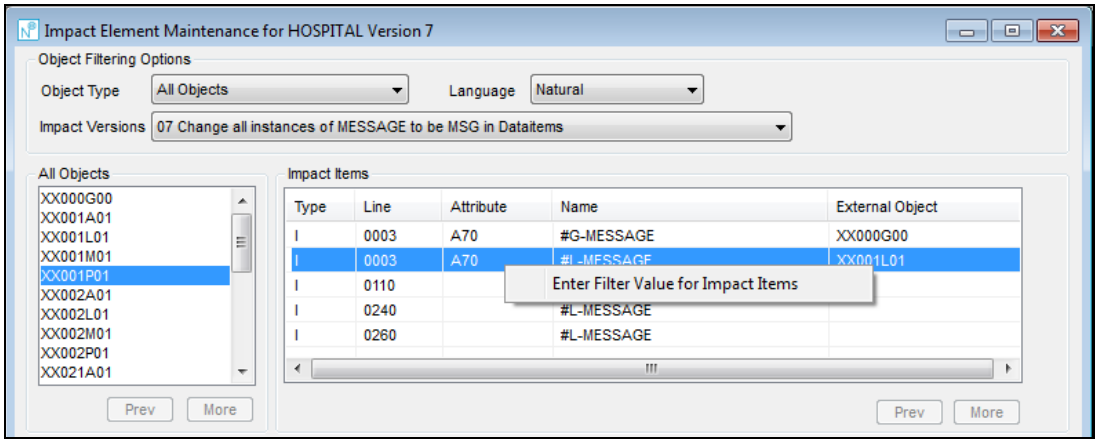

Figure 1-3-19 Impact Items context menu

CONTEXT MENU ITEM	DESCRIPTION
Enter Filter Value for Impact Items	Filters the list of impact items to show impacts for a particular impact item name only. The filter value can be input using either a complete name or part name using an '*' (asterisk) wildcard. The filter value is appended to the impact item list title to highlight the type of filter being applied. Possible filter values are:

Value	Result
' ' (blank)	Reposition to the top of the impact item list.
*****	Reposition to the top of the impact item list.
ABC*	Only show impact items that are prefixed by 'ABC'.
XYZ	Only show impact items that are named 'XYZ'.

Impact Element Maintenance GenTree Structure Analyzer

Impact Element Maintenance makes use of the GenTree Structure Analyzer to display the impacts for each search criteria for the currently selected Impact version in a Tree View.

The GenTree Structure Analyzer is invoked from the Object List context menu. This will only display the details for the object selected.

The GenTree diagram will display all the search criteria specified within the Impact version. For each search criteria, the object or objects that have been impacted will be displayed. For each impacted object, the impacted statement line numbers and search value will be displayed.

From the diagram, it is possible to view the source code, preview the map (map objects only), data definitions and object properties by using the right hand mouse button over each object in the diagram.

Note: For more information on GenTree Structure Analyzer refer to Chapter 2 in the Natural Engineer Reporting manual.

The following Figure 1-3-20 illustrates a GenTree Structure Analyzer diagram for View Structure Diagram for Search Criteria.

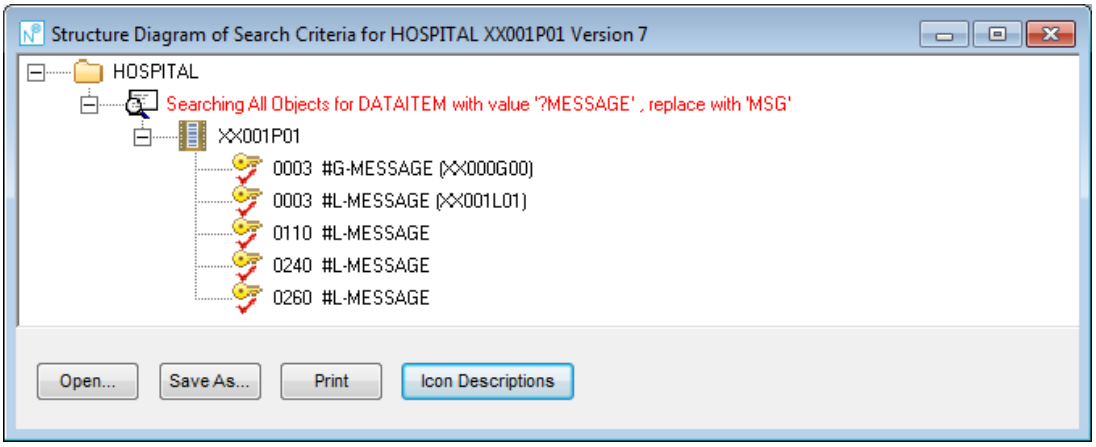

Figure 1-3-20 GenTree Structure Analyzer diagram for View Structure Diagram for Search Criteria

Impact Analysis Inventory

The Analysis Inventory consists of a set of Impact Reports, which provide various types of information concerning the Impact Analysis, including a view of used Search Criteria. Reports are available at the summary, object and detailed data item levels.

The information provided by the Impact reports complements the information found in the Impact Element Maintenance screen.

The Impact Reports can be accessed using the following menu navigation: Analysis➔ Impact Reports.

The following list illustrates the Impact Reports that are available:

- Search Criteria
- Application Impact Summary
- Object Impact Summary
- All Impacts
- Impacts by Impact & Object Type
- Impacted External: Objects
- Impacted External: Interfaces
- Impacted Construct Models
- Impacted Predict Case Components
- Data Item Impact Inventory
- Impacted Steplib Inventory
- Data Item Impact Usage Inventory
- View Impacted Source Code

The Impact Reports option becomes available after Impact Execution has been run.

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

2

MODIFICATION PROCESSES

Chapter Overview

The Modification processes provide all the facilities to modify object source code for the objects within an application held on the Repository. The Modification process relies on the Analysis information generated by the Impact execution and is controlled by the Impact Version process.

Once Modification has been executed, there are various reporting options to view the results either online or using textual reports.

All the Modification processes are available from the Modification menu.

The topics covered in this chapter:

1. Modification Preferences
2. Modification Element Maintenance
3. Execute Modification for All Objects
4. Modification Inventory

Modification Preferences

The Modification Preferences option is used to specify override TLMs for an application and is accessed from the Modification menu.

TLMs are Natural objects with an object type of Text, containing the required processing code to be used during modification. They need to exist on either the modification library specified in the application properties or, can be held on the Natural SYSTEM library.

After the override TLMs have been saved, they need to be defined using the Modification Preferences option in order that Natural Engineer can recognize them and use them during the modification process.

When override TLMs are specified for an application, they will override any site wide TLM settings that are in place, across all versions for the currently selected application.

Note: Site wide TLMs are specified using the Default Text Logic Members option found using the following menu navigation: Options➔Administration➔Default Text Logic Members.

For more information on the Default Text Logic Members option refer Chapter 1 in the Natural Engineer Administration Guide for Windows manual.

Supplied Sample Text Logic Members

Natural Engineer comes supplied with four Text Logic Member (TLM) objects that can be adapted and used for the Modification Preferences option.

The sample TLMs supplied are:

1. TLMSTART
2. TLMCOM
3. TLMAFTER
4. TLMDYNAM

Note: These objects can be found on the Natural Engineer SYSNEE library and will need to be moved to either SYSTEM or modification libraries as required. If moved to the SYSTEM library, they will be available to all modification libraries.

New TLMs can be generated and added to the required Natural library and their usage must correspond to the TLM types that are available.

TLMSTART

This is an example ‘Start’ TLM. It may be used for Structured Mode or Reporting Mode objects.

```
0010 * ------------------------------------------------------------
0020 * TLMSTART - START
0030 * ------------------------------------------------------------
0040 FORMAT PS=24 LS=132
0050 * ------------------------------------------------------------
0060 * TLMSTART - END
0070 * ------------------------------------------------------------
```

This will get inserted before the first line of procedural code in an object e.g., after the definitions in a Structured Mode program.

TLMCOM

This is an example ‘Comment’ TLM. It will get inserted at the start of an object for object types: Program, Subprogram and Subroutine.

```
0010 /* ------------------------------
0020 /* TLMCOM-START
0030 /* ------------------------------
0040 /* Updated by: XX-USER Dated: XX-DATE Time: XX-TIME using NEE
0050 /* ------------------------------
0060 /* TLMCOM -END
0070 /* ------------------------------
```

The XX-USER will be translated to *USER contents
The XX-DATE will be translated to DD/MM/YYYY from *DATX
The XX-TIME will be translated to HH:II:SS from *TIMX
The user can add additional code to their own specifications.

TLMAFTER

This is an example ‘After Field’ TLM for a field that would get inserted after a particular keyword or data item if specified in the Impact Search criteria or on the Modification Element Maintenance screen.

```
0010 * ------------------------------------------------------------
0020 * TLMAFTER - START
0030 * ------------------------------------------------------------
0040 PERFORM ##VALIDATION
0050 * ------------------------------------------------------------
0060 * TLMAFTER - END
0070 * ------------------------------------------------------------
```

TLMDYNAM

This is an example ‘Dynamic operand replacement’ TLM. It is used when a TLM has been used to modify a statement and retain the original statement operands.

For more information refer to the section Dynamic Operand replacement in Text Logic Members.

```
0010 /* ------------------------------------------------------------------
0020 /*
0030 /* Sample TLM to show dynamic replacement of operands.
0040 /* This example shows how to use dynamic operand replacement to
0050 /* convert all MOVE statements to ASSIGNS
0060 /*
0070 /* Impact Criteria = MOVE with replacement TLM set of TLMDYNAM
0080 /*
0090 /* Up to 10 dynamic operands per statement may be specified.
0100 /* Conditional logic is specified via **NEE XX-OPERn and closed via
0110 /* **NEE BLOCK-END
0120 /*
0130 /* START OF TLMDYNAM
0140 /*
0150 /* ------------------------------------------------------------------
0160 ASSIGN XX-OPER2 = XX-OPER1
0170 **NEE XX-OPER3
0180 ASSIGN XX-OPER3 = XX-OPER1
0190 **NEE BLOCK-END
0200 **NEE XX-OPER4
0210 ASSIGN XX-OPER4 = XX-OPER1
0220 **NEE BLOCK-END
0230 **NEE XX-OPER5
0240 ASSIGN XX-OPER5 = XX-OPER1
0250 **NEE BLOCK-END
0260 **NEE XX-OPER6
0270 ASSIGN XX-OPER6 = XX-OPER1
0280 **NEE BLOCK-END
0290 /* ------------------------------------------------------------------
0300 /* END OF TLMDYNAM
0310 /* ------------------------------------------------------------------
```

Dynamic Operand replacement in Text Logic Members

If a TLM is used to modify a statement, it is possible to retain the use of the operands from the original statement within the TLM.

For example, if all MOVE statements are to be replaced by ASSIGN statements, the following search criteria could be specified:

Search Keyword = MOVE,

Replace TLM value = 'TLMDYNAM'

Replace Position set to REPLACE.

Note: For details on the sample TLM: TLMDYNAM refer to the section Supplied Sample Text Logic Members.

Points to note using Dynamic Operand replacement TLMs:

1. Operands in the original statement are referenced in the TLM by using XX-OPERn. A maximum of 10 operands can be replaced this way, i.e., XX-OPER1 to XX-OPER10.
2. The replace TLM will only be activated with position set to REPLACE (search criteria specification). If position AFTER or BEFORE is used, then the dynamic operand replacement will not occur.
3. Conditional logic can be used within the TLM, by using the '**NEE' notation at the start of the statement line.

In the sample TLM above at line 0170, the statement '**NEE XX-OPER3' is an IF condition checking to see if XX-OPER3 exists in the original statement. If it does then the statement ASSIGN XX-OPER3 = XX-OPER1 will be included in the modified object. The '**NEE BLOCK-END' statement acts as an END-IF.

Example of Dynamic Operand replacement in a TLM

The following example illustrates a simple object which will have a modification applied changing the MOVE statement to ASSIGN using the example TLM illustrated above.

Sample object before modification:

```
0010 DEFINE DATA LOCAL
0020 01 #A            (A10)
0030 01 #B            (A10)
0040 01 #C            (A10)
0050 01 #D            (A10)
0060 01 #E            (A10)
0070 END-DEFINE
0080 /*
0090 MOVE #A TO #B #C #D #E
0100 END
```

The operands for dynamic replacement are:

#A for XX-OPER1
#B for XX-OPER2
#C for XX-OPER3
#D for XX-OPER4
#E for XX-OPER5

After modification the object code will look like this:

```
0010 DEFINE DATA LOCAL
0020 01 #A            (A10)
0030 01 #B            (A10)
0040 01 #C            (A10)
0050 01 #D            (A10)
0060 01 #E            (A10)
0070 END-DEFINE
0080 /*
0090 /* MOVE #A TO #B #C #D #E /* NEE OLD CODE
0100 /* ------------------------------------------------------------------
0110 /* START OF TLMDYNAM
0120 /* ------------------------------------------------------------------
0130 ASSIGN #B = #A
0140 ASSIGN #C = #A
0150 ASSIGN #D = #A
0160 ASSIGN #E = #A
0170 /* ------------------------------------------------------------------
0180 /* END OF TLMDYNAM
0190 /* ------------------------------------------------------------------
0200 END
```

Note: The leading comments from the supplied sample TLM: TLMDYNAM have been removed for the purpose of this example.

Modification Preferences Window

All the specifications for the override TLMs are defined using the Modification Preferences screen. This is accessed by using the following menu navigation: Modification➔Modification Preferences from the main Natural Engineer screen.

The following Figure 2-1 illustrates the Modification Preferences screen.

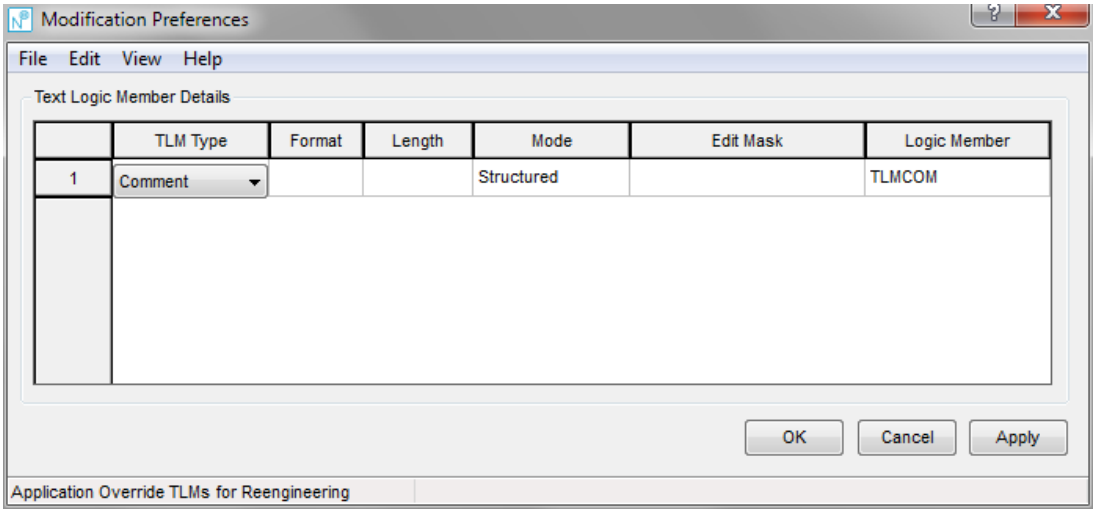

Figure 2-1 Modification Preferences screen

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Exit**	Exit the Modification Preferences screen and return back to the main Natural Engineer screen.
Edit	**Insert Row**	Add a new row into the Text Logic Member Details list box.
	Delete Row	Delete an existing row from the Text Logic Member Details list box.
View	**Site Wide / Application Override**	This option provides the facility to flip between the application override TLM details and the site wide TLM details. The option changes as follows: When application override details being displayed: '**Site Wide TLMS for Reengineering**' When site wide details being displayed: '**Application Override TLMs for Reengineering**' *Note: The site wide TLM details cannot be modified using the Modification Preferences screen. For more information on the site wide TLMs refer to Chapter 1 in the Natural Engineer Administration Guide for Windows manual.*
Help		Invoke the Modification Preferences help.

SCREEN ITEMS	DESCRIPTION	
TLM Type	Specify what type of TLM is defined. Valid values are:	
	START	A TLM to be inserted at the Start of an Object. This is after the definition of the data items in the object.
	DATA	A TLM that provides data items to be included in an object.
	MISC	A TLM that is placed at the end of the object that can contain processing, for example including common routines.
	COMMENT	To be inserted at the start of the object to explain another TLM inserted in the object. The following variables can be specified and will be replaced at remedy execution. • XX-DATE, which will be translated into DD/MM/YYYY • XX-TIME, which will be translated into HH:MM:SS • XX-USER, which will contain the user-id of the person who executed modification for the object.
	CMPT COMM	The Component comment inserted at the start of the new component subprogram that has been created.
	CMPT DATA	Component parameter data inserted as the last parameter passed in the subprogram. The TLM data must be specifically coded in this routine and must contain the following definition first. 01 #EXTRA-PDA If a component TLM is required to pass a data item #RESPONSE between the new subprogram and the object calling it, then the following is the structure for this TLM: 01 #EXTRA-PDA 02 #RESPONSE (A1)

SCREEN ITEMS	DESCRIPTION	
	SAG05 R1	This is the default modification for empty FOR and REPEAT statement blocks. The TLM will insert the keyword IGNORE into the empty block. For Example: 1020 REPEAT 1030 IGNORE 1040 END-REPEAT *Note: Used in Nat 2.2 to Nat 3.1 conversion.*
	SAG05 R2	This TLM type can be used as an alternative to the default SAG05R1. This will comment out the empty statement block but then insert a line of code to set the applicable variable to the maximum value. For Example: FOR #A = 1 TO 10, will insert MOVE 10 TO #A. This will only be applied to a FOR loop block, a REPEAT loop block will only get commented out. Additionally, if this TLM type is selected in the preference screen, then prior to modification the update field button will need to be used on the Modification Element Maintenance screen to ensure the correct TLM is applied during modification. *Note: Used in Nat 2.2 to Nat 3.1 conversion.*
Format	The format of the data item the TLM relates to. (Not used at present.)	
Length	The length of the data item the TLM relates to. (Not used at present.)	
Mode	Programming mode to which the TLM applies. Valid values are:	
	Structured	Structured mode
	Reporting	Reporting mode
Edit Mask	The specific edit mask for the data item that the TLM relates to. (Not used at present.)	
Logic Member	Name of the TLM to be used.	

BUTTON NAME	DESCRIPTION
OK	Save the Modification Preferences settings and close the current screen.
Cancel	Cancel the Modification Preferences process and return back to the main Natural Engineer screen.
Apply	Save the Modification Preferences settings and retain the current screen. *Note: This button is only enabled if any changes have been made.*

STATUS BAR ITEM	DESCRIPTION
The Modification Preferences status bar is divided into 2 individual panes.	
Pane 1	The name of TLM details being displayed.
Pane 2	Any Modification Preferences processing messages.

Modification Element Maintenance

The Modification Element Maintenance option provides the facility to review and modify interactively, the default modifications to be applied to objects from the last Impact execution for the currently selected version within an application. All impacted objects within an application are available for selection; once selected a list of the impacted items within the object are listed.

The Modification Element Maintenance option allows each modification to be updated to change the modification types, categories and replacement values as desired. The Modification changes to be applied can be reviewed before they are implemented, using the Browser.

The Modification Element Maintenance option also provides the facility to review the Impact results in the Browser.

Modifications to single objects can be implemented from this option.

Modification Element Maintenance Window

The Modification Element Maintenance screen is accessed by using the following menu navigation: Modification➔Modification Element Maintenance from the main Natural Engineer screen.

The following Figure 2-2 illustrates the Modification Element Maintenance screen.

