Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE "LICENSE" FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document contains confidential information that 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 TIBCO Software Inc.

TIBCO, The Power of Now, TIB, Information Bus, Rendezvous, TIBCO Rendezvous and Messaging Appliance are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

EJB, Java EE, J2EE, and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

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.

THIS 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 THIS SOFTWARE VERSION 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. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

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 "READ ME" FILES.

Copyright © 1997–2012 TIBCO Software Inc. ALL RIGHTS RESERVED.

TIBCO Software Inc. Confidential Information
Contents

Tables ........................................................................................................................................... v

Preface ........................................................................................................................................... vii
  Changes from the Previous Release of this Guide ................................................................. viii
  Related Documentation ........................................................................................................... ix
    TIBCO Product Documentation ............................................................................................... ix
    Third Party Documentation .................................................................................................... xi
  Typographical Conventions .................................................................................................... xiii
  Connecting with TIBCO Resources ......................................................................................... xvi
    How to Join TIBCO Community ............................................................................................... xvi
    How to Access All TIBCO Documentation ............................................................................. xvi
    How to Contact TIBCO Support .............................................................................................. xvi

Chapter 1 Introduction ................................................................................................................. 1
  Overview .................................................................................................................................. 2
  Hardware Requirements ........................................................................................................... 3
    Storage Requirements ........................................................................................................... 3
  System Software Requirements .............................................................................................. 4
    MVS Environment .................................................................................................................. 4

Chapter 2 Installation .................................................................................................................... 7
  Before Installation ..................................................................................................................... 8
  Distribution Media and Contents ............................................................................................. 9
    Complete Replacement Package ........................................................................................... 9
    Component Code .................................................................................................................... 9
    Obtaining the Installation Media .......................................................................................... 9
  Checklist for Rendezvous Installation ................................................................................... 11
  Uploading the Software .......................................................................................................... 12
    Initial Installation .................................................................................................................. 12
  Rendezvous MVS Installation Procedure .............................................................................. 14
  Verifying the Rendezvous Installation ..................................................................................... 17
  Optional Installation ................................................................................................................ 19
    Compiling C Programs ........................................................................................................... 19
    Compiling COBOL Programs ............................................................................................... 20
<table>
<thead>
<tr>
<th>Contents</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Alternate UNIX installation</td>
<td>20</td>
</tr>
<tr>
<td><strong>Chapter 3  z/OS MVS</strong></td>
<td>21</td>
</tr>
<tr>
<td>Starting Daemons and Applications</td>
<td>22</td>
</tr>
<tr>
<td>User Profile Requirements</td>
<td>22</td>
</tr>
<tr>
<td>JCL and Procedures</td>
<td>23</td>
</tr>
<tr>
<td>Daemon Tasks (Non-Swappable and No CPU Limit)</td>
<td>24</td>
</tr>
<tr>
<td>Sites with Multiple Stacks</td>
<td>24</td>
</tr>
<tr>
<td>Time Zone</td>
<td>24</td>
</tr>
<tr>
<td>Case Sensitivity in Input Parameters</td>
<td>25</td>
</tr>
<tr>
<td>JCL Statements for Daemons</td>
<td>25</td>
</tr>
<tr>
<td>Adding a Certificate</td>
<td>27</td>
</tr>
<tr>
<td>JCL Statements for Applications</td>
<td>27</td>
</tr>
<tr>
<td>JCL Statements for Certified Messaging (RVCM)</td>
<td>28</td>
</tr>
<tr>
<td>Programmer's Checklist</td>
<td>30</td>
</tr>
<tr>
<td>C</td>
<td>30</td>
</tr>
<tr>
<td>COBOL</td>
<td>30</td>
</tr>
<tr>
<td><strong>Appendix A  MVS Batch Examples</strong></td>
<td>33</td>
</tr>
<tr>
<td>Sample Programs</td>
<td>34</td>
</tr>
<tr>
<td><strong>Appendix B  Rendezvous Installed Data Sets</strong></td>
<td>39</td>
</tr>
</tbody>
</table>
# Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>Related Documents</td>
<td>xi</td>
</tr>
<tr>
<td>Table 2</td>
<td>General Typographical Conventions</td>
<td>xiii</td>
</tr>
<tr>
<td>Table 3</td>
<td>Syntax Typographical Conventions</td>
<td>xiv</td>
</tr>
<tr>
<td>Table 4</td>
<td>TCP and UDP parameters in the TCP Profile</td>
<td>5</td>
</tr>
<tr>
<td>Table 5</td>
<td>General Variables</td>
<td>8</td>
</tr>
<tr>
<td>Table 6</td>
<td>Rendezvous Installation Checklist</td>
<td>11</td>
</tr>
<tr>
<td>Table 7</td>
<td>File Tailoring Variables for Installation JCL</td>
<td>14</td>
</tr>
<tr>
<td>Table 8</td>
<td>File Tailoring Variables</td>
<td>16</td>
</tr>
<tr>
<td>Table 9</td>
<td>Rendezvous Assembler Basis Code Segment Examples</td>
<td>34</td>
</tr>
<tr>
<td>Table 10</td>
<td>Rendezvous C Examples</td>
<td>34</td>
</tr>
<tr>
<td>Table 11</td>
<td>Rendezvous COBOL Examples</td>
<td>35</td>
</tr>
<tr>
<td>Table 12</td>
<td>Rendezvous General Examples</td>
<td>37</td>
</tr>
<tr>
<td>Table 13</td>
<td>Rendezvous General Procedures</td>
<td>38</td>
</tr>
<tr>
<td>Table 14</td>
<td>Installed Data Sets</td>
<td>39</td>
</tr>
</tbody>
</table>
Preface

TIBCO Rendezvous® is a messaging infrastructure product.

TIBCO is proud to announce the latest release of TIBCO Rendezvous®. This release is the latest in a long history of TIBCO products that leverage the power of the Information Bus® to enable truly event-driven IT environments. To find out more about how TIBCO Rendezvous and other TIBCO products are powered by TIB® technology, please visit us at www.tibco.com.

This manual describes how to install and configure the TIBCO Rendezvous software. It is primarily intended for system programmers who are involved in setting up and maintaining the system, but is also useful for applications programmers.

Topics

• Changes from the Previous Release of this Guide, page viii
• Related Documentation, page ix
• Typographical Conventions, page xiii
• Connecting with TIBCO Resources, page xvi
Changes from the Previous Release of this Guide

This section itemizes the major changes from the previous release of this guide.

Revised Content

The following have been revised:

- Updated the installation files to the current release. For details, see Obtaining the Installation Media on page 9.
- Updated the list of Rendezvous installed data sets. For details, see Table 14, Installed Data Sets, on page 39.
Related Documentation

This section lists documentation resources you may find useful.

TIBCO Product Documentation

The following documents form the Rendezvous documentation set:

Specific to z/OS

- **TIBCO Rendezvous for z/OS Installation and Configuration**
  Includes step-by-step instructions for installing Rendezvous software on z/OS platforms.

- **TIBCO Rendezvous for z/OS COBOL Reference**
  Detailed descriptions of each datatype and function in the Rendezvous COBOL API. Readers should already be familiar with the COBOL programming language, z/OS, as well as the material in *TIBCO Rendezvous Concepts*.

General

- **TIBCO Rendezvous Concepts**
  Read this book first. It contains basic information about Rendezvous components, principles of operation, programming constructs and techniques, advisory messages, and a glossary. All other books in the documentation set refer to concepts explained in this book.

- **TIBCO Rendezvous C Reference**
  Detailed descriptions of each datatype and function in the Rendezvous C API. Readers should already be familiar with the C programming language, as well as the material in *TIBCO Rendezvous Concepts*.

- **TIBCO Rendezvous C++ Reference**
  Detailed descriptions of each class and method in the Rendezvous C++ API. The C++ API uses some datatypes and functions from the C API, so we recommend the *TIBCO Rendezvous C Reference* as an additional resource. Readers should already be familiar with the C++ programming language, as well as the material in *TIBCO Rendezvous Concepts*.

