

webMethods EntireX

Using EntireX RPC for RPG under IBM i

Version 10.9

April 2023

This document applies to webMethods EntireX Version 10.9 and all subsequent releases.

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

Copyright © 1997-2023 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA, Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://softwareag.com/licenses>.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at <http://softwareag.com/licenses/> and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <http://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

Document ID: EXX-RPC-109-20230403RPG

Table of Contents

1 About this Documentation	1
Document Conventions	2
Online Information and Support	2
Data Protection	3
2 Using EntireX RPC for RPG under IBM i	5
Creating the Client/Server Interface	6
Creating a Sample Server in RPG	7
Compiling and Linking the Server	8
Verifying the Server	8

1 About this Documentation

▪ Document Conventions	2
▪ Online Information and Support	2
▪ Data Protection	3

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Monospace font	Identifies service names and locations in the format <code>folder.subfolder.service</code> , APIs, Java classes, methods, properties.
<i>Italic</i>	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

Product Documentation

You can find the product documentation on our documentation website at <https://documentation.softwareag.com>.

In addition, you can also access the cloud product documentation via <https://www.software-ag.cloud>. Navigate to the desired product and then, depending on your solution, go to “Developer Center”, “User Center” or “Documentation”.

Product Training

You can find helpful product training material on our Learning Portal at <https://knowledge.softwareag.com>.

Tech Community

You can collaborate with Software AG experts on our Tech Community website at <https://tech-community.softwareag.com>. From here you can, for example:

- Browse through our vast knowledge base.
- Ask questions and find answers in our discussion forums.
- Get the latest Software AG news and announcements.
- Explore our communities.
- Go to our public GitHub and Docker repositories at <https://github.com/softwareag> and <https://hub.docker.com/publishers/softwareag> and discover additional Software AG resources.

Product Support

Support for Software AG products is provided to licensed customers via our Empower Portal at <https://empower.softwareag.com>. Many services on this portal require that you have an account. If you do not yet have one, you can request it at <https://empower.softwareag.com/register>. Once you have an account, you can, for example:

- Download products, updates and fixes.
- Search the Knowledge Center for technical information and tips.
- Subscribe to early warnings and critical alerts.
- Open and update support incidents.
- Add product feature requests.

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 Using EntireX RPC for RPG under IBM i

▪ Creating the Client/Server Interface	6
▪ Creating a Sample Server in RPG	7
▪ Compiling and Linking the Server	8
▪ Verifying the Server	8

This chapter describes how to build a server application using the IBM i ILE language RPG. The server will be named `CALC_RPG`. Its functionality and implementation is based on the ILE COBOL server `CALC` as described in the chapter *Using the COBOL Wrapper*.

The function `CALC_RPG` is a calculator that can add, subtract, multiply and divide two binary values `PIC S9(8) BINARY` and return a result.

This chapter covers the following topics:

The EntireX RPC Server for IBM i of EntireX version 7.1.1 running under IBM i (AS/400) is no longer supported. We strongly recommend using the new *RPC Server for AS/400* or the EntireX Adapter. See also *Connection Parameters for AS/400 Connections*.

Creating the Client/Server Interface

The PLIST in the RPG program is a good source of type information for the interface when you create the Software AG IDL file:

```

...
D OPERATOR          S           1A
D OPERAND_1         S           10I 0
D OPERAND_2         S           10I 0
D FCT_RESULT        S           10I 0
*
C      *ENTRY        PLIST
C              PARM          OPERATOR
C              PARM          OPERAND_1
C              PARM          OPERAND_2
C              PARM          FCT_RESULT
.....

```

The assumption is made that the program is implemented in library `EXAMPLE`. Convert the linkage section function above to Software AG IDL syntax as follows:

```

Library 'EXAMPLE' Is
Program 'CALC_RPG' Is
  Define Data Parameter
    1 Operator          (A1) In
    1 Operand_1        (I4) In
    1 Operand_2        (I4) In
    1 Function_Result  (I4) Out
  End-Define

```



Note: A 10-digit RPG integer takes 4 bytes, so it must be mapped to an (I4) IDL field definition.

For details on how IDL field definitions are mapped to RPG elementary field items, see `div-name_rpc_rpg_map`.

Creating a Sample Server in RPG

The server is implemented as an ILE RPG program of type *PGM.

For our IDL example CALC, the implemented server looks similar to the example below.

```

*-----
* Member          CALC_RPG
* Description     Calculation Engine.
*
* Author         (c) Software AG
* Platform       OS/400
*
* UUU YYYY-MM-DD History
* HBA 2003-05-20 Created
*
*----- CALC Interface -----
D OPERATOR       S           1A
D OPERAND_1      S           10I 0
D OPERAND_2      S           10I 0
D FCT_RESULT     S           10I 0
*-----
C   *ENTRY       PLIST
C               PARM          OPERATOR
C               PARM          OPERAND_1
C               PARM          OPERAND_2
C               PARM          FCT_RESULT
C               CLEAR        FCT_RESULT
*
C               SELECT
C               WHEN          OPERATOR = '+'
C               EVAL          FCT_RESULT = OPERAND_1 + OPERAND_2
C               WHEN          OPERATOR = '-'
C               EVAL          FCT_RESULT = OPERAND_1 - OPERAND_2
C               WHEN          OPERATOR = '*'
C               EVAL          FCT_RESULT = OPERAND_1 * OPERAND_2
C               WHEN          OPERATOR = '/'
C               IF            OPERAND_2 <> *ZERO
C               EVAL          FCT_RESULT = OPERAND_1 / OPERAND_2
C               ENDIF
C               ENDSL
*
C   PGM_EX      TAG
C               MOVE          *ON          *INLR

```

Compiling and Linking the Server

Compile the server source using the IBM i command `CRTRPGMOD` (create bound RPG module) and bind it as a dynamically callable program of type `*PGM` using the command `CRTPGM`. Make sure your server fulfills all requirements to be callable by the IBM i host server.

As an alternative to the commands `CRTRPGMOD` and `CRTPGM`, you can use the command `CRTBNDRPG` to compile and bind RPG sources in one step.

Name the resulting server program like the program name in the IDL file and put it in a library whose name corresponds to the library name in the IDL file.

Example

If a client performs an RPC which is based on the IDL program `CALC_RPG` in the IDL library `EXAMPLE`, the *RPC Server for AS/400* or the EntireX Adapter (see *Connection Parameters for AS/400 Connections*) will dynamically try to execute the ILE program `CALC_RPG` in the IBM i library `EXAMPLE`. If no corresponding program can be found, the access fails.

Verifying the Server

To verify the server program `CALC_RPG`, use the *EntireX IDL Tester* when the *RPC Server for AS/400* and EntireX Broker are used.

> To verify the server

- 1 Confirm that an EntireX Broker and an *RPC Server for AS/400* are available in your network.
- 2 Make sure the IBM i host is started on your IBM i machine.
- 3 Create a *Software AG IDL File* in the IDL Editor documentation using the *IDL Editor* as described under [Creating the Client/Server Interface](#).
- 4 Start the IDL Tester using the IDL file created in step 3.



Note: When using the EntireX Adapter, use the **Service Development** perspective of the Software AG Designer for testing.