CST-Shell Model

This section describes the CST-Shell model, which is used to create a template for a model subprogram. The following topics are covered:

- Introduction

- Parameters for the CST-Shell Model

- User Exits for the CST-Shell Model

Introduction

The CST-Shell model generates a template for a model subprogram; it is similar to the supplied Shell model. The main differences between the models are that the CST-Shell model:

- Supports regeneration

- Supports messaging

The CST-Shell model creates the `DEFINE DATA ... END-DEFINE` framework containing definitions for the global data area (GDA), parameter data areas (PDAs), local data areas (LDAs), or views specified on the Standard Parameters panel, as well as the required REPEAT loops and messaging subroutines. You can use this time-saving model to generate startup modules for your model subprograms.

References

- For an example of a generated shell program, refer to CUMPSLFV in the SYSCST library.

- For information about the Shell model, see *Shell Model*, *Natural Construct Generation*.

Parameters for the CST-Shell Model

Use the CST-Shell model to create the shell subprogram. This model has one specification panel, Standard Parameters.

Standard Parameters Panel

```
CUGSMA                     CST-Shell Program                      CUGSMA0
Jul 05                     Standard Parameters                     1 of 1

  Module name ........ CXMPSLFV
  Module type ........ _
  System name ........ NCSTDEMO_____ *

  Title .............. CST module ..._____
  Description ........ This CST module is used for ..._____
                       _____
  Messaging support .. _
  Global data area ... _____ *
  Parameter data area  _____ _____ _____ _____ _____ *
  Local data area .... _____ _____ _____ _____ _____ *
                       _____ _____ _____ _____ _____ *

  Views ........  1    _____ *
                  2    _____ *
                  3    _____ *
                  4    _____ *
                  5    _____ *
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
main  help  retrn quit                                         userX main
```

The input fields on the Standard Parameters panel are:

Field	Description
Module name	Name specified on the Generation main menu. The name of the shell program must be alphanumeric and no more than eight characters in length.
Module type	Code for the type of module for which you are creating the shell program. Valid codes are: ● P (program) ● N (subprogram) ● H (helproutine) ● S (subroutine)
System name	Name of the system (by default, the name of the current library). The system name must be alphanumeric, no more than 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.)
Title	Title for the generated subprogram. The title identifies the subprogram for the List Generated Modules function on the Generation main menu and is used internally for program documentation.
Description	Brief description of the subprogram. The description is inserted in the banner at the beginning of the subprogram and is used internally for program documentation.
Messaging support	Indicates whether the shell program supports the dynamic translation of messages. To support dynamic translation, mark this field.
Global data area	Name of the global data area used by the generated module.
Parameter data area	Names of up to five inline parameter data areas used by the generated module. **Note:** If the Module type is P or S, you cannot specify parameter data.
Local data area	Names of up to 10 inline or external local data areas used by the generated module.
Views	Names of up to five Predict views used by the generated module.

User Exits for the CST-Shell Model

```
CSGSAMPL                        CST-Shell Program                        CSGSM0
Jul 05                              User Exits                           1 of 1

              User Exits                 Exists    Sample   Required Conditional
   -------------------------------- -------- ---------- -------- ------------
   _   CHANGE-HISTORY                                 Subprogram
   _   PARAMETER-DATA
   _   LOCAL-DATA                                     Example
   _   START-OF-PROGRAM
   _   GENERATE-CODE
   _   BEFORE-CHECK-ERROR                             Example
   _   ADDITIONAL-INITIALIZATIONS                     Example
   _   END-OF-PROGRAM
```

For information about these user exits, see Supplied User Exits. For information about using the User Exit editor, see *User Exit Editor*, *Natural Construct Generation*.