- **TIBCO Rendezvous Java Reference**
  Detailed descriptions of each class and method in the Rendezvous Java language interface. Readers should already be familiar with the Java programming language, as well as the material in *TIBCO Rendezvous Concepts*.

- **TIBCO Rendezvous .NET Reference**
Detailed descriptions of each class and method in the Rendezvous .NET interface. Readers should already be familiar with either C# or Visual Basic .NET, as well as the material in TIBCO Rendezvous Concepts.

- **TIBCO Rendezvous COM Reference**
  Detailed descriptions of each class and method in the Rendezvous COM component. Readers should already be familiar with the programming environment that uses COM and OLE automation interfaces, as well as the material in TIBCO Rendezvous Concepts.

- **TIBCO Rendezvous Administration**
  Begins with a checklist of action items for system and network administrators. This book describes the mechanics of Rendezvous licensing, network details, plus a chapter for each component of the Rendezvous software suite. Readers should have TIBCO Rendezvous Concepts at hand for reference.

- **TIBCO Rendezvous Configuration Tools**
  Detailed descriptions of each Java class and method in the Rendezvous configuration API, plus a command line tool that can generate and apply XML documents representing component configurations. Readers should already be familiar with the Java programming language, as well as the material in TIBCO Rendezvous Administration.

- **TIBCO Rendezvous Installation**
  Includes step-by-step instructions for installing Rendezvous software on various operating system platforms.

- **TIBCO Rendezvous Release Notes**
  Lists new features, changes in functionality, deprecated features, migration and compatibility information, closed issues and known issues.
Third Party Documentation

Table 1 lists IBM publications you may find useful. You can find many of the IBM documents in these CD-ROM sets:

- Online Library Omnibus Edition MVS Collection
- Online Library Omnibus Edition z/OS Collection

Table 1  Related Documents

<table>
<thead>
<tr>
<th>Publication Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM RACF User’s Guide</td>
</tr>
<tr>
<td>IBM z/OS Security Server RACF User’s Guide</td>
</tr>
<tr>
<td>IBM RACF Messages</td>
</tr>
<tr>
<td>BM z/OS SecureWay Security Server RACF Messages</td>
</tr>
<tr>
<td>IBM RACF System Programmer’s Guide</td>
</tr>
<tr>
<td>IBM z/OS Security Server RACF System Programmer’s Guide</td>
</tr>
<tr>
<td>IBM MVS/ESA OpenEdition MVS User’s Guide</td>
</tr>
<tr>
<td>IBM z/OS UNIX System Services User's Guide</td>
</tr>
<tr>
<td>IBM MVS/ESA Planning: OpenEdition MVS</td>
</tr>
<tr>
<td>IBM z/OS UNIX System Services Planning</td>
</tr>
<tr>
<td>IBM TCP/IP for OpenEdition MVS Applications Feature Guide</td>
</tr>
<tr>
<td>IBM TCP/IP for MVS: Messages and Codes</td>
</tr>
<tr>
<td>IBM TCP/IP for MVS: Customization and Administration Guide</td>
</tr>
<tr>
<td>IBM TCP/IP for MVS: User’s Guide</td>
</tr>
<tr>
<td>IBM TCP/IP Performance Tuning Guide</td>
</tr>
<tr>
<td>IBM TCP/IP for MVS: Planning and Migration Guide</td>
</tr>
<tr>
<td>IBM ISPF Dialog Developer’s Guide and Reference</td>
</tr>
<tr>
<td>IBM z/OS ISPF Dialog Developer's Guide</td>
</tr>
<tr>
<td>IBM TSO/E CLISTS</td>
</tr>
<tr>
<td>IBM z/OS TSO/E CLISTs</td>
</tr>
</tbody>
</table>
Table 1  Related Documents (Cont’d)

<table>
<thead>
<tr>
<th>Publication Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM REXX/370 User’s Guide</td>
</tr>
<tr>
<td>IBM Online Library Omnibus Edition MVS Collection</td>
</tr>
<tr>
<td>IBM Online Library Omnibus Edition z/OS Collection</td>
</tr>
</tbody>
</table>
## Typographical Conventions

The following typographical conventions are used in this manual.

**Table 2  General Typographical Conventions**

<table>
<thead>
<tr>
<th>Convention</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>TIBCO_HOME</code></td>
<td>Many TIBCO products must be installed within the same home directory. This directory is referenced in documentation as <code>TIBCO_HOME</code>. The value of <code>TIBCO_HOME</code> depends on the operating system. For example, on Windows systems, the default value is <code>C:\tibco</code>.</td>
</tr>
<tr>
<td><code>ENV_HOME</code></td>
<td>Other TIBCO products are installed into an installation environment. Incompatible products and multiple instances of the same product are installed into different installation environments. An environment home directory is referenced in documentation as <code>ENV_HOME</code>. The default value of <code>ENV_HOME</code> depends on the operating system. For example, on Windows systems the default value is <code>C:\tibco</code>.</td>
</tr>
<tr>
<td><code>TIBRV_HOME</code></td>
<td>TIBCO TIBCO Rendezvous for z/OS installs into a version-specific directory inside <code>TIBCO_HOME</code>. This directory is referenced in documentation as <code>TIBRV_HOME</code>. The value of <code>TIBRV_HOME</code> depends on the operating system. For example on Windows systems, the default value is <code>C:\tibco\rv\8.4.0</code>.</td>
</tr>
</tbody>
</table>

**Code font**

Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example:

Use MyCommand to start the foo process.

**Bold code font**

Bold code font is used in the following ways:

- In procedures, to indicate what a user types. For example: Type `admin`.
- In large code samples, to indicate the parts of the sample that are of particular interest.
- In command syntax, to indicate the default parameter for a command. For example, if no parameter is specified, `MyCommand` is enabled:
  
  ```
  MyCommand [enable | disable]
  ```
Typographical Conventions

### Table 2  General Typographical Conventions (Cont’d)

<table>
<thead>
<tr>
<th>Convention</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>italic font</td>
<td>Italic font is used in the following ways:</td>
</tr>
<tr>
<td></td>
<td>• To indicate a document title. For example: See TIBCO FTL Concepts.</td>
</tr>
<tr>
<td></td>
<td>• To introduce new terms For example: A portal page may contain several</td>
</tr>
<tr>
<td></td>
<td>portlets. Portlets are mini-applications that run in a portal.</td>
</tr>
<tr>
<td></td>
<td>• To indicate a variable in a command or code syntax that you must</td>
</tr>
<tr>
<td></td>
<td>replace. For example: MyCommand PathName</td>
</tr>
<tr>
<td>Key</td>
<td>Key name separated by a plus sign indicate keys pressed simultaneously.</td>
</tr>
<tr>
<td>combinations</td>
<td>For example: Ctrl+C.</td>
</tr>
<tr>
<td></td>
<td>Key names separated by a comma and space indicate keys pressed one</td>
</tr>
<tr>
<td></td>
<td>after the other. For example: Esc, Ctrl+Q.</td>
</tr>
<tr>
<td></td>
<td>The note icon indicates information that is of special interest or</td>
</tr>
<tr>
<td></td>
<td>importance, for example, an additional action required only in</td>
</tr>
<tr>
<td></td>
<td>certain circumstances.</td>
</tr>
<tr>
<td></td>
<td>The tip icon indicates an idea that could be useful, for example, a</td>
</tr>
<tr>
<td></td>
<td>way to apply the information provided in the current section to</td>
</tr>
<tr>
<td></td>
<td>achieve a specific result.</td>
</tr>
<tr>
<td></td>
<td>The warning icon indicates the potential for a damaging situation,</td>
</tr>
<tr>
<td></td>
<td>for example, data loss or corruption if certain steps are taken or</td>
</tr>
<tr>
<td></td>
<td>not taken.</td>
</tr>
</tbody>
</table>

### Table 3  Syntax Typographical Conventions

<table>
<thead>
<tr>
<th>Convention</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ ]</td>
<td>An optional item in a command or code syntax. For example:</td>
</tr>
<tr>
<td></td>
<td>MyCommand [optional_parameter] required_parameter</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>chosen. For example, you can select only one of the following</td>
</tr>
<tr>
<td></td>
<td>parameters:</td>
</tr>
<tr>
<td></td>
<td>MyCommand para1</td>
</tr>
</tbody>
</table>
The following command requires two parameters, which can be either the pair `param1` and `param2`, or the pair `param3` and `param4`:

```
MyCommand {param1 param2} | {param3 param4}
```

In the next example, the command requires two parameters. The first parameter can be either `param1` or `param2` and the second can be either `param3` or `param4`:

```
MyCommand {param1 | param2} {param3 | param4}
```

In the next example, the command can accept either two or three parameters. The first parameter must be `param1`. You can optionally include `param2` as the second parameter. And the last parameter is either `param3` or `param4`:

```
MyCommand param1 [param2] {param3 | param4}
```
Connecting with TIBCO Resources

How to Join TIBCOmmunity

TIBCOmmunity is an online destination for TIBCO customers, partners, and resident experts, a place to share and access the collective experience of the TIBCO community. TIBCOmmunity offers forums, blogs, and access to a variety of resources. To register, go to http://www.tibcommunity.com.