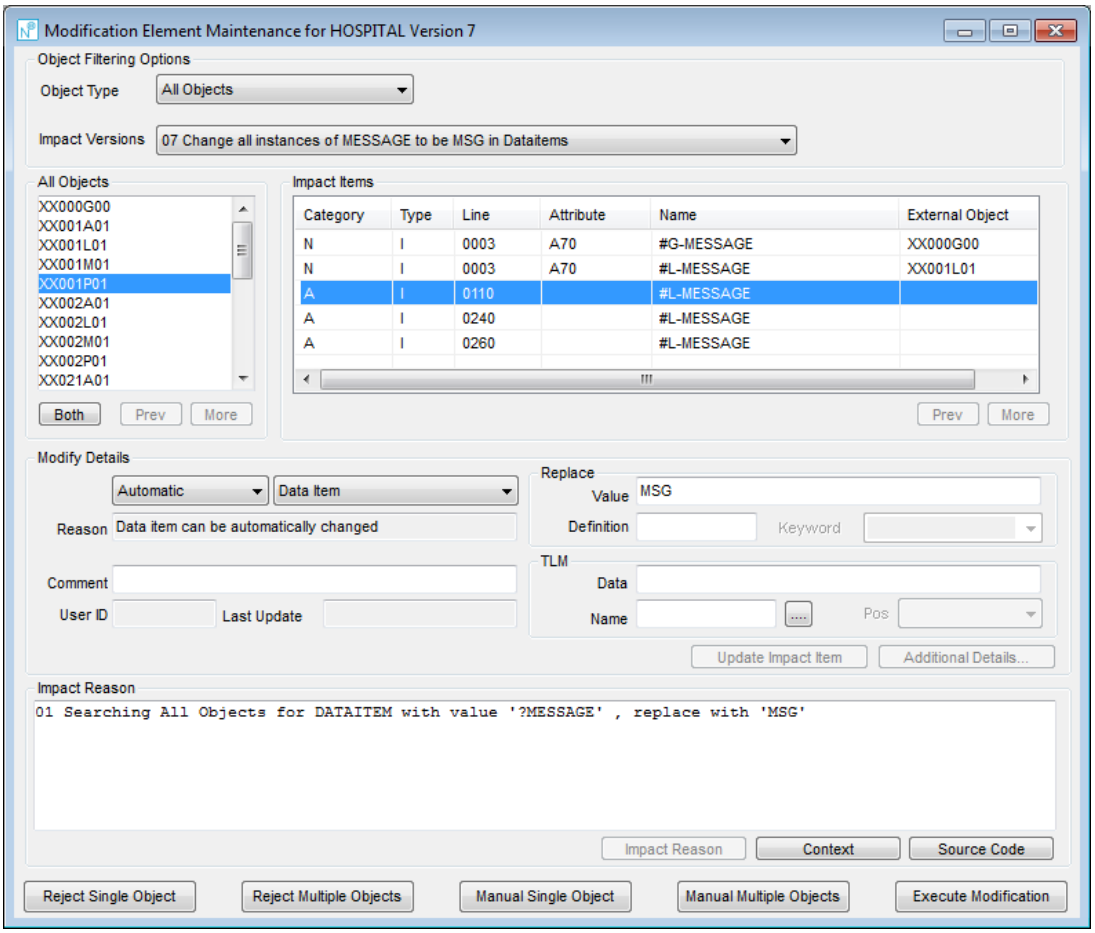

Figure 2-2 Modification Element Maintenance screen

SCREEN ITEMS	DESCRIPTION
Object Filtering group:	
Object Types	Allows you to select the types of object to be listed. The types of object shown are dependent on the objects loaded within your application.
Impact Version	Change the Impact version to review alternate Impact results for the application.
Object List group:	
Object List	Lists all the impacted objects for the current Impact Version. The list of objects can be tailored to your requirements using the options available in the Object Types menu. Further refinement can be made using the option 'Change Start Position of Object List…' from the View menu. The Object List title reflects the Object Types being listed and will append any reposition values that may have been specified. Any objects which have had Modification applied will have an asterisk (*) after the object name. A context menu is available to invoke viewing options View Impacted Code or Preview Modified Code by using the **right hand mouse button** on a selected object. *Note: For more information on the Object List context menu, refer to section Modification Element Maintenance Context Menu.*
Impact Items List group:	
Impact Items List	Lists all the impact items for the selected object. A context menu is available to invoke viewing options Enter Filter Value for Impact Items by using the right hand mouse button on an entry in the Impact Items box to tailor the list to your requirements. The Impact Items List title will append any filter values that may have been specified. The columns available are: **Category** — This is a 1-byte value, which denotes the Category of Modification to be applied for the impact item. *Note: For more information on these categories refer to the section Modification Categories.* **Type** — This is a 2-byte value, which denotes the type of Impact. *Note: For more information on type of impact refer to the section Impact Types.*

SCREEN ITEMS	DESCRIPTION	
	Line	The statement line number for the impact item within the selected object.
	Attribute	The format and length of the impact item if the item is a data item from a data definition area within the object.
	External Object	The name of the object that contains the impact item if the item is in an external object, for example GDA, LDA, PDA or Copycode.
	Name	The name of the impact items impacted within the selected object.
Modify Details group:		
Modify Details	The Modify Details group provides the facility to review and update modification details for a selected impact item. Any changes made need to be saved using the '**Update Impact Item**' button.	
	Modification Category	All the available modification categories are listed. Each one can be selected to change the default category provided by Impact execution. After the '**Update Impact Item**' button is used, the new category code will appear in the Impact Items list box under the 'Cat' column. *Note: For more information on these categories refer to the section Modification Categories.*
	Impact Type	All the available impact types are listed for the search keyword group. Each one can be selected to change the default type provided by Impact execution. After the '**Update Impact Item**' button is used, the new type code will appear in the Impact Items list box under the 'Type' column. *Note: For more information on type of impact refer to the section Impact Types in Chapter 1.*
	Reason	Natural Engineer's reason for assigning the Modification Category and Impact Type.
	Comment	A user-entered comment usually used to explain any changes made to the defaults.
	User ID	The User ID of the last update made to the impact item.

SCREEN ITEMS	DESCRIPTION	
	Last Update	The date and time of the last update made to the impact item.
	Replace Value	The value with which to replace the impact item.
	Replace Definition	The definition to replace the original definition of the impact item.
	Replace Keyword	The keyword to replace the original keyword of the impact item. *Note: This option is only available for keywords that execute other objects, for example, CALLNAT, FETCH and PERFORM.*
	TLM Data	Data for passing to the TLM. The modification process will substitute this value for the TLM data field XX-TLM. *Note: The TLM used must have XX-TLM coded within it for the value to be substituted successfully.*
	TLM Name	The name of the TLM to be used. This will display the name of the TLM that was specified with the Search Criteria, otherwise will be blank. A new TLM name can be input here.
	TLM Pos	The position the TLM will be inserted relative to the impact item. Available values are: ▪ **After** ▪ **Before** ▪ **Replace.**

The Impact Reason list box is a multi-purpose list box used to display, either the Impact Reason, Context or Source Code information for the selected impact item. The display is controlled by the Impact Reason, Context and Source Code buttons.

The default display is Impact Reason.

Impact Reason	Displays information on the reasons for the impact based on the search criteria specified.

SCREEN ITEMS	DESCRIPTION

Context List	Displays the context of the selected data item within the data definition of the selected object and the relative offsets if applicable.
	
	Note: Only displays details for impacted data items from data definition areas.
	
	If the selected data item is a literal string, then no context details will be displayed.
	
	If the selected data item is part of a view definition, group or redefinition, then the context list will show data items one level lower and one level higher in relation to the selected data item.
	
	Some examples based on the following data definition:
	
	`0020 01 #GROUP`
	`0030   02 #ALPHA(A10)`
	`0040   02 REDEFINE #ALPHA`
	`0050     03 #NUMERIC(N6)`
	
	For data item #GROUP, the context list will show:
	
	`0020            01 #GROUP               G`
	`0030    1   10  02 #ALPHA               A10`
	
	For data item #ALPHA, the context list will show:
	
	`0020            01 #GROUP               G`
	`0030    1   10  02 #ALPHA               A10`
	`0040            02 REDEFINE #ALPHA`
	`0050    1    6  03 #NUMERIC             N6`
	
	For data item #NUMERIC, the context list will show:
	
	`0030    1   10  02 #ALPHA               A10`
	`0040            02 REDEFINE #ALPHA`
	`0050    1    6  03 #NUMERIC             N6`

Source Code	Displays all the impacted statement references for the selected impact item. The columns available are: **Object Name** — The name of the object in which the Impact is identified. This will normally be the object selected, but can be an external object such as a GDA, LDA, PDA or Copycode, which is included in the selected object. **Line Number** — The line number of the impacted statement code. **Statement** — The statement code which is impacted.

BUTTON NAME	DESCRIPTION
Object List group:	
Prev	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.
More	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.
Both	This button provides additional refinement of the objects listed in the Object List box. This button has three different states, with the button text changing accordingly: **Both** The default for the screen is to list all objects whether they have had Modification applied or not. **Mod O** Will only list objects that have had Modification applied. **N Mod** Will only list objects that have had no Modification applied.
Impact Item group:	
Prev	Scrolls the impact items to previous page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
More	Scrolls the impact items forward one page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
Modify Details group:	
TLM Name Selection [….]	Invokes the General Selection screen, listing any Objects with object type 'Text' either from the Modification library or the Natural library SYSTEM. *Note: For more information on the General Selection screen refer to Chapter 2 in the Concepts and Facilities manual.*
Update Impact Item	Saves the new Modify Details that have been specified for a selected Impact Item.
Additional Details…	Invokes the Map Parameters screen. This button is only available when dealing with Map objects, and where the impact item is not a data item definition. *Note: For more information refer to the section Map Parameters Window.*

BUTTON NAME	DESCRIPTION
Multi-purpose List box group:	
Impact Reason	Displays information on the reasons for the impact based on the search criteria specified.
Context	Displays the context of the selected data item within the data definition of the selected object.
Source Code	Displays all the impacted statement references for the selected impact item.
Modification Element Maintenance screen:	
Execute Modification	Executes any automatic Modification changes, by copying the code into the defined Modification library and changing the code as required. *Note: This will only apply Modification changes to the currently selected object.* *Note: If the object is from a PAC application, then a set of validations will be performed to check that the modification can be applied. For more information refer to the section PAC Applications.*
Reset Modification Categories within impacted objects to either '**Reject**' or '**Manual**'.Available options are:	
Reject Single Object	Any impact items with Modification Categories of 'Automatic' or 'Manual' will be reset to '**Reject**'. Applied to selected object only.
Reject Multiple Object	Any impact items with Modification Categories of 'Automatic' or 'Manual' will be reset to '**Reject**'. Applied to all impacted objects.
Manual Single Object	Any impact items with Modification Category of 'Automatic' will be reset to '**Manual**'. Applied to selected object only.
Manual Multiple Object	Any impact items with Modification Category of 'Automatic' will be reset to '**Manual**'. Applied to all impacted objects.

Note: For more information on the Automatic, Manual or Reject Modification Categories refer to the section Modification Categories.

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

Modification Categories

The following table shows all of the available modification categories:

Cat	Description
A	**Automatic** These are changes that can be made automatically by the Natural Engineer Modification process.
G	**Generated** Applies to CONSTRUCT and Predict Case objects. These are changes that must be made manually by a user.
M	**Manual** These are changes that must be made manually by a user.
N	**Not Applicable** These are Items that are not relevant for change.
R	**Reject** These are Items that have been Rejected for change by a user.
X	**User Exit** These are changes applied based on customized settings applied using the Natural Engineer supplied User Exit NEEUEX3. *Note: For more information on User Exit usage refer to the section User Exit Modification.*

Modification Element Maintenance Context Menu

The Modification Element Maintenance context menu is invoked by placing the cursor on any of the items listed in the Object or Impact Items lists and using the right hand mouse button with a single click.

Object List Context Menu

The Object List context menu allows viewing options View Impacted Code or Preview Modified Code to be invoked. It also provides the option to reposition the object list.

Note: The Object List context menu is not available when displaying DDM objects.

The following Figure 2-3 illustrates the Object List context menu

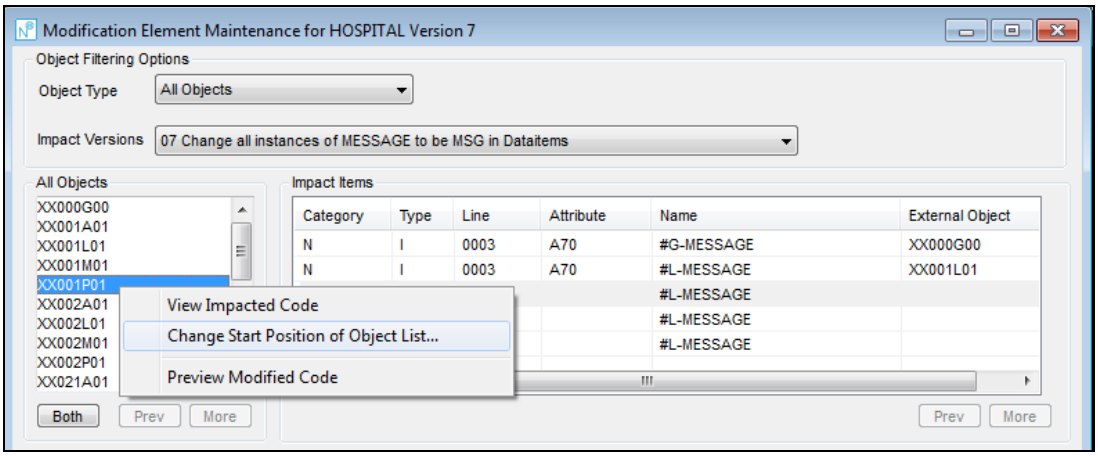

Figure 2-3 Object List context menu

CONTEXT MENU ITEM	DESCRIPTION
Change Start Position of Object List…	Reposition the list of objects to start from a particular object name. The reposition value can be input using either a complete name or part name using an '*' (asterisk) wildcard. The reposition value is appended to the object list title to highlight the type of repositioning being applied. Possible reposition values are: **Value** — **Result** **' ' (blank)** — Reposition to the top of the object list. ***** — Reposition to the top of the object list. **ABC*** — Only show objects that are prefixed by 'ABC'. **XYZ** — Reposition to the first object that either matches or is greater than 'XYZ' and then continue the object list from that point.
View Impacted Code	Display the impacts within an object using the Browser. The whole object source code will be displayed with the impacted items highlighted using the colors set by the SPECIFIED, DERIVED and EXCLUDED parameters in the NATENG.INI file. *Note: For more information on the NATENG.INI file parameters SPECIFIED, DERIVED and EXCLUDED refer to Chapter 1 in the Natural Engineer Administration Guide for Windows manual.*
Preview Modified Code	Display the modified object code using the Browser. The whole object source code will be displayed with the modified code highlighted. This provides the facility to preview the modifications before they are made real.

Impact Items Context Menu

The Impact Items context menu provides the option to filter the Impact Items list.

The following Figure 2-4 illustrates the Object List context menu.

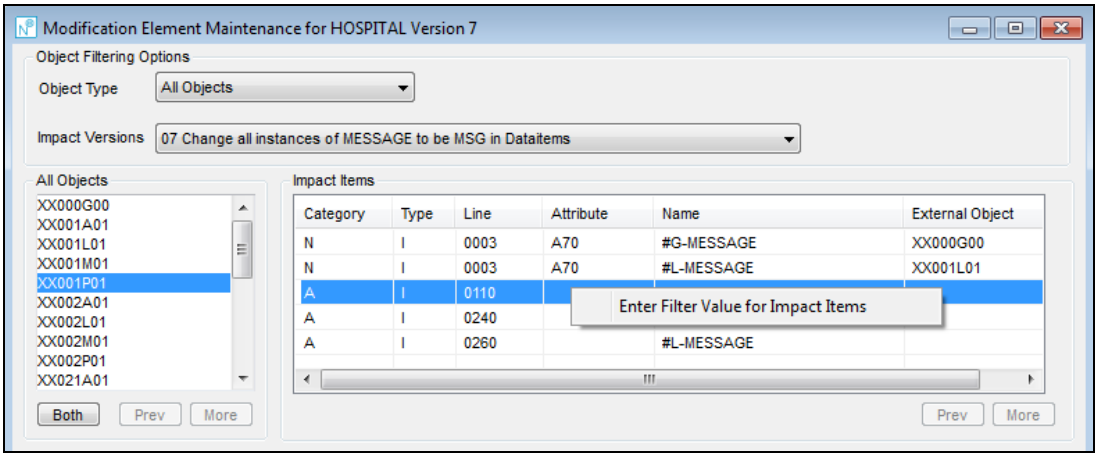

Figure 2-4 Object List context menu

CONTEXT MENU ITEM	DESCRIPTION
Enter Filter Value for Impact Items	Filter the list of impact items to show a particular impact item name. The filter value can be input using either a complete name or part name using an '*' (asterisk) wildcard. The filter value is appended to the impact item list title to highlight the type of filtering being applied. Possible filter values are:

Value	Result
' ' (blank)	Reposition to the top of the impact item list.
*****	Reposition to the top of the impact item list.
ABC*	Only show impact items that are prefixed by 'ABC'.
XYZ	Only show impact items that are named 'XYZ'.

Map Parameters Window

The Map Parameters window is accessed by using the '**Additional Details…**' button from the Modification Element Maintenance screen.

It is only available when dealing with Map objects, where the data item is to be displayed on the map.

The following Figure 2-5 illustrates the Map Parameters screen.

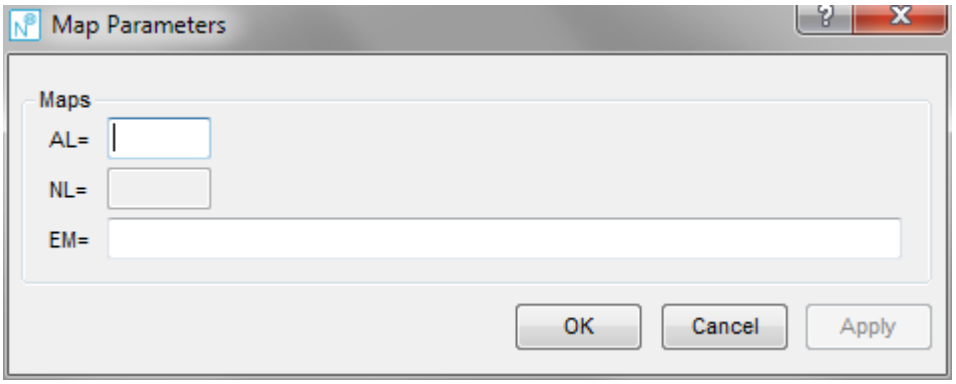

Figure 2-5 Map Parameters screen

SCREEN ITEMS	DESCRIPTION
AL=	Display length for alphanumeric data item.
NL=	Display length for a numeric data item.
EM=	Edit mask for the data item.

BUTTON NAME	DESCRIPTION
OK	Save the Map Parameters settings and close the current screen.
Cancel	Cancel the Map Parameters process and return back to the Modification Element Maintenance screen.
Apply	Save the Map Parameters settings and retain the current screen. *Note: This button is only enabled if any changes have been made.*

User Exit Modification

Customizable modifications can be applied by making use of the supplied user exit module NEEUEX3.

The user exit module can be customized to replace selected Impact items within an object with up to 20 lines of new source code.

Note: The user exit module supplied is named 'NEEUEX3X' on the Natural Engineer SYSNEE library. This is to avoid overwriting any existing (modified) versions on the production SYSNEE library during Natural Engineer installation. If this user exit has not been loaded before, then it will need to be renamed to 'NEEUEX3' before making use of the User Exit Modification functionality.

The user exit module can be invoked during the Modification process by selecting an impacted statement line in the Impact item list on the Modification Element Maintenance screen and changing the Modification Category to 'X' (User Exit).

Example of User Exit Modification

This example is based on the sample code from the supplied user exit module.

Impact Analysis is run against the HOSPITAL application using the following Impact Criteria settings of DBFILE (Impact Type), PATIENT (Keyword Value) and PATIENT-ID (Search Value).

The impacted item at statement line number 0690 within object XX021P01 is selected and the Modification Category changed to 'X' (User Exit).

The Modification process is then invoked for object XX021P01.

Sample source code for object XX021P01 before Modification:

```
0630 *
0640 SET KEY ALL
0650 MOVE *DATN TO #L-TEMP-DATE
0660 DECIDE ON FIRST VALUE OF #G-SELECTED-OPTION
0670 VALUE "A"
0680   MOVE *DATX TO PATIENT.RELEASED
0690   CALLNAT "XXGETID" PATIENT.PATIENT-ID #L-TEMP-DATE-N6
0700   MOVE "              ADD A PATIENT" TO #M-MAP-HEADING
```

Sample source code for object XX021P01 after Modification:

```
0630 *
0640 SET KEY ALL
0650 MOVE *DATN TO #L-TEMP-DATE
0660 DECIDE ON FIRST VALUE OF #G-SELECTED-OPTION
0670 VALUE "A"
0680   MOVE *DATX TO PATIENT.RELEASED
0690 /*   CALLNAT "XXGETID" PATIENT.PATIENT-ID #L-TEMP-DATE-N6 /* NEE OLD
0700 *                                                        /* Changed
0710 * Start of Block of Code from NEEUEX3                   /* Changed
0720 *                                                        /* Changed
0730   CALLNAT "SAGGETID" PATIENT.PATIENT-ID #L-TEMP-DATE-N6  /* Changed
0740 *                                                        /* Changed
0750 * End of Block of Code from NEEUEX3                     /* Changed
0760 *                                                        /* Changed
0770   MOVE "              ADD A PATIENT" TO #M-MAP-HEADING
```

Execute Modification for All Objects

The Execute Modification for All Objects option invokes the Modification process, which will apply the specified Modifications to the object source code for all the objects within an application, held on the Repository in one single operation.

This option is usually executed after review and confirmation that all Modification Categories, Types and other settings are as required, using the Modification Element Maintenance screen and Modification Reports.

Each object is modified as follows:

- The object is found in the application Natural library (i.e., the Natural library from which the Extract process extracted the objects).
- The object is then copied over to the Modification library (i.e., as specified in the Application Properties).
- During the copy phase, the Modification process checks the Impact and Modification data held in the Repository for the object being modified and applies it to the modified version.

The Execute Modification for All Objects option is accessed using the following menu navigation: Modification➔Execute Modification for All Objects. When this option is selected, a confirmation window is displayed.

The following Figure 2-6 illustrates the Execute Modification for All Objects confirmation window.

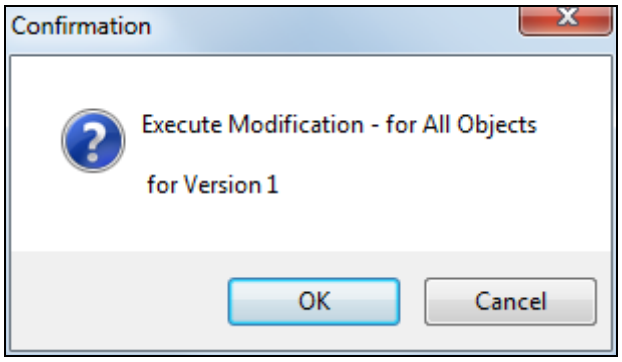

Figure 2-6 Execute Modification for All Object confirmation window

BUTTON NAME	DESCRIPTION
OK	Invoke the Execute Modification for All Objects process.
Cancel	Cancel the Execute Modification for All Objects process and return back to the main Natural Engineer screen.

Warning: For any objects being modified, if the same object already exists in the Modification library, then it will be over written by the new version created during the Modification process.

PAC Applications

This section only applies if Natural Engineer is executing in a remote development environment and PAC version 2.4.2 or above is installed on the mainframe.

If the Modification is to be applied to objects from a PAC Application, then the Modification process will apply a set of validations to each object to check that the object can be modified. If any of the validations fail, then the object will not be modified.

The validations are:

- The object is already 'checked out' by another PAC user.
- The Repository version of the object is different to the PAC version of the object.
- If a PAC Check Out log for the object has not been created.

Modification Inventory

The Modification Inventory consists of a set of Modification Reports, which provide various types of information concerning the Modification process. Reports are available at the summary, object and detailed data item levels.

The Modification Reports can be accessed using the following menu navigation: Modification➔Modification Reports.

