Natural Business Services

Common Model Specifications and Development Tasks

Version 8.2.2

October 2018

ADABAS & NATURAL

This document applies to Natural Business Services Version 8.2.2 and all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: NBS-COMMONSPEC-822-20180930

Table of Contents

Preface

This documentation describes the common specifications for the Natural Construct models and how to perform common development tasks. It is organized under the following headings:

Common Fields on the Standard Parameters Panel	Describes the common fields on the Standard Parameters panel.
Set Up a Password File	Describes how to set up a password file to implement password checking for generated modules.
Change the Default Window Settings	Describes how to change the default window settings for your generated modules, such as the size and position of the window and whether it uses a frame (border). Includes information on testing the new window settings.
Determine Which Condition Codes are Set	Explains how to verify which condition codes have been set for your generated module.

Notes:

1. Although the screen examples used in this documentation are from a mainframe environment, the information applies to all server environments.
2. For information about the additional specification panels for a specific model, see the documentation in which that model is described.

1 About this Documentation

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
`Monospace font`	Identifies service names and locations in the format *`folder.subfolder.service`*, APIs, Java classes, methods, properties.
`Italic`	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
`Monospace font`	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
|	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the | symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at **http://documentation.softwareag.com**. The site requires credentials for Software AG's Product Support site Empower. If you do not have Empower credentials, you must use the TECHcommunity website.

Software AG Empower Product Support Website

If you do not yet have an account for Empower, send an email to empower@softwareag.com with your name, company, and company email address and request an account.

Once you have an account, you can open Support Incidents online via the eService section of Empower at **https://empower.softwareag.com/**.

You can find product information on the Software AG Empower Product Support website at **https://empower.softwareag.com**.

To submit feature/enhancement requests, get information about product availability, and download products, go to **Products**.

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the **Knowledge Center**.

If you have any questions, you can find a local or toll-free number for your country in our Global Support Contact Directory at **https://empower.softwareag.com/public_directory.asp** and give us a call.

Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at **http://techcommunity.softwareag.com**. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

2 Common Fields on the Standard Parameters Panel

The Standard Parameters panel is similar for all program models; it is the first and sometimes the only specification panel. The following example shows the Standard Parameters panel for the Object-Maint-Dialog model:

```
CUOMMA                  Object-Maint-Dialog Program                  CU--MAO
Sep 16                      Standard Parameters                       1 of 4

 Module ............. ________
 System ............. CST82S_____________________________
 Global data area ... CDGDA___ *
 With block ......... ___________________________________

 Title ............. Object Dialog..._________
 Description ........ This program is used to maintain the..._________________
                      ____________________________________________________________
                      ____________________________________________________________
                      ____________________________________________________________

 First header ....... ____________________________________________________
 Second header ...... ____________________________________________________

 Command ............ _
 Message numbers .... _
 Password ........... _

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
right help  retrn quit                                          right main
```

Note: The Standard Parameters panel for other models may not contain all the fields shown.

The fields on the Standard Parameters panel include:

Field	Description
Module	Name of the module (by default, the name specified in the Module name field on the Generation main menu). This is a required field. The module name must be alphanumeric, a maximum of eight characters in length, and cannot contain blanks.
System	Name of the system (by default, the name of the current library). This is a required field. The system name must be alphanumeric, not exceed 32 characters in length, and does not have to be associated with a Natural library ID. (The combination of the module name and system name is used as a key to access help information for the generated module.)
Global data area	Name of the global data area (GDA) used by the generated module. By default, CDGDA is displayed. This is a required field. To allow inter-program communication, generated modules require a small number of global variables. The CDGDA global data area contains the global variables required to test a generated module. Before creating a new application, copy this GDA from the SYSTEM library and rename it to match your naming conventions. Then add any additional global variables your application may require.
With block	Name of the GDA block used by the generated module. This is an optional field. To use a GDA block, you need only specify the lowest level block name; the corresponding path name is determined automatically. For more information about GDA blocks, see the Natural documentation.
Title	Title for the generated module. The title is used internally to identify modules for the List Generated Modules function on the Generation main menu. This is a required field.
Description	Brief description of what the generated module does. This is an optional field and is used internally for documentation purposes.
First header	First header (heading) for the generated module. This header is centered at the top of the generated panel and intensified. This is a required field.
Second header	Second header (heading) for the generated module. This header is centered under the first header and intensified. This is an optional field.
Command	If this field is marked, the generated module supports direct command processing. If this field is blank, the generated module does not support direct commands.
Message numbers	If this field is marked, the generated module uses message numbers rather than message text for all REINPUT and INPUT messages. **Note:** Use the same technique consistently throughout your application, since passing messages between modules using different techniques will not always produce the desired results.