How to Access All TIBCO Documentation

After you join TIBCOmmunity, you can access the documentation for all supported product versions here:

http://docs.tibco.com/TibcoDoc

How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, please contact TIBCO Support as follows.

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit this site:
  http://www.tibco.com/services/support
- If you already have a valid maintenance or support contract, visit this site:
  https://support.tibco.com

Entry to this site requires a user name and password. If you do not have a user name, you can request one.
Chapter 1  Introduction

This manual provides information concerning TIBCO Rendezvous® software for IBM z/OS systems and is primarily intended for system programmers and administration personnel who are responsible for installing and maintaining the software. Some of the information contained here may also be useful for application programmers involved in creating or modifying programs that work in conjunction with this product. We recommend that you reading through this manual before attempting to install the product, and keep a copy handy for future reference.

Topics

- Overview, page 2
- Hardware Requirements, page 3
- System Software Requirements, page 4
Overview

Rendezvous software allows applications running on IBM z/OS systems to pass information between Rendezvous applications that communicate through Rendezvous message passing facilities. This message passing is implemented through one or more long running batch tasks (described as network daemons in other TIBCO publications) that receive Rendezvous messages from the network. These daemons can be run under the traditional z/OS MVS environment, or alternatively can be run in the Unix System Services USS (sometimes referred to as Open Edition) environments. Rendezvous daemons filter incoming messages according to their subjects and route these messages to applications (that register interest in a subject or group of subjects).

Applications that incorporate Rendezvous API function calls into their processing logic must supply the Rendezvous API library during the pre-link edit phase of the compilation. Application programs can then publish and subscribe to messages through the batch tasks running on z/OS to other Rendezvous applications anywhere in the network.

Please refer to *TIBCO Rendezvous Concepts* for a complete description of Rendezvous operations.
Hardware Requirements

Before you can begin using the Rendezvous product, your system must have the appropriate network hardware to support IBM TCP/IP (for example, an ethernet adapter card) should be installed and properly configured.

The network should be broadcast-capable. For messages to be seen in the local subnet, the network interface must support UDP broadcast or multicast. If your network interface does not support UDP broadcast, the Rendezvous software will not function to its full capability.

Storage Requirements

Rendezvous for the MVS environment requires approximately 350 cylinders of Direct-Access Storage Device (DASD) space. This spaces calculation is based on an IBM 3390 DASD device.
System Software Requirements

The Rendezvous software runs as an MVS batch task or as a Unix Systems Services (USS) process. You can install both the MVS and USS versions from the installation materials supplied by TIBCO Software Inc. Both products are supported on z/OS 1.6 and later (64-bit).

MVS Environment

Before installing Rendezvous software in an MVS environment, you must ensure that the following software components are installed and properly configured:

TCP/IP

Rendezvous z/OS only supports the IBM TCP/IP environment. A TIBCO messaging client for z/OS is expected to be installed and operated within a previously existing, fully functional and integrated network. At a minimum, this includes the following:

- The TCP/IP protocol stack within the z/OS Communications Server.
- A DNS server that provides the host name and host address of the TIBCO Rendezvous daemon (if the daemon is not running on the same host).
- Unrestricted network connectivity between the DNS, the TIBCO Rendezvous Client, and the TIBCO Rendezvous daemon (if the daemon is not running on the same host).

Customers running z/OS version 1.6 and above should be running a properly configured and customized RESOLVER address space.

Customers should review their TCPCONFIG parameters to ensure that they meet the installation’s needs. If the customer chooses to run the TIBCO messaging client non-APF authorized, the RESTRICTLOWPORTS parameter should be used. If customers want to control TCP buffering, to limit storage usage, or to support large bandwidth devices, the TCPSENDBFRSIZE, TCPRCVBUFFRSIZE, and TCPMAXRCVBUFFRSIZE parameters may be needed. TIBCO Software typically recommends that customers override the IBM default of 16K by setting TCPRCVBUFFRSIZE and TCPSENDBFRSIZE to a minimum value of 48K.
Next, you must make sure that the following required or recommended TCP and UDP parameters are appropriately set in the TCP Profile:

Table 4  TCP and UDP parameters in the TCP Profile

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCPCONFIG</td>
<td>RESTRICTLOWPORTS</td>
</tr>
<tr>
<td>TCPSENDB</td>
<td>48K</td>
</tr>
<tr>
<td>TCPRCVB</td>
<td>48K</td>
</tr>
<tr>
<td>UDPCONFIG</td>
<td>RESTRICTLOWPORTS</td>
</tr>
<tr>
<td>UDPCHKSUM</td>
<td></td>
</tr>
<tr>
<td>UDPQUEUELIMIT</td>
<td></td>
</tr>
<tr>
<td>UDPSENDB</td>
<td>32768</td>
</tr>
<tr>
<td>UDRPCVB</td>
<td>32768</td>
</tr>
</tbody>
</table>

**LE**

LE is the language environment. The C language and/or COBOL for MVS compilers are required for application development under this environment.

IBM TCP uses the OE socket interface. Therefore, all user id’s running Rendezvous daemons or client programs must have an OMVS UID and GID. Define these values in the OMVS segment of the USERID and GROUP records in RACF.

For more information, please refer to User Profile Requirements on page 22.

**SSL**

Rendezvous z/OS supports the following certificate formats:

- PEM (Privacy Enhanced Mail)
- PKCS#12

For details on SSL usage in Rendezvous, see TIBCO Rendezvous Concepts and TIBCO Rendezvous Administration.
Chapter 2  Installation

This chapter describes how to install the TIBCO Rendezvous z/OS software.

Topics

- Before Installation, page 8
- Distribution Media and Contents, page 9
- Checklist for Rendezvous Installation, page 11
- Uploading the Software, page 12
- Rendezvous MVS Installation Procedure, page 14
- Verifying the Rendezvous Installation, page 17
- Optional Installation, page 19
Before Installation

Table 5 provides a checklist of variables and data set HLQs (High Level Qualifiers) that you should obtain before performing the installation process. All Rendezvous supplied JCL and procedures have a standard naming convention. You must replace TIBCO site-specific installation-defined variables such as JCL SET statements, procedure overrides, SYSIN data and system data set HLQ with user values. You must also replace TIBCO Software variables before the JCL's can be submitted for execution.

It is recommended that you make a copy of the General Variables table below and then review and complete the entries. Use the User Value column to insert appropriate values for your site.

Here is a description of the table column headings:

The Description column provides instructional information.

The Variable column supplies the variable name, as they will appear in the provided JCL(S) and/or the member names. There may be several instances of the same variable within a specific member.

The User Value column is left blank so you can insert the appropriate replacement value.

