

Introduction to Natural Construct

This section introduces you to Natural Construct. It describes how to access the subsystems, use PF-keys, and access the online help. It includes sections on translating to upper case, handling messages, storing saved modules, and using direct commands.

This section covers the following topics:

- What is Natural Construct?
 - Access Natural Construct
 - Use Standard PF-Keys
 - Access Online Help
 - Convert Text to Upper Case
 - Maintain Messages for Generated Programs
 - Store Saved Modules
 - Use Direct Commands
-

What is Natural Construct?

Natural Construct is a set of tools for application developers. Created for Software AG's Natural/Predict environment, it helps application developers achieve higher productivity goals than are obtainable using Natural and Predict alone. At the same time, Natural Construct helps standardize and control the application development process.

Natural Construct provides a series of models you can use to create different Natural modules (objects). The following table lists the advantages of using Natural Construct-generated modules over modules created in Natural alone:

Advantage	Benefits
Standardization and quality	Create a consistent user interface and code structure.
Reusage	Once your model is tested and debugged, it can be used by multiple users, problem free. Models help share your Natural expertise, making optimal use of available talent.
Increase productivity	These benefits include: <ul style="list-style-type: none">● Reduce design considerations● Speed up implementation● Reduce testing requirements
Minimize errors	Avoid errors that are introduced by program cloning.

Natural Construct Subsystems

Natural Construct is comprised of the following subsystems:

Subsystem	Description
Administration	Used by the Natural Construct administrator to define custom models and maintain the models Natural Construct uses to generate programs. The Administration subsystem is described in detail in this documentation.
Generation	<p>Used by the developer to define specifications for the Natural Construct models and generate the following modules:</p> <ul style="list-style-type: none"> ● programs ● subprograms ● help routines ● subroutines ● copycode ● maps ● parameter data areas ● local data areas ● global data areas ● Predict program descriptions ● code blocks ● JCL text (mainframe) ● User exit code <p>For information about this subsystem, refer to <i>Natural Construct Generation</i>.</p>
Help Text	Used by documenters or developers to create and maintain help text at the map and/or input field level. For information about this subsystem, refer to <i>Natural Construct Help Text</i> .

Access Natural Construct

You can access the Administration subsystem in standard or translation mode. Translation mode allows you to create multilingual specification panels for developers, as well as dynamically maintain the panel components.

This section describes how to access each Natural Construct subsystem, how to access the Administration subsystem in standard and translation mode, and how to access the generation facilities from a steplib with Natural Security installed.

Note:

Always terminate Natural Construct by pressing the quit PF-key or entering a period (.) in the input field on the main menu. This method ensures proper cleanup of the environment.

Natural Construct Libraries

While other Software AG products can be accessed from other libraries, they run exclusively in their own product library (SYSSEC, SYSPAC, SYSDIC, for example). Natural Construct does not run exclusively out of its product library, SYSCST. It also must also run out of the application libraries in the FUSER file.

Typically, Natural Construct developers access Natural Construct using the NCSTG or NCSTH command from any library. These commands invoke modules in the SYSLIB and SYSLIBS libraries. The CD-HELP* modules in the SYSLIBS library provide online help for Natural Construct screens.

Natural Construct also allows administrators and modelers to customize the standard Natural Construct models. These users access Natural Construct using the CSTG or CSTH command from the SYSCST library. All changes are confined to this library, which allows administrators to test customizations without affecting developers and their applications.

Since administrators can customize help modules like CD-HELP*, copies of these modules are also stored in the SYSCST library. Any changes to these modules do not affect developers because they use the CD-HELP* routines in the SYSLIBS library. Typically, modelers access Natural Construct using the CSTG or CSTH command.

Once a Natural Construct modeler creates or maintains a model, all customized modules must be copied to the appropriate library.

- If the changes apply to the development environment, copy the modules to the SYSLIB or SYSLIBS library.
- If the changes apply to the runtime environment for a Natural Construct-generated application, copy the modules to a library within the application steplib chain (for example, SYSTEM on the FNAT). It is also common to see CD-HELP* routines in the SYSTEM FNAT library.

