TIBCO® Object Service Broker

Release Notes

Software Release 6.0.0
July 2012

two-second advantage™

TIBCO®
The Power of Now®

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, TIBCO Object Service Broker, and and TIBCO Service Gateway are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or 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.

The TIBCO Object Service Broker technologies described herein are protected under the following patent numbers:

Australia:	-	-	671137	671138	673682	646408
Canada:	2284250	-	-	2284245	2284248	2066724
Europe:	-	-	0588446	0588445	0588447	0489861
Japan:	-	-	-	-	-	2-513420
USA:	5584026	5586329	5586330	5594899	5596752	5682535

Contents

Preface

TIBCO® Object Service Broker is an application development environment and integration broker that bridges legacy and non-legacy applications and data.

Topics

Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions

Convention	Use
TIBCO_HOME *OSB_HOME*	By default, all TIBCO products are installed into a folder referenced in the documentation as *TIBCO_HOME*. On open systems, TIBCO Object Service Broker installs by default into a directory within *TIBCO_HOME*. This directory is referenced in documentation as *OSB_HOME*. The default value of *OSB_HOME* depends on the operating system. For example on Windows systems, the default value is `C:\tibco\OSB`. Similarly, all TIBCO Service Gateways on open systems install by default into a directory in *TIBCO_HOME*. For example on Windows systems, the default value is `C:\tibco\OSBgateways\6.0`. On z/OS, no default installation directories exist.
`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]`
italic font	Italic font is used in the following ways: • To indicate a document title. For example: See *TIBCO ActiveMatrix BusinessWorks Concepts*. • To introduce new terms For example: A portal page may contain several portlets. *Portlets* are mini-applications that run in a portal. • To indicate a variable in a command or code syntax that you must replace. For example: `MyCommand` *PathName*

Table 1 General Typographical Conventions (Cont'd)

Convention	Use
Key combinations	Key name separated by a plus sign indicate keys pressed simultaneously. For example: Ctrl+C. Key names separated by a comma and space indicate keys pressed one after the other. For example: Esc, Ctrl+Q.
	The note icon indicates information that is of special interest or importance, for example, an additional action required only in certain circumstances.
	The tip icon indicates an idea that could be useful, for example, a way to apply the information provided in the current section to achieve a specific result.
	The warning icon indicates the potential for a damaging situation, for example, data loss or corruption if certain steps are taken or not taken.

Table 2 Syntax Typographical Conventions

Convention	Use
`[ ]`	An optional item in a command or code syntax. For example: `MyCommand [optional_parameter] required_parameter`
`|`	A logical `OR` that separates multiple items of which only one may be chosen. For example, you can select only one of the following parameters: `MyCommand para1 | param2 | param3`

Table 2 Syntax Typographical Conventions

Convention	Use

`{ }`	A logical group of items in a command. Other syntax notations may appear within each logical group.
	
	For example, 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

You can access TIBCO documentation here:

http://docs.tibco.com

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.

Release Notes

This document includes release notes for TIBCO Object Service Broker, Software Release 6.0.0.

Check the TIBCO Product Support web site at https://support.tibco.com for product information that was not available at release time. Entry to this site requires a username and password. If you do not have a username, you can request one. You must have a valid maintenance or support contract to use this site.

Topics

New Features

This section lists features added since the last major (6.0.0) release of this product.

Release 6.0

The following are new features in this release.

- **IPV6 Support** All TIBCO Object Service Broker components for all supported platforms now support IPv6 networks. This new TIBCO Object Service Broker feature takes advantages of the TCP/IP dual stacks feature that is support by modern operating systems. To assist customers that continue to use IPV4 addresses in a mixed IP address environment, all TIBCO Object Service Broker components prefer IPV4 addresses over IPV6 addresses when resolving names to IP address.
- **JDBC Support** Java applets, applications, or application servers may now access and modify TIBCO Object Service Broker tables and run stored procedures through a JDBC interface. The JDBC driver may run in 32-bit or 64-bit Java Virtual Machines. For more information, see the section on Accessing TIBCO Object Service Broker Using JDBC in the *TIBCO Object Service Broker for Open Systems External Environments* manual.
- **ODBC for 64-bit Windows Applications** The TIBCO Object Service Broker Adapter for JDBC-ODBC now supports 64-bit Windows applications. The new 64-bit ODBC driver uses the same infrastructure as the JDBC interface. For more information, see the section on Accessing TIBCO Object Service Broker Using 64-bit ODBC in the *TIBCO Object Service Broker for Open Systems External Environments* manual.
- **Test ODBC Adapter Connection Settings** A Test button is now included when configuring the properties for an ODBC Adapter connection.
- **Extended SSL Support** All TIBCO Object Service Broker components, across platforms, can now communcate using Secure Sockets Layer (SSL). This allows all sensitive data, such as user IDs and passwords, to be securely passed between TIBCO Object Service Broker components.
- **TIBCO Object Service Broker UI** The UI now supports debugging of TIBCO Object Service Broker applications in a manner similar to the debugging of other programming languages supported by the Eclipse development environment. For example, breakpoints may be set on rule calls, exceptions, local variables, table accesses, and so on. The application can then be invoked, and its internal state be examined and/or modified whenever a breakpoint is reached. For information about using the TIBCO Object Service Broker UI and this new feature, refer to the TIBCO Object Service Broker UI online help.

TIBCO Object Service Broker for Open Systems

- **Checkpoint I/O Performance Improvements for Open Systems** A new approach is now used to write updated (dirty pages) from the resident page pool to page datasets during check point processing. For systems that experience a heavy update load, you will see fewer threads being used by the DOB, significantly improved write I/O rates, faster checkpoint elapse times, and on average fewer dirty pages in the resident page waiting to be migrated to page datasets. This all results in more optimal use of pages in the resident page pool and for some customers improved transactions rate because of better utilization of the resident page pool.
- **Platform Support** This release of TIBCO Object Service Broker for Open Systems includes support for the following new platforms:
  - Windows Server 2008 (64-bit)
  - Windows 7 Professional or Enterprise Editions (64-bit)
  - Solaris 11

TIBCO Object Service Broker for z/OS

- **Custom Session Types** You can now specify a custom session type in the SMGTRAN field using the tool $SETOPT, allowing for more detailed session accounting and SMF recording.

TIBCO Service Gateway for CICS

- **Containers and Channels** Traditionally, CICS programs have used communication areas (COMMAREAs) to exchange data. Beginning with Version 3 Release 1, CICS Transaction Server for z/OS added the Containers and Channels, which is an improved method of transferring data between programs in amounts that far exceed the 32KB limit that applies to COMMAREAs.

  The TIBCO Service Gateway for CICS now supports this facility, so that large amount of data can be passed in and out of the Object Service Broker session and to and from the CICS external routines within the session, using new shareable tools.

TIBCO Service Gateway for Oracle

- **Platform Availability** TIBCO Service Gateway for Oracle is available as a 32-bit application that can be used with Red Hat Enterprise Linux Server 5.6 for x86_64.

Changes in Functionality

This section lists changes in functionality since the last major (6.0.0) release of this product.

Release 6.0

The following are changes in functionality in this release.

TIBCO Object Service Broker for Open Systems

- **Maximum Resident Page Pool Size** The maximum resident page pool size was increased to 1 GB on Open Systems, matching the maximum size on z/OS.
- **Permission Requirements for Data Object Broker Users** As a result of improvements in security in Windows 7 and Windows 2008 Server, the user that starts the Data Object Broker must have the user right "Create Global Objects". By default, only Windows system administrators have this user right. If you choose not to use a system administrator account to start the Data Object Broker, you must use the Security Configuration Manager to modify this local security policy to grant the "Create Global Objects" right to the user that starts the Data Object Broker.

Deprecated and Removed Features

This section describes deprecated features (if any), and lists features, if relevant, that may be useful alternatives to the deprecated features. Any use of a deprecated feature should be discontinued as it may be removed in a future release. You should avoid becoming dependent on deprecated features and become familiar with the suggested alternative features.

This section also lists features that are removed (if any).

Release 6.0

Deprecated Features

No features are deprecated in this release.

Removed Features

The following features are removed in this release.

- **Platform Change** TIBCO Object Service Broker is no longer supported on Microsoft Windows Server 2003.

Migration from Previous Releases on z/OS

This section explains how to migrate from Release 5.x on a z/OS platform. To migrate to TIBCO Object Service Broker Release 6.0.0, do the following:

- Run the Installation Procedure
- Prepare the DOB for Database Migration
- Migrate the Database
- Customize the Migrated Operating Environment (Optional)

Before proceeding, note the following:

- You do not need to modify the journal and segment definitions.
- The allocations for data sets for TIBCO Object Service Broker components in Release 5.x are sufficient. The allocations include backup and dump data sets and supporting data sets for the DOB.
- You do not need to install TIBCO Object Service Broker type-3 SVC, which is unchanged from Release 5.x.
- We recommend that segment 0 contain a minimum of 5,000 free pages before the start of migration.

Run the Installation Procedure

To run the installation procedure, do the following:

1. Obtain the Installation Media
2. Prepare and Upload the Product File and Utilities
3. Install the OSTAREDC Program
4. Determine Your System Environment Values
5. Edit the Properties File
6. Install the Software

Obtain the Installation Media

Download TIBCO Object Service Broker from the TIBCO Software site by following these steps:

1. Contact TIBCO Software Inc. for a password, directory information, and so forth.
2. Connect to the TIBCO site with the required information.
3. Download this compressed file: `TIB_osb_6.0.0_zos.zip`.

Prepare and Upload the Product File and Utilities

1. Transfer the `TIB_osb_6.0.0_zos.zip` file to a PC that can connect to the z/OS host system.
2. Unzip the file to a temporary location on the PC. The file contains the following files:
   - `osb.xm1`, a compressed file with the TIBCO Object Service Broker libraries.
   - `metastr.xm1`, a file not used for migration.
   - `install.bin`, the `REXX EXEC` to perform the installation.
   - `ostarrec.bin`, the `REXX EXEC` to uncompress the `.xm1` files.
   - `property.bin`, the sample template of the properties file.
   - `OSTAREDC`, a load module to improve the performance of `OSTARREC`
3. Preallocate a PDS, fixed block data set on the z/OS host system, with the following name:

   HLQ`.INSTALL`

   where *HLQ* is a valid multi-level qualifier that will be used during installation. This qualifier should indicate that this is an TIBCO Object Service Broker distribution dataset for software release 6.0.0. For example, `userid.DIST600`, or `something.OSBD600`.