Table 5 General Variables

<table>
<thead>
<tr>
<th>Description</th>
<th>Variable</th>
<th>User Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obtain a High Level Qualifier for your Rendezvous libraries. Typically,</td>
<td>#USERHLQ</td>
<td></td>
</tr>
<tr>
<td>TIBCO.RVB is used.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Obtain a hard disk Volume Serial where the Rendezvous files are to be</td>
<td>#USERVOL</td>
<td></td>
</tr>
<tr>
<td>installed.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Replace these &lt;JOBNAME&gt; variables with user defined values.</td>
<td>#JOBNAME</td>
<td></td>
</tr>
</tbody>
</table>
Distribution Media and Contents

The most convenient way to get a copy of the Rendezvous software is to download it directly from the TIBCO web site, or obtain the product on a CD. Rendezvous mainframe software is distributed in IBM XMIT format.

Product distribution for IBM platforms that do not have DFSMSdss are supplied only upon request. Please request a downloadable package from TIBCO Support (see Connecting with TIBCO Resources on page xvi).

Complete Replacement Package

By default, the Rendezvous software is delivered as a non SMP/E stand-alone, base function package. Subsequent modifications and updates are version and product dependent. Therefore, a complete product replacement is preformed with each installation.

Component Code

The component code for the TIBCO Rendezvous products is RVB. This component code is registered with the IBM product codes division.

Obtaining the Installation Media

To obtain the Rendezvous software, you can download it from the web or you can request a cartridge.

Download from the Web Site

To download the Rendezvous software from the TIBCO web site, follow these steps:

- Contact TIBCO Software Inc. for a password, directory information, and so forth.
- Connect to the TIBCO web site with the required information.
- Download the appropriate files, which are in the standard IBM XMIT format.
If you do not want to download the software over the network, you can obtain a CD containing the Rendezvous software and load it directly onto your system. The minimum time period for delivery of a CD is five working days. Contact TIBCO Support and request the desired media for your environment. For details, see Connecting with TIBCO Resources on page xvi.

### Requesting a CD

If you do not want to download the software over the network, you can obtain a CD containing the Rendezvous software and load it directly onto your system. The minimum time period for delivery of a CD is five working days. Contact TIBCO Support and request the desired media for your environment. For details, see Connecting with TIBCO Resources on page xvi.
Checklist for Rendezvous Installation

Table 6 provides a checklist of variables or required items that should be obtained before performing the installation process. The data set suffix and member names are included as reference to where they occur.

Here is a description of the table column headings:

- The **Description** column provides instructional information.
- The **Suffix** column lists the suffix of the data set containing the reference.
- The **Member** column contains the variable in question. There may be several instances of the same variable within a specific member.

### Table 6  Rendezvous Installation Checklist

<table>
<thead>
<tr>
<th>Description</th>
<th>Suffix</th>
<th>Member</th>
</tr>
</thead>
<tbody>
<tr>
<td>If you are transferring the Rendezvous installation files from a Server, obtain the IP address or Server name for the FTP process.</td>
<td>INSTALL</td>
<td>RVB1FTP</td>
</tr>
<tr>
<td>Obtain the data set and or member name where your site FTPDATA information resides.</td>
<td>INSTALL</td>
<td>RVB1FTP</td>
</tr>
<tr>
<td>Obtain the data set and or member name where your site TCPDATA information resides.</td>
<td>PROC</td>
<td>RVD, RVRD, RVSD, RVSRD, RVA, and RVCACHE</td>
</tr>
</tbody>
</table>
Uploading the Software

If you have acquired the Rendezvous software by downloading it from the TIBCO web site rather than via a cartridge, you will need to upload the Rendezvous software to the MVS host system using the FTP file transfer utility. Following is a sample JCL that details how to perform this file transfer:

Initial Installation

Sample JCL for an Upload

The sample JCL illustrates the process, using IBM's FTP to copy the Rendezvous installation file(s) from a LAN server or PC to the MVS host.

```plaintext
//<JOBNAME> JOB (&SYSUID),'RENDEZVOUS FTP',CLASS=A
//*
// SET FTPADDR=10.10.2.99   < IP Address or Host Name
//*
//FTP EXEC PGM=FTP,REGION=0M,PARM='&FTPADDR (TIMEOUT 20'
//SYSPRINT DD SYSOUT=* 
//OUTPUT DD SYSOUT=* 
//SYSTCPD DD DISP=SHR,DSN=TCPIP.PROFILE(FTPDATA)  < Host Member 
//INPUT DD *
<lan-userid>
<lan-password>
cd /<lan-file-dir>
binary
LOCSITE RECFM=FB LRECL=80 BLKSIZE=27920 TR PRI=60 SEC=15
get #srvhlq.xmit.inst' <USERHLQ>.XMIT.INST' (REPLACE quit 
/*
//
```
Receive the Initial Installation File

This step always needs to be performed during a Rendezvous installation on the MVS Host.

The <USERHLQ>.XMIT.INST file contains JCL that is required to receive other files and to create the appropriate libraries for Rendezvous. To complete this process, the TIBCO supplied variables must be substituted with the appropriate site-specific user values.

Follow these steps:

1. From the ISPF Command Shell Panel, type the following command to receive a file containing the JCL required to begin a first-time installation:

   receive inda('<USERHLQ>.XMIT.INST')

   TSO will prompt with the following:

   INMR906A Enter restore parameters or 'DELETE' or 'END' +

2. Supply the following:

   DA(''<USERHLQ>.INSTALL')

   After a successful receive, the following message is displayed:

   INMR001I Restore successful to dataset '<USERHLQ>.INSTALL'

   The data set name supplied in step 2 is output and populated with the members that are required in order for you to continue with the installation.
Rendezvous MVS Installation Procedure

The following steps are required to install the MVS components for Rendezvous. All members reside in the `<USERHLQ>.INSTALL` dataset.

Edit the data sets and selected members listed in the steps below. Substitute the variables, follow any instructions that are contained in the member and then submit the JCL for execution.

**STEP 1:** Make CLIST temporary file

Change the `#JOBNAME`, `#USERHLQ`, `#USERVOL` statement variables according to your site requirements.

Member: RVB$1MKT

**STEP 2:** Execute File Tailoring clist for installation JCL.

If an error is made during input, press the attention key (PA1) and start again at Step 1.

Member: RVB$2EX1 (EX member)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Default and structure</th>
<th>User Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>USERHLQ</td>
<td>TIBCO.RVB.VxRxMx</td>
<td></td>
</tr>
<tr>
<td>VOLSER</td>
<td>TIB999</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Volume Serial</td>
<td></td>
</tr>
<tr>
<td>JOBNAME</td>
<td>RVB84</td>
<td>First 5 characters are used</td>
</tr>
<tr>
<td>JOBCLASS</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>MSGCLASS</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

**STEP 3:** Upload all installation files to z/OS MVS.

Member: For DFSMSdss sites, use RVB$3FT.

For non DFSMSdss sites, use RVB$3FT@.

TIBCO Rendezvous for z/OS Installation and Configuration
STEP 4a: **Rendezvous configuration files**
This member will allocate the Rendezvous daemons’ configuration files.

**Member:** RVB$4ACF

STEP 4b: **Define and allocate Rendezvous product libraries**
Only to be performed on initial installation.

**Member:** RVB$4ALL

STEP 5: **Convert product installation files to appropriate datasets**
The installation can be restarted multiple times at this step if required.

**Member:** For DFSMSdss sites, use RVB$5RC.
For non DFSMSdss sites, use RVB$5RC@.
The following table defines the user substitution variables within JCL and parameter members that are required during the file tailoring process. Ensure that you have all the necessary information before executing the next step (Step 6).

Table 8  File Tailoring Variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Default and Structure</th>
<th>User Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Assembler Libraries HLQ</td>
<td>ASM</td>
<td>*.SASMMAC2</td>
</tr>
<tr>
<td>COBOL for MVS Libraries HLQ</td>
<td>IGY</td>
<td>*.SIGYCOMP</td>
</tr>
</tbody>
</table>

STEP 6: Execute File Tailoring clist to update the product supplied variables.

If an error is made during input, press the attention key (PA1) and execute the clist again.

Member: RVB$6EX2 (EX member)

STEP 7: Populate the Rendezvous data sets with all the update and downloaded members.

Member: RVB$7UPD
Verifying the Rendezvous Installation

This section provides information that allows you to tests verify that Rendezvous components have been correctly installed and configured.