Copies of Natural Construct are stored in the following libraries:

graphics/cst-libraries.png

Each library is available to different users and contains different subsystems. The libraries are:

- SYSLIBS Library
- SYSTEM (FNAT) Library
- SYSCST Library
- SYSCSTX Library
- SYSCSTDE, SYSCSTD2, SYSCSTDV, and SYSCSTDS Libraries
- USERLIB Library
- Execute Generation Facilities from a Steplib with Natural Security Installed

SYSLIBS Library

The SYSLIBS library contains modules used by Natural Construct. The following table indicates who can use the library, the subsystems it contains, and the command entered at the Next prompt to invoke each subsystem:

Authorized Users	Subsystems	Command to Invoke Each Subsystem
All users	Generation	"ncstg "
	Help Text	"ncsth"

SYSTEM (FNAT) Library

The SYSTEM library contains modules used by Natural Construct-generated applications. The following table indicates who can use the library, the subsystems it contains, and the command entered at the Next prompt to invoke each subsystem:

Authorized Users	Subsystems	Command to Invoke Each Subsystem
All users	Generation	"ncstg "
	Help Text	"ncsth"

SYSCST Library

The SYSCST library is used to modify the supplied models or create new ones. The following table indicates who can use the library, the subsystems it contains, and the command entered at the Next prompt to invoke each subsystem:

Authorized Users	Subsystems	Command to Invoke Each Subsystem
Administrators	Administration	"menu" (standard mode)
		"menut" (translation mode)
	Generation	"ncstg"
	Help Text	"ncsth"

SYSCSTX Library

The SYSCSTX library contains sample routines provided with Natural Construct. The routines can be used as is or modified as desired.

- To customize a routine, create a copy of the routine in the SYSCST library.
- To make the routine active, move the object code to the SYSLIBS library.

SYSCSTDE, SYSCSTD2, SYSCSTDV, and SYSCSTDS Libraries

These libraries contain the Natural Construct demo system for different systems. To invoke the demo system, enter "menu" at the Next prompt in the applicable library.

***USERLIB* Library**

This library is created by Natural Construct users.

Execute Generation Facilities from a Steplib with Natural Security Installed

With Natural Security installed, you can access the Natural Construct generation facilities from a steplib. This allows you to override the supplied model subprograms at a higher level steplib without disturbing the modules supplied by Natural Construct.

For example, you can define the following steplibs in your development library:

- *CSTMODS* (your modification library)
- SYSCST
- SYSLIBS
- SYSTEM

Using this configuration, you can easily change your standards without disturbing the supplied modules. To modify any modules in the SYSCST or SYSTEM library that are affected by changes, copy them into the *CSTMODS* library.

Note:

You can also define multiple modification libraries in the steplib chain (to reflect corporate versus application standards).

When accessing Natural Construct from a steplib, the highest level steplib should contain a replacement for the NCSTG program. For example:

```
FETCH 'CSTG'  
END
```

Otherwise, the NCSTG program invokes the version of Natural Construct stored in the SYSLIBS library.

Note:

If Natural Security is not installed, refer to USR1025P in the SYSEXT library for an example of how to set up your steplibs.

Use Standard PF-Keys

Throughout the Natural Construct system, certain PF-keys have standard functions (pressing the PF1 key invokes online help, for example). The PF-key lines, which are typically located at the bottom of panels, display the PF-key functions for that panel.

Notes:

1. PF-keys 13 to 24 are equivalent to PF-keys 1 to 12, respectively. However, only PF1 to PF12 are displayed.
2. You can change the function and/or description associated with each key (for more information, see Access the Administration Main Menu). Within this documentation, the default values are used.