The following list illustrates the Modification Reports that are available:

- Application Modification Summary
- Object Modification Summary
- Category / Type Summary
- PREDICT Changes
- Data Item Inventory Modification
- Data Item Inventory for Automatic Modification
- Data Item Inventory for Manual Modification
- Impacted Objects Not Directly Modified
- Construct Models Not Directly Modified
- Database Data Requirements Modification
- Preview Modified Code

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

In addition to the Modification Reports, all modification changes generate audit trail records of the before and after images of the changed source code. These audit trail records can be reviewed using the Change Management Tracking option from the Utilities menu.

Note: For more information on the Change Management Tracking option refer to Chapter 4 in the Natural Engineer Utilities for Windows manual.

3

COMBINATION SEARCH KEYWORDS

Chapter Overview

This chapter describes the combination search keywords that can be used when specifying Search Criteria for the Impact Analysis option.

The combination search keywords are not Natural keywords in themselves, but are used to qualify a group of Impact Criteria, which may encompass several Natural keywords in the process.

Note: The combination search keywords are only available for Natural objects.

The topics in this chapter are:

ADJUST

The combination search keyword ADJUST can be used to change the name of an object within an application.

Impact will identify the object that has been specified within an application and using the consistency option with this search keyword, will result in all references to the object being identified within the other objects in the application.

Modification will copy and rename the specified object to the modification library and then change all impacted references within objects to use the new object name.

Specifying Adjust

Select Impact Type ADJUST from the Criteria Details tab screen.

The following Figure 3-1 illustrates the Criteria Detail tab screen for combination keyword ADJUST.

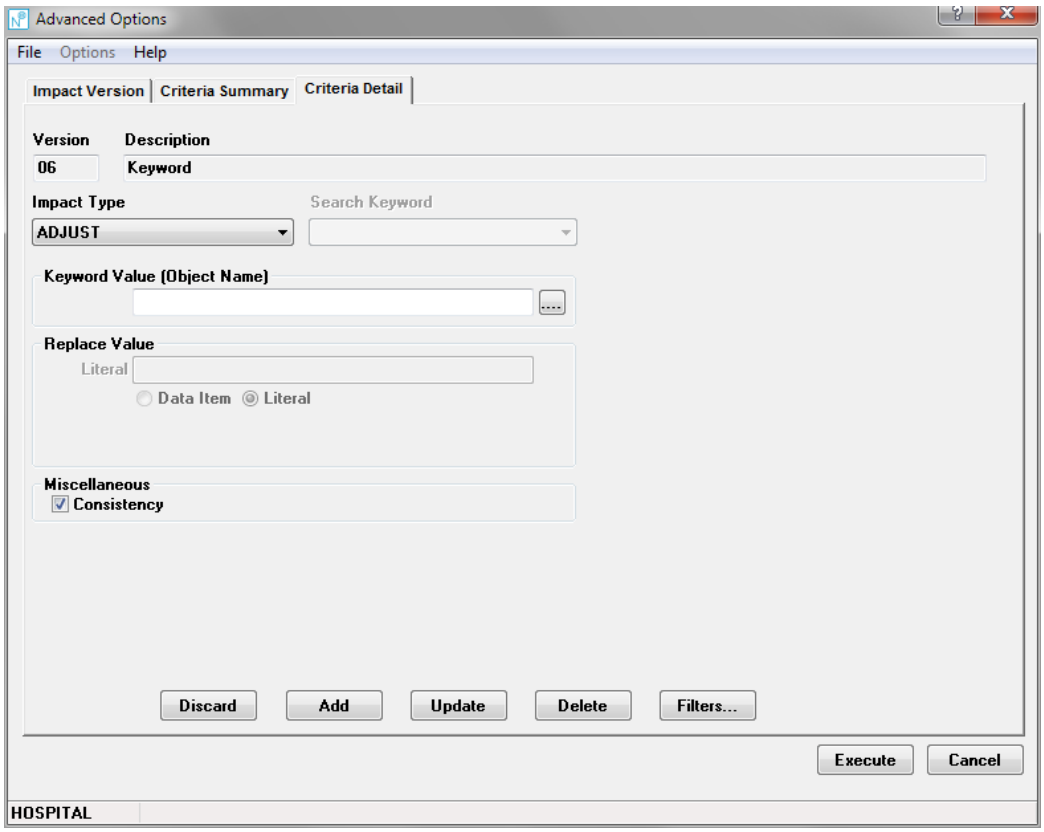

Figure 3-1 Criteria Detail tab screen for combination keyword ADJUST

The following only describes the Criteria Detail tab screen options relevant to the combination search keyword ADJUST.

Note: For more information on the Criteria Details tab screen refer to the section Criteria Detail tab screen in Chapter 1.

SCREEN ITEMS	DESCRIPTION
Keyword Value group:	
Keyword Value	The name of an object. The object name can be typed in or selected by using the Keyword Value Selection button [….].
Replace Value group:	
Data Item / Literal	The replacement name to be used for the selected object.
Miscellaneous group:	
Consistency	If not selected, then the object specified in the keyword value will be renamed using the replacement name. All calling references within the application will not be modified. If selected, then the object specified in the keyword value will be renamed using the replacement name and all calling references within the application will be modified to use the new name.

Setting Object Filters

The ADJUST criteria can be further refined by only applying the criteria to specific object types and/or object names during Impact execution.

This can be set by using the 'Filters…' button on the Criteria Details tab screen to invoke the Filters window.

Note: For more information on Filters refer to the Filters Window section.

Example to illustrate the use of Adjust

This example will use the search keyword ADJUST to change the name of object XX021P01 to be SAG21P01. Consistency will be activated so that all references within the sample application HOSPITAL for object XX021P01 are found and changed to be SAG21P01.

Step 1 Create a new version of the impact search criteria and using an Impact Type of ADJUST, a Keyword value of 'XX021P01', a Replace Value' of 'SAG21P01' and Consistency activated.

The following Figure 3-2 illustrates the Criteria Detail tab screen with the specified criteria for ADJUST.

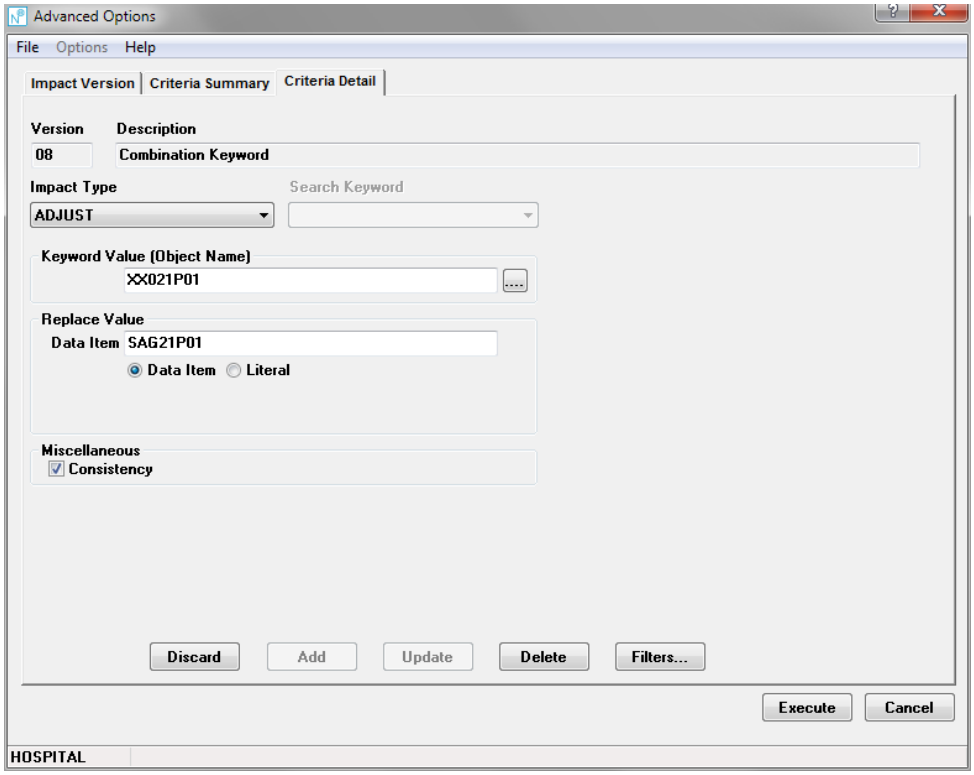

Figure 3-2 Criteria Detail tab screen with the specified criteria for ADJUST

Step 2 Execute Impact analysis.

Step 3 Review the Impact results using the Impact Element Maintenance screen. Impact Analysis will find impacts for three objects: XX002P01, XX021P01 and XX025P01.

The following Figure 3-3 illustrates the Impact Element Maintenance screen displaying the impacts for search keyword: ADJUST.

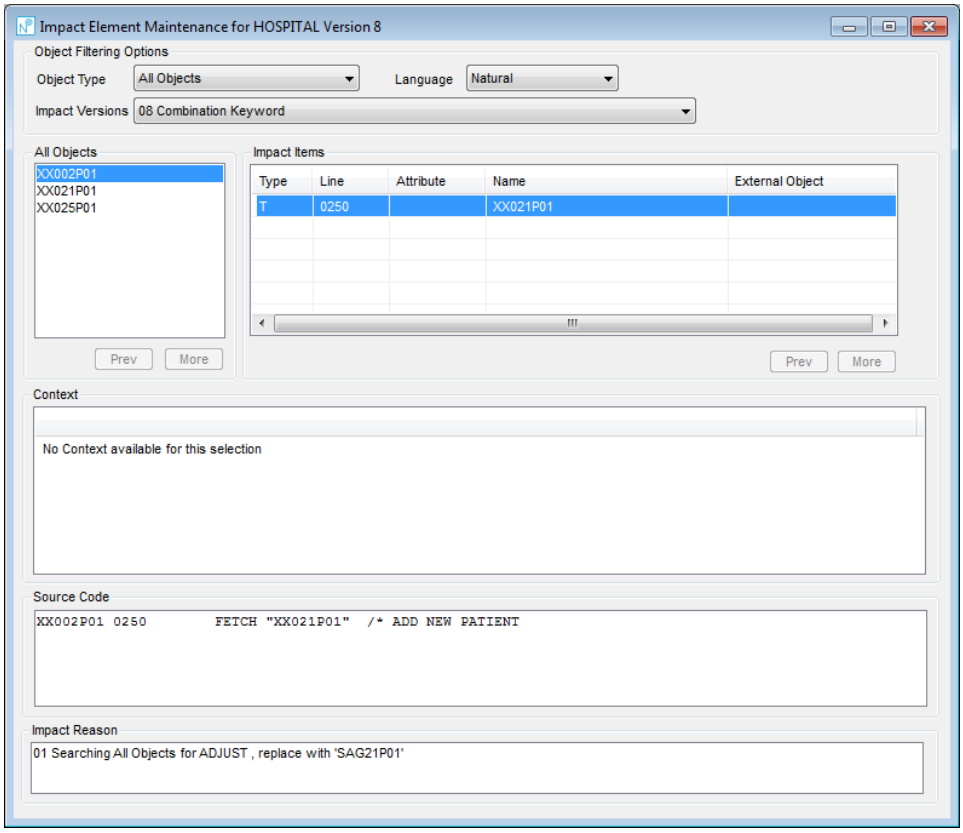

Figure 3-3 Impact Element Maintenance screen with impact results for ADJUST

Object XX021P01 has been impacted for the object rename. Objects XX002P01 and XX025P01 have been impacted because they make reference to object XX021P01.

Step 4 Review modification information by selecting objects using the Modification Element Maintenance screen. Select each object to view the modification strategy. All object changes with a category of 'A' will be automatically completed by Natural Engineer. All others must be reviewed.

The following Figure 3-4 illustrates the Modification Element Maintenance screen displaying the modification details for search keyword: ADJUST.

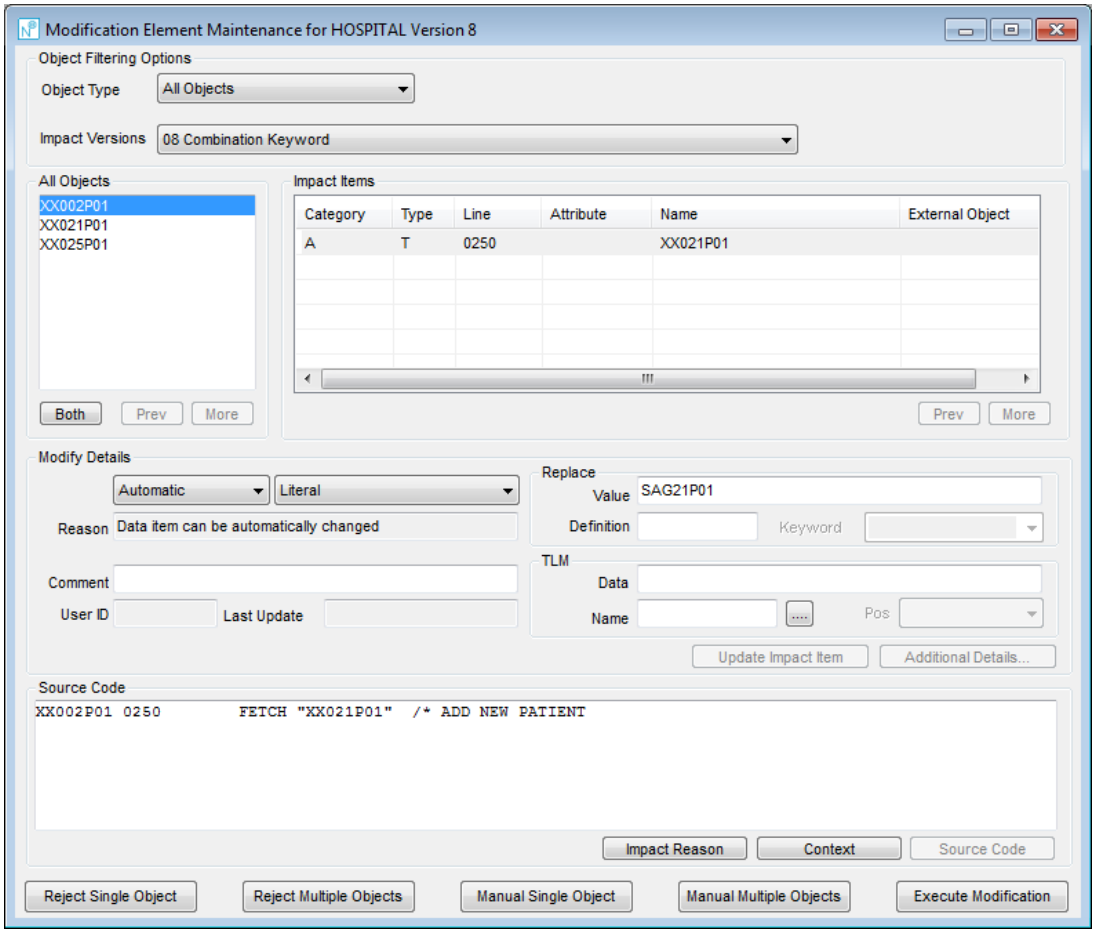

Figure 3-4 Modification Element Maintenance screen displaying modification details for ADJUST

Step 5 Execute modification individually for each object from the Modification Element Maintenance screen. (Alternatively, use the Execute Modification for All Objects option from the Modification menu to apply all modifications in one single operation.).

This will copy each object to the Modification library HOSPITAX, and apply the changes that have been specified. The changes applied are:

1. Rename object XX021P01 to SAG21P01,
2. In object XX002P01, change statement at line number 0250:

   From: FETCH “XX021P01”

   To FETCH “SAG21P01”.
3. In object XX025P01, change statement at line number 1080 :

   From: FETCH RETURN "XX021P01" #M-PATIENT-ID (#W-LOOP)

   To: FETCH RETURN “SAG21P01" #M-PATIENT-ID (#W-LOOP).

Step 6 Review the modifications applied by opening the modification library HOSPITAX in Natural.

The following Figure 3-5 illustrates the modified objects in the modification library HOSPITAX.

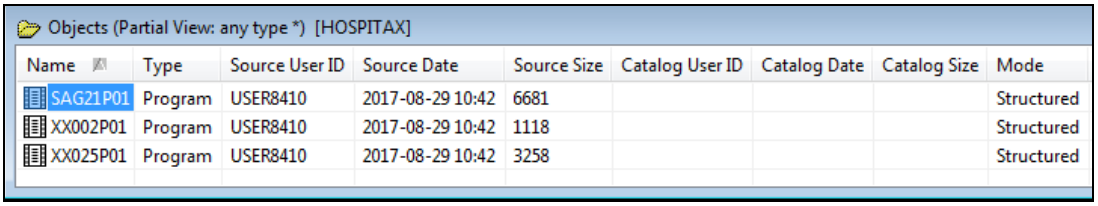

Figure 3-5 Modified objects in modification library HOSPITAX

From the object list it can be seen that object XX021P01 has been renamed to SAG21P01.

The following Figure 3-6 illustrates the modified source code within object XX002P01.

```
0010 DEFINE DATA
0020 GLOBAL
0030 USING XX000G00
0040 LOCAL
0050 USING XX002L01
0060 END-DEFINE
0070 *
0080 SET KEY ALL
0090 *
0100 REPEAT
0110 *
0120 INPUT USING MAP "XX002M01"
0130 RESET #M-MESSAGE
0140 *
0150 DECIDE ON FIRST VALUE OF *PF-KEY
0160 *
0170 VALUE "PF3", "PF15"
0180     FETCH "XX001P01"
0190 VALUE "PF12", "PF24"
0200     PERFORM XXEXIT
0210 VALUE "ENTR"
0220     DECIDE ON FIRST VALUE OF #M-OPTION
0230     VALUE "A"
0240        MOVE "A" TO #G-SELECTED-OPTION
0250 /*        FETCH "XX021P01"  /* ADD NEW PATIENT /* NEE OLD CODE
0260        FETCH "SAG21P01"  /* ADD NEW PATIENT /* NEE MODIFIED
```

Figure 3-6 Modified source code within object XX002P01

The following Figure 3-7 illustrates the modified source code within object XX025P01.

```
1030  DEFINE SUBROUTINE PROCESS-SELECTED
1040  *
1050  FOR #W-LOOP = 1 TO 15
1060      DECIDE FOR FIRST CONDITION
1070      WHEN #M-SELECTED(#W-LOOP) = " "
1080           ESCAPE TOP
1090      WHEN #M-SELECTED(#W-LOOP) NOT = " "  AND
1100           #M-PATIENT-ID(#W-LOOP) = 0
1110           MOVE "YOU SELECTED A BLANK LINE - TRY AGAIN!" TO #G-MESSAGE
1120           ESCAPE ROUTINE
1130      WHEN #M-SELECTED(#W-LOOP) = "D" OR = "M"
1140           MOVE #M-SELECTED(#W-LOOP) TO #G-SELECTED-OPTION
1150  /*          FETCH RETURN "XXO21PO1" #M-PATIENT-ID(#W-LOOP) /* NEE OLD CODE
1160           FETCH RETURN "SAG21P01" #M-PATIENT-ID(#W-LOOP) /* NEE MODIFIED
1170           MOVE " " TO #M-SELECTED(#W-LOOP)
1180      WHEN NONE
1190           MOVE "INVALID CHARACTER FOUND IN SELECTION ARRAY" TO #G-MESSAGE
1200           ESCAPE ROUTINE
1210      END-DECIDE
1220  END-FOR
1230  PERFORM READ-DATA
1240  END-SUBROUTINE
1250  END
```

Figure 3-7 Modified source code within object XX025P01

Step 7 Copy the rest of the application HOSPITAL objects, taking care that the objects XX002P01 and XX025P01 do not get over written. Apply a CATALL to the HOSPITAX library and then execute the HOSPITAL system using the modified objects.

CODE IMPROVEMENT

The combination search keyword CODE IMPROVEMENT is used to locate certain Natural statements that may cause performance considerations within Natural Applications.

This search keyword uses a sub-set of criteria, which can be refined by selecting/ deselecting the available options.

Once impacted, then for some cases it may be possible to modify the code to correct the inefficiencies found, using the Modification process.

Specifying Code Improvement

Select Impact Type CODE IMPROVEMENT from the Criteria Details tab screen.

Note: Only one set of CODE IMPROVEMENT criteria is allowed per Impact Version.

The following Figure 3-11 illustrates the Criteria Detail tab screen for combination keyword CODE IMPROVEMENT.

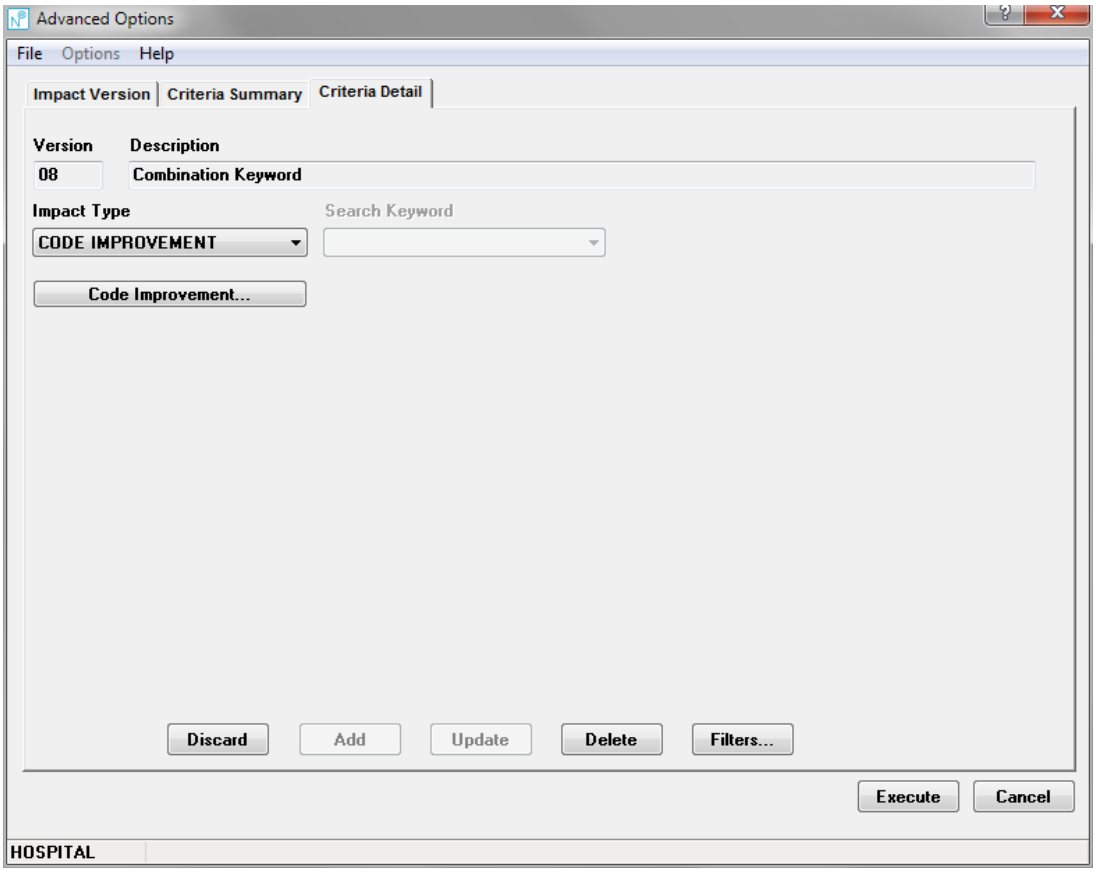

Figure 3-11 Criteria Detail tab screen for combination keyword CODE IMPROVEMENT

Note: For more information on the Criteria Details tab screen refer to the section Criteria Detail tab screen in Chapter 1.

