Natural Remote Procedure Call (RPC)

Natural Remote Procedure Call (RPC) is available as a separate subcomponent of Natural. It has its own version number. This measure takes into account that Natural RPC is a cross-platform component and makes it possible to provide new Natural RPC versions independent of new Natural versions for the various platforms supported.

As of Version 6.3 of Natural Remote Procedure Call (RPC), the enhancements listed in this chapter are available. The following topics are covered:

- New Documentation Structure and Contents
- Migration
- New Features
- Enhanced Features
- Changed Features
- SYSRPC Utility Enhancements
- Changes to the Service Directory
- Limitations on OpenVMS

New Documentation Structure and Contents

The Natural RPC documentation has been restructured and revised completely, also taking into account the new features that are now available with EntireX Communicator (EntireX Broker).

Important:

Information on prerequisites and restrictions that you should observe before you start to implement or modify a Natural RPC environment has been summarized in the following sections of the *Natural Remote Procedure Call (RPC)* documentation:

- *Prerequisites and Preparatory Information*
- *Restrictions and Limitations*

Migration

EBCDIC/ASCII Conversion Removed

With Natural RPC Version 5.1.1 (and below), an RPC built-in conversion from EBCDIC to ASCII and vice versa took place if the EntireX attrribute `TRANSLATION` or `CONVERSION` was missing in the Broker attribute file and the encoding of the client data had to be translated to the required encoding of the server platform.

The RPC built-in conversion from EBCDIC to ASCII and vice versa has been removed. The conversion of client data from EBCDIC to ASCII and vice versa is already part of the EntireX functionality. In addition, the EntireX conversion is more versatile than the RPC built-in conversion. Specify the EntireX attributes `TRANSLATION` or `CONVERSION` in the Broker attribute file to ensure an appropriate conversion.

Setting of Steplib SYSRPC Omitted on Client Side

With Natural RPC Version 5.1.1 (and below), the library `SYSRPC` is implicitly set as a steplib when the service is accessed on the client side.

The library `SYSRPC` is no longer set as a steplib during execution of a remote `CALLNAT` request. This is to improve the performance and to avoid that an old service directory in the library `SYSRPC` is accidentally accessed. The `SYSRPC` utility places the service directory in the current user library instead of library `SYSRPC` in the `FNAT` system file. Therefore, the library `SYSRPC` should not contain any user-specific information, including the service directory.

If you want to continue to use a service directory that has been placed in library `SYSRPC` before, you have the following choices:

- Move the service directory from the library `SYSRPC` to the client library.
- Move the service directory from the library `SYSRPC` to the library `SYSTEM` in the `FUSER` system file.
- Define the library `SYSRPC` as an additional steplib using Natural Security.
- Use the application programming interface `USR3025N` in the library `SYSEXT` to set the library `SYSRPC` as an additional steplib in an environment without Natural Security.

New Features

Support of the New EntireX RPC Protocol RPC2000

The new EntireX RPC protocol RPC2000 introduces the following new features that are supported by Natural RPC Version 6.3:

- The new header layout of the RPC2000 protocol has removed all length restrictions for names. However, Natural-specific names such as library names, program names and user IDs remain limited to 8 characters (this feature requires that RPC2000 is enabled).
- Base 64 encoding is used for marshalling of binary data. This saves up to 30 percent space in the send/receive buffer (this feature requires that RPC2000 is enabled).
- If the parameter data area (PDA) contains arrays of groups, the physical field sequence of a PDA may be different on the client and server sides if the PDA contains arrays of groups. With Natural RPC Version 6.3, the group structure of a PDA may be kept when an RPC server is called by a Natural client. This is to ensure the same physical field sequence in the PDA for both the client and the server.

X-Arrays

X-arrays are supported in the parameter list of a remote `CALLNAT` execution. The server may increase or decrease the number of occurrences.

Restrictions:

- In case of a multidimensional array, all dimensions of the array must be extensible.

- The lower bound must not be extensible, that is only extensible upper bounds are allowed.

Format U Data

Variables of format U (Unicode data in UTF-16) are supported in the parameter list of a remote `CALLNAT` execution. The variables may either be fixed length variables or dynamic variables.

If U data is sent from a platform with big endian encoding to a platform with little endian encoding or vice versa, the encoding is adapted so that it conforms with the encoding on the receiving platform. For example, when U data in little endian encoding arrives on a big endian platform, this data is converted to big endian encoding before it is handed over to the program. When this data is sent back, it is converted back to little endian encoding.

Optional Parameters

Optional parameters (*n*`X` notation in the `CALLNAT` statement) are supported in the parameter list of a remote `CALLNAT` execution.

Support of EntireX ACI Version 6 and ACI Version 7 Features

With Natural RPC Version 6.3, the following features available with ACI Version 6 and ACI version 7 are supported:

- Data compression to compress the send/receive buffer on client and server side. This feature reduces the amount of data that is to be transferred from client to server and vice versa. To use this feature, ACI Version 7 is required.