Step I1: Network daemon startup JCL

Replace the Rendezvous variables with the appropriate user values. This member will start the Rendezvous daemon.

Data Set: <USERHLQ>.JCL
Member: RVBIRVD

After you have started the network daemon, you need to determine if it is operating correctly by performing a sample publish-and-subscribe operation.

You can use an http browser interface as a method for verifying that the daemon is running before continuing with the following steps.

For example, specify the following:

http://hostname_or_IP_address:7580/

Please refer to TIBCO Rendezvous Administration for a more complete description of the http browser interface.

Step I2: Reliable messaging subscriber startup JCL

Replace the Rendezvous variables with the appropriate user values. This member will initiate a Rendezvous listener that receives messages from the publisher.

Data Set: <USERHLQ>.JCL
Member: RVBILSN

Step I3: Reliable messaging publisher startup JCL

Replace the Rendezvous variables with the appropriate user values. This member will initiate a Rendezvous sender that publishes a message to an already executing listener.

Data Set: <USERHLQ>.JCL
After you have examined the output of the previous execution, you may consider testing certified messaging (RVCM) by performing the IVPs for RVCM publish and subscribe.

**Step I4: Certified messaging (CM) ledger file(s) allocation JCL**

Replace the Rendezvous variables with the appropriate user values. This member will allocate the Rendezvous certified messaging ledger files.

Data Set: `<USERHLQ>.JCL`

Member: RVGDEFLF

**Step I5: Certified messaging (CM) subscriber startup JCL**

Replace the Rendezvous variables with the appropriate user values. This member will initiate a Rendezvous certified messaging (CM) listener that receives messages from the certified messaging (CM) publisher. The listener sends the confirmation back to the publisher.

Data Set: `<USERHLQ>.JCL`

Member: RVBICMLS

**Step I6: Certified messaging (CM) publisher startup JCL**

Replace the Rendezvous variables with the appropriate user values. This member will initiate a Rendezvous certified messaging (CM) sender that publishes messages.

Data Set: `<USERHLQ>.JCL`

Member: RVBICMSN
Optional Installation

Rendezvous sample programs are supplied as source and executable modules as part of the installation. However, users may desire to change the source code to perform additional functions. JCLs and procedures have been supplied to compile the sample programs that normally have to be modified to be used within a user’s environment.

Edit the data sets and selected members listed in the steps below. Substitute the variables, follow any instructions that are contained in the member and then submit it for execution.

In addition, this section describes an alternate UNIX installation.

Compiling C Programs

Step 1a: C language options file

This member is the Options File used for Rendezvous C compiles that needs to include members from the system C and Rendezvous libraries. Substitute the appropriate values and installation variables that are located within this member.

Data Set: <USERHLQ> . CNTL
Member: OPTFRVBC

Step 1b: Rendezvous C language compilation JCL

The JCL listed below uses a procedure, which may have to be modified so that it conforms to your site’s requirements and executes correctly.

Data Set: <USERHLQ> . JCL
Member: RV$3CPL
Compiling COBOL Programs

**Step 1c: Rendezvous COBOL compilation JCL**

The JCL listed below uses a procedure, which may have to be modified so that it conforms to your site's requirements and executes correctly. This member will compile COBOL supplied sample program(s).

Data Set: <USERHLQ>.JCL
Member: RV$CCPL

Alternate UNIX installation

You can install the standard UNIX-based z/OS install package as an alternative to the standard Rendezvous for z/OS software. However, this method installs software only for use in a UNIX System Services environment.

**Installation Files for z/OS USS Version 1.6 and Later**

Obtain this installation package:

```
TIB_rv_8.4.0_zos16uss_zSeries.tar.gz
```

The installation procedure is identical to other UNIX installations. For details, see the UNIX installation sections in *TIBCO Rendezvous Installation*.

**Rendezvous Java Libraries**

Rendezvous Java libraries are available for the IBM z/OS operating system in a UNIX System Services environment. These libraries require IBM Java 1.5.0. To check the Java support level on your host computer, execute the following command:

```
../J5.0/bin/java -version
```

For z/OS platforms, compare the command output with the following requirements:

```
java version "1.5.0"
Java(TM) 2 Runtime Environment, Standard Edition (build pmz31devifx-20071025 (SR6b))
IBM J9 VM (build 2.3, J2RE 1.5.0 IBM J9 2.3 z/OS s390-31 j9vmmz3123-20071007 (JIT enabled)
J9VM - 20071004_14218_bHdSMr
JIT - 20070820_1846ifx1_r8
GC - 200708_10)
JCL - 20071025
```

*TIBCO Rendezvous for z/OS Installation and Configuration*
This chapter describes administration and use of Rendezvous software using MVS procedures.

Before doing any of these administrative procedures, you must have completed the installation of the MVS version of Rendezvous software, and executed the IVP steps successfully, as described in Verifying the Rendezvous Installation on page 17.

Topics.

- Starting Daemons and Applications, page 22
- Programmer's Checklist, page 30
Starting Daemons and Applications

User Profile Requirements

Any userid used to run the Rendezvous application must have a valid RACF OMVS segment defined.

- If you are starting Rendezvous as a batch job, verify that the submitter's userid has a valid OMVS segment defined.

- If you are starting Rendezvous as a started task, verify that the userid assigned by the RACF STARTED class or ICHRIN03 started procedures table has a valid OMVS segment. Verification can be done by using the RACF LISTUSER command.

TSO LISTUSER TIBCO01 OMVS NORACF

USER=TIBCO01

OMVS INFORMATION
UID= 0000012345
HOME= /tibco/tibco01
PROGRAM= /bin/sh
CPUTIMEMAX= NONE
ASSIZEMAX= NONE
FILEPROCMA X= NONE
PROCUSERMAX= NONE
THREADSMAX= NONE
MMAPAREAMAX= NONE
***

TIBCO Software Inc. recommends running Rendezvous daemons with a userid without resource limits specified in the OMVS segment, as shown in the example userid TIBCO01. This allows Rendezvous to run within the limits defined to the overall system in BPXPRMxx. There are no special requirements for UID, HOME, or PROGRAM in order to run Rendezvous.

If an existing userid does not have an OMVS segment, you can define one by using the RACF ALTUSER command. For more information regarding the use of the various RACF commands and the setup of an OMVS segment, refer to "SecureWay Security Server RACF Command Language Reference" or equivalent documentation published by IBM.
If the installation is using the FACILITY class profile BPX.DEFAULT.USER instead of defining OMVS segments for every userid, the userid defined in the FACILITY class profile BPX.DEFAULT.USER should be examined to ensure that it does not have resource limits defined if the user submits RV jobs. For more information about managing the FACILITY class profile BPX.DEFAULT.USER, see "SecureWay Security Server RACF Security Administrator’s Guide" documentation published by IBM.

JCL and Procedures

After installation, Rendezvous daemons and example programs reside in the <USERHLQ>.LOAD load library. Rendezvous daemons can be started from JCL or catalogued procedures that have been supplied during the installation processes. JCL has been provided to execute the example programs.

Refer to Appendix A on page 33 for a list example programs and JCL members.

We recommend that you maintain catalogued JCL procedures to run the Rendezvous daemons, and define them as started tasks, so the system operator at the master console can start and monitor them.

The following steps are required to define daemons as started tasks.

1. First, you need to copy the RVD / RVRD procedures from <USERHLQ>.PROC into a procedure library that is defined in the JES2 startup.

2. Modify the variables <USERHLQ> and the data set name that contains the TCPDATA member.

3. The daemon procedures can now be started from the MVS console.

The daemon procedures are long-running applications. The Rendezvous daemons support operational commands to terminate them. For the syntax and usage of these commands, refer to z/OS MVS System Commands.

Examples:

F < [jobname.]identifier >,SHUT
P < [jobname.]identifier >
(Stop an active daemon)

When you start a daemon, the following messages are written to the system log:

RVB1000I Rendezvous Daemon Started.
RVB1001I Rendezvous Daemon ports. Listen:7500 Http:7580

When you issue the command to stop a daemon, you will see the following message in the system log:

RVB1009I Rendezvous daemon shutdown in progress
Daemon Tasks (Non-Swappable and No CPU Limit)

We recommend that you run the Rendezvous daemon programs set as non-swappable and no CPU time limit, especially in a production environment. For a Rendezvous daemon program to be made non-swappable and no CPU time limit, you need to add the following to the system parameter library.

