

CST-Proxy Model

This section describes the CST-Proxy model, which is used to generate a client or server proxy to remotely access a subprogram on the server.

This section covers the following topics:

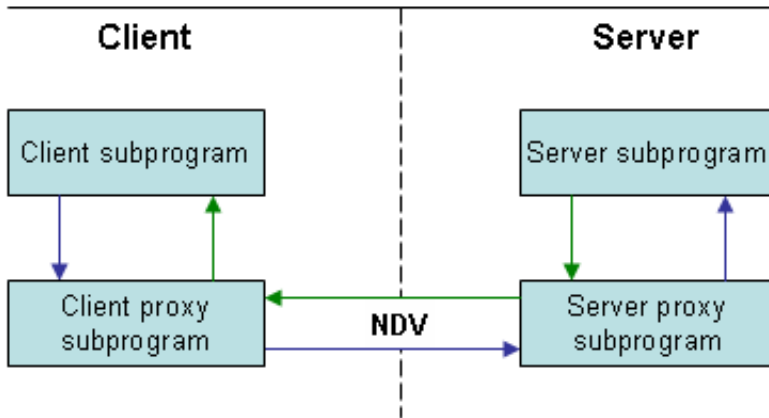
- Introduction
 - Parameters for the CST-Proxy Model
 - User Exits for the CST-Proxy Model
-

Introduction

The CST-Proxy model generates either a client or server proxy to access a subprogram on the server. The proxy acts as a bridge between a subprogram on the client and a subprogram on the server. When a request to the server is initiated from the client (for example, when a user requests active help for a field on a panel), the following process occurs:

1. The client subprogram issues a request, which invokes the client proxy subprogram.
2. The client proxy subprogram converts the data into the network transfer format and identifies the name of the server proxy to be invoked.
3. The data is sent to the server via NDV (Natural Development Server).
4. The server proxy subprogram converts the data to Natural data format and invokes the server subprogram.
5. The server subprogram completes the request.
6. The server proxy subprogram converts the data into network transfer format.
7. The data is sent to the client via NDV.
8. The client proxy subprogram converts the data to Natural data format and returns the information to the client subprogram.

The following diagram illustrates this process. Blue arrows indicate data sent to the server subprogram; green arrows indicate data returned to the client subprogram:



Parameters for the CST-Proxy Model

Use the CST-Proxy model to generate either a client or server proxy to access a subprogram on the server. This model has one specification panel, Standard Parameters.

Standard Parameters Panel

```

CUGXMA                      CST-PROXY Subprogram                      CUGXMA0
Apr 02                      Standard Parameters                        1 of 1

Module ..... MYPROXY_
System ..... CNDPRO_____

Title ..... Proxy for..._____
Description ..... Description of proxy that..._____
                                     _____
                                     _____
                                     _____

Subprogram ..... _____ *
Gen client proxy ... _
Server proxy subp .. _____ *

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---
      help  retrn quit          1:V                                userX main
    
```

The input fields on the Standard Parameters panel are:

Field	Description
Module	Name of the proxy subprogram you are creating (by default, the name specified in the Module name field on the Generation main menu). The name must follow standard Natural naming conventions, must be alphanumeric, and cannot be more than eight characters in length.
System	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 module and system names is used as a key to access help information.)
Title	Title for the proxy 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 proxy subprogram. The description is inserted in the banner at the beginning of the subprogram and is used internally for program documentation.
Subprogram	Name of the subprogram for which you are generating the proxy.
Gen client proxy	Indicates whether the generated proxy is a client proxy or a server proxy. By default, a server proxy is generated. To generate a client proxy, mark this field and specify the name of the corresponding server proxy in Server proxy subp. Note: The specified server proxy subprogram must exist before you can generate the client proxy subprogram; generate the server proxy first.
Server proxy subp	Name of the server proxy subprogram.

Specify the Number of Occurrences Returned

If the proxy handles 1:V arrays, specify the maximum number of 1:V arrays that can be returned to the client for each request. A 1:V array can consist of either one-dimensional data, such as a list of repeating values, or two-dimensional data, such as a row of record data.

To specify the maximum number of occurrences to return for each request:

1. Press PF5 (1:V) on the Standard Parameters panel.

The 1:V Overrides window is displayed. For example:

```

                                1:V Overrides
01 >>

  1 Structure .....
    Field .....
    Occurrences ..... ____ / ____ / ____

  2 Structure .....
    Field .....
    Occurrences ..... ____ / ____ / ____

  3 Structure .....
    Field .....
    Occurrences ..... ____ / ____ / ____

  4 Structure .....
    Field .....
    Occurrences ..... ____ / ____ / ____
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---P
      help retrn                retrv      bkwrtd frwrtd
    
```

Note:

If no fields in the target subprogram use the 1:V notation, a message is displayed. Otherwise, the model determines these values and displays their names.

- Specify the maximum number of occurrences that can be returned to the client with each call to the server.

Press PF5 (retrv) to update the information from the server.

- Press PF2 (retrn) to return to the Standard Parameters panel.

User Exits for the CST-Proxy Model

CSGSAMPL		NATURAL CONSTRUCT			CSGSM0
Apr 02		User Exits			1 of 1
User Exit	Exists	Sample	Required	Conditional	

— CHANGE-HISTORY		Subprogram			
— LOCAL-DATA		Example			
— ON-ERROR-MSG-NR		Example		X	
— START-OF-PROGRAM		Example			
— BEFORE-CALL-SERVER		Example		X	
— AFTER-CALL-SERVER		Example		X	
— BEFORE-CALL-OBJECT		Example		X	
— AFTER-CALL-OBJECT		Example		X	
— SET-DATA-LENGTH		Example			
— SET-RETURN-BLOCKS		Example			
— BEFORE-COMPRESS-OUTPUT		Example			
— AFTER-COMPRESS-OUTPUT		Example			
— BEFORE-EXPAND-INPUT		Example			
— AFTER-EXPAND-INPUT		Example			
— MISCELLANEOUS-SUBROUTINES		Example			
— END-OF-PROGRAM		Example			
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---					
help retrn quit		bkwrtd frwrtd			

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*.