- Data that is exchanged between client and server may be encrypted for reasons of data security. To use this feature, ACI Version 6 is required.

Application Programming Interface to Set the Target Library for CALLNAT Execution on the Server Side

To unbundle the client and the server environment, an application programming interface (API) is provided. This API (`USR4008`) enables the client to set the library in which the `CALLNAT` request is to be executed on the server side. The `LOGON` option must be set to `ON` to use this feature.

Application Programming Interface to Set EntireX Specific Parameters

A new application programming interface (`USR4009`) is provided to enable the client to set EntireX-specific parameters for RPC request execution.

Application Programming Interface to Terminate the EntireX Broker Service

The new application programming interface `USR2075N` enables you to terminate the EntireX Broker Service from within your application. See *Terminating an EntireX Broker Service* in the *Natural Remote Procedure Call (RPC)* documentation.

New RPC User Exits Called by Natural RPC Server at Certain Events

With Natural RPC Version 6.3, two optional user exits are introduced. These exits are called by the Natural RPC server at certain events and must be implemented by the user as a subprogram. The following two user exits have been introduced:

- A before-service-execution user exit that is called after the data is unmarshalled and before the service (the remote `CALLNAT`) is executed.
- An after-service-execution user exit that is called on return from the service (the remote `CALLNAT`) and before the data is marshalled.

New Parameter CPRPC

The new parameter `CPRPC` is used to specify the name of the code page used by the EntireX Broker. `CPRPC` is applied to the Natural RPC facility when the transport protocol ACI (that is EntireX Broker) is used. See also *Migrating Applications to Version 6.3*.

Wait Time of RPC Server

The new parameter `SRVWAIT` is used to specify the number of seconds the server is to wait for an RPC client request.

Enhanced Features

Server Trace Enhanced to Aid in Error Diagnostics

With Natural RPC Version 6.3, additional information is logged in the Natural RPC server trace file to aid in error diagnostics.

Information about Server Status Provided to Aid in Error Diagnostics

For better error diagnostics, the server will provide additional status information. An application programming interface (API) is provided to return this information to the client.

Status information includes:

- the `STEPLIB` chain,
- the Natural buffer pool that is used,
- important Natural profile parameter settings (`FNAT`, `FUSER`, RPC parameters, ...).

Performance of Request Processing Improved

Several administrative database calls, that have been issued after processing of the client RPC request has been completed, have been eliminated with Natural RPC Version 6.3. This improves the performance of the RPC server especially for those requests which generate only a small workload.

Natural Security Password Change via RPC

The Natural RPC server has been enhanced by the possibility to change the Natural Security (NSC) password on the `FSEC` by a remote procedure call.

The new application programming interface `USR2074N` allows a Natural RPC client which is running in a non-NSC environment to change the NSC password on the `FSEC` via a remote procedure call. It is no longer required for the client to use a terminal emulation to change the password on the Natural logon screen.

Similar interfaces are provided by EntireX to enable an EntireX RPC client to change his NSC password.

Case-Sensitive User ID and Password

The sample callers `USR1071P`, `USR2071P` and `USR2072P` of the application programming interfaces `USR1071N`, `USR2071N` and `USR2072N` allow the use of case-sensitive user IDs and passwords.

By default, user IDs and passwords are translated into upper case for compatibility reasons.

Prevent Password Attacks on Server

To prevent password attacks if a Natural RPC server is running under Natural Security, each invalid logon attempt is now recorded on the FSEC. If the limit for logon attempts is reached for a user, this user is locked and any further logon attempt (valid or invalid) is rejected. The counter for the logon attempts is reset for a specific user after a successful logon of this user.

Please note that this feature leads to the intended incompatibility that a user must be unlocked after the limit of logon attempts has been reached. In previous versions, the number of attempts was unlimited.

Refer to *RPC Server Session Options* in *Library and User Preset Values* in the *Natural Security* documentation for more details.

Logon Forced after Timeout

If your application has once issued an EntireX Broker logon using the application programming interface `USR2071N`, a new logon is enforced after a timeout. In previous versions, the behavior after a timeout depends on the `AUTOLOGON` setting in the EntireX Broker attribute file.

- With `AUTOLOGON=YES` an implicit logon is issued.
- With `AUTOLOGON=NO` a new logon is enforced.

This feature avoids problems with external security systems such as RACF if `AUTOLOGON=YES` is specified but an explicit logon is required.

Support of Simplified EntireX Security Installation Provided with ACI Version 8

If the profile parameter ACIVERS is set to 8, the Natural RPC issues a KERNELVERS call to get the correct value for the ACI field KERNELSECURITY. In this case, it is no longer required to perform the special installation steps to install the EntireX Broker stubs for EntireX Security (install the secuexit module).

This enhancement applies to the Natural RPC server as well as to the Natural RPC client. In case of a Natural RPC client, the KERNELVERS call is done as part of the EntireX logon using the application programming interface USR2071N.