'SYSPARMLIB(SCHEDxx)'
PPT PGMNAME(daemon_program_name)
NOSWAP
SYST

Additionally, the Rendezvous daemon programs load module must reside in an APF authorized library. When the Rendezvous daemon program has been restarted it should be running as NON-SWAPPABLE (NS).

Sites with Multiple Stacks

When the system uses multiple TCP/IP stacks, insert the following JCL statement before the Rendezvous application execution steps:

//TCAFF EXEC PROC=TIBTCAFF,STACK=TCPIP-STACK-NAME

This statement identifies the stack that the application will use.

Time Zone

The daemon receives timestamps from the z/OS LE environment. If you have not configured the LE environment with the correct time zone, then messages from the daemon report time in GMT format.

You can dynamically set the time zone by setting the TZ environment variable in the daemon’s startup JCL or PROC. Use either of the following examples as a model (adjusting for your actual time zone):

//RVD EXEC PGM=RVD,PARM='ENVAR("TZ=GMT-7"),/&TIBPARM'
//RVD EXEC PGM=RVD,PARM='ENVAR("TZ=PST8PDT"),/&TIBPARM'

The first example calculates the time zone as an offset from GMT (either + or -). The second example uses the time zone name.

You can find the appropriate time zone values in UNIX System Services Command Reference, Setting the Local Time Zone with the TZ Environment Variable.
Case Sensitivity in Input Parameters

Rendezvous program parameters are case sensitive. Be sure to turn CAPS OFF when editing JCL and parameter members that are edited and used during execution of Rendezvous daemons and applications.

For example, the Rendezvous daemon (rvd) accepts the -listen parameter. The parameter must be in lower case letters.

Any application program that uses Rendezvous API calls must have the correct case when specifying input to the API call.

JCL Statements for Daemons

The following models illustrate the general form of the JCL EXEC statement for any Rendezvous daemon or agent (rvd, rvsd, rurd, rvsrd, rva, or rvcache):

```
//RVD EXEC PGM=RVD,PARM='-no-multicast'
//STEPLIB DD DISP=SHR,DSN=<USERHLQ>.LOAD
//TIXFILE DD DISP=SHR,DSN=<USERHLQ>.CNTL
```

or:

```
//RVRD EXEC PGM=RVRD,PARM='-store "DD:RVRDCFG"'
//STEPLIB DD DISP=SHR,DSN=<USERHLQ>.LOAD
//TIXFILE DD DISP=SHR,DSN=<USERHLQ>.CNTL
//RVRDCFG DD DISP=SHR,DSN=<USERHLQ>.CFG.RVRD
```

The case-sensitive parameter -no-multicast specifies that the daemon disables multicast (and broadcast) communication. It changes daemon behavior in the following ways:

- When a client sends a message to a public subject, the daemon does not multicast it (nor broadcast it) to the network.
- When a routing daemon receives multicast or broadcast messages from the network, it does not forward them to other daemons within the local network.

When multicast communication is disabled, daemons continue to operate as follows:

- Point-to-point messages continue to flow in both directions between clients and the network.
- All messages (including public subjects) flow among all the clients of the daemon.
- All messages (including public subjects) flow in both directions between local clients of a routing daemon and the daemon’s neighbors.
The **TIXFILE** DD statement specifies the location of the z/OS Rendezvous license ticket file.

The **STEPLIB** DD statement specifies the location of the Rendezvous daemon.

Daemons other than 'RVD' can take the '-store' parameter to store configuration data. The installation allocates the configuration files for the daemons, (for example, "<USERHLQ>.CFG.RVRD"). The configuration file must reside on a single DASD volume.

The daemons configuration files are opened internally with a C function call, `fopen()`. The qualifier(s) referenced in the '-store' parameter does not require that the standard "//" be before the **DD:RVBCFG**.

Example:

```c
PARM=''-store "DD:RVRDCFG"'
```
Adding a Certificate

The following models illustrate the general form for specifying a certificate file when entering the certificate through the HTTP interface:

- To specify as a z/OS dataset: `'HLQ.SLQ(MEMB)'`
- To specify as a DDNAME: `DD:DDNAME`
- To specify as a USS file: `/dir/dir/file.ext`

Certificate files have the following data set restrictions:

- z/OS PEM certificate files must be EBCDIC and in 80 record length data sets.
- z/OS PK12 certificate files must be in a dataset with record length greater than 80.
- USS PEM certificate files can be ASCII or EBCDIC.

JCL Statements for Applications

This model illustrates the general form of JCL for running any Rendezvous program (that is, any program that calls Rendezvous API functions).

**Optional**

When the system uses multiple TCP/IP stacks, insert the following JCL statement, which identifies the stack that the application will use:

```
//SYSTCPD DD DISP=SHR,DSN=your.TCPDATA.file
```

**General Form**

```
//RVPGM EXEC PGM=<RVPGM>,PARM='<RV Transport parameters>'
//STEPLIB DD DISP=SHR,DSN=your.user.loadlib
```

**Alternate Form**

Consider this example from `<USERHLQ>.JCL(RVBCLSN)`:  
```
//RVLISTEN EXEC PGM=RVBCLSN,PARM='rv.test.subject'
//STEPLIB DD DISP=SHR,DSN=your.user.loadlib
```

The program uses the default transport settings and the user-supplied parameter `rv.test.subject` is a substitution variable.
JCL Statements for Certified Messaging (RVCM)

CM Ledger File - Allocation characteristics

When executing a RVCM program that uses the disk resident ledger file, you may use the RVGDEFLF JCL member as an example to allocate one or more ledger files. The ledger file must reside on a single DASD volume, due to Language Environment restrictions in the support of multi-volume data sets.

If you allocate a ledger file using another method, refer to this member for the correct DCB information.

Except for the DSNTYPE, RECFM and LRECL parameters all other values can be changed to suit your applications RVCM ledger requirements.

For MVS batch, a ledger file that exists on disk must be defined as a PDS. Failure to adhere to this requirement will result in an error condition when executing Rendezvous example certified messaging programs.

From <USERHLQ>.JCL(RVGDEFLF):

```plaintext
// * ALLOCATE THE TIBCO RVCM SENDER'S LEDGER DATA SET
// *
//ALLOCAT1 EXEC PGM=IKJEFT01, DYNAMNBR=300
//SYSTSPRT DD SYSOUT=* 
//SYSTSIN DD *
ALLOCATE                        -
   DSNAME( '<USERHLQ>.CMLEDGER.SENDER01') -
   NEW CATALOG -
   RECFM(F) -
   LRECL(512) -
   BLKSIZE(6144) -
   DSORG(PS) -
   NEW CATALOG -
   SPACE(5,1) CYL -
   DIR(0) -
   UNIT(3390) 
/*
// * ALLOCATE THE TIBCO RVCM LISTENER'S LEDGER DATA SET
// *
//ALLOCAT2 EXEC PGM=IKJEFT01, DYNAMNBR=300
//SYSTSPRT DD SYSOUT=* 
//SYSTSIN DD *
ALLOCATE                        -
   DSNAME( '<USERHLQ>.CMLEDGER.LISTEN01') -
   NEW CATALOG -
   RECFM(F) -
   LRECL(512) -
   BLKSIZE(6144) -
   DSORG(PS) -
```
CM Ledger File - JCL **ddname** specification

When using a disk file ledger, an application input parameter with the following format is required:

```
[-ledger "//DD:ddname"]
```

where **ddname** is the name of an existing **DDNAME** file referenced in your JCL jobstream.

The default is NULL. A NULL value causes a non-persistent, memory-based ledger file to be used.