The standard PF-keys and functions are:

PF-Key	Name	Function
PF1	help	<p>Displays help for a particular panel or field.</p> <ul style="list-style-type: none"> ● When the cursor is in a field followed by an asterisk (*), displays a window from which you can select a valid value for the field. For information, see Field-Level Help. ● When the cursor is in a field not followed by an asterisk (*), displays help information for that field. For information, see Panel-Level Help. ● When the cursor is anywhere on the panel except a field, displays help for the entire panel <p>Note: An asterisk is the default help indicator for Natural Construct. The help indicator for your organization may be different.</p>
PF2	retrn	Displays the previous panel. Pressing PF2 is equivalent to entering a period (.) in the Function field on a menu.
PF3	quit	Terminates the Natural Construct session. In most cases, a confirmation window is displayed when you press PF3. Press PF3 again to complete the termination process.
PF7	bkwrđ	Scrolls backward (up) through data.
PF8	frwrđ	Scrolls forward (down) through data.
PF10	left	Displays the panel to the left of the current panel. If you are currently on the first panel in a series of panels, pressing PF10 displays the last panel in the series.
PF11	right	Displays the panel to the right of the current panel. If you are currently on the last panel in a series of panels, pressing PF11 displays the first panel in the series.
PF12	main	Displays the Natural Construct Administration main menu.

Help and Return Codes on Menus

On each Natural Construct menu, you are given the options “?” and “.” as valid menu codes. Typing a question mark (?) in the Function field and pressing Enter displays help for that panel. It is equivalent to pressing PF1 (help). Typing a period (.) and pressing Enter terminates the current program and returns you to the previous menu. It is equivalent to pressing PF2 (retrn).

Access Online Help

Natural Construct provides extensive online help. You can display both general help information for each panel (panel-level help) or help for a specific field (field-level help). This section covers the following topics:

- Panel-Level Help
- Field-Level Help

Panel-Level Help

While you are using Natural Construct, you can display help information about the current panel by moving the cursor anywhere on the panel (except an input field) and pressing PF1 (help).

Note:

If the cursor is positioned in an input field when you request help, Natural Construct displays help information for that field. For more information, see Field-Level Help.

The following example shows the panel-level help for the Administration main menu:

```

                                Panel Help
                        Administration Main Menu

This menu lists the functions available within the Administration
subsystem; you use these functions to perform various administrative
duties within Construct.

For translation mode details, see:
  <<Administration Main Menu>>

For example, you use these functions to:
- maintain the Construct control record defaults, such as the
  default PF-key settings and dynamic attribute characters
- maintain the Construct components, such as the code frames and
  subprograms used by each model
- invoke the supplied utilities to compare models or code frames
- use the supplied driver programs to invoke many of the internal
  Construct subprograms
Page ... : 1 / 2
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF1
frwr help  retrn quit                                bkwr frwr
Help for: P/CS/CSDMMN0/1

```

- To scroll forward through the pages of help text, either enter a number in the Page field, press PF8 (frwr), or press Enter.

- To scroll backward, either enter a number in the Page field or press PF7 (bkwrđ).
- To return to the main screen, press PF2 (retrn).
- To display help about how to use online help, press PF1 (help) in any help window.
- To display information about a topic enclosed within angle brackets (<< >>), move the cursor over the name and press Enter. A window is displayed, containing help information about the selected topic.

Field-Level Help

Natural Construct has two types of field-level help: passive and active. Passive field-level help displays a description of a field on a panel. Active field-level help displays a selection window containing the valid values for a field. If active help is available, the field is followed by an asterisk (*).

Passive

Start of instruction set **To display passive field-level help:**

1. Move the cursor to any field that is not followed by an asterisk (*).
2. Press PF1 (help).

Note:

You can also type a question mark (?) in the first-character position of any field that is not followed by an asterisk (*) and press Enter (mainframe).

Active

Start of instruction set **To display active field-level help:**

1. Move the cursor to a field that is followed by an *.
2. Press PF1 (help).

Note:

You can also type a question mark (?) in the first-character position of any field that is followed by an asterisk (*) and press Enter (mainframe).