DFS Profile Parameter Enhanced

The profile parameter DFS has been enhanced to prevent the Natural RPC from accessing and using a service directory. This avoids that a service directory has to be created before Natural RPC can be used. Using this feature will result in a reduction of administrative overhead and a corresponding performance enhancement.

TRACE Profile Parameter Enhanced

The TRACE profile parameter has been enhanced by an option to write a trace entry only if an error has occurred. This enables you to analyze errors on the Natural RPC server side without the need to have the Natural RPC trace permanently turned on.

ACIVERS Profile Parameter Enhanced

The EntireX ACI Version 9 is now supported by ACIVERS. This allows you to enable the EntireX Broker stub to send additional environmental information about client and server to the EntireX Broker.

Changed Features

Reset Natural Limits on Server

The following Natural limits apply to the Natural RPC server:

- LT - Limit for Processing Loops,
- MAXCL - Maximum Number of Program Calls,
- MADIO - Maximum DBMS Calls between Screen I/O Operations.

This makes it difficult to enforce certain limits on every CALLNAT request.

With Natural RPC Version 6.3, these limits are reset after each remote CALLNAT execution to avoid unpredictable error situations when the execution of a client request reaches one of the limits due to the processing of a previous client.

Long Names for Node and Server

The maximum length for node and server names in the sample callers `USR2007P` and `USR2071P` of the application programming interfaces `USR2007N` and `USR2071N` has been increased from 8 to 32.

SYSRPC Utility Enhancements

The following changes/enhancements have been provided in the `SYSRPC` utility:

- Calculate Space Requirements without Stub Generation
- Generate Stubs in Command Mode
- Comment Added to Parameters during Stub Generation
- Maximum Length of Format Buffer Increased
- Stub Generation Supports Dynamic Variables
- Stub Generation Supports Group Structure
- Stub Generation Supports X-Arrays
- Number of Parameters for Stub Generation Increased
- Stub Generation Supports Format U Data
- Allow Wildcard Notation for Subprogram Names
- Search Function Provided
- Terminate Server Tasks

Calculate Space Requirements without Stub Generation

For stubless RPC calls, the required space for the send/receive buffers can be calculated without generating a stub to allow the user to determine the space requirements.

Generate Stubs in Command Mode

For stubs with the default field attribute M (inout), the stub generation can now also be executed in command mode with wildcard notation for a set of stubs. A stub can be generated in batch mode on the basis of an existing subprogram. In this case, all field attributes will be M.

Comment Added to Parameters during Stub Generation

When a stub is generated, a comment is added to the parameters to indicate whether a parameter is an input or an output parameter.

Maximum Length of Format Buffer Increased

The format buffer length limit is dynamically calculated depending on the buffer space currently used for data.

Stub Generation Supports Dynamic Variables

It is now possible to generate stubs that contain dynamic variables in their parameter data area (PDA). The **Stub Generation** screen has been adapted accordingly.

Stub Generation Supports Group Structure

It is now possible to enter field levels and groups on the **Stub Generation** screen. The definition of groups is required to access certain EntireX RPC servers.

Only contiguous groups are supported.

Stub Generation Supports X-Arrays

It is now possible to generate stubs that contain X-arrays in their parameter data area (PDA). The **Stub Generation** screen has been adapted accordingly.

Number of Parameters for Stub Generation Increased

In the **Stub Generation** screen (which is used to enter the parameter definitions that are used in the stub subprogram), you can now specify a maximum of 5000 parameters. See *Specifying Parameters* in the SYSRPC section of the *Tools and Utilities* documentation.

Stub Generation Supports Format U Data

It is now possible to generate stubs that contain format U variables in their parameter data area (PDA). The **Stub Generation** screen has been adapted accordingly.

Allow Wildcard Notation for Subprogram Names

To reduce the size of the service directory and to enhance its readability, it will be possible to use wildcard notation to specify subprogram names.

Search Function Provided

To ease the inspection of a large service directory, a search function is provided for subprogram names.

Terminate Server Tasks

It is now possible to terminate all server tasks associated with a given EntireX Broker service. See *Terminating a Server* in the *SYSRPC* part of the *Utility* documentation. In previous versions, it was only possible to terminate a single server.

Changes to the Service Directory

Support Local Directory Layout by Remote Directory Server

With Natural RPC Version 6.3, an application programming interface (API) is provided to enable access to information stored in the local directory located on the remote server.

Version Number and Copyright Notice Added to Service Directory

A comment is added to the service directory containing the Natural version that was used to generate the service directory. In addition, a copyright notice of Software AG will be contained in the service directory.

Limitations on OpenVMS

In some places, the documentation set for Natural for OpenVMS mentions features which are (currently) not supported with Natural for OpenVMS. Such features are:

Integrated Authentication Framework (IAF)

The Integrated Authentication Framework (IAF), which is an optional feature that can be used on the Natural RPC server side and which is controlled by Natural Security, is not supported on OpenVMS platforms. Therefore, the following section in the *Natural Remote Procedure Call (RPC)* documentation does not apply on OpenVMS: *Using the Integrated Authentication Framework*.