   See the sample JCL in the next step.
4. Preallocate the following sequential data set on the z/OS host system:

   HLQ`.OSB.XM1` (Size: 77,570 KB)

   Use the same *HLQ* as the previous data set. Below is sample JCL to allocate these data sets. Provide a `JOB` card and submit the JCL.

```
//ALLOC EXEC PGM=IEFBR14
//DD1   DD DSN=<HLQ>.INSTALL,
//      DISP=(,CATLG,DELETE),UNIT=SYSDA,
//      DCB=(RECFM=FB,LRECL=80,BLKSIZE=0),
//      SPACE=(TRK,(1,5,1))
//DD2   DD DSN=<HLQ>.OSB.XM1,
//      DISP=(,CATLG,DELETE),UNIT=SYSDA,
//      DCB=(RECFM=FB,LRECL=1024,BLKSIZE=0,DSORG=PS),
//      SPACE=(TRK,(2000,50))
```

5. FTP `install.bin`, `ostarrec.bin`, and `property.bin` to your z/OS system in `BIN` mode to the *HLQ*`.INSTALL` data set. Name these utilities `INSTALL`, `OSTARREC`, and `PROPERTY`, respectively.
6. FTP the `osb.xm1` file in `BIN` mode to the *HLQ*`.OSB.XM1` data set.

Install the OSTAREDC Program

1. Upload the `OSTAREDC` file to z/OS in binary format to a data set with `LRECL=80` and `RECFM=FB`.
2. In ISPF 3.4, type the following against this data set:

   `"RECEIVE INDA(/)"`

   When prompted, specify `DA('`*HLQ*`.INSTLOAD')` as the name of the load library where you want the `OSTAREDC` program restored.
3. Edit `OSTARREC` as follows:
   - Run the command `FIND OSTAREDC 1`.
   - Change the constant after the equal sign to contain the full name of the program's data set. The string must start with a double quote and a single quote; and end with a single quote and a double quote. The double quotes delimit the string; the single quotes tell TSO that the name of the data set is fully qualified. For example, change the following:

     `OSTAREDC = "'`*HLQ*`.INSTLOAD(OSTAREDC)'"`

     to

     `OSTAREDC = "'`*your.load.library*`(OSTAREDC)'"`

     where *your.load.library* is the name of the library referenced in step 2.

Determine Your System Environment Values

Before installation, review the information on the system environment in Table 3 and determine your values. **Note**: The Default Value column lists the default values for an initial installation of the base component.

Table 3 OSEMOD Variables

Description	OSEMOD Variable	Default Value	Your Value
High-level qualifier for uploaded data sets `INSTALL` and `OSB.XM1`.	`$HLQ$`	Specified on upload	
High-level qualifier for non-VSAM and VSAM data sets you are authorized to create.	`$HLQNONV$` `$HLQVSAM$`	`TIBCO.TESTNV` `TIBCO.TESTVS`	
Second-level qualifier for install files.	`$INSTVER$`	`INS60`	
Second-level qualifier for TIBCO Object Service Broker system files.	`$SLQ$`	`OSB60`	
Second-level qualifier for SMP/E libraries	`$SMP$`	`SMP60`	
For SMS Shops – `managementclass`, `dataclass`, and `storageclass`, if required			

For new non-VSAM data sets.	`$NMGTCLAS`
	`$NDATCLAS$`
	`$NSTOCLAS$`
		`STANDARD`
		`STANDARD`
		`S6BNONV`	

For new VSAM data sets.	`$VMGTCLAS`
	`$VDATCLAS$`
	`$VSTOCLAS$`
		`STANDARD`
		`STANDARD`
		`S6BVSAM`	

High-level qualifier of Language Environment libraries for `SCEELKED` and `SCEEBIND`.	`$CEELIB$`	`CEE`	
High-level qualifier of IBM's Callable Services library `CSSLIB`.	`$CSSLIB$`	`SYS1`	
COMMID for DOB.	`$TDS$`	`OSBDOB`	

For details, see *Appendix A, Installation Variables*, in the *TIBCO Object Service Broker for z/OS Installing and Operating* manual.

Edit the Properties File

Use the PROPERTY member in *HLQ*.INSTALL as a template and modify it to suit your requirements. Table 4 describes the keywords in the properties file that correspond to the system environment variables in Table 3.

Table 4 Properties File Keywords

Keyword	Description
INSTALL=	To migrate your TIBCO Object Service Broker system, specify MIGRATION: INSTALL=MIGRATION
HLQNONV=	High-level qualifier for non-VSAM data sets.
HLQVSAM=	High-level qualifier for VSAM data sets.
INSTVER=	Second-level qualifier for install files.
SLQ=	Second-level qualifier for TIBCO Object Service Broker system files.
SMP=	Second-level qualifier for SMP/E libraries.
SMS=	YES for SMS site, NO for non-SMS site.
COMPAT=	Use if SMS=YES. The valid values are YES for SMS-compatible class names of data sets and NO for non-SMS-compatible class names of data sets. If COMPAT=NO, specify the following: • NMGTCLAS – MANAGEMENTCLASS for non-VSAM data sets • NDATCLAS – DATACLASS for non-VSAM data sets • NSTOCLAS – STORAGECLASS for non-VSAM data sets • VMGTCLAS – MANAGEMENTCLASS for VSAM data sets • VDATCLAS – DATACLASS for VSAM data sets • VSTOCLAS – STORAGECLASS for VSAM data sets

Table 4 Properties File Keywords

Keyword	Description
VOLSER=	If SMS=YES, specify one DASD volume for VSAM data set allocation. The default is USER01. If SMS=NO, specify these three DASD volumes, separated by commas. • vol1 – DASD volser for temporary work files • vol2 – DASD volser for install files • vol3 – DASD volser for TIBCO Object Service Broker system files
CEELIB=	High-level qualifier of Language Environment libraries.
CSSLIB=	High-level qualifier of IBM's Callable Services library CSSLIB.
TDS=	COMMID for DOB.

Install the Software

To exit the interactive session at any time after executing the REXX exec INSTALL, do the following:

1. Press PA1.
2. Enter hi.
3. Press ENTER twice.

Step 1: **Execute File Tailoring EXEC to start installation.**

Member in: *HLQ*.INSTALL

Member: INSTALL (EX member)

Step 1 verifies that files can be allocated successfully with the values in the PROPERTY file. Test files of type sequential, PDS, PDSE, and VSAM are allocated and then deleted. Installation stops if any test allocation fails, in which case you should investigate the cause, correct the condition, and repeat step 1.

Step 2: **Run the job OSB.JCL.**

This job unpacks the XM1 data sets and creates working copies. It ends with RC=0.

JCL in: *HLQ*.OSB.JCL

Data Set: *HLQ*.OSB.JCL (Edit the JOB card and SUB data set.)

Step 3: **Edit OSEMOD.**

Change the values for TDS, MDL, and $MIGRPSW$ to those specific to your installation. They are the COMMID, MDLAPPL, and SYSADMIN passwords for the system being migrated.

Member in: *HLQ*.FILECLS

Member: OSEMOD

Step 4: **Customize the supplied data sets.**

Member in: *HLQ*.OSB.FILEI

Member: S6A1CUST (EX member)

Step 5: **Perform SMP apply and accept tasks and configure the parameters with the *HLQNONV*.*INSTVER*.MIGRATE data set.**

1. We recommend that you accept the base OSB FMID.

   To do that later, change the status of S6Z3ACPT in member JOBSZ from INSTALL to FUTURE.

2. Edit job S6Z3CFGR to reference your existing TIBCO Object Service Broker 5.x files, such as the APF load library, into which to copy and the Execution Environment configuration parameter members PARMBAT, PARMTSO, PARMNEE, and so on.

   In particular, review the recommendations for parameters in the section Migrate the Database on page 18.

3. Ensure the APF load library is previously emptied.

   Member in: *HLQ*.*INSTVER*.MIGRATE

   Member: S6Z1RUNJ (EX member)

Jobs S6Z2APLY and S6Z3ACPT complete with a return code of 4. Job S6Z3CFGR completes with a return code of 0.

If any job fails, do the following (members reside in *HLQ*.*INSTVER*.MIGRATE):

1. Fix the problem.
2. Perform a cleanup task.

   Member: S6Z9CLEN (SUB member)
3. Rerun as follows:

   Member: S6Z1RUNJ (EX member)

If doing a restart, do the following (members reside in *HLQNONV*.*INSTVER*.JCL):

1. Perform a cleanup task.

   Member: S6Z9CLEN (SUB member)
2. Manually delete the following working copies:

   HLQNONV.*INSTVER*.OSB.JOBS
   HLQNONV.*INSTVER*.CNTL
   HLQNONV.*INSTVER*.CLIST
   HLQNONV.*INSTVER*.JCL
   HLQNONV.*INSTVER*.MIGRATE
3. Execute File Tailoring EXEC.

   Member: S6Z1RUNJ (EX member)

Step 6: **Upgrade the optional Service Gateways.**

If you plan to install any other optional products, be sure to perform the accept task in Step 5.

For details on installing the Service Gateways for CICS, IMS TM, and WMQ, see the *TIBCO Object Service Broker for z/OS Installing and Operating* manual. For the Service Gateway for Files, refer to the *TIBCO Object Service Broker Managing External Data* manual. For all other gateways, refer to the gateway-specific manuals.

Step 7: **Apply the latest hotfix level.**

Refer to Late Breaking News on the TIBCO Support site (http://support.tibco.com) to determine the latest hotfix level.

Download it and apply any required SMP maintenance at this point according to the hotfix documentation.

Step 8: **Customize the migrated TIBCO Object Service Broker.**

Customize the following:

- Date display mask
- Century range
- Unicode processing

For details, see the *TIBCO Object Service Broker for z/OS Installing and Operating* manual.

To start the installation over, manually delete the following data sets in the specified sequence and restart at step 1.

