



# **TIBCO Rendezvous®**

## Using the IBM i Samples

Version 8.8.0 | February 2025

# Contents

---

<b>RPG Samples for Rendezvous .....</b>	<b>3</b>
<b>Listen Program .....</b>	<b>4</b>
<b>Send Program .....</b>	<b>6</b>
<b>Prototype File .....</b>	<b>8</b>
<b>Compiling the Samples .....</b>	<b>9</b>
<b>Binding Applications .....</b>	<b>10</b>
<b>Executing the Samples .....</b>	<b>11</b>
Executing from the QSHELL Environment .....	11
Executing in Batch .....	11
<b>Rendezvous Programming in RPG .....</b>	<b>12</b>
<b>TIBCO Documentation and Support Services .....</b>	<b>15</b>
<b>Legal and Third-Party Notices .....</b>	<b>17</b>

# RPG Samples for Rendezvous

---

The Rendezvous installation package for IBM i includes ILE RPG samples (source files and executables).

The samples `tibrvlisten.rpgle` and `tibrvsend.rpgle` parallel the functionality in `tibrvlisten.c` and `tibrvsend.c`.

The file `TIBRV.rpgle` contains API prototypes for the calls in these samples.

- [Listen Program](#)
- [Send Program](#)
- [Prototype File](#)
- [Compiling the Samples](#)
- [Executing the Samples](#)
- [Rendezvous Programming in RPG](#)

# Listen Program

---

The program `tibrvlisten` receives Rendezvous messages and outputs them.

Copies of `tibrvlisten.rpgle` exist both in `QRPGLSRC(TIBRVLISTE)` and in the IFS (at `RV_IFS_HOME/src/examples/rpg`).

This implementation of `tibrvlisten` illustrates an RPG-style call interface with positional parameters.

For simplicity, the program uses the `DSPLY` operation to the `QSYSOPR` message queue for all output. To monitor this information without explicitly displaying the message queue, consider setting the queue to `*BREAK` mode:

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK) SEV(0)
```

## Syntax

```
call tibrvlisten service network daemon subject1  
                                     subject_list
```

## Parameters

The first four parameters all accept either a specific value, or the keyword `*DEFAULT`.

Parameter	Description
<i>service</i>	Specify the service number. For details, see Service Parameter in TIBCO Rendezvous Concepts.  The program interprets the keyword <code>*DEFAULT</code> as 7500.
<i>network</i>	Specify the network name. For details, see Network Parameter in TIBCO Rendezvous Concepts.  The program interprets the keyword <code>*DEFAULT</code> as the null string.

Parameter	Description
<i>daemon</i>	Specify the daemon parameter. For details, see Daemon Parameter in TIBCO Rendezvous Concepts.  The program interprets the keyword *DEFAULT as localhost:7500.
<i>subject1</i>	The program listens for messages with this subject name.  The program interprets the keyword *DEFAULT as 'your.subject'.
<i>subject_list</i>	Optional.  When present, the program also listens to these subject names.  You may supply between zero and four additional subjects.

## Call Examples

The first three examples specify equivalent behavior—the program uses the default values for all parameters.

```
ADDLIBLE tibrv_lib
CALL PGM(TIBRVLISTE)
ADDLIBLE tibrv_lib
CALL PGM(TIBRVLISTE) PARM(*DEFAULT *default *DEFAULT *DEFAULT)
ADDLIBLE tibrv_lib
CALL PGM(TIBRVLISTE) PARM(*DEFAULT *DEFAULT *DEFAULT )
```

In the next example, the user supplies a non-default network parameter, and a second listen subject.

```
ADDLIBLE tibrv_lib
CALL PGM(TIBRVLISTE) PARM('7500' *DEFAULT 'myhost.my_
enterprise.com:7500' 'our.subject' 'your.subject')
```

# Send Program

---

The send program sends Rendezvous messages.

Copies of `tibrvsend.rpgle` exist both in `QRPGLESRC(TIBRVSEND)` and in the IFS (at `RV_IFS_HOMERV_IFS_HOME/src/examples/rpg`).

In contrast to the listen program, this implementation illustrates a C-style call interface with parameter switches. The code uses fixed-column calculation specifications.

For simplicity, the program uses the `DSPLY` operation to the `*EXT` message queue to communicate with the user.

## Syntax