Code Improvement Preferences Window

The Code Improvement Preferences screen allows you to control which Code Improvements are actually checked for during Impact execution.

The Code Improvement Preferences are accessed from the Criteria Detail tab screen by using the 'Code Improvement…' button. Alternatively, existing Code Improvement criteria can be selected from the Criteria Summary tab screen by double clicking with the left mouse button.

The Following Figure 3-12 illustrates the Code Improvement Preferences screen.

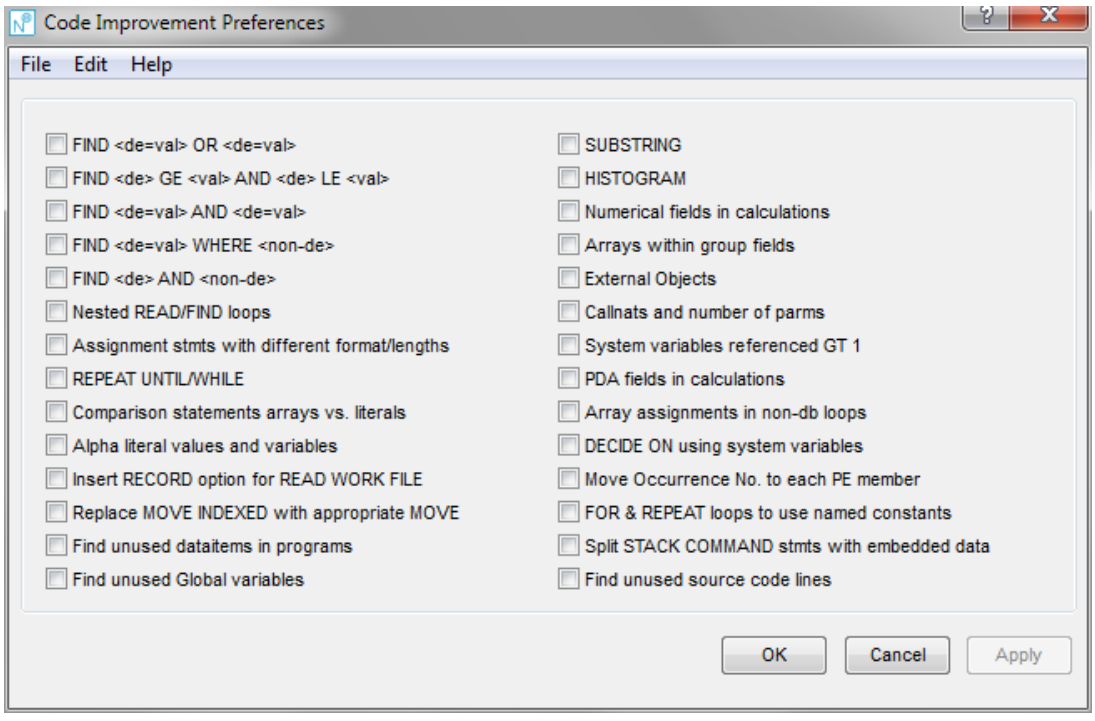

Figure 3-12 Code Improvement Preferences screen

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Exit**	Exit the Code Improvement Preferences screen and return back to the Criteria Details tab screen.
Edit	**Select All**	Select all the Code Improvement Preferences.
	Deselect All	Deselect all the Code Improvement Preferences.
Help	Invoke the Code Improvement Preferences help.	

SCREEN ITEMS	DESCRIPTION
Code Improvement Preferences	Each Code Improvement option is listed. A tick in the check box next to each option indicates that the option will be checked for. If a check box is blank then that option will not be referenced during Impact execution. *Note: At least one Code Improvement option must be selected.* Available selections are: ▪ **FIND <de=val> OR <de=val>** ▪ **FIND <de> GE <val> AND <de> LE <val>** ▪ **FIND <de=val> AND <de=val>** ▪ **FIND <de=val> WHERE <non-de>** ▪ **FIND <de> AND <non-de>** ▪ **Nested READ/FIND loops** ▪ **Assignment stmts with different format/lengths** ▪ **REPEAT UNTIL/WHILE** ▪ **Comparison statements arrays vs. literals** ▪ **Alpha literal values and variables** ▪ **Insert RECORD option for READ WORK FILE** ▪ **Replace MOVE INDEXED with appropriate MOVE** ▪ **Find unused dataitems in programs** ▪ **Find unused Global variables** ▪ **SUBSTRING** ▪ **HISTOGRAM** ▪ **Numerical fields in calculations** ▪ **Arrays within group fields** ▪ **External Objects** ▪ **Callnats and number of parms** ▪ **System variables referenced GT 1** ▪ **PDA fields in calculations**

SCREEN ITEMS	DESCRIPTION
	▪ **Array assignments in non-db loops** ▪ **DECIDE ON using system variables** ▪ **Move Occurrence No. to each PE member** ▪ **FOR & REPEAT loops to used named constants** ▪ **Split STACK COMMAND stmts with embedded data** ▪ **Find unused source code lines** *Note: For more information on each option refer to the section Code Improvement Preferences Explained.*

BUTTON NAME	DESCRIPTION
OK	Save the Code Improvement Preferences settings and close the current screen.
Cancel	Cancel the Code Improvement Preferences process and return back to the Criteria Details tab screen.
Apply	Save the Code Improvement Preferences settings and retain the current screen. *Note: This button is only enabled if any changes have been made.*

Setting Object Filters

The CODE IMPROVEMENT criteria can be further refined by only applying the criteria to specific object types and/or object names during Impact execution.

This can be set by using the 'Filters…' button on the Criteria Details tab screen to invoke the Filters window.

Note: For more information on Filters refer to the Filters Window section.

Code Improvement Preferences Explained

The Code Improvement Preferences help identify inefficiencies within Natural objects that may cause a degradation of performance.

Some of the inefficiencies can be modified using the Modification process, others will only have the Impact process available and manual modifications may need to be applied to each of these.

FIND <de=val> OR <de=val>

Impact	Impact will look for any FIND statements that have a WITH clause that uses "<descriptor=value> OR < descriptor=value>", where the same descriptor is used. Example: `0230 FIND VEHICLES WITH MAKE = 'BMW' OR MAKE = 'VW'`
Impact Type	7A
Modification	Manual.

FIND <de> GE <val> AND <de> LE <val>

Impact	Impact will look for any FIND statements that have a WITH clause that uses "<descriptor> GE <value> AND < descriptor> LE <value>", where the same descriptor is used. Example: `0230 FIND VEHICLES WITH MAKE GE 'BMW' AND MAKE LE 'VW'`
Impact Type	7B
Modification	Manual.

FIND <de=val> AND <de=val>

Impact	Impact will look for any FIND statements that have a WITH clause that uses "<descriptor=value> AND < descriptor=value>". Example: `0230 FIND VEHICLES WITH MAKE = 'BMW' AND COLOR = 'RED'`
Impact Type	7C
Modification	Manual.

FIND <de=val> WHERE <non-de>

Impact	Impact will look for any FIND statements that have a WITH clause that uses "<descriptor=value>" and a WHERE clause that uses "< non-descriptor>". Example: `0230 FIND VEHICLES WITH MAKE = 'BMW' WHERE MODEL = 'M3'`
Impact Type	7D
Modification	Manual.

FIND <de> AND <non-de>

Impact	Impact will look for any non-descriptor searches. For any non-descriptors found, Impact will show the DDM and any FIND statements referencing them. There are two types of non-descriptor searched for:
	
	[1] Database fields that have descriptor type set to 'N' in the DDM being referenced.
	
	Example:
	`0230 FIND GSL-VEH WITH MAKE = 'BMW' AND MODEL = 'M3'`
	
	Note: The descriptor MODEL in this example has a descriptor type of 'N' (non-descriptor) set in the DDM GSL-VEH.
	
	[2] Database fields that have a descriptor type set to 'D' in the DDM, but are not marked as descriptors in the FDT.
	
	Note: For this Impact, the database must be active and contain the relevant FDT.
	
	Example:
	`0230 FIND GSL-EMP WITH NAME = 'PAUL' AND MAR-STAT = 'S'`
	
	Note: The descriptor MAR-STAT in this example has a descriptor type of 'D' (descriptor) set in the DDM GSL-EMP, but is not marked as a descriptor in the FDT (EMPLOYEES).

Impact Type	7E
Modification	Manual.

Nested READ/FIND loops

Impact	Impact will look for any FIND statements, including FIND FIRST and FIND (1), which are within READ loops.
	
	Both the READ and FIND statements are impacted.
	
	If the FIND statement is part of an internal subroutine and is invoked by using a PERFORM statement within a READ loop, then PERFORM statement will also be impacted.
	
	Example:
	`0420 READ EMPLOYEES`
	`0430   FIND VEHICLES WITH PERSONNEL-ID ='12345678'`
	`0440     WRITE '=' PERSONNEL-ID(0420) '=' NAME(0420)`
	`0450     WRITE '=' MAKE(0430)`
	`0460       '=' MODEL(0430) '=' REG-NUM(0430)`
	`0470   END-FIND`
	`0480 END-READ`
	`::::`
	`0530 READ EMPLOYEES`
	`0540   MOVE PERSONNEL-ID TO #EMP-PERS-ID`
	`0550   WRITE '=' PERSONNEL-ID '=' NAME`
	`0560   PERFORM ##FIND-VEHICLES-ONLY`
	`0570 END-READ`
	`::::`
	`0780 DEFINE SUBROUTINE ##FIND-VEHICLES-ONLY`
	`0790 FIND VEHICLES WITH PERSONNEL-ID = #EMP-PERS-ID`
	`0800   WRITE '=' MAKE '=' MODEL '=' REG-NUM`
	`0810 END-FIND`
	`0820 END-SUBROUTINE`

Impact Type	7F
Modification	Manual.

Assignment stmts with different format/lengths

Impact	Impact will look for any assignment statements that have fields of either different format or, fields that have the same format but different lengths.
	
	Example:
	`0010 01 #ALPHA5 (A5)`
	`0020 01 #ALPHA10 (A10)`
	`0030 01 #NUMERIC5 (N5)`
	`0040 01 #NUMERIC10 (N10)`
	`0050 MOVE #NUMERIC5 TO #ALPHA5`
	`0060 MOVE #ALPHA5 TO #ALPHA10`
	
	Statement at line 0050 will be impacted because the fields have different formats (7G)
	
	Statement at line 0060 will be impacted because the fields have same format but different lengths (7S).
	
	ASSIGN and COMPUTE statements that are 'assignments' will be impacted, but 'computational' ASSIGN and COMPUTE statements will not be impacted.
	
	Example:
	`0070 ASSIGN #ALPHA5 = #NUMERIC5`
	`0080 ASSIGN #NUMERIC5 = #NUMERIC10 - #NUMERIC5`
	`0090 COMPUTE #ALPHA5 = #NUMERIC5`
	`0100 COMPUTE #NUMERIC5 = #NUMERIC10 - #NUMERIC5`
	
	Statements 0070 and 0090 will be impacted, but statements 0080 and 0100 will not.

Impact Type	7G (different format) 7S (same format, different length)
Modification	Manual.

REPEAT UNTIL/WHILE

Impact	Impact will look for any REPEAT statements that use UNTIL or WHILE clauses.
	
	Example:
	`0210 REPEAT UNTIL #COUNT EQ 2`
	`0220   ADD 1 TO #COUNT`
	`0230 END-REPEAT`
	`::::`
	`0310 REPEAT WHILE #COUNT EQ 2`
	`0320   ADD 1 TO #COUNT`
	`0330 END-REPEAT`
	
	If the UNTIL or WHILE clause is placed near the bottom of the loop block, then the corresponding REPEAT statement is also impacted.
	
	Example:
	`0490 REPEAT`
	`0500   ADD 1 TO  #COUNT`
	`0510   UNTIL #COUNT EQ 2`
	`0520 END-REPEAT`
	`::::`
	`0590 REPEAT`
	`0600   ADD 1 TO  #COUNT`
	`0610   WHILE #COUNT EQ 2`
	`0620 END-REPEAT`

Impact Type	7H
Modification	Manual.

Comparison statements arrays vs. literals

Impact	Impact will look for any alpha literal strings that are being compared against a field defined as an array.
	Example:
	`0010 01 #A (A1/10)`
	`0020 01 #B (A1)`
	`::::`
	`0050 IF 'A' EQ #A(1)`
	`::::`
	`0100 IF 'B' EQ #B`

Impact Type	7I
Modification	Manual.

Alpha literal values and variables

Impact	Impact will look for any alpha literal string being moved to alpha variables that are defined with a length greater than the length of the literal string.
	Example:
	`0010 01 #TEMP (A10)`
	`0020 01 #WORK (A5)`
	`0030 MOVE 'ABCDE' TO #TEMP`
	`0040 MOVE 'VWXYZ' TO #WORK`
	Note: It is more efficient if length of the alpha literal string is equal to the length of the alpha variable it is being moved to.

Impact Type	7T

Modification	Automatic.
	Modification will pad the literal string value with spaces to match the length of the alpha variable.
	Example:
	`0010 01 #TEMP (A10)`
	`0020 01 #WORK (A5)`
	`0030 /* MOVE 'ABCDE' TO #TEMP /* NEE OLD CODE`
	`0040 MOVE 'ABCDE     ' TO #TEMP /* NEE MODIFIED`
	`0050 MOVE 'VWXYZ' TO #WORK`

Insert RECORD option for READ WORK FILE

Impact	Impact will look for any READ WORK FILE statements that only use one user-defined variable.
	Example:
	`::::`
	`0130 READ WORK FILE 1 #WKF01`
	`0140 END-WORK`
	`::::`

Impact Type	7V

Modification	Automatic.
	Modification will add the RECORD clause to these READ WORK FILE statements.
	Example:
	`::::`
	`0130 /* READ WORK FILE 1 #WKF01 /* NEE OLD CODE`
	`0140 READ WORK FILE 1 RECORD #WKF01 /* NEE MODIFIED`
	`0150 END-WORK`
	`::::`

Replace MOVE INDEXED with appropriate MOVE

Impact	Impact will look for any MOVE INDEXED statements. *Note: Only applies to Reporting Mode objects.*
Impact Type	7X

Modification	Automatic.
	Modification will perform one of the following:
	▪ Standard MOVE INDEXED statements will be replaced with the correct MOVE statement.
	▪ MOVE INDEXED statements utilizing contiguous storage will produce a message "Object 'object-name' contains a contiguous MOVE INDEXED statement. This is marked for manual change." The MOVE INDEXED statement will be marked for manual change and have a comment of "REASON: CONTIGUOUS MOVE INDEXED" appended.
	Example of contiguous MOVE INDEXED statement:
	`0010 RESET #FIELD-1(A10) #FIELD-2(A10)`
	`0020 RESET #DISPLAY(A10)`
	`0030 MOVE 'CORRECT' TO #FIELD-1`
	`0040 MOVE 'WRONG' TO #FIELD-2`
	`0050 MOVE INDEXED #FIELD-1<2> TO #DISPLAY`
	`0060 WRITE #DISPLAY`
	▪ MOVE INDEXED statements for DDM fields will produce a message "Object 'object-name' contains a view with a MOVE INDEXED statement. This is marked for manual change." The MOVE INDEXED statement will be marked for manual change and have a comment of "REASON: FIELD IS FROM A VIEW" appended.
	Example of MOVE INDEXED statement for DDM fields:
	`0010 RESET #DISPLAY-ADDRESS(A20)`
	`0020 READ EMPLOYEES BY NAME`
	`0030   OBTAIN ADDRESS-LINE(1:4)`
	`0040   MOVE INDEXED ADDRESS-LINE<1>`
	`0050     TO #DISPLAY-ADDRESS`
	`0060 LOOP`

Find unused dataitems in programs

Impact	Impact will look for any unused data items. These can be user-defined variables or logical view variables.
	
	Example:
	`0010 #A (A5)`
	`0020 #B (A5)`
	`0030 #C (A5)`
	`0040 MOVE 'ABCDE' TO #A`
	`0050 MOVE #A TO #B`
	`0060 WRITE #B`
	`0070 END`

Impact Type	7Z

Modification	Automatic.
	
	Modification will comment out any unused data items.
	
	Example:
	`0010 #A (A5)`
	`0020 #B (A5)`
	`0030 /* #C (A5) /* UNUSED DATA ITEM /* NEE OLD CODE`
	`0040 /* UNUSED */ #C (A5) /* UNUSED DATA ITEM /* NEE MODIFIED`
	`0050 MOVE 'ABCDE' TO #A`
	`0060 MOVE #A TO #B`
	`0070 WRITE #B`
	`0080 END`
	
	Note: LDA and PDA objects with unused data items will not be modified.

Find unused Global variables

Impact	Impact will look for any unused Global variables defined in a GDA, across a whole application. Impacts are only marked within the GDA objects and not individual objects.
Impact Type	72
Modification	Manual.

SUBSTRING

Impact	Impact will look for any statements that use the SUBSTRING clause.
	Example:
	`0400 ASSIGN #TEMP-DATE = SUBSTRING(#CONST-DATE,3,6)`
	`::::`
	`0600 IF SUBSTR(#TEXT,1,2) EQ 'SM'`
	`0610  IGNORE`
	`0620 END-IF`
	`::::`
	`0700 MOVE SUBSTR(#TEXT,1,4) TO SUBSTR(#TEXT2,5,4)`
	Note: The internal evaluation of the SUBSTRING clause is a performance issue, as there are three operands (argument, start offset and length value) that need to be resolved, as well as a 'range in fields bounds' check. Performance can be improved by use of variable redefinitions to substitute for the SUBSTRING clause.

Impact Type	7J
Modification	Manual.

HISTOGRAM

Impact	Impact will look for any HISTOGRAM statements.
	Example:
	`0210 HISTOGRAM EMPLOYEES NAME END-HISTOGRAM`
	`::::`
	`0350 HISTOGRAM EMPLOYEES NAME STARTING 'SMITH'`
	`0360 END-HISTOGRAM`
	`0460 HISTOGRAM EMPLOYEES NAME 'JONES' THRU 'SMITH' 0470 END-HISTOGRAM`

Impact Type	7K
Modification	Manual.

Numerical fields in calculations

Impact	Impact will look for any fields defined as numeric (format 'N') that are used within arithmetic statements.
	
	The following Natural keywords are included:
	
	ADD, ADD ROUNDED, SUBTRACT, SUBTRACT ROUNDED, DIVIDE, DIVIDE ROUNDED, MULTIPLY, MULTIPLY ROUNDED, ASSIGN, ASSIGN ROUNDED, COMPUTE, COMPUTE ROUNDED.
	
	Example:
	`0010 01 #VAL1 (N7)`
	`0020 01 #VAL2 (N7)`
	`0030 01 #TEMP (N7)`
	`::::`
	`0050 ADD #VAL1 TO #VAL2`
	`0060 SUBTRACT ROUNDED #VAL2 FROM #VAL1 GIVING #TEMP`
	`0070 DIVIDE #VAL1 INTO #VAL2 GIVING #TEMP`
	`0080 MULTIPLY ROUNDED #VAL1 BY #VAL2 GIVING #TEMP`
	
	ASSIGN and COMPUTE statements that are 'computational' will be impacted, but 'assignment' ASSIGN and COMPUTE statements will not be impacted.
	
	Example:
	`0100 ASSIGN #VAL1 = #VAL2`
	`0110 ASSIGN #TEMP = #VAL1 - #VAL2`
	`0120 COMPUTE #VAL1 = #VAL2`
	`0130 COMPUTE #TEMP = #VAL1 - #VAL2`
	
	Statements 0110 and 0130 will be impacted, but statements 0100 and 0120 will not.

Impact Type	7L
Modification	Manual.

Arrays within group fields

Impact	Impact will look for any group fields that have been defined as an array. Example: **0210 01 #GROUP-ARRAY(5)** 0220 02 #GROUP-FIELD1 (A2) 0230 02 #GROUP-FIELD2 (A2) 0240 02 #GROUP-FIELD3 (A2) 0250 02 #GROUP-FIELD4 (A2) :::: 0370 01 #GROUP-GROUP-ARRAY 0380 02 #GGA-NAME (A10) 0390 02 #GGA-PASSWORD (A8) **0400 02 #GGA-GROUP-OF-OPTIONS(1:4)** 0410 03 #GOO-OPTION1 (A2) 0420 03 #GOO-OPTION2 (A2) *Note: Only applies to Structured Mode objects.*
Impact Type	7M
Modification	Manual.

External Objects

Impact	Impact will look for any statements that reference external objects.
	The following Natural keywords are included:
	FETCH, FETCH RETURN, FETCH REPEAT, CALLNAT, PERFORM (external subroutines only).
	Example:
	`0190 FETCH RETURN 'PGM1' #PARM-FIELD1`
	`::::`
	`0330 CALLNAT 'SUBPGM1' #PARM-FIELD1 #PROCESS-OPTION`
	`::::`
	`0490 PERFORM ##EXTERNAL-SUBROUTINE1`
	Note: Calls to external objects can degrade performance as each called object will need to use the object load process. For example, calls to an external subroutine are more expensive than using internal subroutines.