The following ledger file example shows use of the **ddname** for Listeners and Senders:

```
*--------------- START OF NEXT EXAMPLE ------------------------
/**
/* ** CM Ledger file (for Listener) example
/**
//RUNCCMLC EXEC PGM=RVBCCMLC,
  PARM='-ledger "//DD:RVCMLFL1" rvcm-mvs RVCM-MVS'
/**
//STEPLIB DD DISP=SHR,DSN=&USERHLQ..LOAD
//RVCMLFL1 DD DISP=SHR,DSN=&USERHLQ..CMLEDGER.LISTEN01
/**
**------------------ END OF NEXT EXAMPLE ------------------------
/**
/* ** CM Ledger file (for Sender) example
/**
//RUNCCMSN EXEC PGM=RVBCCMSN,
  PARM='-ledger "//DD:RVCMLFS1" RVCM-MVS "RVCM Test Message "'
/**
//STEPLIB DD DISP=SHR,DSN=&USERHLQ..LOAD
//RVCMLFS1 DD DISP=SHR,DSN=&USERHLQ..CMLEDGER.SENDER01
/**
```

CM Examples-Listener and Sender programs

To initiate the RVCM Listener and Sender example applications, submit JCL members.

- For COBOL, use RVBCCMLC / RVBCCMLS, then RVBCCMSN.
- For C, use RVB3CMLS, then RVB3CMSN.
Programmer's Checklist

All application programmers in an MVS environment should make the following checks.

- **User options file**
  A Language Environment user options object is supplied as part of the installation. When compiling Rendezvous enabled programs, it is required to include the object during the link-edited step. The installation-supplied procedures already cater for this.

  Including the supplied user options object ensures that `POSIX(ON)` is set at run-time and memory allocations are optimized for applications using the Rendezvous API.

- **Execution**
  The Rendezvous API uses the TCP/IP socket interface. Most installations require a `SYSTCPD DD` statement in the execution JCL. Contact your TCP specialist for details.

**C**

C application programmers in an MVS environment can use this checklist.

- **Code**
  Include the appropriate Rendezvous C header files. (See the section, Include These Header Files, *TIBCO Rendezvous C Reference*.)

- **Compile/Pre-link/Link**
  Sample JCL for compiling C programs is member RV$3CPL.
  
  — Application source files normally reside in the.C dataset.
  
  — Requirements are specified in the applicable compile procedure.

**COBOL**

COBOL application programmers in an MVS environment can use this checklist.

- **Code**
  Include the appropriate Rendezvous COBOL copybooks.

- **Compile/Pre-link/Link**
Sample JCL for compiling COBOL programs is member RV$CCPL.

Application source files normally reside in the .COBOL dataset.

Requirements are specified in the applicable compile procedure.
Appendix A  MVS Batch Examples

This appendix describes members required to build and run MVS Batch transactions that use the Rendezvous Reliable API.

Topics

- Sample Programs, page 34
Sample Programs

The following are members related specifically to MVS Batch.

These Assembler examples are not ready for compilation and they are not executable.

Table 9  Rendezvous Assembler Basis Code Segment Examples

<table>
<thead>
<tr>
<th>Suffix</th>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASM</td>
<td>RVBACMLS</td>
<td>Sample code segments of RV CM Listener</td>
</tr>
<tr>
<td></td>
<td>RVBACMSN</td>
<td>Sample code segments of RV CM Sender</td>
</tr>
<tr>
<td></td>
<td>RVBALS</td>
<td>Sample code segments of RV Basic Listener</td>
</tr>
<tr>
<td></td>
<td>RVBASND</td>
<td>Sample code segments of RV Basic Sender</td>
</tr>
</tbody>
</table>

Table 10  Rendezvous C Examples

<table>
<thead>
<tr>
<th>Suffix</th>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>RVB3CLNT</td>
<td>Sample of RV RPC client-side</td>
</tr>
<tr>
<td></td>
<td>RVB3CML</td>
<td>Sample of RV CM Listener</td>
</tr>
<tr>
<td></td>
<td>RVB3CMSN</td>
<td>Sample of RV CM Sender</td>
</tr>
<tr>
<td></td>
<td>RVB3DISP</td>
<td>C program of RV multiple dispatcher threads</td>
</tr>
<tr>
<td></td>
<td>RVB3DQLS</td>
<td>C program of generic DQ RV subscriber</td>
</tr>
<tr>
<td></td>
<td>RVB3FTMN</td>
<td>Sample of RV FT Group Monitor.</td>
</tr>
<tr>
<td></td>
<td>RVB3FTTM</td>
<td>Sample of RV FT Timestamp example</td>
</tr>
<tr>
<td></td>
<td>RVB3INIV</td>
<td>C program get initial values out of rvcache</td>
</tr>
<tr>
<td></td>
<td>RVB3LSN</td>
<td>Sample of RV Basic Listener</td>
</tr>
<tr>
<td></td>
<td>RVB3PTY</td>
<td>C program of queues, groups and priorities</td>
</tr>
<tr>
<td></td>
<td>RVB3SN</td>
<td>Sample of RV Basic Sender</td>
</tr>
</tbody>
</table>
Table 10  Rendezvous C Examples

<table>
<thead>
<tr>
<th>Suffix</th>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RVB3SNMF</td>
<td>Sample of RV multi-field message publisher</td>
</tr>
<tr>
<td></td>
<td>RVB3SRVR</td>
<td>Sample C RV RPC server-side</td>
</tr>
<tr>
<td></td>
<td>RVB3USRT</td>
<td>C program of RV custom message types</td>
</tr>
<tr>
<td></td>
<td>RVPERFM</td>
<td>Sample of RV performance tester - master side</td>
</tr>
<tr>
<td></td>
<td>RVPERFS</td>
<td>Sample of RV performance tester - slave side</td>
</tr>
<tr>
<td>H</td>
<td>TIBRV</td>
<td>This header file for RV C API</td>
</tr>
<tr>
<td>CM</td>
<td></td>
<td>This header file for the RV certified message delivery and distributed queue C API.</td>
</tr>
<tr>
<td>FT</td>
<td></td>
<td>This header file for the RV fault tolerance C API.</td>
</tr>
<tr>
<td>JCL</td>
<td>RVB3CMLS</td>
<td>JCL to execute RVCM Listener</td>
</tr>
<tr>
<td></td>
<td>RVB3CMSN</td>
<td>JCL to execute RV CM Sender</td>
</tr>
<tr>
<td></td>
<td>RVB3LSN</td>
<td>JCL to execute RV Basic Listener</td>
</tr>
<tr>
<td></td>
<td>RVBCLSMF</td>
<td>JCL to execute RV multi-field Listener</td>
</tr>
<tr>
<td></td>
<td>RVB3SND</td>
<td>JCL to execute RV Basic Sender</td>
</tr>
<tr>
<td></td>
<td>RVB3SNMF</td>
<td>JCL to execute RV multi-field Sender</td>
</tr>
</tbody>
</table>

Table 11  Rendezvous COBOL Examples

<table>
<thead>
<tr>
<th>Suffix</th>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COBOL</td>
<td>RVBCCMLC</td>
<td>RV CM Listener using automated call-back function</td>
</tr>
<tr>
<td></td>
<td>RVBCCMLS</td>
<td>Sample of RV CM Listener</td>
</tr>
<tr>
<td></td>
<td>RVBCCMSN</td>
<td>Sample of RV CM Sender</td>
</tr>
<tr>
<td></td>
<td>RVBCFTMN</td>
<td>Sample of RV FT Group Monitor.</td>
</tr>
</tbody>
</table>
### Table 11  Rendezvous COBOL Examples