1. HLQ.ASM
2. HLQ.COBOL
3. HLQ.FILECLS
4. HLQ.FILECTL
5. HLQ.FILEEM1
6. HLQ.FILEEM2
7. HLQ.FILEJCL
8. HLQ.FILEMIG
9. HLQ.FILEOBJ
10. HLQ.FILETRK
11. HLQ.FILEUP4
12. HLQ.FILEUP5
13. HLQ.FILEXML
14. HLQ.MACRO
15. HLQ.OSB.FILEI
16. HLQ.UNICODE
17. $HLQNONV$.$INSTVER$.CLIST
18. $HLQNONV$.$INSTVER$.CNTL
19. $HLQNONV$.$INSTVER$.JCL
20. $HLQNONV$.$INSTVER$.MIGRATE
21. $HLQNONV$.$INSTVER$.OSB.JOBS

Prepare the DOB for Database Migration

Perform the following tasks before migrating the database.

Before proceeding, review the MIGRATE data set and verify that the following jobs—

- S6Z2APLY (RECEIVE and APPLY the base component)
- S6Z3ACPT (ACCEPT the base component)
- S6Z3CFGR (create Execution Environment configuration modules)

—have completed successfully.

Step 1: **Query the Capabilities of Your CPU. (Optional)**

The S6BBRHDW member in the JCL data set contains JCL to run the S6BBRHDW (CPU Capabilities Report) utility, to report on the availability of the hardware instructions required to run Release 6.0. Produce the CPU Capabilities Report; it ends with RC=0.

If the report shows that your system is missing instructions, then it has some restrictions in running this version of the software; contact TIBCO Support to resolve the issue.

Member in: *HLQ*.*INSTVER*.JCL

Member: S6BBRHDW (SUB member)

Step 2: **Back up the Release 5.x MetaStor.**

Back up segment 0. The BACKUP member of the *HLQ*.*INSTVER*.JCL data set of your Release 5.x system contains the JCL to take the initial system backup of segments 0 and 1. The system backup is required in the spin verification process.

Perform the following:

1. Shut down the DOB.
2. Point the STEPLIB DD statement to the Release 5.x load library.
3. Submit BACKUP.

   The job ends with RC=0.

Step 3: **Spin the journals.**

You must spin (empty) the journals with the DOB shut down, since starting or stopping the DOB results in additional journal records being written. The `SPIN01` and `SPIN02` members of the *HLQ*`.`*INSTVER*`.JCL` data set of your Release 5.*x* system contain the necessary JCL.

Perform the following:

1. Shut down the DOB if it is running. The DOB should terminate with no errors.
2. Ensure that the `STEPLIB DD` statement still points to the Release 5.x load library.
3. In the step which calls S6BSPJEX add the DD statement:

   ```
   //S6BRELAY DD DUMMY
   ```

4. Submit SPIN01 for the first journal, followed by `SPIN02` for the other journal.

   Spin jobs end with `RC=4`.

Step 4: **Convert your 5.x `DBDLIB` to a 6.0.0 `DBDLIB`.**

This step is required for Release 6.0.0. You can customize sample JCL member `DBJCL` in *HLQNONV*`.`*INSTVER*`.JCL` to perform the conversion.

Perform the following:

1. Ensure that the DOB is still down.
2. Verify that the `SYSLIB DD` concatenation for `$HLQNONV$.$SMP$.MAC` in step ASM points to the newly installed 6.0.0 macro library or to *HLQ*`.MAC`.
3. Copy your current database definitions to the customized `DBJCL` job.

   Note: If you are migrating from 5.0, note that the DBGEN macro in 6.0 does not accept operands.

4. Submit the job.

   It ends with `RC=0`.

Step 5: **Customize tables `@PROM_CONSTANTS` and `@PROM_BACKUPDS`.**

Do the following:

1. Change the `STEPLIB DD` statement of the Release 5.x DOB job to point to the Release 6.0.0 load library:

   HLQNONV`.`*INSTVER*`.AUTH`

2. Change the DBDLIB statement of the Release 5.x DOB job to point to the 6.0.0 DBDLIB you created in step 3 above.

3. Start the DOB and respond to a WTOR prompt.

4. Ensure that no peer and external database servers are connected and that all user access is disabled. To disable user access, choose the Suspend users option from one of the Promotions Administration menus.

5. Log in as `SYADMIN`. Use the newly supplied `USER` exec to log in with the following command:

   `TSO EX '`*HLQNONV*`.`*INSTVER*`.CLIST(USER)' 'U=SYSADMIN'`

6. Edit `@PROM_CONSTANTS`, ensuring that an occurrence exists with the following values:

   ```
   NODENAME = 'HURONPTF'
   CH_PREFIX = 'HLQNONV.CH'
   SUFFIX = '.UPGRADE'
   RESTRICT_RIGHTS = 'S'
   EXE_LIB = 'VXPXXUPG'
   DEV_LIB = 'COMMON'
   ALLOC_DATASET = 'N'
   SAVE_APPLY_LOG = 'Y'
   ```

7. Edit `@PROM_BACKUPDS($NODNAM$)`, where `$NODNAM$` is the node name of your Release 5.x DOB. Ensure that a table occurrence exists with the following values:

   ```
   SOURCENODE = 'HURONPTF'
   BK_PREFIX = 'HLQNONV.BK'
   SUFFIX = '.UPGRADE'
   ALLOC_DATASET = 'N'
   ```

Migrate the Database

The table below shows the database PTFs to apply, based on your current database maintenance level (first column). The corresponding job members (third column) in the `MIGRATE` data set contain JCL to apply the database migration PTFs to your Release 5.x system.

Before running the upgrade jobs, do the following:

1. Set the batch EE configuration parameters to at least the minimum values, as follows:

   ```
   TRANMEMMAX=100M
   SESSIONMEMMAX=100M
   TRANMAXNUM=9
   ```

2. Ensure the following DOB parameters are *not* set less than the default values:

   ```
   DATAPAGELIMIT
   TRXREADLIMIT
   WORKINGSET
   ```

For more information, see the solution `SOL1-A8HKWO` on support.tibco.com.

Database Level	#	Job	Job Description	RC	Status (Default)
Release 5.0.0/DB00 Run #1 – 7	1	S6Z51D0A	Apply V5.1.0/DB00 database PTFs (Part 1).	0	OPTIONAL
	2	S6Z51D0B	Apply V5.1.0/DB00 database PTFs (Part 2).	0	OPTIONAL
	3	S6Z51D0X	Delete V5.1.0/DB00 extract and backup data sets.	0	OPTIONAL
	4	S6Z52D0A	Apply V5.2.0/DB00 database PTFs (Part 1).	0	OPTIONAL
	5	S6Z52D0B	Apply V5.2.0/DB00 database PTFs (Part 2).	0	OPTIONAL
	6	S6Z52D0C	Apply V5.2.0/DB00 database PTFs (Part 3).	0‘	OPTIONAL
	7	S6Z52D0X	Delete V5.2.0/DB00 extract and backup data sets.	0	OPTIONAL
Release 5.2.0/DB00 Run #8 – 13	8	S6Z53D0A	Apply V5.3.0/DB00 database PTFs.	0	OPTIONAL

Database Level	#	Job	Job Description	RC	Status (Default)
	9	S6Z53D0X	Delete V5.3.0/DB00 extract and backup data sets.	0	OPTIONAL
	10	S6Z55D0A	Apply V5.5.0/DB00 database PTFs.	0	OPTIONAL
	11	S6Z55D0X	Delete V5.5.0/DB00 extract and backup data sets.	0	OPTIONAL
	12	S6Z60D0A	Apply V6.0.0/DB00 database PTFs.	0	OPTIONAL
	13	S6Z60D0X	Delete V6.0.0/DB00 extract and backup data sets.	0	OPTIONAL

Do the following:

Step 1: **Copy the Release 6.0.0 load library created by way of SMP to the library to be used by the DOB.**

Ensure the load library is authorized.

Step 2: **Start the DOB.**

Step 3: **Determine your current database maintenance level.**

If you are unsure of your database level, log in to the DOB and `EX DBMAINTLVL` to return the last complete database maintenance level and any additional database PTFs applied.

Locate your database maintenance level in the first column of the above table.

Step 4: **Modify the status of the jobs to run.**

The jobs to run are listed directly below the database maintenance level determined in step 2. For example, if your level is Release 5.0.0/DB00, run the jobs numbered 01 through 13 (`S6Z51D0A` through `S6Z60D0X`) in sequence. To run these jobs manually, proceed to step 5.

Change the default status (listed in column 6) of all jobs that will be run to `INSTALL`.

Member in: *HLQNONV*.*INSTVER*.MIGRATE

Member: JOBSZ (EDIT member)

Note: Do not edit member JOBSZ once the migration jobs are initiated in step 6.

Step 5: **Specify the load library.**

Because the maintenance jobs listed in step 4 are set up to run against the load library built by way of SMP in the section Install the Software, you can do one of the following:

- Authorize this library for the duration of the database maintenance. (**Note:** Once the upgrade is complete, all Execution Environments must use the same load library as their DOB.)
- Modify the database maintenance jobs to use the load library being used by the DOB.

For subsequent runs of the database maintenance against a different DOB, modify the TDS=xxx parameter of all the jobs listed in step 4 to point to the DOB that is to be upgraded.

Step 6: **Run the migration jobs.**

Run the jobs listed directly below the maintenance level determined in step 3. For example, if your level is Release 5.0.0/DB00, you will run the jobs numbered 01 through 13 in sequence.

If you wish to run these jobs manually, submit the jobs one at a time in sequence. Each job should complete with an RC=0. Continue to the next job only if RC=0. If RC is nonzero, fix the problem and rerun the failed job.

After successful completion of all the DB maintenance jobs, proceed to step 7.

Member in: *HLQNONV*.*INSTVER*.MIGRATE

Member: S6Z1RUNJ (EX member)

If RC nonzero, perform the following:

1. Fix the problem
2. Rerun the failed job.

   Member: S6Z1RUNJ (EX member)

Step 7: **Apply the latest hotfix level.**

If any database maintenance exists with the hotfix downloaded earlier, apply it now. See the hotfix documentation.

Step 8: **Shut down the DOB.**

At this point, the DOB has been upgraded to Release 6.0.0.

Step 9: **Back up the migrated system.**

Back up segment 0. The BACKUP member of your Release 5.x system contains the JCL to take the initial system backup of segments 0 and 1. The system backup is required in the spin verification process. For details, see the *TIBCO Object Service Broker for z/OS Installing and Operating* manual.

Do the following:

1. Point the STEPLIB DD statement to the Release 6.0.0 load library.
2. Submit BACKUP.

   It ends with RC=0.