```
call  tibrvsend -service service -network network -daemon daemon
      subject
      msg_list
```

## Parameters

The first three parameters (`-service`, `-network` and `-daemon`) are switched parameters; pair each switch with its value. You may enter these three pairs in any order. If you omit any of these three parameters, the program supplies default values.

Parameter	Description
<code>-service</code> <i>service</i>	Optional.  Specify the service number. For details, see Service Parameter in TIBCO Rendezvous Concepts.  When absent, the default value is 7500.
<code>-network</code> <i>network</i>	Optional.  Specify the network name. For details, see Network Parameter in TIBCO

Parameter	Description
	<p>Rendezvous Concepts.</p> <p>When absent, the default value is the null string.</p>
<p><code>-daemon</code> <i>daemon</i></p>	<p>Optional.</p> <p>Specify the daemon parameter. For details, see Daemon Parameter in TIBCO Rendezvous Concepts.</p> <p>When absent, the default value is <code>localhost:7500</code>.</p>
<i>subject</i>	<p>Required.</p> <p>The program interprets the first non-switched parameter as the send subject name.</p>
<i>msg_list</i>	<p>Required.</p> <p>The program interprets all subsequent parameters as messages. It sends these messages on the send subject.</p> <p>The program parses the message strings, and supplies field names. You must supply the characters <code>**</code> to delimit each message string. Embedded space characters are permitted.</p>

## Call Example

Notice that this example omits the `-service` and `-network` parameters. It supplies two messages.

```
ADDLIBLE tibrv_lib
CALL PGM(TIBRVSEND) PARM('-daemon' 'myhost.my_enterprise.com:7500'
'b.c.d' 'message 1**' 'message 2* foo **' )
```

# Prototype File

---

The file `TIBRV.rpgle` contains the prototypes of the Rendezvous calls used in the two RPG examples (which constitute only a subset of the full Rendezvous API).

Copies of `TIBRV.rpgle` exist both in `QRPGLSRC(TIBRV)` and in the IFS (at `RV_IFS_HOME/src/examples/rpg`).

# Compiling the Samples

---

To compile the samples, use the following command(s):

## Procedure

1. Find the include file:

```
ADDLIB tibrv_lib
```

2. Create the module:

```
CRTPGMOD MODULE(lib_name/TIBRVSEND) SRCFILE(tibrv_lib/QRPGLESRC)
```

3. Bind the programs:

```
CRTPGM PGM(lib_name/TIBRVSEND) BNDSRVPGM((tibrv_lib/LIBTIBRVI) (tibrv_lib/LIBTIBRV))
```

# Binding Applications

---

It is possible that several versions of Rendezvous are installed. In this situation, you must bind and run applications with the correct version of Rendezvous. Two options are available for binding a product version.

## Option A

Statically bind applications to a specific service program. For example:

```
CRTPGM PGM(yourlib/TIBRVSEND) MODULE(yourlib/TIBRVSEND) BNDSRVPGM( (tibrv_  
lib/LIBTIBRV) (tibrv_lib/LIBTIBRVI))
```

## Option B

Dynamically bind applications with the service program, and determine the version of Rendezvous at run time. First add one of the Rendezvous product libraries to the library list, and then bind your program. For example:

```
ADDLIBLE tibrv_lib  
CRTPGM PGM(yourlib/TIBRVSEND) MODULE(yourlib/TIBRVSEND) BNDSRVPGM  
(((*LIBL/LIBTIBRV) (*LIBL/LIBTIBRVI))
```

To verify the binding of your application, you may execute the following command (to see the library in which service program LIBTIBRV is found):

```
DSPPGM PGM(yourlib/TIBRVSEND) DETAIL(*SRVPGM)
```

# Executing the Samples

---

You can execute the RPG sample programs either from the QSHELL environment or in batch.

Multi-threading is required in either environment.

## Executing from the QSHELL Environment

### Procedure

1. Enable multi-threading (before starting the QSHELL environment):

```
ADDENVVAR ENVVAR(QIBM_MULTI_THREADED) VALUE(Y) LEVEL(*SYS)
```

2. Start the QSHELL environment:

```
STRQSH
```

3. Start the program:

```
system "CALL PGM(libr_name/TIBRVLISTE) PARM('7500' *DEFAULT  
'myhost.my_enterprise.com:7500' 'our.subject' 'your.subject')"
```

## Executing in Batch

Start the program:

```
SBMJOB CMD(CALL PGM(libr_name/TIBRVLISTE) PARM('7500' *DEFAULT 'myhost.my_  
enterprise.com:7500' 'our.subject' 'your.subject')) LOG(4 0 *SECLVL)  
ALWMLTTHD(*YES)
```

# Rendezvous Programming in RPG

Coding your own applications in RPG requires that you extend the prototype file to include all the Rendezvous calls that your application uses. Use the information in this section as a guide when you extend the prototype file, and when you code Rendezvous calls in RPG.

Parameter passing protocol in C differs from the parameter passing protocol in RPG. By default RPG passes parameters by reference, unless you explicitly code to pass by value. In contrast, C passes parameters by value (though you can explicitly supply a pointer referencing a location).

The following table guides the translation from C to RPG. Each section of the table addresses a specific type of parameter passing in C, followed by the equivalent constructs in RPG.

When C Passes a Non-String by Value	
C Prototype	<pre>foo(int order)</pre>
C Program Call	<pre>int order; foo(order);</pre>
RPG Prototype	<pre>...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 d foo                pr d order                10i 0 value</pre>
RPG Program Call	<pre>...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 D order                S                10i 0 inz(37) C                callp        foo(order)</pre>

## When C Passes a Non-String by Reference

C Prototype	<code>foo(int* order)</code>
C Program Call	<code>int order; foo(&amp;order);</code>
RPG Prototype	<pre> ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 d foo          pr d order                      10i 0 </pre>
RPG Program Call	<pre> ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 D order          S          10i 0 inz(37) C              callp      foo(order) </pre>

## When C Passes a String by Value

C Prototype	<code>foo(char* desc)</code>
C Program Call	<code>char* desc; foo(desc);</code>
RPG Prototype	<pre> ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 d foo          pr d desc                      *   value </pre>
RPG Program Call	<pre> ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 D desc          S          11A /free   desc = 'TIBRVLISTE' + X'00';   foo(%addr(desc)); /end-free </pre>

## When C Passes a String by Reference

C Prototype	<pre>foo(char** desc)</pre>
C Program Call	<pre>char* desc; foo(&amp;desc);</pre>
RPG Prototype	<pre>...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 d foo          pr d desc                      *</pre>
RPG Program Call	<pre>...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 D desc          S          11A D pDesc         S          *   inz(%addr (desc)) /free   desc = 'TIBRVLISTE' + X'00';   foo(pDesc); /end-free</pre>

# TIBCO Documentation and Support Services

---

For information about this product, you can read the documentation, contact TIBCO Support, and join TIBCO Community.

## How to Access TIBCO Documentation

Documentation for TIBCO products is available on the [Product Documentation website](#), mainly in HTML and PDF formats.

The [Product Documentation website](#) is updated frequently and is more current than any other documentation included with the product.

## Product-Specific Documentation

The documentation for this product is available on the [TIBCO Rendezvous® Product Documentation](#) page.

## How to Contact Support for TIBCO Products

You can contact the Support team in the following ways:

- To access the Support Knowledge Base and getting personalized content about products you are interested in, visit our [product Support website](#).
- To create a Support case, you must have a valid maintenance or support contract with a Cloud Software Group entity. You also need a username and password to log in to the [product Support website](#). If you do not have a username, you can request one by clicking **Register** on the website.

## How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature

requests from within the [TIBCO Ideas Portal](#). For a free registration, go to [TIBCO Community](#).

# Legal and Third-Party Notices

---

SOME CLOUD SOFTWARE GROUP, INC. (“CLOUD SG”) SOFTWARE AND CLOUD SERVICES EMBED, BUNDLE, OR OTHERWISE INCLUDE OTHER SOFTWARE, INCLUDING OTHER CLOUD SG SOFTWARE (COLLECTIVELY, “INCLUDED SOFTWARE”). USE OF INCLUDED SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED CLOUD SG SOFTWARE AND/OR CLOUD SERVICES. THE INCLUDED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER CLOUD SG SOFTWARE AND/OR CLOUD SERVICES OR FOR ANY OTHER PURPOSE.

USE OF CLOUD SG SOFTWARE AND CLOUD SERVICES IS SUBJECT TO THE TERMS AND CONDITIONS OF AN AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER AGREEMENT WHICH IS DISPLAYED WHEN ACCESSING, DOWNLOADING, OR INSTALLING THE SOFTWARE OR CLOUD SERVICES (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH LICENSE AGREEMENT OR CLICKWRAP END USER AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE SAME TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of Cloud Software Group, Inc.

TIBCO, the TIBCO logo, the TIBCO O logo, TIB, Information Bus, FTL, eFTL, Rendezvous, and LogLogic are either registered trademarks or trademarks of Cloud Software Group, Inc. in the United States and/or other countries.

Java and all Java based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only. You acknowledge that all rights to these third party marks are the exclusive property of their respective owners. Please refer to Cloud SG’s Third Party Trademark Notices (<https://www.cloud.com/legal>) for more information.

This document includes fonts that are licensed under the SIL Open Font License, Version 1.1, which is available at: <https://scripts.sil.org/OFL>

Copyright (c) Paul D. Hunt, with Reserved Font Name Source Sans Pro and Source Code Pro.

Cloud SG software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the “readme” file

for the availability of a specific version of Cloud SG software on a specific operating system platform.

THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. CLOUD SG MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S), THE PROGRAM(S), AND/OR THE SERVICES DESCRIBED IN THIS DOCUMENT AT ANY TIME WITHOUT NOTICE.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "README" FILES.

This and other products of Cloud SG may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at <https://www.cloud.com/legal>.

Copyright © 1997-2025. Cloud Software Group, Inc. All Rights Reserved.