Impact Type	7N
Modification	Manual. *Note: Impacted calls to external objects can be reviewed to see if incorporating them into calling object is feasible. Consideration should be given to any maintenance overheads that may result.*

Callnats and number of parms

Impact	Impact will look for any CALLNAT statements and the number of fields used in the parameter list. Example: `0100 CALLNAT 'SUBPGM01'` **`0110 CALLNAT 'SUBPGM02' #P-FIELD1 #P-FIELD2`** **`0120 CALLNAT 'SUBPGM03' #PARM-GROUP`** *Note: The transfer of parameter fields to an external object can be a performance issue if the parameter list contains a large number of fields. Performance can be improved by rationalizing the number of fields in the parameter list.*
Impact Type	7O
Modification	Manual.

System variables referenced GT 1

Impact	Impact will look for any System Variables that are referenced more than once within an object. This applies to all System Variables except for: *ISN, *COUNTER, *NUMBER. Example: **`0340 IF *PROGRAM EQ 'PROGRAM1'`** **`0350   MOVE *PROGRAM TO #STORE-PROGRAM1`** `0360 END-IF` *Note: A System Variable is used to access data that resides in one of the Natural control structures. The data for these variables is accessed using special internal Natural access modules, which impacts the performance as it is slower than accessing a scalar variable. Performance can be improved by copying the System Variable to a scalar variable.*
Impact Type	7P
Modification	Manual.

PDA fields in calculations

Impact	Impact will look for any Parameter Data Area fields that are used within arithmetic statements.
	
	The following Natural keywords are included:
	
	ADD, ADD ROUNDED, SUBTRACT, SUBTRACT ROUNDED, DIVIDE, DIVIDE ROUNDED, MULTIPLY, MULTIPLY ROUNDED, ASSIGN, ASSIGN ROUNDED, COMPUTE, COMPUTE ROUNDED.
	
	Example:
	`0010 DEFINE DATA PARAMATER`
	`0020 01 #VAL1 (N7)`
	`0030 01 #VAL2 (N7)`
	`0040 01 #TEMP (P7)`
	`0050 END-DEFINE`
	`::::`
	`0150 ADD #VAL1 TO #VAL2`
	`0160 SUBTRACT ROUNDED #VAL2 FROM #VAL1 GIVING #TEMP`
	`0170 DIVIDE #VAL1 INTO #VAL2 GIVING #TEMP`
	`0180 MULTIPLY ROUNDED #VAL1 BY #VAL2 GIVING #TEMP`
	
	ASSIGN and COMPUTE statements that are 'computational' will be impacted, but 'assignment' ASSIGN and COMPUTE statements will not be impacted.
	
	Example:
	`0200 ASSIGN #VAL1 = #VAL2`
	`0210 ASSIGN #TEMP = #VAL1 - #VAL2`
	`0220 COMPUTE #VAL1 = #VAL2`
	`0230 COMPUTE #TEMP = #VAL1 - #VAL2`
	
	Statements 0210 and 0230 will be impacted, but statements 0200 and 0220 will not.

Impact Type	7Q
Modification	Manual.

Array assignments in non-db loops

Impact	Impact will look for any 'assignment' statements within non-database loops that use fields defined as arrays.
	Note: There must be an array field on both sides.
	The following Natural keywords are included:
	MOVE, MOVE ALL, MOVE EDITED, MOVE ROUNDED, MOVE LEFT, MOVE RIGHT, MOVE INDEXED, MOVE BY NAME, MOVE BY POSITION, ASSIGN, ASSIGN ROUNDED, COMPUTE, COMPUTE ROUNDED.
	Example:
	`::::`
	`0300 MOVE #ARRAY1(*) TO #ARRAY2(*)`
	`::::`
	`0400 FOR #INDEX EQ #LOOP-START TO #LOOP-END`
	`0410   MOVE #ARRAY1(#INDEX) TO #ARRAY2(#INDEX)`
	`0420 END-FOR`
	`::::`
	`0500 READ (10) VEHICLES`
	`0510   MOVE #ARRAY1(*) TO #ARRAY2(*)`
	`0520 END-READ`
	Statement 0300 will not be impacted as it is not bound by a non-database loop.
	Statement 0510 will not be impacted as it is bound by a database loop.

Impact Type	7R
Modification	Manual.

DECIDE ON using system variables

Impact	Impact will look for any DECIDE ON statements that reference system variables, for example *PF-KEY, *DATE.
	
	Example:
	`::::`
	`0120 DECIDE ON FIRST VALUE OF *PF-KEY`
	`0130   VALUE 'PF1'`
	`0140     WRITE NOTITLE 'PF1'`
	`0150   NONE`
	`0160     IGNORE`
	`0170 END-DECIDE`
	
	Note: When using a DECIDE ON statement for a system variable, it is more efficient to use a temporary variable of the same format and length.

Impact Type	7U

Modification	Automatic.
	
	Modification will generate a temporary variable and move the system variable to this temporary variable. The DECIDE ON statement will then be changed to reference the temporary variable rather than the system variable.
	
	Example:
	`0100 01 #NEE-PF-KEY (A04) /* NEE MODIFIED`
	`::::`
	`0140 /* DECIDE ON FIRST VALUE OF *PF-KEY /* NEE OLD CODE`
	`0150 MOVE *PF-KEY TO #NEE-PF-KEY /* NEE MODIFIED`
	`0160 DECIDE ON FIRST VALUE OF #NEE-PF-KEY /* NEE MODIFIED`
	`0170   VALUE 'PF1'`
	`0180     WRITE NOTITLE 'PF1'`
	`0190   NONE`
	`0200     IGNORE`
	`0210 END-DECIDE`

Move Occurrence No. to each PE member

Impact	Impact will look for any logical view data definitions where the occurrences for a PE group are defined at the group level rather than for each child level within that group.
	
	Example:
	`0090 01 REP1 VIEW GSREPOSITORY`
	`0100   02 REC-TYPE`
	`0110   02 PROGRAM`
	`0120   02 LIBRARY`
	`0130   02 FIELD-GROUP(1:5)`
	`0140     03 FIELD-NAME`
	`0150     03 FIELD-VALUE`
	
	Note: Certain operations will perform faster if referenced array fields are defined in a contiguous manner.

Impact Type	7W

Modification	Automatic.
	
	Modification will change the PE group definition so that the number of occurrences is at each child level within the group and remove the number of occurrences from the PE group variable.
	
	Example:
	`0090 01 REP1 VIEW GSREPOSITORY`
	`0100   02 REC-TYPE`
	`0110   02 PROGRAM`
	`0120   02 LIBRARY`
	`0130 /*   02 FIELD-GROUP(1:5) /* NEE OLD CODE`
	`0140   02 FIELD-GROUP /* NEE MODIFIED`
	`0150 /*     03 FIELD-NAME /* NEE OLD CODE`
	`0160     03 FIELD-NAME (1:5) /* NEE MODIFIED`
	`0170 /*     03 FIELD-VALUE /* NEE OLD CODE`
	`0180     03 FIELD-VALUE (1:5) /* NEE MODIFIED`

FOR & REPEAT loops to use named constants

Impact	Impact will look for any FOR and REPEAT statements that make use of numeric literal values. Example: **0180 FOR #IDX EQ 1 TO 10** 0190 WRITE NOTITLE 'HELLO' 0200 END-FOR
Impact Type	7Y
Modification	Automatic. Modification will change the impacted FOR and REPEAT statements to use variables defined with CONSTANT values. Example: **0120 01 #NEE-1 (P09) CONST <1> /* NEE MODIFIED** **0130 01 #NEE-10 (P09) CONST <10> /* NEE MODIFIED** :::: **0200 /* FOR #IDX EQ 1 TO 10 /* NEE OLD CODE** **0210 FOR #IDX EQ #NEE-1 TO #NEE-10 /* NEE MODIFIED** 0220 WRITE NOTITLE 'HELLO' 0230 END-FOR

Split STACK COMMAND stmts with embedded data

Impact	Impact will look for any STACK COMMAND statements and check if both data and commands have been stacked with the same statement. Example: `0270 STACK COMMAND 'COMMAND' #COMMAND1` *Note: For improved performance, any use of the Natural command stack should separate the commands from the data. All commands are copied to the command input buffer, while the data is stacked for input processing within the object, resulting in improved efficiency.*
Impact Type	71

Modification	Automatic.
	Modification will split the STACK COMMAND statements impacted into STACK DATA and STACK COMMAND statements.
	Example:
	`0270 STACK TOP DATA #COMMAND1 /* NEE MODIFIED`
	`0280 /* STACK COMMAND 'COMMAND' #COMMAND1 /* NEE OLD CODE`
	`0290 STACK COMMAND 'COMMAND' /* NEE MODIFIED`

Find unused source code lines

Impact	Impact will look for any unused source code lines within programming objects, across a whole application.
	
	Any source code that is driven by event rather than position is ignored. For example AT BREAK, AT END OF PAGE, WRITE TITLE.
	
	Unused source code within internal subroutines is included, but for external subroutines the Unused Objects report should be referenced.
	
	Note: Only Structured Mode objects will be impacted. For Reporting Mode objects, it is recommended that they are converted to Structured Mode first using the Mode Conversion function, and then impacted.
	
	Example:
	`0290 IF #CHOICE = 'X'`
	`0300   MOVE 'COPTZP4S' TO #FIELD-A`
	`0310   RESET #FIELD-N`
	`0320   FETCH 'COPTZP4S' #FIELD-A #FIELD-N`
	`0330   RESET #FIELD-N`
	`0340   MOVE 'XYZ'TO #FIELD-A`
	`0350 END-IF`

Impact Type	73
Modification	Manual.

OBJECT BUILDER

The combination search keyword OBJECT BUILDER is used to specify line ranges within an object, which will then be used to create a new subprogram object containing the specified lines and a Parameter Data Area object containing any parameter data that is required. The original object is then modified to call the new subprogram.

This process is part of the Application Restructuring processes available within Natural Engineer.

Note: For more information on the Object Builder process refer to the Natural Engineer Application Restructuring for Windows manual.

Specifying Object Builder

Select Impact Type OBJECT BUILDER from the Criteria Details tab screen.

Note: OBJECT BUILDER criteria are mutually exclusive to all other Impact Types within a single Impact Version.

The following Figure 3-14 illustrates the Criteria Detail tab screen for combination keyword OBJECT BUILDER.

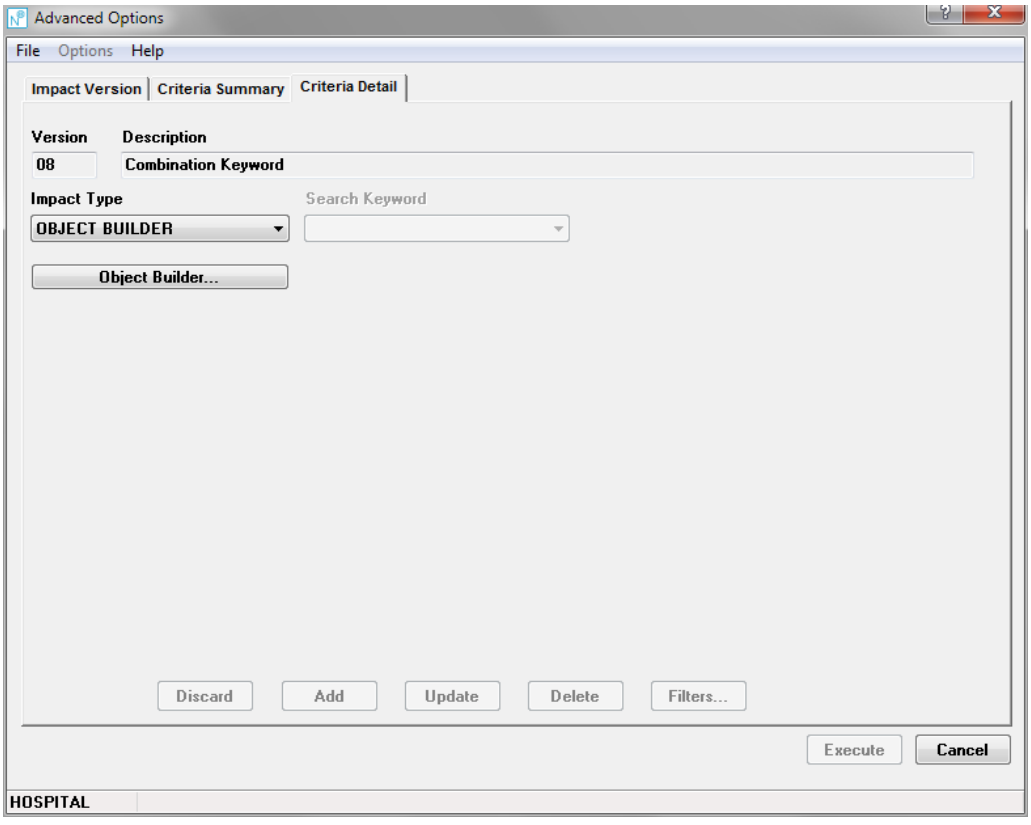

Figure 3-14 Criteria Detail tab screen for combination keyword OBJECT BUILDER

Note: For more information on the Criteria Details tab screen refer to the section Criteria Detail tab screen in Chapter 1.

Object Builder Window

The Object Builder screen can be accessed from the Criteria Detail tab screen by using the 'Object Builder…' button. Alternatively, existing Object Builder criteria can be selected from the Criteria Summary tab screen by double clicking with the left mouse button.

The following Figure 3-15 illustrates the Object Builder screen.

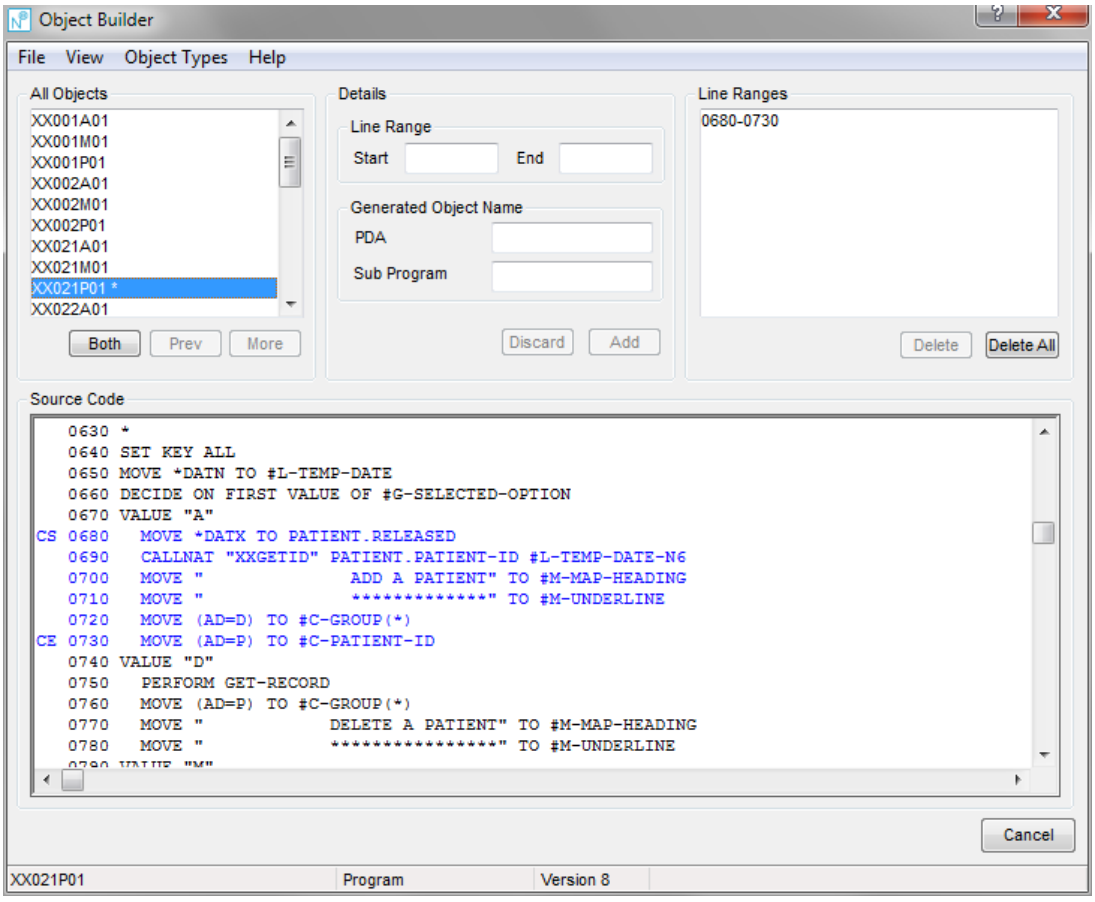

Figure 3-15 Object Builder screen

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Exit**	Exit the Object Builder screen and return back to the Criteria Detail tab screen.
View	**Change Start Position of Object List…**	Reposition the list of objects to start from a particular object name. The reposition value can be input using either a complete name or part name using an '*' (asterisk) wildcard. The reposition value is appended to the object list title to highlight the type of repositioning being applied. Possible reposition values are:

Value	Result
' ' (blank)	Reposition to the top of the object list.
*****	Reposition to the top of the object list.
ABC*	Only show objects that are prefixed by 'ABC'.
XYZ	Reposition to the first object that either matches or is greater than 'XYZ' and then continue the object list from that point.

Object Types Allows you to select the types of object to be listed.

Available selections are:

- **All Objects**
- **Copycodes**
- **Functions**
- **Helproutines**
- **Maps**
- **Programs**
- **Subprograms**
- **Subroutines**

Help Invoke the Object Builder help.

SCREEN ITEMS	DESCRIPTION
Object List	List of all the objects used by the currently selected application. The list of objects can be tailored to your requirements using the options available in the Object Types menu. Further refinement can be made using the option 'Change Start Position of Object List…' from the View menu. The Object List title reflects the Object Types being listed and will append any reposition values that may have been specified. Objects can be selected using the left mouse button with a single click. Any objects that have had line ranges saved will show with an asterisk (*) to the right of the object name. For example: XX021P01 *.
Line Range group:	
Start	The start of range statement line number.
End	The end of range statement line number.
Generated Object Name group:	
PDA	The generated PDA object name override.
SubPgm	The generated Subprogram object name override.
Line Ranges group:	
Line Ranges List	List of all the line ranges and generated object name overrides that have been saved for an object. Selecting any of the line ranges will reposition the source code to show the start of range.
Source Code	Display the selected source code for the currently selected object. Any source code lines that are part of a saved line range will be colored blue and will have the following in the first 2 bytes of the line: **CS** Indicates that the line is the start of a range. **CE** Indicates that the line is the end of a range. *Note: If a single line has been selected for start and end ranges, e.g., 0100-0100, then the indicator will show only CS.*

BUTTON NAME	DESCRIPTION
Object List group:	
Both	This button provides additional refinement of the objects listed in the Objects List box. This button has three different states, with the button text changing accordingly: **Both** The default for the screen is to list all objects whether they have line ranges saved or not. **OEM O** Only list objects that have line ranges saved, i.e., OEM data available. **N OEM** Only list objects that have no line ranges saved, i.e., no OEM data available.
Prev	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.
More	Scrolls the object list to forward one page. This button will be available/unavailable depending on the value specified in the LISTBOXMAX parameter in the NATENG.INI file.
Details group:	
Discard	Reset the current Line Range and Generated Object name details. If the details are new, then reset blank out all inputs in the Details group. If the details are update, reset them to previous state (when selecting existing saved Line Ranges).
Add / Update	Add / Update the current Line Range and Generated Object name details. Button text will read 'Add' for new details and 'Update' for any existing details (when selecting existing saved Line Ranges).
Line Ranges group:	
Delete	Delete the selected line range.
Delete All	Delete all line ranges.
Object Builder screen:	
Cancel	Cancel the Object Builder process and return back to the Criteria Details tab 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.

STATUS BAR ITEM	DESCRIPTION
The Object Builder status bar is divided into 4 individual panes.	
Pane 1	Name of the selected object. If the selected object is from a steplib, then the steplib library name will also be shown here.
Pane 2	Object type of the selected object.
Pane 3	The currently selected Impact Version number.
Pane 4	Any Object Builder processing messages.

NATRPC

The combination search keyword NATRPC can be used to locate statements that may be affected by future versions of Natural, in particular with the Natural Remote Procedure Call (RPC). The impacted statements for Natural RPC are detailed below showing their respective current and future functionality:

Current State

Statement	Description
TERMINATE	Using this statement causes the server to be terminated, regardless of conversations that may still be open.
FETCH,	Using these statements causes the CALLNAT context to get lost.
RUN, **STOP**	Upon a FETCH, RUN or STOP statement, the server detects that it has lost its CALLNAT context and returns a corresponding Natural error message to the client; at that time, however, the statement has already been executed by the server.
	Exception: This does not apply to FETCH RETURN.
INPUT	Input values are unpredictable when the input data are read from a file (and not from the stack).

Future State

Statement	Description
FETCH, RUN, INPUT	Not Permitted.
STOP, TERMINATE	Same as ESCAPE ROUTINE.

Natural Engineer Analysis and Modification

Using the search keyword NATRPC, Natural Engineer will identify those statements that may be affected with future versions of Natural.

Automatic modification will also be performed for instances of STOP and TERMINATE where the statement will be replaced with ESCAPE ROUTINE.

Specifying NATRPC

Select Impact Type NATRPC from the Criteria Details tab screen.

Note: Only one set of NATRPC criteria is allowed per Impact Version.

The following Figure 3-16 illustrates the Criteria Detail tab screen for combination keyword NATRPC.