You can now start the Release 6.0.0 DOB.

Customize the Migrated Operating Environment (Optional)

Depending on your requirements, the tasks in this section are optional.

- If necessary, modify the DOB startup JCL (for example, OSRUN) and other CLISTs to point to the Release 6.0.0 load library if it is not installed in the same location as the Release 5.0.0 load library. For details, see the *TIBCO Object Service Broker for z/OS Installing and Operating* manual.
- Some changes to CICS resources between 5.0.0 and 6.0.0 have occurred. The full specification of all required resources is in *HLQNONV*.*INSTVER*.JCLSAMP(CICSDEFS). To use the 6.0.0 version of Service Gateway for CICS, either create a new DFHCSD file with the latest CICSDEFS or make the following changes to a previously defined DFHCSD:
  - Transaction HOPT has been removed from the group associated with the application owning region (AOR).

- A new transaction called HREL has been added to the groups associated with the non-MRO, AOR, and TOR.
- Program S6BCSOPT in group S6BT1 (previously known as OSTART1) has been changed from RESIDENT(YES) to RESIDENT(NO).
- Program S6BCSREL has been added.

Between 5.0.0 and 6.0.0, some cosmetic changes have also been made to group names. Use CICSDEFS as the definitive source of information.

Migration from Previous Releases on Open Systems

This section explains how to migrate from Release 5.x on supported Windows or Solaris platforms.

If you are upgrading from Release 5.x on a Windows system other than Windows 2008 R2 64-bit or Windows 7 64-bit, see the section Migrating from Unsupported Windows Platforms on page 31.

Installation Modes

You can run the installer for migrating to TIBCO Object Service Broker Release 6.0.0 in the following modes.

- **GUI mode** – In GUI mode, the installer presents panels that enable you to make choices about product selection, product location, and so on. When you invoke the installer by double-clicking the icon, GUI mode is in effect.
- **Console mode** – Console mode enables you to run the installer from the command prompt or terminal window, primarily for non-Windows environments.
- **Silent mode** – Silent mode installs without prompting you for information. To use this mode, first generate a *response file* (in GUI mode or Console mode) that contains the input values for the installation.

Base Components and SDK Clients

The TIBCO Object Service Broker installation is a two-part process. The first part installs the base TIBCO Object Service Broker binary and database components, including the MetaStor. The second part, optionally launched from the first process, installs the TIBCO Object Service Broker SDK Clients. These clients are as follows:

- TIBCO Object Integration Gateway: J2EE for Windows and Solaris
- TIBCO Object Integration Gateway COM and .NET components: for Windows only
- ODBC Adapter: for Windows only

The C++ and Java SDKs are part of the base components for all platforms.

Preinstallation

Prerequisite Software

- Your current TIBCO Object Service Broker system must be at Release 5.0 GA or higher. To verify the release level of your current installation, run this command from a command prompt:

  ```
  hrncr state
  ```

  The output shows the TIBCO Object Service Broker release level (for example, Release 5.0.0.0.0XX).

- The installers make use of any installed JRE or JDK at the version 1.6 or 1.7 level set in the `JAVA_HOME` environment variable.

Installer Disk Space Requirements

- **Microsoft Windows** – The entire package is extracted into a temporary folder, typically `SystemDrive:\Temp` or `SystemDrive:\Documents and Settings\`*user_name*`\Local Settings\Temp`. The installer requires 60 MB of free space in the temporary directory.
- **Solaris** – When a regular (nonroot) user installs a TIBCO product, the installation registry is maintained in the user's home directory. As more products are installed, entries are added. The user's home directory must at least have 500 KB of free disk space.

Installer Account

On Microsoft Windows, you must have administrator privileges for the machine on which TIBCO Object Service Broker is installed. Otherwise, the installer exits. You must then log out of the system and log in as a user with the required privileges or request your system administrator to assign the privileges to your account.

To install the product on a network drive, ensure that the account for installation has permission to access the network drive.

For an installation on a Windows terminal server, there are two modes: `Execute` and `Install`. By default, all users are in `Execute` mode, which enables them to run the applications. To install TIBCO Object Service Broker for use by everyone, the Administrator must change to `Install` mode.

The best way to install TIBCO Object Service Broker is through the Add/Remove Programs control panel applet, which automatically sets the mode to `Install` during the installation and then back to `Execute` at the end. Alternatively, you can manually change your mode to `Install` by typing:

```
C:\> change user /install
```

Change back to execute:

```
C:\> change user /execute
```

Check your current mode:

```
C:\> change user /query
```

If you install in the `Execute` mode, the installation registry is maintained in your user home directory. If you install in the `Install` mode, the installation registry is maintained in the *%SystemRoot%* folder.

On Solaris, you can install TIBCO Object Service Broker as a regular (nonroot) user or superuser (root). Different users can install the same product at different locations.

Obtain the Installation Media

Installation file names for TIBCO Object Service Broker vary by version number and platform according to the following format:

`TIB_osb_`*`version_platform`*

where *version* is the three-digit version number for this TIBCO Object Service Broker release and *platform* is an abbreviated form of the hardware platform for which the executable is intended.

The installation files for this release for Windows are as follows:

`TIB_osb_6.0.0_win_x86.exe`	Base product
`TIB_osbC_6.0.0_win_x86.exe`	SDK Clients

The installation files for this release for Solaris are as follows:

`TIB_osb_6.0.0_solaris.bin`	Base product
`TIB_osbC_6.0.0_solaris.bin`	SDK Clients
`OSBSolarisInstall.sh`	Install script

Download TIBCO Object Service Broker from the TIBCO Software site by following these steps:

1. Contact TIBCO Software for a password, directory information, and so on.
2. Connect to the TIBCO Software web site with the required information.
3. Download the installation file appropriate for your platform.

Shut Down the DOB

Before shutting down the DOB, bring down all connected components, such as Execution Environments, external database servers, batch clients, and others. Follow your normal shutdown procedure to terminate the DOB.

Backup the MetaStor (Segment 0)

Back up the MetaStor (segment 0) by following your site's standard TIBCO Object Service Broker backup procedure. If your MetaStor is more than 95-percent utilized, you should resize it. For details on system backup, see the TIBCO Object Service Broker *for Open Systems Managing Backup and Recovery* manual.

Prepare to Run the Installer

Before running the installer, verify the following:

- The *HURON* environment variable points to the directory in which your current copy of TIBCO Object Service Broker resides. This is where your upgraded copy of the software will be installed.
- Ensure that all other TIBCO Object Service Broker-related environment variables (`HURONDIR` and `OS_ROOT`) correctly point to the target system to be upgraded and that `PATH` also reflects its presence.
- You have at least 90 MB of free space on the disk for installing the software.
- Your database has 128 or fewer page files in each segment.
- Your `crparm` file contains no `CTABHANDLES` or `RESHANDLES` entries. If those parameters are in the `crparm` file, delete them; they are no longer valid.
- The Base and SDK Clients installers should reside together in the same folder.

Installation Verification Procedure (IVP) Parameters

The installation of the TIBCO Object Service Broker SDK Clients includes an IVP, which requires values for the following parameters:

Parameter	Description
Host Name	The host on which a DOB is active to serve IVP requests.
Port Number	The osMon port number.
OSB Userid	The user ID to use in each IVP request.

Parameter	Description
Password	The password associated with the user ID.
Java Home	The JRE/JDK path to be used by the IVP process.

Choose an Installation Mode

Table 5 describes the modes for installing the product on Windows and Solaris.

Table 5 Installation Modes on Windows and Solaris

Installation Mode	Description

GUI mode	Enables you to input values in panels. Run the following command:
	On Windows:
	`TIB_osb_6.0.0_win_x86.exe`
	On Solaris:
	`cd installation-directory`
	`OSBSolarisInstall.sh`

Console mode	Enables you to install the software without a GUI. The installer prompts you for values. Run the following command:
	On Windows:
	`TIB_osb_6.0.0_win_x86.exe -console`
	On Solaris:
	`cd installation-directory`
	`OSBSolarisInstall.sh -console`

Table 5 Installation Modes on Windows and Solaris

Installation Mode	Description
Silent mode	To use this mode, you must first generate a response file (in GUI mode or Console mode) that contains the input values for the installation. You generate a response file with the following command: *installer* `-options-record` *responseFileName* where *installer* is the installer executable (or script file on Solaris), and *responseFileName* is the name of the response file to be generated. For an example, see "Install and generate a response file" below. Type the following at the command prompt to use this mode: On Windows: `TIB_osb_6.0.0_win_x86.exe -silent -options` *responseFileName* On Solaris: `cd installation-directory` `OSBSolarisInstall.sh -silent -options` *responseFileName*
Install and generate a response file	You can generate a response file during installation which you can later use to invoke the installer with the selected values as default values (GUI mode) or as selected values (silent mode). To install and generate a response file, run the following command: On Windows: `TIB_osb_6.0.0_win_x86.exe -options-record` *responseFileName* On Solaris: `./TIB_osb_6.0.0_solaris.bin` *-mode* `-options-record` *responseFileName* where *-mode* is `-gui`, `-console`, or `-silent`. However, you cannot generate a response file with the `-silent` option.

Table 5 Installation Modes on Windows and Solaris

Installation Mode	Description
GUI or console mode installation with a response file	You can use a previously generated response file for installation. For nonsilent modes, the response file determines the defaults that are presented. For silent mode, the response file determines what will be installed. To install with a response file, run the following command: On Windows (GUI mode): `TIB_osb_6.0.0_win_x86.exe -options` *responseFileName* On Windows (Console mode) `TIB_osb_6.0.0_win_x86.exe -console -options` *responseFileName* On Solaris (GUI mode): `./OSBSolarisInstall.sh -options` *responseFileName* On Solaris (Console mode): `./OSBSolarisInstall.sh -console -options` *responseFileName*
Combination of options	You can combine the various options. For example, to install in console mode and generate a response file, run the following command: On Windows: `TIB_osb_6.0.0_win_x86.exe -console -options-record` *responseFileName* On Solaris: `./OSBSolarisInstall.sh -console -options-record` *responseFileName*

Installing the Software

Before proceeding, you should have determined which components you will upgrade as described in Base Components and SDK Clients on page 23. If you plan to install the SDK Clients, ensure that the SDK Clients installer file is in the same folder as the base components installer and that the folder is writable *before* invoking the base product installer. You need approximately 5 KB of available space.