Field	Description
Password	If this field is marked, the generated module performs password checking. If this field is blank, the generated module does not perform password checking. To include password checking, you must set up a password file. For information, see ***Set Up a Password File***.

3 Set Up a Password File

You can specify password checking for many of the generated modules. Natural Construct builds the mechanism for password checking into your module by including the CCPASSW copycode member. Within this copycode, the CDPASSW subprogram is invoked and passed the module and system names.

To include password checking, you must set up a password file. The file is keyed on the module name used to catalog the module and the system name used to generate the module.

The password file can be a view of any file with Natural-Construct-Password as the data definition module name. The view must contain the following fields:

Field	Format
PASSWORD-KEY	A40 (32-character system name, plus an 8-character module name)
PASSWORD	A8 (8-character password)

When a user attempts to invoke the module, the CDPASSW subprogram reads the password file. If the module/system name combination exists in the file and does not have a password, the user can invoke the module. If the module/system name combination exists and has a password, the user must enter the correct password before the module is invoked. If a user enters five incorrect passwords, execution is aborted.

If you specify password checking, you must modify the CDPASSW subprogram to include a valid password view and any final processing you want to perform and then catalog the modified subprogram. For more specific password checking, you can modify the CCPASSW copycode member (to call a different subprogram) or modify the CDPASSW subprogram (to refine your security standards).

4 Change the Default Window Settings

To change the default window settings for your generated module

1 Press PF5 (windw).

The Window Parameters window is displayed. For example:

```
CU--DWM           Natural Construct           CU--DWMO
Oct 21            Window Parameters             1 of 1

          Size .........  Height ....... ___
                          Width ........ ___

          Position .....  Line ......... ___
                          Column ....... ___

          Frame OFF .... _
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---
      help  retrn quit  test
```

2 Use the following fields to build the DEFINE WINDOW statement:

Field	Description
Size	■ Height Number of lines the window spans. ■ Width Number of columns the window spans.
Position	■ Line Number of lines from the top of the panel to the top of the window. ■ Column Number of columns from the left side of the panel to the left side of the window. The line and column values form the top left corner of the window.
Frame OFF	If this field is marked, the window is displayed without a border (frame).

3 Press Enter to confirm the changes.

Test the Modified Window Settings

You can view a test version of the window with the characteristics specified in the Window Parameters window.

To test the modified window settings

- Press PF4 (test) in the Window Parameters window.

5 Determine Which Condition Codes are Set

To determine which condition codes are set

1 Mark the `Condition codes` field in the Optional Parameters window.

2 Press PF2 (retrn).

The Generation main menu is displayed.

3 Generate the module.

After generation is complete, the Condition Codes Trace window is displayed, showing the condition codes set for the current module. For example:

```
CSGCTRAC                 Natural Construct                  CSGCTRC0
Jan 13                 Condition codes Trace                  1 of 1

   Condition codes                     Condition codes
   ------------------------------      ------------------------------
1  X MULTIPLE-ENTITIES              14   DB2-TIMESTAMP
2  X KEY-NOT-DEFINED-AS-UQ          15   CONFINED-KEY-PREFIX
3    KEY-IS-ALPHA                   16   RESET-REDEFINES
4    KEY-IS-LOGICAL                 17   LOGGING-UPDATES
5    HOLD-FIELD-EQ-TIME             18   DB2-REFERENTIAL-INTEGRITY
6  X PRIME-FILE-IS-ADABAS           19   DB2-INTRA-CASCADE-DELETE
7    PRIME-FILE-IS-DB2              20   KEY-IS-REDEFINED
8    CLAUSE-IS-DB2-SPECIFIC         21   DERIVED-DATA
9    PRIME-FILE-IS-IMS              22   NAT-REFACTORING
10   PRIME-FILE-IS-VSAM             23 X GET-BY-ISN
11   KEY-IS-A-SUPER                 24 X HASH-LOCKING
12 X USE-MSG-NR                     25   NAT4-OR-HIGHER
13   KEY-IS-PRIME
```

Tip: If the condition code is not marked, ensure that the code is being generated in a Natural V4 (Natural V6 on open systems) or higher environment and that the option is set to True

in the CSXDEFLT subprogram. CSXDEFLT is installed in the SYSCSTX library. To use the code, compile the subprogram in the SYSCST library and then copy the object code to the SYSLIBS library. For information on CSXDEFLT, see *Use CSXDEFLT Overrides* in *Natural Construct Administration and Modeling*.