<table>
<thead>
<tr>
<th>Suffix</th>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RVBCFTTM</td>
<td>Sample of RV FT Timestamp example</td>
</tr>
<tr>
<td></td>
<td>RVBCLSC</td>
<td>RV Basic Listener using automated call-back function</td>
</tr>
<tr>
<td></td>
<td>RVBCLSN</td>
<td>Sample of RV Basic Listener</td>
</tr>
<tr>
<td></td>
<td>RVBCLSMF</td>
<td>COBOL program of RV multi-field message listener</td>
</tr>
<tr>
<td></td>
<td>RVBCSND</td>
<td>Sample of RV Basic Sender</td>
</tr>
<tr>
<td></td>
<td>RVBCSNMF</td>
<td>Sample of RV multi-field message publisher</td>
</tr>
<tr>
<td>COPYBOOK</td>
<td>TIBRV</td>
<td>Copybook for RV enumerated types</td>
</tr>
<tr>
<td></td>
<td>TIBRVCM</td>
<td>Copybook for RV CM definitions and constants</td>
</tr>
<tr>
<td></td>
<td>TIBRVFT</td>
<td>Copybook for RV FT definitions and constants</td>
</tr>
<tr>
<td></td>
<td>TIBRVTEQ</td>
<td>Copybook for RV enumerated constants</td>
</tr>
<tr>
<td>JCL</td>
<td>RVBCCMLC, RVBCCMLS</td>
<td>JCL to execute RV CM Listener</td>
</tr>
<tr>
<td></td>
<td>RVBCCMSN</td>
<td>JCL to execute RV CM Sender</td>
</tr>
<tr>
<td></td>
<td>RVBCLSC, RVBCLSN</td>
<td>JCL to execute RV Basic Listener</td>
</tr>
<tr>
<td></td>
<td>RVBCLSMF</td>
<td>JCL to execute RV multi-field Listener</td>
</tr>
<tr>
<td></td>
<td>RVBCSND</td>
<td>JCL to execute RV Basic Sender</td>
</tr>
<tr>
<td></td>
<td>RVBCSNMF</td>
<td>JCL to execute RV multi-field Sender</td>
</tr>
</tbody>
</table>
Table 12  Rendezvous General Examples

<table>
<thead>
<tr>
<th>Suffix</th>
<th>Member</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>RVB3LGRR</td>
<td>Sample Rendezvous CM ledger file review</td>
</tr>
<tr>
<td>JCL</td>
<td>RVGCACHE</td>
<td>JCL to execute RV Cache daemon</td>
</tr>
<tr>
<td></td>
<td>RVGDEFLF</td>
<td>JCL to allocate CM Ledger File</td>
</tr>
<tr>
<td></td>
<td>RVGRVA</td>
<td>JCL to execute RV Agent</td>
</tr>
<tr>
<td></td>
<td>RVGRVD</td>
<td>JCL to execute RV daemon</td>
</tr>
<tr>
<td></td>
<td>RVGRVRD</td>
<td>JCL to execute RV routing daemon</td>
</tr>
<tr>
<td></td>
<td>RVGRVSD</td>
<td>JCL to execute RV secure daemon</td>
</tr>
<tr>
<td></td>
<td>RVGRVSRD</td>
<td>JCL to execute RV secure routing daemon</td>
</tr>
<tr>
<td></td>
<td>RVUINIFT</td>
<td>JCL to execute &quot;INIFTST&quot; (card)</td>
</tr>
<tr>
<td></td>
<td>RVULDGRR</td>
<td>JCL to execute RV CM ledger review</td>
</tr>
<tr>
<td></td>
<td>RVUPERFM</td>
<td>JCL to execute RV performance tester - master side</td>
</tr>
<tr>
<td></td>
<td>RVUPERFS</td>
<td>JCL to execute RV performance tester - slave side</td>
</tr>
<tr>
<td></td>
<td>RV$CCPL</td>
<td>JCL to compile COBOL RV API applications</td>
</tr>
<tr>
<td></td>
<td>RV$3CPL</td>
<td>JCL to compile C RV API applications</td>
</tr>
<tr>
<td>Suffix</td>
<td>Member</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>PROC</td>
<td>RVA</td>
<td>Procedure to execute RV Agent daemon</td>
</tr>
<tr>
<td></td>
<td>TIBTCAFF</td>
<td>Procedure to set TCPIP stack affinity</td>
</tr>
<tr>
<td></td>
<td>RVCACHE</td>
<td>Procedure to execute RV Cache daemon</td>
</tr>
<tr>
<td></td>
<td>RVD</td>
<td>Procedure to execute RV daemon</td>
</tr>
<tr>
<td></td>
<td>RVRD</td>
<td>Procedure to execute RV routing daemon</td>
</tr>
<tr>
<td></td>
<td>RVSD</td>
<td>Procedure to execute RV secure daemon</td>
</tr>
<tr>
<td></td>
<td>RVSRD</td>
<td>Procedure to execute RV secure routing daemon</td>
</tr>
<tr>
<td></td>
<td>TIBCNCPL</td>
<td>Procedure to compile prelink &amp; link COBOL application</td>
</tr>
<tr>
<td></td>
<td>TIB3NCPL</td>
<td>Procedure to compile prelink &amp; link C application</td>
</tr>
</tbody>
</table>
Appendix B  

Rendezvous Installed Data Sets

This appendix lists all of the installed data sets that are associated with Rendezvous.

Table 14  Installed Data Sets

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Space (1st/2nd)</th>
<th>DSORG</th>
<th>RECFM</th>
<th>LRECL</th>
<th>BLK SIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;USERHLQ&gt;.ASM</td>
<td>CYL (3/1)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.C</td>
<td>CYL (5/1)</td>
<td>PO</td>
<td>VB</td>
<td>100</td>
<td>27998</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.CFG.RVA</td>
<td>CYL (2/1)</td>
<td>PS</td>
<td>F</td>
<td>512</td>
<td>6144</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.CFG.RVCACHE</td>
<td>CYL (2/1)</td>
<td>PS</td>
<td>F</td>
<td>512</td>
<td>6144</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.CFG.RVPERFM</td>
<td>CYL (1/1)</td>
<td>PS</td>
<td>F</td>
<td>512</td>
<td>6144</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.CFG.RVPERFS</td>
<td>CYL (1/1)</td>
<td>PS</td>
<td>F</td>
<td>512</td>
<td>6144</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.CFG.RVRD</td>
<td>CYL (2/1)</td>
<td>PS</td>
<td>F</td>
<td>512</td>
<td>6144</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.CFG.RVSD</td>
<td>CYL (2/1)</td>
<td>PS</td>
<td>F</td>
<td>512</td>
<td>6144</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.CFG.RVSRD</td>
<td>CYL (2/1)</td>
<td>PS</td>
<td>F</td>
<td>512</td>
<td>6144</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.CMLEDGER.LISTEN01</td>
<td>CYL (5/2)</td>
<td>PS</td>
<td>FB</td>
<td>512</td>
<td>6144</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.CMLEDGER.SENDER01</td>
<td>CYL (5/1)</td>
<td>PS</td>
<td>FB</td>
<td>512</td>
<td>6144</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.CNTL</td>
<td>CYL (5/1)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.CNTLSAMP</td>
<td>CYL (2/1)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.COB</td>
<td>CYL (3/1)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.COPY</td>
<td>CYL (2/1)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.H</td>
<td>CYL (2/1)</td>
<td>PO</td>
<td>VB</td>
<td>100</td>
<td>27998</td>
</tr>
<tr>
<td>Data Set</td>
<td>Space (1st/2nd)</td>
<td>DSORG</td>
<td>RECFM</td>
<td>LRECL</td>
<td>BLK SIZE</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>----------</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.INSTALL</td>
<td>CYL (8/2)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.JCL</td>
<td>CYL (5/1)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.JCLSAMP</td>
<td>CYL (3/1)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.LIBRARY</td>
<td>CYL (25/5)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>3120</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.LOAD</td>
<td>CYL (150/0)</td>
<td>PO</td>
<td>U</td>
<td>0</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.OBJ</td>
<td>CYL (3/0)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.PROC</td>
<td>CYL (3/1)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.PROCSAMP</td>
<td>CYL (1/1)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.SKEL</td>
<td>CYL (1/1)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.SSL.LIBRARY</td>
<td>CYL (20/5)</td>
<td>PO</td>
<td>FB</td>
<td>80</td>
<td>3120</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.XMIT.INST</td>
<td>CYL (3/1)</td>
<td>PS</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
<tr>
<td>&lt;USERHLQ&gt;.XMIT.MAIN</td>
<td>CYL (71/5)</td>
<td>PS</td>
<td>FB</td>
<td>80</td>
<td>27920</td>
</tr>
</tbody>
</table>