Base Components Installation (with or without SDK Clients installation)

To upgrade TIBCO Object Service Broker, do the following:

1. Initiate installation by issuing the command of the desired installation mode.
2. When prompted, accept the license agreement.
3. When prompted, select Upgrade.
4. When prompted to confirm the upgrade of the system installed in the directory pointed to by the `HURON` environment variable, click Next.
5. When prompted for a TIBCO Object Service Broker user ID and password (the default is `SYSADMIN`), specify your values and click Next.
6. When instruction and reminder panels are displayed, click Next until the actual database upgrade process begins. The upgrade process shows its progress as each database level and PTF is applied.
7. Once the upgrade completes, the base installer launches the SDK Client installation process. Click the Install button.
8. When the SDK Clients installation finishes, the IVP installer is automatically launched. To run the IVP, specify the field values as described in Installation Verification Procedure (IVP) Parameters. Then select the checkbox to launch the IVP and click Next.

   In the Windows environment, the black window for osMon displays as the upgraded system is brought up to provide the host connectivity to carry out the IVP processing.
9. If a dialog box regarding `osbinstall`.data message displays, click Yes.
10. When the SDK Clients/IVP installation is complete, click Finish to exit the installers for those processes.
11. Click Finish again to exit the installation wizard.

Activate the Object Integration Gateway Environment

To activate the Object Integration Gateway environment, do the following:

1. Log in to TIBCO Object Service Broker as `SYSADMIN`.
2. Execute the rule `ECTSSETUP`.
3. Log off TIBCO Object Service Broker.

Reconfigure the System (Optional)

After upgrading TIBCO Object Service Broker, consider reconfiguring the DOB and user sessions and, if you use distributed data access, your communications environment. To configure a TIBCO Object Service Broker ODBC data source, see the TIBCO Object Service Broker *for Open Systems External Environments* manual.

Migrating from Unsupported Windows Platforms

If you are migrating from a Windows system other than Windows 2008 R2 64-bit or Windows 7 64-bit, follow these steps to perform the migration:

1. Install TIBCO Object Service Broker software release 6.0.0 on the target Windows machine, using the `TIB_osb_6.0.0_win_x86_64.exe` installation program.
2. After the installation is complete, rename or remove the TIBCO Object Service Broker 6.0 system database folder just created.
3. Copy your TIBCO Object Service Broker 5.x database folder to the directory where the 6.0 directory was.
4. Edit the following DATABASE files to update the system administrator user ID and the hostname.
   - `CRPARM`
   - `SECPARM`
   - `HURON.DIR`

   These files are located in the `%HURON%\DATABASE` directory.
5. Change the `start_osmon.bat` file in `%HURON%\utils` to point to the correct DOB and MON.

6. In the Windows Start Menu, update the Stop OSMon shortcut with the OSMon name used in the 5.x TIBCO Object Service Broker 6.0 installation.
   a. Right-click the Stop OSMon Shortcut in the Windows Start Menu and select Properties.
   b. Edit the argument in the Target value to specify the value from your OSB 5.x installation.
7. Bring up an OSB enabled command prompt (using the shortcut from the new OSB 6.0's installation). This ensures that the HURON environment variable points to the correct system for the next step.
8. Invoke the `TIB_osb_6.0.0_win_x86_64.exe` installer and select **Upgrade** in the Install Type panel.
9. Respond to the subsequent installation prompts, providing the proper SYSADMIN password to begin the Database Upgrade process.

After the upgrade process has completed, the resulting TIBCO Object Service Broker 6.0 installation will use your 5.x system's assigned DOB name, ports, and so forth.

Closed Issues

The tables in the sections below list issues that were closed in the named releases:

- Closed Issues for TIBCO Object Service Broker for Open Systems, page 33
- Closed Issues for TIBCO Object Service Broker for z/OS, page 37
- Closed Issues for TIBCO Service Gateways, page 47

Closed Issues for TIBCO Object Service Broker for Open Systems

Table 6 lists the closed issues for the TIBCO Object Service Broker for Open Systems.

Table 6 Closed Issues for TIBCO Object Service Broker for Open Systems

Closed in Release	Key	Summary
6.0.0	OSBO-1704 OSBO-1528 1-D1VL7D 1-BF8A53	Undesired change in behavior of the Report Server between release 5.0 and 5.2 has been reversed.
6.0.0	OSBO-1693 1-CM2NQV	Fixed an error that prevented a manual journal spin from completing when the command was issued from a prompt, and DOB ran as a Windows Service on Server 2003 or 2008.
6.0.0	OSBO-1584 1-BT3DAI	Fixed a problem that caused the Service Gateways to fail on Open Systems when attached to a z/OS DOB.
6.0.0	OSBO-1583 1-BEQNB1 1-BYD1CF	Fixed an error that sometimes caused the Data Object Broker to hang without warning messages. This error occurred when the redo log became nearly full, and a checkpoint was requested while under heavy load. Two new messages with identifiers, S6BUL214W and S6BUL215I, have been added to record the potential full redo log condition being reached and resolved.
6.0.0	OSBO-1573 1-BPTIKD	The OIG .Net component may be used by 64-bit applications on the supported Windows platforms. Previously, this component was supported only on 32-bit applications.
6.0.0	OSBO-1538 1-BJ1SPN	Fixed an error that could cause the DOB to abent following a CALL to FORALLA with an invalid selection string.

Table 6 Closed Issues for TIBCO Object Service Broker for Open Systems

Closed in Release	Key	Summary
6.0.0	OSBO-1524 1-BD7IC5	Fixed an error that caused a screen table field with a NULL value to be treated differently on Open Systems, compared to the behavior on z/OS.
6.0.0	OSBO-1523 1-BD9BO3	Fixed an error that could cause a mis-match between field length and the display mask on a screen table field results in a buffer overlay.
6.0.0	OSBO-1521 1-BCE46T	Fixed an error that caused inconsistent messages across platforms for screens when scrolling left/right or up/down.
6.0.0	OSBO-1520 1-BBMLJH	This release adds the ability to establish a 3270 session to TIBCO Object Service Broker on Open Systems using Attachmate Reflection Standard Edition 2011 emulator.
6.0.0	OSBO-1519 1-BCAGLV	Previously, the behavior of screens containing multiple scrollable tables in Open Systems differed from the behavior on z/OS. When scrolling to the end of a screen table, the cursor automatically moved to the first field of the next screen table. This has been corrected to match the behavior z/OS, where the cursor does not automatically move.
6.0.0	OSBO-1515 1-BAXLT7	Fixed a problem that could cause incorrect values to display in the Reference Table for a Screen field, after changing the Reference Table options. This occurred on Open Systems when using `USERPROMPT=Y`.
6.0.0	OSBO-1514 1-BB10MV	Previously, `CURSOROCC#` behaved differently on z/OS and Open Systems. On z/OS, placing a cursor on a field returned the occurrence number. On Open Systems, it returned zero. This has been corrected. The actual occurrence number is now returned on both z/OS and Open Systems.
6.0.0	OSBO-1499 1-B8E68I	Fixed a problem that could cause fields to display incorrectly in Open Systems when migrating from z/OS.
6.0.0	OSBO-1493 1-B3M2DP	Previously, repeat invocations of eCTSSession.RunTrans within a session could fail because the ECTSPARMS table was not populated with the full set or arguments. This has been fixed.
6.0.0	OSBO-1424	Fixed an error that prevented the base installer from generating nonconflicting parameters for `osMon` and the DOB, while installing multiple instances of TIBCO Object Service Broker.

Table 6 Closed Issues for TIBCO Object Service Broker for Open Systems

Closed in Release	Key	Summary
6.0.0	OSBO-1524 1-BD7IC5	Fixed an error that caused a screen table field with a NULL value to be treated differently on Open Systems, compared to the behavior on z/OS.
6.0.0	OSBO-1523 1-BD9BO3	Fixed an error that could cause a mis-match between field length and the display mask on a screen table field results in a buffer overlay.
6.0.0	OSBO-1521 1-BCE46T	Fixed an error that caused inconsistent messages across platforms for screens when scrolling left/right or up/down.
6.0.0	OSBO-1520 1-BBMLJH	This release adds the ability to establish a 3270 session to TIBCO Object Service Broker on Open Systems using Attachmate Reflection Standard Edition 2011 emulator.
6.0.0	OSBO-1519 1-BCAGLV	Previously, the behavior of screens containing multiple scrollable tables in Open Systems differed from the behavior on z/OS. When scrolling to the end of a screen table, the cursor automatically moved to the first field of the next screen table. This has been corrected to match the behavior z/OS, where the cursor does not automatically move.
6.0.0	OSBO-1515 1-BAXLT7	Fixed a problem that could cause incorrect values to display in the Reference Table for a Screen field, after changing the Reference Table options. This occurred on Open Systems when using `USERPROMPT=Y`.
6.0.0	OSBO-1514 1-BB10MV	Previously, `CURSOROCC#` behaved differently on z/OS and Open Systems. On z/OS, placing a cursor on a field returned the occurrence number. On Open Systems, it returned zero. This has been corrected. The actual occurrence number is now returned on both z/OS and Open Systems.
6.0.0	OSBO-1499 1-B8E68I	Fixed a problem that could cause fields to display incorrectly in Open Systems when migrating from z/OS.
6.0.0	OSBO-1493 1-B3M2DP	Previously, repeat invocations of eCTSSession.RunTrans within a session could fail because the ECTSPARMS table was not populated with the full set or arguments. This has been fixed.
6.0.0	OSBO-1424	Fixed an error that prevented the base installer from generating nonconflicting parameters for `osMon` and the DOB, while installing multiple instances of TIBCO Object Service Broker.