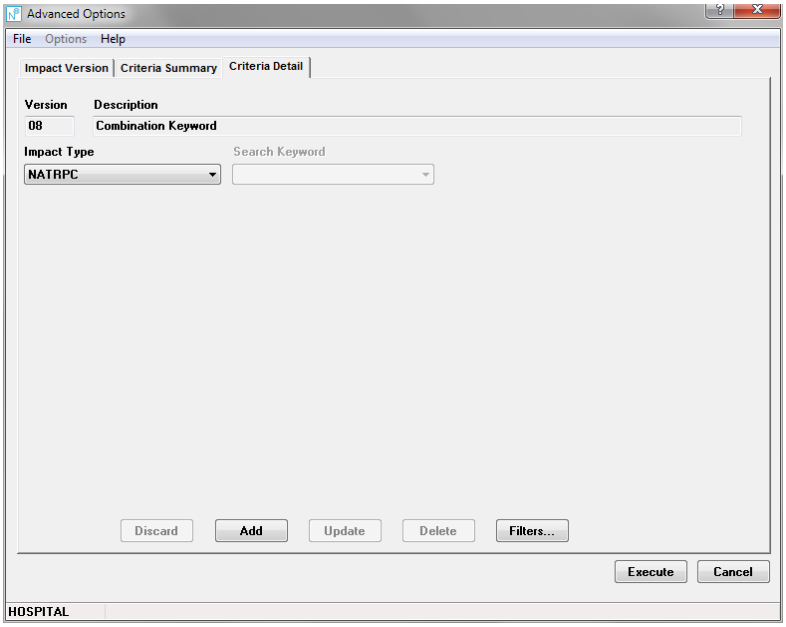

Figure 3-16 Criteria Detail tab screen for combination keyword NATRPC

Note: For more information on the Criteria Details tab screen refer to the section Criteria Detail tab screen in Chapter 1.

Setting Object Filters

The NATRPC criteria can be further refined by only applying the criteria to specific object types and/or object names during Impact execution.

This can be set by using the 'Filters…' button on the Criteria Details tab screen to invoke the Filters window.

Note: For more information on Filters refer to the Filters Window section.

MULTI SEARCH

The combination search keyword MULTI SEARCH allows for more advanced search criteria to be specified, including, conditions that are to be included in the analysis, or excluded from the analysis.

The combination search keyword MULTI SEARCH allows up to 500 search conditions to be combined. These may be DATAITEM searches, DBFILE searches, LITERAL searches and/or attribute searches.

The MULTI SEARCH criteria can be saved to a PC text file, allowing them to be re-used across applications. These files will have a file extension of '.ISC'. By default these files will be saved to the data folder where Natural Engineer is installed.

Note: Natural Engineer comes supplied with a default MULTI SEARCH criteria file ###DEF01.ISC. For more information refer to Chapter 3 in the Natural Engineer Administration Guide for Windows manual.

Specifying MULTI SEARCH

Select Impact Type MULTI SEARCH from the Criteria Details tab screen.

Note: MULTI SEARCH criteria are mutually exclusive to all other Impact Types within a single Impact Version.

The following Figure 3-17 illustrates the Criteria Detail tab screen for combination keyword MULTI SEARCH.

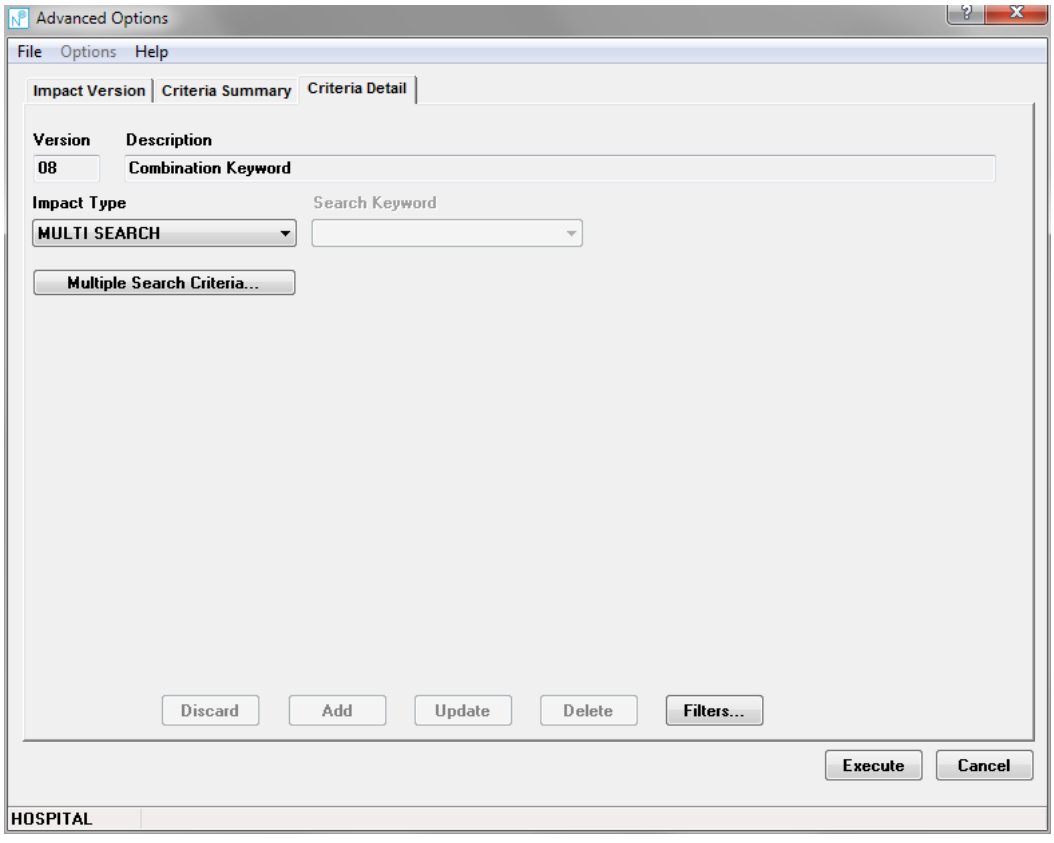

Figure 3-17 Criteria Detail tab screen for combination keyword MULTI SEARCH

Note: For more information on the Criteria Details tab screen refer to the section Criteria Detail tab screen in Chapter 1.

Multi Search Criteria Window

The Multi Search Criteria screen can be accessed from the Criteria Detail tab screen by using the 'Multi Search Criteria…' button. Alternatively, existing Multi Search criteria can be selected from the Criteria Summary tab screen by double clicking with the left mouse button.

The following Figure 3-18 illustrates the Multi Search Criteria screen.

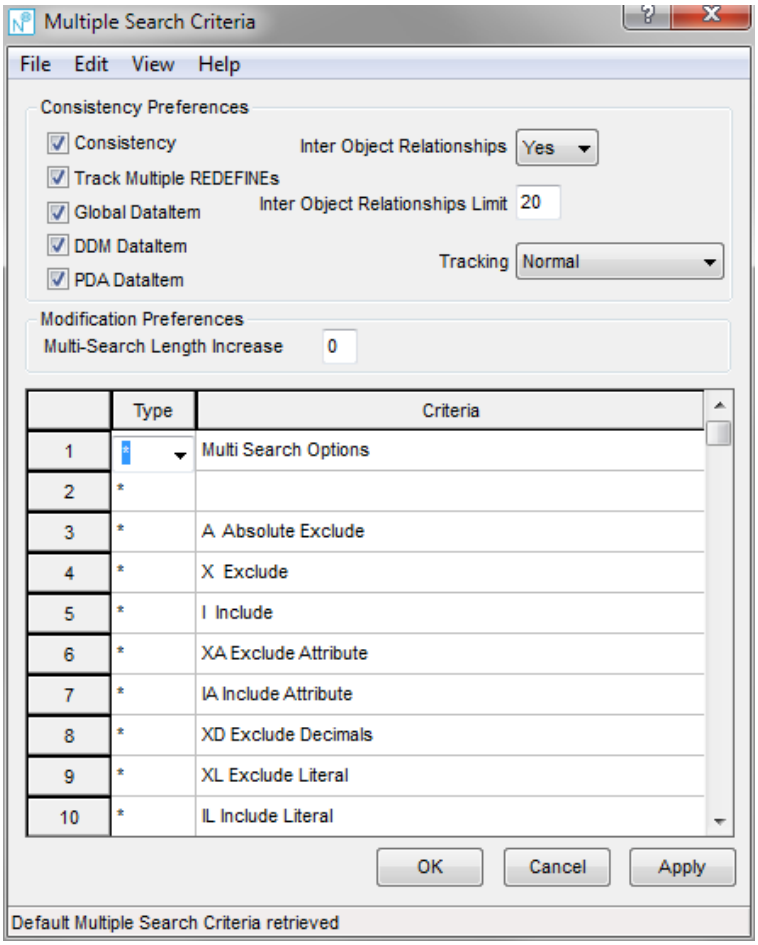

Figure 3-18 Multi Search Criteria screen

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Open…**	Open a Multi Search Criteria PC file. These files will have a file extension of '.ISC'. By default these files are located in the data folder where Natural Engineer is installed.
	Open Default Criteria	Open the default Multi Search Criteria file '###DEFnn.ISC', where nn refers to the language code specified in the LANG parameter in the NATENG.INI file. *Note: For more information on the NATENG.INI file parameter LANG refer to Chapter 1 in the Natural Engineer Administration Guide for Windows manual.*
	Save…	Save the current Multi Search Criteria for the currently selected version to a PC file. These files will have a file extension of '.STD'. By default these files will be saved to the data folder where Natural Engineer is installed.
	Exit	Exit the Multi Search Criteria screen and return back to the Criteria Details tab screen.
Edit	**Insert Row**	Add a new row to the criteria list.
	Delete Row	Delete an existing row from the criteria list.

MENU ITEMS	OPTIONS	DESCRIPTION
View	**Sort Criteria**	Sort all the Multi Search criteria into the following criteria type order: ▪ **Absolute Excludes (A)** ▪ **Exclude Field (X)** ▪ **Include Field (I)** ▪ **Exclude Format (XA)** ▪ **Include Format (IA)** ▪ **Exclude Decimals (XD)** ▪ **Exclude Literal (XL)** ▪ **Include Literal (IL)** *Note: Comments (*) maintain position for the criteria type that they precede. For example:* ** Exclude MESSAGE fields* *X ?MESSAGE?* *Would always be kept together as a pair.* *Any comments found in line 1 will always be retained at line 1 and can be used as a description line for the criteria set.*
Help	Invoke the Multi Search Criteria help.	

SCREEN ITEMS	DESCRIPTION
Consistency Preferences Group:	
Consistency	The processing level to be applied to the Multi Search criteria during Impact Analysis execution. If marked impact will perform a multiple iteration process matching the specified search criteria and identifying any derivative. If not marked impact will perform a single iteration process only matching the specified search criteria. No derivative tracking will be performed.
Inter Object Relationships	If set to Y, data elements are tracked across object boundaries following the impact process. Setting the value to ONLY will ensure that Natural Engineer only performs inter object relationships. Please note that this setting should only be used following a successful multiple impact or to restart a previously failed Inter Object Relationship (IOR) process. Possible values Y,N, ONLY

SCREEN ITEMS	DESCRIPTION
Inter Object Relationships Limit	This is the number of Iterations that IOR will track objects across object boundaries.
Tracking	Controls the tracking direction for a variable. If set to Forwards, a variable is tracked in a forward direction, showing all the derivatives being populated from the variable. If set to Backwards, a variable is tracked in a backward direction, showing where the variable and derivatives have been populated. If set to Normal, both the forward and backward directions will be shown. *For more information refer to section Forward/Backward Tracking*
Track Multiple REDEFINEs	If selected, multiple redefines are tracked.
Global Dataitem	Used when impacts have been made to Global Data Areas. If selected, Natural Engineer will track these fields, and derivations of these fields, until all possible impacts have been identified.
DDM Dataitem	Used when impacts have been made to DDMs. If selected, Natural Engineer will track these fields, and derivations of these fields, until all possible impacts have been identified.
PDA Dataitem	Used when impacts have been made to Parameter Data Areas. If selected, Natural Engineer will track these fields, and derivations of these fields, until all possible impacts have been identified.
Modification Preferences Group:	
Multi-Search Length Increase	Used by Multi Search to increase the default length of an impacted field when Modification is executed. For example: Field #A is defined as (N2) and Multi-Search Length Increase is set to 2. After Modification, #A will have its length increased to (N4).
Criteria List:	
Criteria Number	Sequential number for each criteria starting from 1.
Criteria Type	The type of Multi Search criteria. *Note: For more information refer to section Multi Search Criteria Types.*
Criteria Value	The criteria that are to be applied. These can be full or partial field names. They can also be full or partial values of literal strings and numerics. If partial names or values are used, they must be entered using wild card '?' *Note: For more information refer to section Criteria Values.*

BUTTON NAME	DESCRIPTION
OK	Save the Multi Search Criteria settings and close the current screen.
Cancel	Cancel the Multi Search Criteria process and return back to the Criteria Details tab screen.
Apply	Save the Multi Search Criteria settings and retain the current screen. *Note: This button is only enabled if any changes have been made.*

STATUS BAR ITEM	DESCRIPTION
Pane	Any Multi Search Criteria processing messages.

.

Setting Object Filters

The MULTI SEARCH criteria can be further refined by only applying the criteria to specific object types and/or object names during Impact execution.

This can be set by using the 'Filters…' button on the Criteria Details tab screen to invoke the Filters window.

Note: For more information on Filters refer to the Filters Window section.

Multi Search Criteria Types

There are nine criteria types that can be used:

1. **'*' – Comment Line**

- These criteria types allow for comments to be placed anywhere within the criteria list for any documentation notes or as separators between the criteria.
- They are ignored by the Impact process.

2. **'A' – Absolute Exclude Field**

- These criteria types will result in the Impact process excluding any fields that are an exact match for the value specified. These fields will be permanently excluded from the Impact process.
- These criteria types must be at the top of the criteria list, before any other criteria types.

3. **'X' – Exclude Field**

- These criteria types will result in the Impact process excluding parts of a field name, while the remainder of the field name is still searched for inclusion.
- These criteria types make use of the 'Byte-Lockout' process.
  Note: For more information refer to the section Byte-Lockout Process.
- These criteria types must be located before the 'I' – Include Field criteria types in order to benefit from the 'Byte-Lockout' process.

4. **'I' – Include Field**

- These criteria types will result in the Impact process including any fields that match the value specified.
- If a criteria value is specified without the use of wild card '?', then it is an absolute include. For example: 'I DATE'.
- If the INI file parameter DEF-REM-LEN is set, then the impacted fields will have their length increased by the value specified during Modification.
- These criteria types must be located after the 'X' – Exclude Field criteria types.

5. 'XA' – Exclude Format

- These criteria types will result in the Impact process excluding any field that matches the format value. For example: 'XA A001' would exclude all fields with a format of A001.
- These criteria types can have format ranges specified. For example:

'XA N1-N20' will exclude any fields that have a format of N and a length between 1 and 20. This would include any fields that have decimal places, e.g., N6.2.

'XA P5.2-P9.2' will exclude any fields with format N and length between 5.2 and 9.2.
Note: If a comma is used to reference a decimal place, then Natural Engineer will convert them to decimal points for Impact to use (i.e., 5,3 becomes 5.3). For Modification they will be converted back to a comma.

'XA N-N999' will exclude any numeric type field.

- These criteria types must be located before the 'IA' – Include Format criteria types.

6. 'IA' – Include Format

- These criteria types will result in the Impact process including any field that matches the format value. For example: 'XA A001' would include all fields with a format of A001.
- These criteria types can have format ranges specified. For example:

'IA N1-N20' will include any fields that have a format of N and a length between 1 and 20. This would include any fields that have decimal places, e.g., N6.2.

'IA P5.2-P9.2' will include any fields with format N and length between 5.2 and 9.2.
Note: If a comma is used to reference a decimal place, then Natural Engineer will convert them to decimal points for Impact to use (i.e., 5,3 becomes 5.3). For Modification they will be converted back to a comma.

'IA N-N999' will include any numeric type field.

- These criteria types must be located after the 'XA' – Exclude Format criteria types.

7. 'XL' – Exclude Literal

- These criteria types will result in the Impact process excluding any literal string that matches the criteria value. For example: 'XL ABCDEF' would exclude all literal strings that match the value 'ABCDEF'.

  Note: If a partial value is used, then the literal string, may still be included by any subsequent 'IL' - Include Literal criteria. For example:

  using the literal string 'ABCDEF'

  *Criteria '**XL ?ABC?**' and '**IL ?DEF?**' would result in the literal string being included in the Impact Analysis.*

  *Criteria '**XL ABCDEF**' and '**IL ?DEF?**' would result in the literal string being excluded in the Impact Analysis.*

- These criteria types must be located before the 'IL' – Include Literal criteria types.

8. 'IL' – Include Literal

- These criteria types will result in the Impact process including any literal string that matches the criteria value. For example: 'IL ABCDEF' would include all literal strings that match the value 'ABCDEF'.

- These criteria types must be located after the 'XL' – Exclude Literal criteria types.

9. 'XD' – Exclude Decimal

These criteria types will result in the Impact process excluding any fields that have been defined to hold decimals. For example: N3.1, N1.4 etc.

Criteria Values

Criteria Values specify the search value for the criteria type being used. These can be entered using full or partial values.

Full Value Criteria

If full values are used then the Impact process will only match if the exact value specified matches the item. These types of criteria value are known as absolute values, i.e., an exact match must be made. For example:

0010 01 #ALPHA (A10)

0020 01 #ALPHA-BET (A26)

0030 MOVE'ABCDEF' TO #ALPHA-BET

0040 MOVE 'ABC' TO #ALPHA

Criteria 'I #ALPHA' will impact line 0010 AND 0040 only.

Criteria 'IL ABC' will impact line 0040 only.

Partial Value Criteria

Partial Value criteria can be specified by using the wildcard '?' as part of the value specification. These types of criteria value allow a range to be specified for field names or literal strings.

There are three positions the wildcard can be placed:

1. At the beginning of the value.

   Impact will look for a match for any field or literal that ends with the specified value. The format is ?value.

2. At the end of the value.

   Impact will look for a match for any field or literal that starts with the specified value. The format is value?.

3. At both the beginning and the end of the value.

Impact will look for a match for any field or literal that contains the specified value. The format is ?value?.

Examples:

0010 01 #INDEX-ONE (I02)

0020 01 #ONE-INDEX (I02)

Criteria ‘I ?ONE’ will impact line 0010 only.

Criteria ‘I ONE?’ will impact line 0020 only.

Criteria ‘I ?ONE?’ will impact line 0010 and 0020.

Additional option

For the criteria types Include (I) and Exclude (X), it is also possible to include or exclude by name and length by adding a format and length to the criteria value. This can be specified as an absolute format or a range.

For example:

‘**X ?FRED? (A10)**’ will exclude any field with a name that includes FRED and with a format and length of A10.

‘**I #TOTAL (N2-N9)**’ will include any field with a name of #TOTAL and a format and length that falls between the range N2 to N9.

Byte Lockout Process

This is a unique technique to handle parts of field names that may be included or excluded in the impact reports, depending on the defined search criteria.

For example, if you define UPDATE to be excluded but DATE to be included in the search, the UPDATE part of the field UPDATE-DATE will be excluded, but the DATE part will be included for further processing. Conversely, the field UPDATE will be excluded as defined, and not be included merely on the basis of the inclusion of DATE. This process works for literal values as well as fields.

MVSNAT22TO31

The combination search keyword MVSNAT22TO31 can be used to identify and modify incompatibilities between the syntax for Natural 2.2 and 3.1.

Note: The incompatibilities that exist between Natural 2.2.8 and Natural 3.1 are the same as those between Natural 2.2.8 and Natural 2.3.

This search keyword uses a sub-set of criteria, which can be refined by selecting/ deselecting the available options.

Specifying MVSNAT22TO31

Select Impact Type MVSNAT22TO31 from the Criteria Details tab screen.

Note: Only one set of MVSNAT22TO31 criteria is allowed per Impact Version.

The following Figure 3-19 illustrates the Criteria Detail tab screen for combination keyword MVSNAT22TO31.

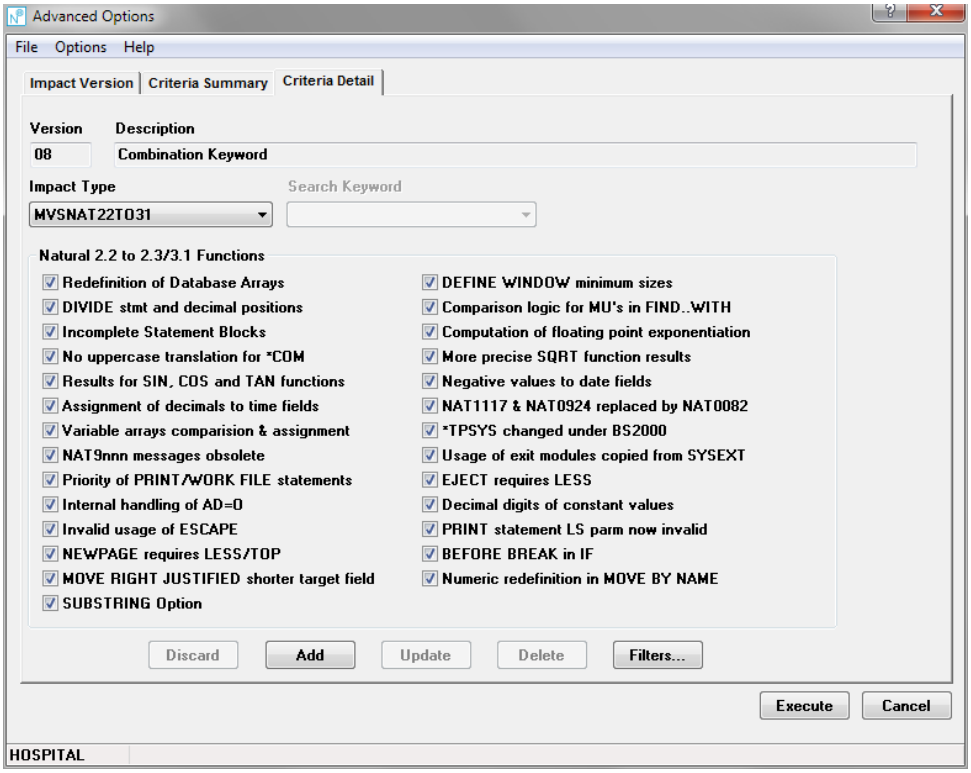

Figure 3-19 Criteria Detail tab screen for combination keyword MVSNAT22TO31

The following only describes the Criteria Detail tab screen options relevant to the combination search keyword MVSNAT22TO31.