The following example shows the active help window for the Relationship name field:

CPHRL	Natural Construct	CPHRL0
Aug 20	Select Predict Relationship	1 of 1
Relationship		Relationship type
-----		-----
NCST-CUSTOMER-ORDER-HEADER		Natural Construct
NCST-LINE-HAS-DISTRIBUTION		Natural Construct
NCST-ORDER-HAS-LINES		Natural Construct
NCST-POLICY-COVERS-VEHICLES		Natural Construct
NCST-POLICY-HAS-INQUIRIES		Natural Construct
NCST-POLICY-IS-FOR-CUSTOMER		Natural Construct
NCST-PRODUCT-ORDER-LINES		Natural Construct
NCST-VEHICLES-HAVE-COVERAGES		Natural Construct
NCST-VEHICLES-MUST-EXIST		Natural Construct
NCST-WAREHOUSE-CUSTOMER		Natural Construct
Relationship		
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9-		
help retrn		bkwrđ frwrđ
Position cursor or enter screen value to select		

Start of instruction set **To select a value from the help window:**

1. Move the cursor to the line containing the value.
2. Press Enter.

You are returned to the original panel and the selected value is displayed in the field for which you requested help.

Convert Text to Upper Case

Natural Construct automatically performs the commands to convert text from lower or mixed case to upper case where appropriate. Headings are displayed exactly as entered (lower or upper case), but if certain specifications must be in upper case, Natural Construct converts them. When Natural Construct ends, the case setting is restored to the default value.

Note:

If you are a mainframe user, specify your teleprocessing (TP) monitor's command for lower case. In Com-Plote, for example, issue the LOW command.

Maintain Messages for Generated Programs

Natural Construct supports multilingual messages for your generated programs. If you use message numbers, the message text for the specified language is retrieved at execution time. If you use message text, the text for the specified language is inserted into the program at generation time.

- Messages 8000 to 8200 are stored in the SYSTEM and SYSCST libraries
- Messages 8300 to 8500 are stored in the CSTAPPL library
- Messages 1 to 9999 (error message text) are stored in the CSTMSG library

- Messages 1 to 9999 (screen prompt text) are stored in the CSTLDA library
- Messages 1 to 9999 (text for Actions) are stored in the CSTACT library
- Messages 1 to 9999 (text for PF-keys) are stored in the CSTPFK library

You can change or add to these messages using the SYSERR utility. For all REINPUT and INPUT message numbers, you can also use the SYSERR utility to add other languages. Generation and CDUTRANS messages are stored in the CSTAPPL library. For information about defining references, see Define SYSERR References.

Note:

Natural Construct sounds an alarm and displays warning messages for errors. Ensure the alarm on your terminal is set to an audible volume.

Store Saved Modules

Any module generated by the default generators and saved by Natural Construct is stored as a Natural structured mode object in the current library. You can edit this module as you would any structured mode Natural object.

Use Direct Commands

To navigate within the Administration subsystem, you can enter codes on menus, press PF-keys, or issue direct commands. Direct commands take you to any function or menu within the subsystem without using intervening menus. They are useful for experienced users who know the menu structure, valid menu codes, and the required parameters at each menu level. The following example shows the Command line:

Command _____

You can string together as many commands as you like. If one of the codes is not valid on the corresponding menu, Natural Construct displays that menu so you can enter a valid code.

The following diagram illustrates a sample direct command:

graphics/sample-direct-command.png

This direct command accesses the Code Frame Menu (menu code F on the Administration main menu) and the Edit Code Frame function (menu code E on the Code Frame menu) and displays the code frame called “FRAME” with the description, “DESCRIPTION”, in the Code Frame editor.

A direct command contains the codes you enter on successive menus. Each direct command must begin with a valid menu code. When entering a direct command, leave a space between menu codes to indicate a new menu or level. To indicate parameters that are at the same level, use a slash (/) to separate them.

When you enter direct commands on the command line for a menu, Natural Construct first determines whether the code is a valid option on that menu. If no code on the current menu matches the first code in the direct command, Natural Construct checks the main menu for a match.

You can also issue direct commands at the Natural Next prompt (Direct command box for Unix). While you are in the SYSCST library, for example, you can enter the following direct command to access the Administration subsystem (MENU) and edit the code frame, FRAME, with the description, DESCRIPTION:

```
MENU F E /FRAME /DESCRIPTION
```