Table 6 Closed Issues for TIBCO Object Service Broker for Open Systems

Closed in Release	Key	Summary
6.0.0	OSBO-1422	Previously, the uninstaller did not check if the DOB was running before proceeding. This has been fixed.
6.0.0	OSBO-1417	Fixed an error that could cause the following error message during installation of an upgrade: *path*\bin\hrncr.exe is in use. Unlock this file to allow the installation to continue.
6.0.0	OSBO-1410	Previously, an unecessary error message could be displayed in the Eclipse UI. When closing a project after adding a new object to the project view, this error message would be displayed: Unable to save the file workset.osb because of the exception. This has been fixed.
6.0.0	OSBO-1404	Previously, running uninstall in the installer did not clean up the Start menu. This has been fixed.
6.0.0	OSBO-1399	Closing the installer window while copying files caused errors. This has been fixed.
6.0.0	OSBO-1219	Fixed an issue that caused upgrades on Open Systems to fail when the JOURNALS and REDO folders were located outside their default location in the HURON/database directory.
6.0.0	OSBO-1071 1-9J1I5J	Stricter password access to the workbench is now enforced, allowing the implementation of increased password complexity requirements. To implement, specify SECURITY=OBJECTSTAR or SECURITY=MIXED.
6.0.0	OSBO-499 1-8232I9	Fixed an error that could cause a connection to fail even though no parsing errors were issued. This occurred when huron.dir was improperly coded. Parsing errors have been implemented to help diagnose errors in the coding.
6.0.0	OSBO-465 1-82B4MJ	A Test button is now included when configuring the properties for an ODBC Adapter connection.

Table 6 Closed Issues for TIBCO Object Service Broker for Open Systems

Closed in Release	Key	Summary
6.0.0	OSBO-424 1-7ZVV7L 1-8D9O2V	Fixed a problem that could cause the DOB to crash when the `spinsubmit` command was issued. If a spin was forced using the command `hrncr spinsubmit=i` while a checkpoint was in progress, the DOB would crash because the `spinsubmit` command attempted to start another checkpoint. The command has been corrected to function as on z/OS. The `spinsubmit` command now waits until the checkpoint has completed.
6.0.0	OSBZ-1254 1-BA4UFB	External syntax V fields assigned NULL are given different values on z/OS and Open Systems. The value on z/OS is `0x'40'`, and on open systems it is `0x'00'`. A new EE parameter has been introduced to allow the installation to specify the padding character to be used for external syntax V fields. On z/OS, we use an EBCDIC blank (X'40') as a pad character, and on Open Systems we use a null (X'00') as a pad character. A new parameter has been added to Open Systems to permit installations to choose the same behaviour as on z/OS. The new parameter is `SYNTAXVBLANKPAD`. It is a logical value indicating whether blank padding is to be used for external syntax V fields. The default value is NO, which corresponds to the current Open Systems behaviour, where we pad with nulls. If a value of YES is specified, we will pad with blanks, as is done on z/OS.

Closed Issues for TIBCO Object Service Broker for z/OS

Table 7 lists the closed issues for the TIBCO Object Service Broker for z/OS.

Table 7 Closed Issues for TIBCO Object Service Broker for z/OS

Closed in Release	Key	Summary
6.0.0	OSBZ-1513 1-CWRI4D 1-CYBKHM	Fixed a problem that resulted in repeated S0C1 abends in an EE filling up the spool.
6.0.0	OSBZ-1511 1-CK4QRR	Problems that caused inefficient access to IMS/DB have been addressed.

Table 7 Closed Issues for TIBCO Object Service Broker for z/OS

Closed in Release	Key	Summary
6.0.0	OSBZ-1510 1-CK4QRR	Problem that caused some uses of an incorrect IMS table definition to result in an abend has been fixed.
6.0.0	OSBZ-1505 1-A9RRIF	Issues that caused unnecessary table sweeps when using secondary keys and partial primary keys have been addressed.
6.0.0	OSBZ-1497 1-BDTYE5	Fixed an error that could cause an `Insufficient security clearance` message from the ODBC Adapter when attempting to access tables by way of an OSB system configured with NLS support for non-English code pages.
6.0.0	OSBZ-1496 1-CO8CU5	Customers who have used custom encryption exits to encrypt passwords, may have experienced abends. This has been fixed.
6.0.0	OSBZ-1481 1-CHWN07 1-CJY7A9	Fixed an error that sometimes caused the REMAINDER function to give incorrect results.
6.0.0	OSBZ-1467 1-CDUPUV	After using DB2 Table Definer to view a table definition, subsequent DB2 table accesses used the wrong SERVERID. This has been fixed.
6.0.0	OSBZ-1459 1-CDP0RJ	Previously, ENVIRONMENT-scope storage obtained in a CICS TAM API Session was never released even though the @MAP reference was cleared. This caused a memory leak, which has now been fixed.
6.0.0	OSBZ-1453 1-C9KUC3	After an upgrade from 5.0 to 5.2, customer applications using @MAP tables to access unregistered memory with @USERS_SECURITY.CATAGORIES set to -1 fail with message '`An attempt was made to access unregistered memory`'. This has been fixed.
6.0.0	OSBZ-1406 1-C3WP3B	Fixed an error that prevented GDG datasets associated with an IMP table to be freed after use and the user's EE Session was ended.
6.0.0	OSBZ-1403 1-C3G1WA	In software release 5.2, CLRTAB requests were turned down in BROWSE mode. This has been fixed.
6.0.0	OSBZ-1402 1-C3JN1D	Fixed an error that could cause an S0C4 abend and DOB failure when the DOB STARTPEER command was issued.
6.0.0	OSBZ-1401 1-BVFOUP	The SMF records for software releases 5.0 and 5.5 have invalid release levels in all DOB SMF records. This has been fixed for the 6.0.0 release.

Table 7 Closed Issues for TIBCO Object Service Broker for z/OS

Closed in Release	Key	Summary
6.0.0	OSBZ-1396 1-C1F1RP	Fixed an error that prevented some Admin General statistics PF keys from being displayed on certain 3270 screen models.
6.0.0	OSBZ-1395 1-C0KB53	A change intended to improve the promotion process caused problems for some customers with custom promotions interfaces. This has been fixed.
6.0.0	OSBZ-1392 1-BX2OVP	A lost server condition resulted in a hung end-user session that could not be cancelled. This has been fixed.
6.0.0	OSBZ-1391 1-BX2OVP	A problem where an ADMIN session could be left hanging in the DOB after listing table locks has been fixed.
6.0.0	OSBZ-1390 1-BVFOUP	A small number of macros to map SMF records were missing from the product distribution. This has been fixed.
6.0.0	OSBZ-1376 1-8T1RNN 1-8XQFNL	Previously, the Eclipse UI did not permit users to edit a table with composite primary key with 16 fields. This has been fixed. You are now able to browse and edit TDS, EES, SES tables with 16 primary keys.
6.0.0	OSBZ-1374 1-AXDRXO	A problem caused the message "`S6BPX016E POSIX task initialization error; Set of host and wire encoding failed with status code %`" not to appear in the system log when appropriate. This has been fixed.
6.0.0	OSBZ-1366 1-BW4FDQ	Fixed an error that could cause the DOB to fail with `S6BKS089A HURP ABEND S0C4/U0000 REAS=00000010 DETECTED IN MODULE APPLSRB` after a user canceled his EE session.
6.0.0	OSBZ-1365 1-BVFO0S	In very rare instances, FILETASK could run out of saveareas and abend. This has been fixed. The number of FILETASK saveareas acquired at startup has increased from 10 to 15.
6.0.0	OSBZ-1364 1-BUBUU9	Previously, low CPU availability could in rare cases prevent midnight processing from running. The priority of the TIMRTASK has been increased to the highest below the main task, so it is dispatched immediately.

Table 7 Closed Issues for TIBCO Object Service Broker for z/OS

Closed in Release	Key	Summary
6.0.0	OSBZ-1359 1-BVFOUP	Data layout descriptions for SMF records have been removed from the documentation and replaced with assembler macro sources delivered with every release of the product. This addressed the problem of the documentation being rendered obsolete by interim releases. See pages 84 and 96 of the *TIBCO Object Service Broker for z/OS Monitoring Performance* manual, software release 6.0.0.
6.0.0	OSBZ-1356 1-BTUB4V	Because of a storage overlay, S6BCSRUN could suffer a S0C4 abend at transaction terminal timeout. This has been fixed. Abend HIEA is now forced if storage corruption is detected.
6.0.0	OSBZ-1350 1-BT6Q6D	Fixed a problem that would on rare occasions cause an abend with the message "`HRNKX001E-HRF VSAM HEX REQ=01 RC=08 FB=18 R15=00000008`".
6.0.0	OSBZ-1343 1-BS69Z7	Fixed an error that prevented trailing blanks from being removed from SYSTEMID return values.
6.0.0	OSBZ-1341 1-BS005R 1-C34VFB	Previously, the screen definer could not handle fields larger than QP8. This has been corrected. Support for P9 through P16 field lengths in both the screen definer and report definer/generator has been added.
6.0.0	OSBZ-1333 1-BQ35OJ	Fixed an error that could cause the audit record written to ACCESSLOG contains an invalid length and data for PARM1 field. This error occurred when a DISPLAY_MENU invocation refers to a SCREEN for which the user does not have permissions.

Table 7 Closed Issues for TIBCO Object Service Broker for z/OS

Closed in Release	Key	Summary
6.0.0	OSBZ-1331 1-BPXKO4 1-CIBR3F	Fixed four issues associated with the LOADER utility: • Replace of a table definition would fail if the user running LOADER was not either the owner of the existing table or a level 7 user. • Replace of a table definition failed if BORROWER RIGHTS were held for the existing table, irrespective of whether those rights were held by the same user running LOADER. • LOAD of the complete file (rather than using SELECT) could fail when using PF3 from the Main 'File Loader Utility' screen. • Using LOADER to replace (IR) an existing definition with a modified one containing, for example, a new field, and then attempting to load DATA to that modified table at the same time failed as the old definition is still being reference at the time the DATA is loaded.
6.0.0	OSBZ-1328 1-BC3K0P	Fixed an error that resulted in the message S6BDC031I SESSION ENDED received after initial workbench displayed with IMS/TM. Terminal session ended but not cancelled in NEE or DOB.
6.0.0	OSBZ-1302 1-BNYFMV	Fixed an error that could cause a level 1 user attempting to access a subview of @USERS_SECURITY to fail with 'Denied VIEW_USER access' message.
6.0.0	OSBZ-1299 1-BMGITZ	AUDITLOG has been modified to allow processing by way of an input file as well as from ACCESSLOG. This allows ACCESSLOG data to be used for auditing purposes.
6.0.0	OSBZ-1295 1-BC3K0P	3270 Table Definer now supports Syntax RD for IMS Tables.
6.0.0	OSBZ-1294 1-BLHQJ7	Calling the shareable tool $RULENAME with a negative transaction count can no longer cause a S0C4.
6.0.0	OSBZ-1285 1-BGVGLB	Spin job at startup can complete before DOB is up. As a result, the journal would not be enabled and it was possible to run out of journals and become quiesced. This has been fixed.