Note: For more information on the Criteria Details tab screen refer to the section Criteria Detail tab screen in Chapter 1.

SCREEN ITEMS	DESCRIPTION
Natural 2.2 to 2.3/3.1 Functions	Each MVSNAT22TO31 option is listed A tick in the check box next to each option indicates that the option will be checked for. If a check box is blank then that option will not be referenced during Impact execution. Available selections and their respective modification types are: ▪ **Redefinition of Database Arrays (SAG01)** ▪ **DEFINE WINDOW minimum sizes (SAG02)** ▪ **DIVIDE stmt and decimal positions (SAG03/SAG3)** ▪ **Comparison logic for MU's in FIND..WITH (SAG04)** ▪ **Incomplete Statement Blocks (SAG05 R1 /SAG05 R2)** ▪ **No uppercase translation for *COM (SAG07)** ▪ **Computation for floating point exponentiation (SAG08/SAG13)** ▪ **Results for SIN, COS and TAN functions (SAG09)** ▪ **More precise SQRT function results (SAG10)** ▪ **Assignment of decimals to time fields (SAG11)** ▪ **Negative values to date fields (SAG12)** ▪ **Variable arrays comparison & assignment (SAG14)** ▪ **NAT1117 & NAT0924 replaced by NAT0082 (SAG15)** ▪ **NAT9nnn messages obsolete (SAG16)** ▪ ***TPSYS changed under BS2000 (SAG17)** ▪ **Priority of PRINT/WORK FILE statements (SAG18)** ▪ **Usage of exit modules copied from SYSEXT (SAG19)** ▪ **Internal handling of AD=O (GSL01)** ▪ **EJECT requires LESS (GSL02)** ▪ **Invalid usage of ESCAPE (GSL03)** ▪ **Decimal digits of constant values (GSL04)** ▪ **NEWPAGE requires LESS/TOP (GSL05)** ▪ **PRINT statement LS parm now invalid (GSL06)** ▪ **MOVE RIGHT JUSTIFIED shorter target field (GSL07)** ▪ **BEFORE BREAK in IF (GSL08)** ▪ **SUBSTRING Option (GSL09)** ▪ **Numeric redefinition in MOVE BY NAME (GSL10)** *Note: For more information on each of these options refer to the section Natural 2.2 / 3.1 Incompatibilities.*

Setting Object Filters

The MVSNAT22TO31 criteria can be further refined by only applying the criteria to specific object types and/or object names during Impact execution.

This can be set by using the 'Filters…' button on the Criteria Details tab screen to invoke the Filters window.

Note: For more information on Filters refer to the Filters Window section.

Natural 2.2 / 3.1 Incompatibilities

Natural Engineer identifies the following incompatibilities between Natural 2.2 and Natural 3.1.

Some of the incompatibilities can be modified using the Modification process, others will only have the Impact process available and manual modifications may need to be applied to each of these.

Note: The NEE Modification Type relates to the modification types and codes that are available from the Modification Element Maintenance screen. For more information refer to the section Modification Element Maintenance.

1. Redefinition of Database Arrays

Description	It is no longer possible to specify a variable index range in the redefinition of a periodic-group or multiple value fields.
NEE Impact	NEE will identify any variable index range specified where the field is a redefinition of a periodic-group or multiple value field.
NEE Modification Type	3a SAG01
NEE Modification	Automatic. NEE will replace the variable index range with a 1 by default. A different replacement value may be specified by setting the appropriate value required in the 'Replace Value' section on the Modification Element Maintenance screen.

2. DEFINE WINDOW Minimum Window Size

Description	The minimum size of a window definition in the SIZE clause of the DEFINE WINDOW statement is 2 lines by 10 columns without a frame and 4 lines by 13 columns with a frame. In Natural 2.2 a size smaller than the minimum may be specified. In Natural 3.1 a NAT1167 compilation error will occur if the lines are wrong, NAT1166 if columns.
NEE Impact	Natural Engineer will identify DEFINE WINDOW statements where the line or column values are less than the minimum allowed.
NEE Modification Type	3b SAG02
NEE Modification	Automatic. Natural Engineer will replace the invalid line or column value with the minimum defined values.

3. DIVIDE statement with GIVING and REMAINDER Clauses

Description	In Natural 3.1, if a DIVIDE statement has both GIVING and REMAINDER options then if the dividend (Operand 2) has more or fewer decimal positions than the result field, then different results will be shown for the REMAINDER field.
NEE Impact	Natural Engineer will identify all DIVIDE statements containing GIVING and REMAINDER where the number of decimals of the dividend is greater or fewer than that of the result. It will also identify the data definition of the relevant result field.
NEE Modification Type	3c SAG03 3h SAG03
NEE Modification	Automatic. The DIVIDE statement is identified by the SAG03 type. There is however no modification to be applied to the actual DIVIDE statement so the modification category is set to Not Applicable. The data definition for the result field is identified by the SAG03 type. This is automatically changed so that the decimal places of the data definition for the result field is the same as that of the dividend.

4. Comparison Logic for MU's in FIND..WITH

Description	The comparison logic for multiple value fields in the WITH clause of the FIND statement has been changed to be in line with other comparison logic in other statements, for example IF. 1. FIND XYZ-VIEW WITH MU = ’A’ With Natural 2.2 and 3.1, this statement returns records in which at least one occurrence of MU has the value 'A'. 2. FIND XYZ-VIEW WITH MU NOT EQUAL ’A’ With Natural 2.2, this statement returns records in which no occurrence of MU has the value 'A' (same as 4.). With Natural 3.1, this statement returns records in which at least one occurrence of MU does not have the value 'A'. 3. FIND XYZ-VIEW WITH NOT MU NOT EQUAL ’A’ With Natural 2.2, this statement returns records in which at least one occurrence of MU has the value 'A' (same as 1). With Natural 3.1, this statement returns records in which every occurrence of MU has the value 'A'. 4. FIND XYZ-VIEW WITH NOT MU = ’A’ With Natural 2.2 and 3.1, this statement returns records in which no occurrence of MU has the value 'A'. This means that if you newly compile under Natural 3.1 existing Natural 2.2 programs containing FIND statements of the type 2 and 3., they will return different results.
NEE Impact	Natural Engineer will identify all FIND statements using an MU field in the WITH clause like: 1) MU NOT EQUAL Value 2) NOT MU NOT EQUAL Value
NEE Modification Type	3d SAG04
NEE Modification	Automatic. The modification will be applied as follows: 1) MU NOT EQUAL Value to NOT MU = value 2) NOT MU NOT EQUAL Value to MU = Value

5. Empty Statement Blocks for FOR and REPEAT

Description	In Natural 2.2 an empty statement block e.g. FOR or REPEAT may not lead to compilation errors. In Natural 3.1, this is removed.
NEE Impact	
NEE Modification Type	3e SAG05R1 3@ SAG05R2
NEE Modification	Automatic. The default modification for this problem is SAG05R1. This will insert into the empty statement block an IGNORE statement based on the TLM N31R05T1. An alternative modification is SAG05R2. This will comment out the empty statement block but then insert a line of code to set the applicable variable to the maximum value. For Example: FOR #A = 1 TO 10, will insert MOVE 10 TO #A. This is based on the TLM N31R05T2. The insertion of this TLM will only apply to FOR loops. REPEAT loops will only be commented out and NO TLM will be inserted. Before applying this modification, the TLM's N31R05T1 and N31R05T2 need to be copied from the SYSNEE library to SYSTEM or the modification library of the Natural Engineer Application. In addition Modification Preferences will need to be defined for the specific Modification Type to assign the TLM's to the type.

6. No Upper-case translation for *COM

Description	In Natural 2.2 *COM may be specified with an AD=T attribute. This is ignored at runtime but not rejected at compile. In Natural 3.1 this will lead to a compile error NAT0335.
NEE Impact	Natural Engineer will identify all occurrences of *COM with an AD=T attribute.
NEE Modification Type	3g SAG07
NEE Modification	Automatic. The AD=T attribute is removed from the *COM data item.

7. Computation of Floating-Point Exponentiation Corrected

Description	With Natural 2.2, if, in an exponentiation, both the base and the exponent are of floating-point format, the length of the exponent is used for the computation of the result. With Natural 3.1, if, in an exponentiation, both the base and the exponent are of floating-point format, the length of the base is used for the computation of the result. With Natural 2.2, if, in an exponentiation, the exponent is of floating-point format and the base is not, the base is internally converted to format/length F4 or F8, depending on the length of the base. With Natural 3.1, if, in an exponentiation, the exponent is of floating-point format and the base is not, the base is internally always converted to format/length F8 so as to get the greatest possible precision. Both the above corrections may in some cases lead to different results; however, these results will be of a greater precision.
NEE Impact	Natural Engineer will identify all relevant occurrences.
NEE Modification Type	3n SAG13/SAG08
NEE Modification	Manual.

8. Results for SIN, COS and TAN functions

Description	With Natural 2.2, when the mathematical functions SIN, COS and TAN (sine, cosine and tangent) are applied to very large numbers (equal to or greater than 10++17), they may in some cases return incorrect results. With Natural 3.1, for numbers equal to or greater than 10++17 the sine will be 0, the cosine will be 1 and the tangent will be 0. This may in some cases lead to different results.
NEE Impact	Natural Engineer will identify all SIN, COS and TAN references.
NEE Modification Type	3i SAG09
NEE Modification	Manual.

9. More precise SQRT function results

Description	With Natural 3.1, the computation of the mathematical function SQRT (square root) has been improved for floating-point operands. This may in some cases lead to different results. However, these results will be of a greater precision.
NEE Impact	Natural Engineer will identify all relevant SQRT references and all data items used in these statements.
NEE Modification Type	3j SAG10
NEE Modification	Manual.

10. Assignments of Numbers with Decimal Digits to Time Fields

Description	With Natural 2.2, if numbers (format N or P) decimal positions are assigned/moved to a time field (format T), the entire number is assigned/moved as an integer; that is, the decimal point is ignored. With Natural 3.1, this error has been corrected: The positions after the decimal point will be truncated, or rounded (if the ROUNDED option is used in the corresponding COMPUTE or MOVE statement). This may lead to different results, which will, however, be correct.
NEE Impact	Natural Engineer will identify all COMPUTE, MOVE statements moving a P, N variable or a value with decimal digits to a time field.
NEE Modification Type	3k SAG11
NEE Modification	Manual.

11. Negative Values to Date Fields

Description	It is not allowed to assign a negative value to a date field (format D) or a time field (format T). With Natural 2.2, however, such invalid assignment at runtime may in some cases not be intercepted. With Natural 3.1, this has been corrected: The assignment of a negative value to a date or time field will always lead to an error (NAT1319).
NEE Impact	All negative assignments to Date or Time fields are identified.
NEE Modification Type	3m SAG12
NEE Modification	Manual.

12. More precise Results for Floating Point Conversions

Description	The format conversion for the transfer of data from floating-point fields (format F) to packed numeric fields (format P) and vice versa, as well as from floating-point fields to alphanumeric fields (format A) and vice versa, has been improved. This may in some cases lead to different results. However, these results will be of a greater precision than with Natural 2.2.
NEE Impact	Natural Engineer will identify all relevant occurrences.
NEE Modification Category	3n SAG13/SAG08
NEE Modification	Manual.

13. Comparison and Assignment of Variable Array Ranges

Description	With Natural 3.1, a comparison or assignment involving arrays with variable indexes will lead to an error at runtime (NAT1317) if an array range turns out to be actually a scalar once the actual values are assigned to the index variables. With Natural 2.2, such a comparison or assignment is allowed, but it is not consistent with the handling of constant scalars (as shown in the following example). Example (assuming j = i + 1): Natural 2.2: 1. IF #A(i:j) = #B(m) is resolved as: IF #A(i) = #B(m) OR #A(j) = #B(m) 2. IF #A(i:j) = #B(m:n) is resolved as: IF #A(i) = #B(m) AND #A(j) = #B(n) This means that if the values of 'm' and 'n' are equal, comparison 2 is resolved inconsistently. Natural 3.1: If the values of 'm' and 'n' are equal, comparison 2. Will cause a runtime error.
NEE Impact	Natural Engineer will identify all comparisons or assignments using variable indexed arrays.
NEE Modification Type	3o SAG14
NEE Modification	Not Applicable

14. Error Messages NAT1117 and NAT0924 replaced by NAT0082

Description	In situations where Natural 2.2 displays error message NAT1117 (requested map not available) or NAT0924 (subroutine, GDA or external report not found), Natural 3.1 displays message NAT0082. This will lead to different results if you interrogate these message numbers in your applications.
NEE Impact	Natural Engineer will identify any literal strings containing 1117 and 924.
NEE Modification Type	3p SAG15
NEE Modification	Manual (default). This can be updated to 'Automatic' via the Modification Element Maintenance screen and then applied by executing the Modification process. All references to 1117 and 924 will be modified to be 0082.

15. Obsolete Error Messages

Description	The following error messages have become obsolete; they no longer exist with Natural 3.1: NAT9000, NAT9100, NAT9101 and NAT9200.
NEE Impact	Natural Engineer will identify all references of NAT9000, NAT9100, NAT9101 and NAT9200.
NEE Modification Type	3q SAG16.
NEE Modification	Manual.

16. Changed System Variable *TPSYS

Description	Under TIAM (BS2000) *TPSYS contains 'TIAM' instead of 'RTIO'
NEE Impact	Natural Engineer will identify all references of 'RTIO'.
NEE Modification Category	3r SAG17.
NEE Modification	Automatic. All references of 'RTIO' will be modified to 'TIAM'.

17. Priority of PRINT/WORK FILE statements

Description	The NATPARM definitions of Print/Load files have priority over the JCL definitions. Special purpose ZAP NA32116 puts back Natural 2.2 functionality.
NEE Impact	Natural Engineer will identify all PRINT/WRITE WORK FILE statements. *Note: This may produce a lot of impacts. If you wish to remove this search from Natural Engineer then please modify the impact search criteria by de-selecting the option 'Priority of PRINT/WORK FILE statements' from the MVSNAT22TO31 Preferences screen.*
NEE Modification Category	3s SAG18
NEE Modification	Manual.

18. User Exit Modules copied from SYSEXT

Description	In general, the user exits (USR****N) located on FUSER (Natural 2.2) have to be replaced with the corresponding module from library SYSEXT on the FNAT of Natural 3.1.
NEE Impact	Natural Engineer will identify all USR..... (CALLNAT) references.
NEE Modification Type	3t SAG19
NEE Modification	Manual.

19. Internal Handling of AD=O

Description	With Natural 3.1, the internal handling of AD=O has changed. A CALLNAT/PERFORM parameter marked with AD=O is no longer passed to the subprogram/subroutine 'by reference' (that is, via its address) but 'by value'.
NEE Impact	Natural Engineer will identify all CALLNAT/PERFORM statements with parameter AD=O Specified.
NEE Modification Type	3u GSL01
NEE Modification	Manual.

20. LESS clause of EJECT Statement

Description	To enhance the clarity of programs and avoid possible ambiguities in the source code, the keyword LESS in Syntax 2 of the EJECT statement is no longer optional, but required. With Natural 2.2, the shortest possible form is: EJECT operand1 With Natural 3.1, it is: EJECT LESS operand1
NEE Impact	Natural Engineer will identify all EJECT statements without the LESS clause.
NEE Modification Type	3v GSL02
NEE Modification	Automatic. The LESS clause will be added to the EJECT statement.

21. ESCAPE TOP within AT START OF DATA ESCAPE TOP and ESCAPE BOTTOM not allowed in ON ERROR blocks

Description	In Natural 3.1, you are no longer allowed to place an ESCAPE TOP statement within an AT START OF DATA statement block. It is also not permitted to place either an ESCAPE TOP or an ESCAPE BOTTOM within an ON ERROR BLOCK.
NEE Impact	Natural Engineer will identify all ESCAPE TOP statements in AT START OF DATA Blocks. Also, ESCAPE TOP and ESCAPE BOTTOM statements within ON ERROR blocks are identified.
NEE Modification Type	3w GSL03
NEE Modification	Manual.

22. Decimal Digits of Constant Values

Description	If the constant value specified after CONSTANT or INIT has more digits after the decimal point than the corresponding field, this does not lead to an error with Natural 2.2. With Natural 3.1, such inconsistency leads to error NAT0094 at compilation.
NEE Impact	Natural Engineer will identify all statements where the constant/initial value has more digits after the decimal point than the corresponding field.
NEE Modification Type	3x GSL04
NEE Modification	Manual.

23. TOP and LESS clauses of NEWPAGE statement

Description	To enhance the clarity of programs and avoid possible ambiguities in the source code, the keywords TOP and LESS OF the NEWPAGE statement are no longer optional, but required. With Natural 2.2, the shortest possible forms are: NEWPAGE EVEN NEWPAGE operand1 With Natural 3.1, they are: NEWPAGE EVEN TOP NEWPAGE LESS operand1
NEE Impact	Natural Engineer will identify all NEWPAGE statements without a TOP or LESS clause.
NEE Modification Type	3y GSL05
NEE Modification	Automatic. The TOP or LESS clause will get inserted into the NEWPAGE statement.

24. LS parameter and PRINT statement

Description	It is no longer possible to specify the LS parameter with the PRINT statement (as it has no effect anyway). With Natural 2.2, this does not lead to an error. With Natural 3.1, it leads to error NAT0934.
NEE Impact	Natural Engineer will identify all PRINT statements with the LS parameter specified.
NEE Modification Type	3z GSL06
NEE Modification	Automatic. The LS parameter will be removed from the PRINT statement.

25. MOVE RIGHT JUSTIFIED where target field is shorter than source field

Description	If the target field in a MOVE RIGHT JUSTIFIED statement is smaller than the sending field length, the resulting #value in the target field is truncated from the start of the sending field data.
	Example :
	`01 #A(A10) INIT <'ABCDEHIJKL'>`
	`01 #B(A05)`
	`*`
	`MOVE RIGHT JUSTIFIED #A TO #B`
	`*`
	Result under Natural 2.2, #B = ABCDE
	Result under Natural 3.1, #B = HIJKL

NEE Impact	Natural Engineer will identify all MOVE RIGHT JUSTIFIED statements where the target field is shorter than the source.
NEE Modification Type	3l GSL07
NEE Modification	Manual.

26. BEFORE BREAK within IF condition

Description	At compile time, it is no longer possible to code a BEFORE BREAK statement within an IF condition. This syntax compiles under OS390 for Natural 2.2, 2.3 and 3.1. However, there is a ZAP (NA44082) to make Natural 3.1.4 compatible with PC Natural 4.1.2. Compilation error NAT0309 – Invalid positioning of AT BREAK/END condition, will be returned during compilation.
NEE Impact	Natural Engineer will identify all BEFORE BREAK statements within an IF condition.
NEE Modification Type	31 GSL08
NEE Modification	Manual.

27. Invalid settings for SUBSTRING

Description	Since Natural 3.1, Natural checks at compile time that the SUBSTRING options are valid. The value of the offset plus the length of the sub-string must be less than or equal to string length. If this is not the case, then compilation error 'NAT0471 Invalid operands in SUBSTRING option' will be returned during compilation. Under Natural 2.2, the user would receive a runtime error.
NEE Impact	Natural Engineer will identify where the offset plus the length used in a SUBSTRING clause, exceed the length of the field. This is applicable to EXAMINE, MOVE, EXAMINE TRANSLATE, COMPRESS, COMPUTE, ASSIGN and SEPARATE statements. *Note: If the offset is a variable, no checking is carried out. Also, if the length is a variable, then the check is the offset against the field length.*
NEE Modification Type	32 GSL09
NEE Modification	Manual

28. MOVE BY NAME with numeric redefinition

Description	If a MOVE BY NAME statement references fields which have been redefined from alpha to numeric and both the source and target fields are the same length, then you will receive different results under Natural 3.1.
	Example:
	`DEFINE DATA LOCAL`
	`01 #GROUP1`
	`  02 #ALPHA (A10)`
	`  02 REDEFINE #ALPHA`
	`    03 #ALPHA1 (A1)`
	`    03 #NUMERIC (N9)`
	`01 #GROUP2`
	`  02 #ALPHA (A10)`
	`  02 REDEFINE #ALPHA`
	`    03 #ALPHA1 (A1)`
	`    03 #NUMERIC (N9)`
	`END-DEFINE`
	`MOVE BY NAME #GROUP1 TO #GROUP2`
	`WRITE 'RESULT:' #GROUP1 #GROUP2`
	`END`
	Result under Natural 2.2, RESULT:
	Result under Natural 3.1, RESULT: 000000000
	(Note: the numeric redefinition now contains zeros.)

NEE Impact	Natural Engineer will identify any MOVE BY NAME statements if the source and target fields are: ▪ Both part of a redefine of an alpha field. ▪ Both have a format of 'N' (numeric). ▪ Both have the same length.
NEE Modification Type	33 GSL10
NEE Modification	Manual

PORTING

The combination search keyword PORTING is used to identify any statements that may affect an application being migrated to alternate platforms.

This search keyword uses a sub-set of criteria, which can be refined by selecting/ deselecting the available options.

No automatic modification is available for this search keyword.

Specifying PORTING

Select Impact Type PORTING from the Criteria Details tab screen.

Note: Only one set of PORTING criteria is allowed per Impact Version.

The following Figure 3-20 illustrates the Criteria Detail tab screen for combination keyword PORTING.