Table 7 Closed Issues for TIBCO Object Service Broker for z/OS

Closed in Release	Key	Summary
6.0.0	OSBZ-1284 1-BFU7L7	Database PTFs 13077 and 13080 will no longer fail if data exists in the SLK extract tables.
6.0.0	OSBZ-1278 1-BFGP2N	Fixed an error that caused RULEPRINTER Print Tree option to fail for table definition type SUB. PRINT_TREE failed if 'Detailed print options' is set to Y, and a table of type SUB exists within the tree structure.
6.0.0	OSBZ-1277 1-BEEQUB	Previously, the entry for EES_TABLE is missing from @ALLOWED_TYPES. As a result, it was not possible to copy tables of type EES. This has been fixed.
6.0.0	OSBZ-1275 1-BBLZ49 1-C1CYDT	Previously, the value of TCPNAME parameter in Relay XML file was not being passed to RLYTCPCN. This resuilted in the INITAPI request using the default value for TCPNAME. If a site did not use the default name, the INITAPI failed with ERRNO=1011 (Mesages: S6BRA040 S6BRA005 S6BRA012). This has been fixed.
6.0.0	OSBZ-1274 1-BF6YEL	The logic introduced by PTF 13441 whereby LE initialization and termination occurs under the same TCB is now extended to Batch EE. This should benefit the IDMS/DC region and avoid a possible U4094-18 abend problem.
6.0.0	OSBZ-1271 1-BEERFP	Previously, Table Definer does not allow PARAMETER greater than QP8. This has been fixed.
6.0.0	OSBZ-1268 1-BDF9MH 1-BJQFPZ	Fixed an error in the execution environment that caused an EX057 error message duirng screen validation, and rules fail and return to the workbench.
6.0.0	OSBZ-1267 1-BCEFTZ	Correct error messaging when a blank string is used in rule or table name indirection.

Table 7 Closed Issues for TIBCO Object Service Broker for z/OS

Closed in Release	Key	Summary
6.0.0	OSBZ-1265 1-BC3K0P	Previously, when IMS databases, segments and fields that are initialized to low-values were accessed by TIBCO Object Service Broker, these low values could cause syntax and conversion errors, such as "Data error detected in CONCATENATION. Conversion of PACKED DECIMAL to VARYING CHARACTER failed". TIBCO Object Service Broker now provides an exit point in the IMS server gateway to allow users to process the IMS segment before it is passed back to Object Service Broker and before it is written to IMS again.
6.0.0	OSBZ-1264 1-BBINRL	Previous to 5.2, Object security was not enforced for a reference table check. That is, a reference check was completed properly whether VIEW_DEFN setting for the reference table was 'N' or 'Y'. However, security is now enforced during reference check such that if VIEW_DEFN for a reference table is set to 'N' then reference check is aborted.
6.0.0	OSBZ-1259 1-BAVCC5	Previously, SHUTI could cause abend S378 reason code x'14' 20 in execution environment. This has been fixed.
6.0.0	OSBZ-1257 1-BC3K0P	Platform differences in the handling of external syntaxes X and Z were eliminated.
6.0.0	OSBZ-1256 1-BAPQ63	Fixed a documentation error that incorrectly identified the Maximum Occurrence Length for a MAP table as 32767. The correct length is 31744. This value has been corrected in the *TIBCO Object Service Broker Programming in Rules* manual.
6.0.0	OSBZ-1255 1-BARGUI	Fixed an error that could cause abend U1979 RC=0024029C when running with SRBMODE=Z and no zIIP.
6.0.0	OSBZ-1237 1-A9RRIF	Fixed an error that caused the RE@CALC routine of TDSM1PGS to produce incorrect statistics.
6.0.0	OSBZ-1235 1-B7742F	Previously, the DBJCL job could fail when the DBJCL member of the JCL library created by the upgrade process had OFFSET=UTC on the DBGEN macro. This ocurred because, as of release 5.2, DBGEN does not accept parameters. This has been fixed. the DBJCL no longer has the invalid DBGEN OFFSET=UTC.

Table 7 Closed Issues for TIBCO Object Service Broker for z/OS

Closed in Release	Key	Summary
6.0.0	OSBZ-1234 OSBZ-1249 1-B65VET 1-B65VET	Loop following msg +S6BDR001E ERROR=000F4 S6BDRSES.TAMSYNC +00CA0 VERSION=06/08/07 19.01 V5R0M0123 Fixed an error that sometimes caused the SNAPMAX# limit for HDRSBSNP to be disregarded. As a result, a large number of SNAP dumps were taken when transaction storage was exhausted. A flag is now checked when storage is exhausted, and the SNAPMAX# limit is honored.
6.0.0	OSBZ-1223 1-B65VET	Fixed an error that could cause the message S6BDR001E to show an incorrect "ERROR=" value for programmed S0C6s initiated via $ABEND macro.
6.0.0	OSBZ-1221 1-B6BO3B	Fixed a defect in the S6BKF051I message documentation. The explaination of the message has been updated in the messages manual.
6.0.0	OSBZ-1220 1-B6BO3B	Previously, certain error messages did not correctly report BWO status, and datasets which were not eligible for BWO procesing did not indicate that in the error messages. This has been fixed.
6.0.0	OSBZ-1219 1-B48EWR	Fixed a problem that caused logon to fail when a user attempted to log on from OAI with a rare combination of characters in the user name.
6.0.0	OSBZ-1217 1-B3MBF9 1-B6DLR3	An unexpected delay in processing a remote CLI SDK client request in z/OS Execution Environment could sometimes lead to various failures, including: • abend U2168 • message: `S6BST001E STORAGE MANAGER ERROR INVALID DESCRIPTOR FOR A RELEASE REQUEST` This problem has been fixed.
6.0.0	OSBZ-1216 1-A9RRIF	Spurious table sweeps when using secondary keys and partial primary keys eliminated.
6.0.0	OSBZ-1215 1-B3NIER	Fixed an error that could cause an abend after an `Insufficient memory to run transaction` message.

Table 7 Closed Issues for TIBCO Object Service Broker for z/OS

Closed in Release	Key	Summary
6.0.0	OSBZ-1214 1-B3APDJ	Fixed an error that could cause an abend following the message `S6BRA027E ACCEPT of new connection failed with errno=5.`
6.0.0	OSBZ-1177 1-76SF3H	When a rule accesses a table with a trigger rule and the original rule is Fixed an error that could cause an unrecoverable error in rules when the rule accessed a table using a trigger rule. This would occurr when the original rule was run under debugger.
6.0.0	OSBZ-979 1-A9RRIF	On a very large table, hundreds of millions of rows, the DOB changed from access by way of a secondary index to primary key access. This resulted in the request response time going from subsecond to over an hour. This has been fixed.
6.0.0	OSBZ-978 1-BL6QR0	Internal documentation in supplied CICS definitions improved.
6.0.0	OSBZ-972	The installer did not delete its root item from the Start menu on Windows. This has been fixed.
6.0.0	OSBZ-954	Previously, the manual installation option was available only if you were installing the base TIBCO Object Service Broker product and only if you will not install any of the gateways. Automatic installation was required if you planed to also install the local gateways, such as CICS, DB2, IMS/TM, Files, IMSDB, IDMSDB, and WMQ Both manual and automatic installation are now available for all installation configurations.
6.0.0	OSBZ-934	If you set the language under Regional and Language Options in the TIBCO Object Service Broker installer to a non-English language, that language was in effect until the Installation Type page only. Beyond that page, the language reverted to English. This has been fixed.
6.0.0	OSBZ-932 1-AXQ8G3	Fixed an error that could cause failure or unexpected results when processing PARSER for GRAMMARS('BROWSE_COMMANDS'). Errors occured when the EXPRESSION string matches the name of a known token type such as NUM or ULIT.

Table 7 Closed Issues for TIBCO Object Service Broker for z/OS

Closed in Release	Key	Summary
6.0.0	OSBZ-900 OSBZ-909 1-ANBZJ3	An insufficient storage message followed by a U2168 abend was sometimes seen when using the TIBCO ActiveMatrix BusinessWorks Plug-in for OSB and CICS remote client interface. This has been fixed. Additionally, two new messages were added to indicate the different error conditions for the U2168 abend: • `SESSMEM - SESSIONMEMMAX exceeded` • `SESSMEM - Storage Manager "OBTAIN" error; RC=nn`
6.0.0	OSBZ-898 1-ASFAZX 1-B2TBPD 1-B5L1SP 1-BL6X4V	Previously, a large number of I/Os are performed to the steplib dataset when a large OIG workload was run into CICS. This has been fixed.
6.0.0	OSBZ-867 1-AP32IN	Fixed an error that sometimes caused errors when processing HXCATESP.
6.0.0	OSBZ-861 1-AO1V3H	Fixed an error that could cause an abend during the CLOSEOUT routine of RLYTCPCN. This occurred when the socket was previously closed. The CLOSEOUT routine now verifies that the socket was not previously closed before proceeding.
6.0.0	OSBZ-858 1-ANDWIF	A fatal invalid document structure error can be generated when parsing an XML document, if the document encoding is incorrectly specified in the header. When passing XML documents as string from a rule, the string will be automatically converted to UTF-16. If a different encoding is specified in the header, parsing fails. This behavior is now correctly documented in the TIBCO Object Service Broker *Tools* manual.
6.0.0	OSBZ-850 1-AMGG2U	Changed message HRNKA006 to U1979 abend.
6.0.0	OSBZ-840 1-AIYYQJ	Users can now replace the V5R0M0 qualifier in the distribution datasets with a different value. See the *TIBCO Object Service Broker for z/OS Installing and Operating* guide.
6.0.0	OSBZ-395 1-8X0Q77	Previously, the message `HRNID041E IDMSSRVR: VTAM communication error` incorrectly identified VTAM as the comms method even though XMS or TCP/IP might in fact be the failing component. This has been fixed.

Table 7 Closed Issues for TIBCO Object Service Broker for z/OS

Closed in Release	Key	Summary
6.0.0	OSBZ-374 1-8U10XJ	Fixed an error that prevented diagnostics from being captured on some messages.
6.0.0	OSBZ-217 1-853IMZ	Additional information has been added to parameter error message EX110 when it is issued by HRNDRSES.INTERP. The message now includes information identifying the parameter in error and whether it was calling or on return from the exteral routine.