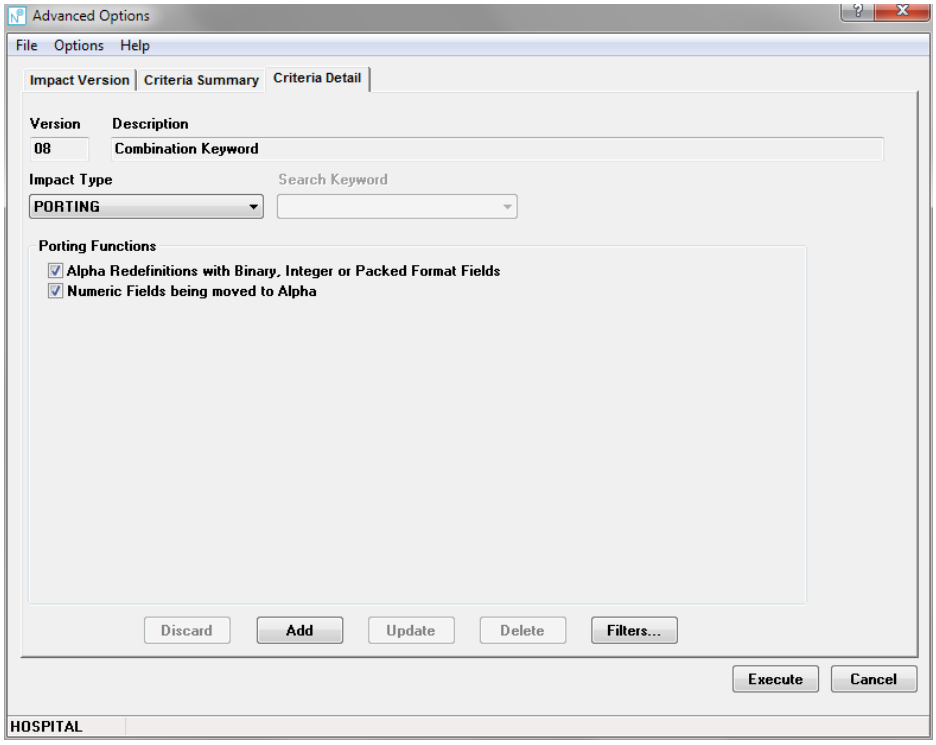

Figure 3-20 Criteria Detail tab screen for combination keyword PORTING

The following only describes the Criteria Detail tab screen options relevant to the combination search keyword PORTING.

Note: For more information on the Criteria Details tab screen refer to the section Criteria Detail tab screen in Chapter 1.

SCREEN ITEMS	DESCRIPTION
Porting Functions	Each PORTING option is listed A tick in the check box next to each option indicates that the option will be checked for. If a check box is blank then that option will not be referenced during Impact execution. Available selections are: **Alpha Redefinitions with Binary, Integer or Packed Format Fields** Impact will look for any alpha variables that have redefinition variables defined using format of Binary, Integer or Packed. No automatic modification is available. **Numeric Fields being moved to Alpha** Impact will look for any numeric variables that are being moved to alpha variables. No automatic modification is available.

Setting Object Filters

The PORTING criteria can be further refined by only applying the criteria to specific object types and/or object names during Impact execution.

This can be set by using the 'Filters…' button on the Criteria Details tab screen to invoke the Filters window.

Note: For more information on Filters refer to the Filters Window section.

REFACTORING

The combination search keyword REFACTORING can be used to locate certain key situations that may need to be addressed when reviewing the organization of a Natural application.

This search keyword uses a sub-set of criteria, which can be refined by selecting/de-selecting the available options.

Once impacted, then for some key situations it may be possible to modify the code using the Modification process.

Specifying Refactoring

Select Impact Type REFACTORING from the Criteria Details tab screen.

Note: Only one set of REFACTORING criteria are allowed per Impact Version.

The following Figure 3-21 illustrates the Criteria Detail tab screen for combination keyword REFACTORING.

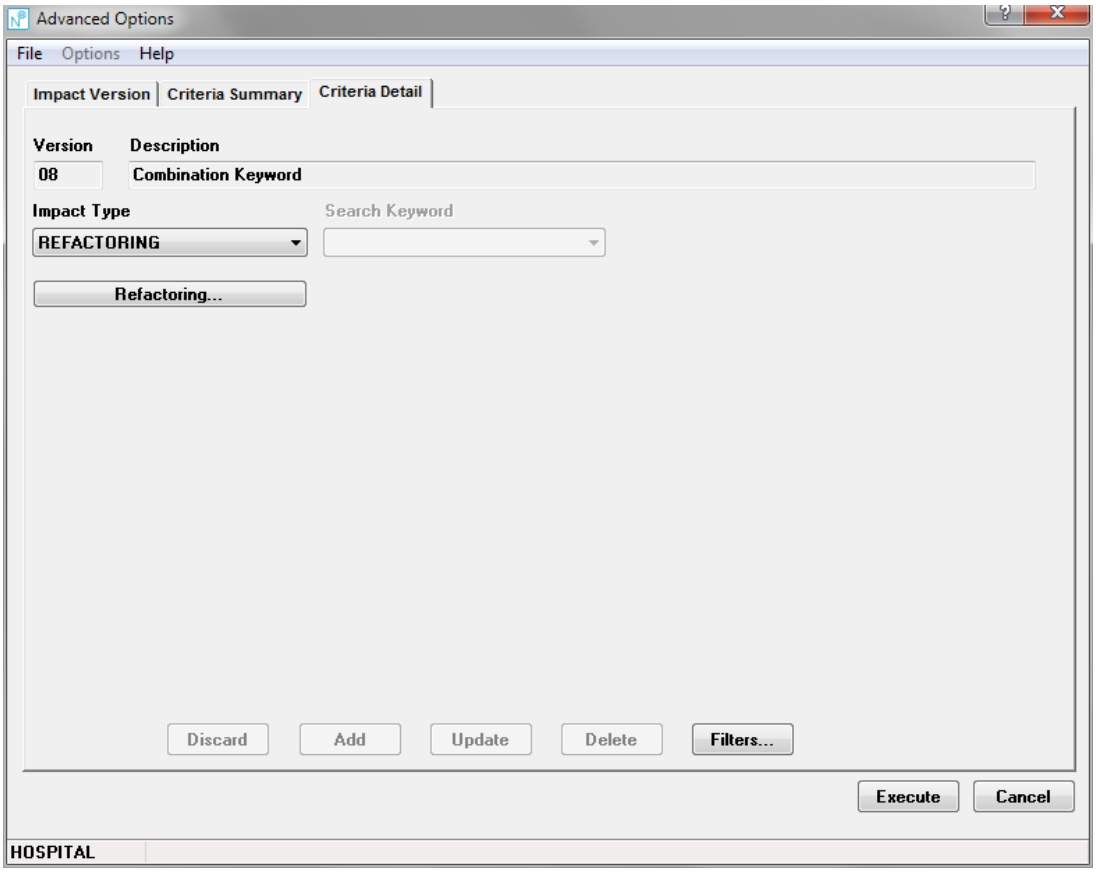

Figure 3-21 Criteria Detail tab screen for combination keyword REFACTORING

Note: For more information on the Criteria Details tab screen refer to the section Criteria Detail tab screen in Chapter 1.

Refactoring Preferences Window

The Refactoring Preferences screen allows you to control which Refactoring options are actually checked for during Impact execution.

The Refactoring Preferences are accessed from the Criteria Detail tab screen by using the 'Refactoring…' button. Alternatively, existing Refactoring criteria can be selected from the Criteria Summary tab screen by double clicking with the left mouse button.

The following Figure 3-22 illustrates the Refactoring Preferences screen.

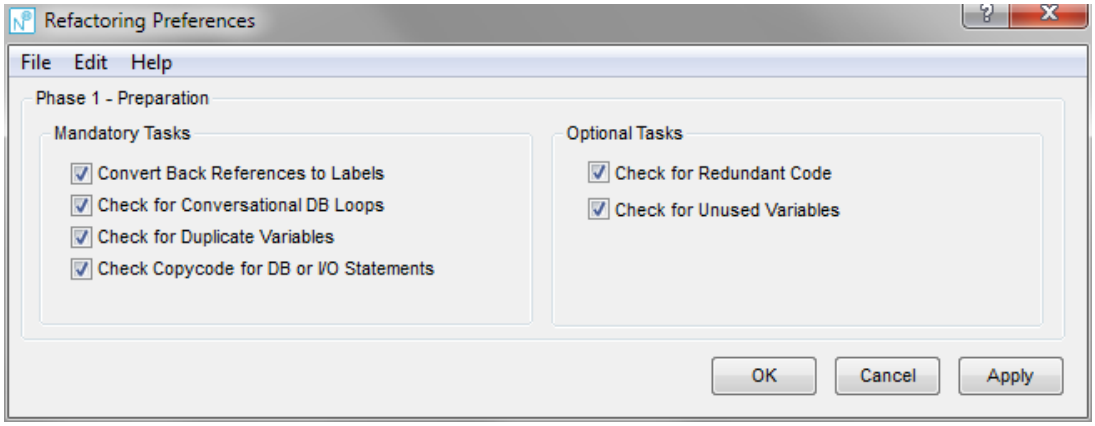

Figure 3-22 Refactoring Preferences screen

MENU ITEMS	OPTIONS	DESCRIPTION
File	**Exit**	Exit the Refactoring Preferences screen and return back to the Criteria Details tab screen.
Edit	**Select All**	Select all the Refactoring Preferences.
	Deselect All	Deselect all the Refactoring Preferences.
Help	Invoke the Refactoring Preferences help.	

SCREEN ITEMS	DESCRIPTION
Refactoring Preferences	Each Refactoring option is listed. A tick in the check box next to each option indicates that the option will be checked for. If a check box is blank then that option will not be referenced during Impact execution. *Note: At least one Refactoring option must be selected.* Available selections are: ▪ **Convert Back References to Labels** ▪ **Check for Conversational DB Loops** ▪ **Check for Duplicate Variables** ▪ **Check Copycode for DB or I/O Statements** ▪ **Check for Redundant Code** ▪ **Check for Unused Variables** *Note: For more information on each of these options, refer to the section Refactoring Preferences Explained.*

Note: The group titles 'Mandatory Tasks' and 'Optional Tasks' are used as guidance when using the Refactoring Workflow function found under the Utilities menu.

BUTTON NAME	DESCRIPTION
OK	Save changes and close the current screen.
Cancel	Cancel the Refactoring Preferences process and return back to the Criteria Details tab screen.
Apply	Save changes and retain the current screen. *Note: This button is only enabled if any changes have been made.*

Refactoring Preferences Explained

The Refactoring Preferences help identify certain key situations within Natural objects that may need to be addressed when reviewing the organization of a Natural application.

The available Refactoring Preferences cater for the following key situations:

- Database and non-database loops with no labels and any numeric back references.
- Conversational database loops. Database loops that contain any screen I/O statements.
- Any duplicate variable names within group structures.
- Copycodes that contain any database access or screen I/O statements.
- Redundant code. Code that is no longer executable within an object.
- Unused variables.

Some of these situations can be modified using the Modification process, others will only have the Impact process available and manual modifications may need to be applied to each of these.

Convert Back References to Labels

Impact	Impact will look for any database and non-database loops that do not have labels, and any numeric back references which can be converted to labels.
	
	Example:
	`0130 /*`
	`0140 READ VEHICLES`
	`0150   DISPLAY MAKE(0140) MODEL(0140) COLOR(0140)`
	`0160 END-READ`
	`::::`
	`0220 REPEAT UNTIL #INDEX GT 12`
	`0230   MULTIPLY #INDEX BY #MULTIPLY-2 GIVING #ANSWER`
	`0240   ADD 1 TO #INDEX`
	`0250 END-REPEAT`
	`::::`
	
	Statement 0140 is impacted for a database loop with no label.
	
	Statement 0150 is impacted three times, once for each numeric back reference.
	
	Statement 0220 is impacted for a non-database loop with no label.

Impact Type	82 (Missing database labels.) 83 (Missing non-database labels.) 84 (Convert numeric back references to labels.)

Modification	Automatic.
	
	Modification will add labels for database and non-database loops without labels, and change any numeric back references to labels.
	
	Example:
	`0130 /*`
	`0140 READ-0140. /* NEE MODIFIED`
	`0150 READ VEHICLES`
	`0160 /*   DISPLAY MAKE(0140) MODEL(0140) COLOR(0140) /* NEE OLD CODE`
	`0170 DISPLAY MAKE (READ-0140.) MODEL (READ-0140.) COLOR (READ-0140.) /* NEE MODIFIED`
	`0180 END-READ`
	`::::`
	`0240 REP-0220. /* NEE MODIFIED`
	`0250 REPEAT UNTIL #INDEX GT 12`
	`0260   MULTIPLY #INDEX BY #MULTIPLY-2 GIVING #ANSWER`
	`0270   ADD 1 TO #INDEX`
	`0280 END-REPEAT`
	`::::`

Check for Conversational DB Loops

Impact	Impact will look for any screen I/O statements within database processing loops. If the screen I/O statements are located within an inline subroutine, then the PERFORM statement for that inline subroutine will also be impacted.
	Database processing loops are identified by the Natural keywords:
	BROWSE, FIND, HISTOGRAM and READ.
	Screen I/O statements are identified by the Natural keywords:
	DISPLAY, INPUT, PRINT, REINPUT and WRITE.
	Example:
	`::::`
	`0100 READ VEHICLES`
	`0110   DISPLAY MAKE MODEL`
	`0120 END-READ`
	`0130 /*`
	`0140 FIND PERSONNEL WITH NAME = 'ADKINSON'`
	`0150   PERFORM ##WRITE-DETAILS`
	`0160   PERFORM ##PRINT-DETAILS`
	`0170 END-FIND`
	`0180 /*`
	`0190 DEFINE SUBROUTINE ##WRITE-DETAILS`
	`0200 WRITE PERSONNEL-NUMBER NAME`
	`0210 END-SUBROUTINE`
	`0220 /*`
	`0230 DEFINE SUBROUTINE ##PRINT-DETAILS`
	`0240 PRINT PERSONNEL-NUMBER NAME`
	`0250 END-SUBROUTINE`
	`::::`

Impact Type	85
Modification	Manual.

Check for Duplicate Variables

Impact	Impact will look for any duplicate variable names within group structures.
	Note: Impact will not check Data Areas, Copycodes or Dialogs.
	Example:
	`::::`
	`0040 DEFINE DATA LOCAL`
	`0050 01 #GROUP-A`
	`0060 02 #B (A10)`
	`0070 /*`
	`0080 01 #GROUP-D`
	`0090 02 #B (A5)`
	`0100 02 #C (A5)`
	`0110 /*`
	`0120 END-DEFINE`
	`0130 /*`
	`0140 MOVE 'ABCDEFGHIJ' TO #GROUP-A.#B`
	`0150 /*`
	`0160 MOVE '12345' TO #GROUP-D.#B`
	`0170 MOVE '67890' TO #GROUP-D.#C`
	`0180 /*`
	`0190 WRITE #GROUP-A #GROUP-D`
	`::::`
	Impact will find the field #B at line numbers 0060 and 0090.

Impact Type	87
Modification	Manual.

Check Copycode for DB or I/O Statements

Impact	Impact will look for any database and/or screen I/O stetments that are present within a copycode object. The copycode object must be in use within another object, for example a Program. Unused copycode objects are not impacted.
	
	Example:
	
	Program using copycode object CCODEDB:
	`::::`
	`0040 DEFINE DATA LOCAL`
	`0050 /*`
	`0060 01 CARS VIEW VEHICLES`
	`0070  02 COLOUR`
	`0080 /*`
	`0090 01 #COLOUR-COUNT (N7)`
	`0100 /*`
	`0110 END-DEFINE`
	`0120 /*`
	`0130 INCLUDE CCODEDB`
	`> /* ----------`
	`> /* COPYCODE CONTAINING db STATEMENT`
	`> /* ----------`
	`> READ CARS`
	`>  IF COLOUR NE 'RED'`
	`>  ESCAPE TOP`
	`>  END-IF`
	`>  ADD 1 TO #COLOUR-COUNT`
	`> END-READ`
	`0140 /*`
	`0150 IF #COLOUR-COUNT EQ 0`
	`0160  WRITE 'NO MATCHING DATA FOUND'`
	`0170 ELSE`
	`0180  WRITE 'MANAGED TO FIND' #COLOUR-COUNT 'RECORDS'`
	`0190 END-IF`
	`::::`
	
	Impact will find the READ CARS statement within the copycode object CCODEDB.
	
	Note: The Impact is marked within the object using the copycode. The copycode object will not be impacted.

Impact Type	88 (I/O statements in copycode.) 89 (Database statements in copycode.)
Modification	Manual.

Check for Redundant Code

Impact	Impact will look for any unused source code lines within programming objects, across a whole application.
	
	Any source code that is driven by event rather than position is ignored. For example AT BREAK, AT END OF PAGE, WRITE TITLE.
	
	Unused source code within internal subroutines is included, but for external subroutines the Unused Objects report should be referenced.
	
	Note: Only Structured Mode objects will be impacted. For Reporting Mode objects, it is recommended that they are converted to Structured Mode first using the Mode Conversion function, and then impacted.
	
	Example:
	`0290 IF #CHOICE = 'X'`
	`0300  MOVE 'COPTZP4S' TO #FIELD-A`
	`0310  RESET #FIELD-N`
	`0320  FETCH 'COPTZP4S' #FIELD-A #FIELD-N`
	`0330  RESET #FIELD-N`
	`0340  MOVE 'XYZ'TO #FIELD-A`
	`0350 END-IF`

Impact Type	85
Modification	Manual.

Check for Unused Variables

Impact	Impact will look for any unused variables. These can be user-defined variables or logical view variables.
	
	Example:
	`0010 #A (A5)`
	`0020 #B (A5)`
	`0030 #C (A5)`
	`0040 MOVE 'ABCDE' TO #A`
	`0050 MOVE #A TO #B`
	`0060 WRITE #B`
	`0070 END`

Impact Type	86

Modification	Automatic.
	
	Modification will comment out any unused variables.
	
	Example:
	`0010 #A (A5)`
	`0020 #B (A5)`
	`0030 /* #C (A5) /* UNUSED DATA ITEM /* NEE OLD CODE`
	`0040 /* UNUSED */ #C (A5) /* UNUSED DATA ITEM /* NEE MODIFIED`
	`0050 MOVE 'ABCDE' TO #A`
	`0060 MOVE #A TO #B`
	`0070 WRITE #B`
	`0080 END`
	
	Note: LDA and PDA objects with unused data items will not be modified.

Setting Object Filters

The REFACTORING criteria can be further refined by only applying the criteria to specific object types and/or object names during Impact execution.

This can be set by using the 'Filters…' button on the Criteria Details tab screen to invoke the Filters window.

Note: For more information on Filters refer to the Filters Window section.

SYSTEM FUNCTIONS

The Combination Search Keyword SYSTEM FUNCTIONS can be used to locate Natural System Functions used within a Natural application.

This search keyword uses a sub-set of criteria, which can be refined by selecting/de-selecting the available types of System Functions using the Criteria Detail screen.

No automatic modification is available for this search keyword.

Specifying SYSTEM FUNCTIONS

Select Impact Type SYSTEM FUNCTIONS from the Criteria Details tab screen.

The following Figure 3-23 illustrates the Criteria Detail tab screen for combination keyword SYSTEM FUNCTIONS.

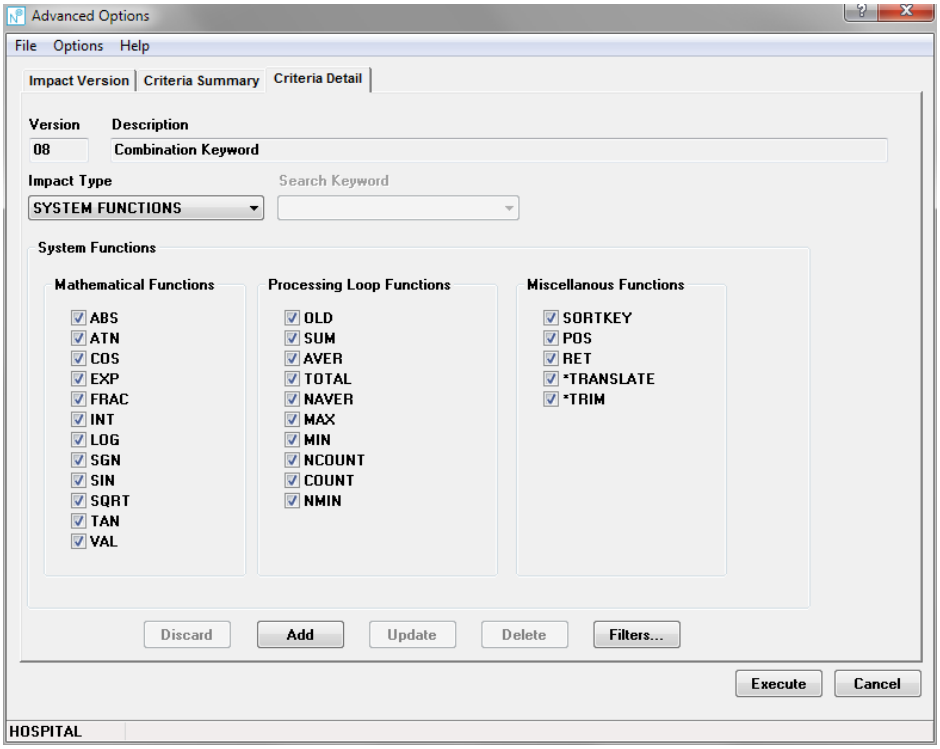

Figure 3-23 Specifying SYSTEM FUNCTIONS Impact Type

The following only describes the Criteria Detail tab screen options relevant to the combination search keyword SYSTEM FUNCTIONS.

Note: For more information on the Criteria Details tab screen refer to the section Criteria Detail tab screen in Chapter 1.

SCREEN ITEMS	DESCRIPTION
System Functions	Each SYSTEM FUNCTIONS option is listed A tick in the check box next to each option indicates that the option will be checked for. If a check box is blank then that option will not be referenced during Impact execution.

The System Functions available are:

Category	System Function	Impact Type
Mathematical Functions	ABS	0A
	ATN	0B
	COS	0C
	EXP	0D
	FRAC	0E
	INT	0F
	LOG	0G
	SGN	0H
	SIN	0I
	SQRT	0J
	TAN	0K
	VAL	0L
Processing Loop Functions	OLD	0M
	SUM	0N

	AVER	0O
	TOTAL	0P
	NAVER	0Q
	MAX	0R
	MIN	0S
	NCOUNT	0T
	COUNT	0U
	NMIN	0V
Miscellaneous Functions	SORTKEY	0W
	POS	0X
	RET	0Y
	*TRANSLATE	0Z
	*TRIM	01

Setting Object Filters

The SYSTEM FUNCTIONS criteria can be further refined by only applying the criteria to specific object types and/or object names during Impact execution.

This can be set by using the 'Filters…' button on the Criteria Details tab screen to invoke the Filters window.

Note: For more information on Filters refer to the Filters Window section.

INDEX