Closed Issues for TIBCO Service Gateways

Table 8 lists the closed issues for the TIBCO Service Gateway products:

- TIBCO Service Gateway for CICS
- TIBCO Service Gateway for DB2
- TIBCO Service Gateway for IDMS/DB
- TIBCO Service Gateway for IMS/DB
- TIBCO Service Gateway for WMQ (Open Systems)

Table 8 Closed Issues for TIBCO Service Gateway

Closed in Release	Key	Summary
TIBCO Service Gateway for CICS		
6.0.0	SGCI-9 1-B8VXF3	In software release 5.2, migration from 5.0 could fail on step one of installation with a message directing that the base S6A4ACPT job in JOBSA be set to INSTALL. This has been fixed for migration to 6.0.0.
TIBCO Service Gateway for DB2		
6.0.0	SGDB-92 1-D0383J	Fixed an error that could cause Static SQL to fail when complex table/access patterns were in use.
6.0.0	SGDB-91 1-CDG5BV 1-D8LTKF	Several issues regarding correct generation of static SQL have been corrected.

Table 8 Closed Issues for TIBCO Service Gateway

Closed in Release	Key	Summary
6.0.0	SGDB-86 1-CHI03B	Problem where the code generated by the shareable tool @STATICSQL cause assembly errors has been fixed.
6.0.0	SGDB-84 1-CDG5AP	Fixed an error that caused conversion errors when STATIC SQL was enabled.
6.0.0	SGDB-83 1-C1OLB7	Problem where non-matching default date masks for Object Service Broker and DB2 could cause a CONVERSION error has been fixed.
6.0.0	SGDB-77 1-BPKZ7S	Generated static SQL handlers now use the BIND PACKAGE command to provide compatibility with future DB2 release.
6.0.0	SGDB-75 1-BEAKB0 1-CDG5AP	Previously, when use of static SQL was rejected by the DB2 Gateway, no notification was issued and CPU use increased significantly. This release add more comprehensive checking on the Static SQL handler compatibility. An error message is also issued if static SQL is rejected.
6.0.0	SGDB-74 1-BEAKB0 1-CDG5AP	Problem where the lack of FOR UPDATE clause on the GET statement built for a DELETE resulting in cursor ambiguity, which causes a SQLCODE -510 failure, has been fixed.
6.0.0	SGDB-73 1-B2DKZF	Problem in mapping a DB2 table to a table in a remote DB2 subsystem in the Table Definer has been fixed.
TIBCO Service Gateway for IDMS/DB		
6.0.0	SGID-18 1-9GNY9Z	Problem where conflicting trace settings in the Service Gateway for IDMS/DB and the Object Service Broker could cause an abend has been fixed.
TIBCO Service Gateway for IMS/DB		
6.0.0	SGIM-27 1-CK4QRR	Fixed a problem that resulted in a S0C4 abend in EE, resulting from inadequate error handling in CONVERTI.
6.0.0	SGIM-26 1-CK4QRR	Several internal issues combining to affect access efficiency in the Service Gateway for IMS/DB have been addressed.
6.0.0	SGIM-22 1-C3FAIP	Previously, the Table Definer did not warn that updates to an IMS table would be disallowed when its definition was saved, and the definition contained overlapping fields. This has been fixed.

Table 8 Closed Issues for TIBCO Service Gateway

Closed in Release	Key	Summary
TIBCO Service Gateway for WMQ (Open Systems)		
6.0.0	SGWO-4 1-B8E689	Previously, the @MOMINIT cleared the area holding the buffer address and length before saving and reusing old record buffers on subsequent calls. This orphaned session scope buffer storage. The behaviour has been corrected and the resulting memory leak eliminated.

Known Issues

The table in this section lists known issues in this release.

Key	Summary/Workaround
OSBO-1773	**Summary** The OSB client installer does not offer the following installation parameters when run in console mode: • SQL Service Host • SQL Service Port **Workaround** None.
OSBO-1772	**Summary** On Windows, the silent and console modes of the installer display a dialog box before entering console mode. If the installer is invoked from a telnet client (rather than a command prompt), it stops immediately. **Workaround** Start the installer from the command prompt and ignore the dialog box.
OSBO-1771	**Summary** On Windows systems, the silent installer does not honor the node name, DOB port, osMon port or TN3270 port supplied by the user in the options file. **Workaround** After completing the silent installation, manually edit these files to specify the correct parameters: • `crparm` — the `NODENAME` parameter • `huron.dir` — edit the `port` parameter in the node definition for the DOB. • `mon.prm` — edit the `port` and `tn3270port` parameters in the osMon configuration.
OSBO-1769	**Summary** On Windows and Solaris, an attempt to stop osMon (running as a service or standalone application), while successfully stopping osMon and its dependent processes, can sometimes produce a series of error messages in the log ending with message S6BDR038E (Unexpected exception was caught). These messages may effectively be ignored. **Workaround** None.
OSBO-1745	**Summary** The OSB console mode installer displays diagnostic messages during execution. They do not affect operation of the installer. **Workaround** None.

Key	Summary/Workaround
OSBO-1583	**Summary** Very long-running and update-intensive applications may exhaust `RESIDENTPAGES` and encounter the message "`S6BUL033W No free buffers in resident page pool`" one or more times, after which the DOB becomes unresponsive and must be recycled. **Workaround** Increase the value of `RESIDENTPAGES` in `crparm`. Additionally, break the workload of the application up in smaller sections and run them sequentially. Recycle the DOB after each section of work completes.
SGOD-25	**Summary** TIBCO Service Gateway for ODBC is a 32-bit application that can only use 32-bit third-party ODBC drivers for its outbound connections. **Workaround** None.
OSBO-1423	**Summary** The TIBCO Object Service Broker Service Gateway for Files (Open Systems) installer does not provide an upgrade path. **Workaround** In an enablement-mode installation, first uninstall the TIBCO Object Service Broker Release 5.0.0 Gateway and then proceed as if it were a new installation. For a standalone installation, also proceed as if it were a new installation since the Gateway Files installer places the files in a separate 6.0.0 folder in the *TIBCO_HOME*'s OSB Gateways folder.
OSBO-1396	**Summary** After an uninstallation of TIBCO Object Service Broker 6.0.0 on Solaris, `the _uninst`, `database`, `logs`, `runtime`, `_uninstc`, `log`, and `utils` folders together with the `osbinstall.info` file are left behind. **Workaround** Run the uninstaller from a directory other than the one in the installation directory for TIBCO Object Service Broker and refer to the directory's full path. TIBCO recommends that you specify `/tmp` as the current working directory. Delete the directory that contains the uninstaller if you plan to use it for a new installation of the same software. Otherwise, the new installation will place the uninstaller in a newly created `_uninst2` folder. The `_uninst`, `database`, `logs`, `_uninstc`, and `log` folders, as well as the `osbinstall.info` file, are preserved by design.

Key	Summary/Workaround
OSBO-1394	**Summary** On Solaris 10, uninstallation of TIBCO Object Service Broker fails after an upgrade from Release 5.0.0 HotFix 5 to Release 6.0.0 in GUI, console, or silent mode. **Workaround** Run the uninstaller from a directory other than the one in the installation directory for TIBCO Object Service Broker and refer to the directory's full path. TIBCO recommends that you specify `/tmp` as the current working directory. Delete the directory that contains the uninstaller if you plan to use it for a new installation of the same software. Otherwise, the new installation will place the uninstaller in a newly created `_uninst2` folder.
OSBZ-584	**Summary** `S6BRUNXT` allocates the `JOBSA` member with `DISP=SHR` but updates it. **Workaround** This issue is too complex to fix. We have added a warning to the documentation on installation against editing the `JOBSx` member while running an installation job.
OSBO-706	**Summary** A problem with session pooling on OIG/BusinessWorks Plug-in exists: When an EE is restarted (mainframe or portable), the existing sessions in the session pool are not aware that the EE has been reset until the next time the application attempts to use a session. This first attempt results in an error and the access to TIBCO Object Service Broker fails. At this point, the session pool is reset and all subsequent accesses are successful until another EE recycle. The session pool should be able to detect that the EE has been recycled and remove all the sessions. **Workaround** Do not use pooling or do not restart the EE when connected pooled sessions are present. Alternatively, restart the OIG application or BusinessWorks process when restarting the EE.
OSBZ-377	**Summary** The TIBCO Object Service Broker UI cannot cope with TIBCO Object Service Broker security groups. When using TIBCO Object Service Broker security groups to access objects that are controlled by a security group other than the one to which you currently logged in, you must switch to the other security group. **Workaround** On the 3270 workbench, you can change the security group with the UP option. From the TIBCO Object Service Broker UI, you must shut down the UI, log in through a green screen to the workbench, change the security group, log off from TIBCO Object Service Broker, and restart the UI.

Key	Summary/Workaround
OSBO-1482	**Summary** Calling the method `moveToInsertRow()` on a `WebRowSet` object without first establishing the schema type causes a `NullPointerException` error. **Workaround** Call `setSchemaType("ROWSET")` on the `WebRowSet` object before calling `moveToInsertRow()`.
OSBZ-455	**Summary** When using the Rule Editor, a combination of deleting condition statements `AND` with the command-line change command (`CH`) within the same Rule Editor session might cause subsequent editing of the same rule to fail. **Workaround** Avoid deleting condition statements and running the change command within the same Rule Editor session.
OSBZ-1192	**Summary** In remote promotions, a successful Refresh Disposition on the remote system is overwritten by the disposition from the source system on a subsequent apply, resulting in constant returns of `Disposition Mismatch`. **Workaround** Ensure that the disposition of objects is correct on the source system before submitting the change request. For details, see Appendix B, Disposition Consistency Across Systems, in the *TIBCO Object Service Broker Managing Deployment* manual.
OSBZ-520	**Summary** This is a request for enhancing `ALLOWUSERKEYCSA(NO)` so that you can change the z/OS option dynamically to support testing of other software without affecting TIBCO Object Service Broker. **Workaround** You should never dynamically modify `ALLOWUSERKDYCSA` after an IPL once a TIBCO Object Service Broker region has initialized. If all TIBCO Object Service Broker regions have been terminated, you might be able to modify `ALLOWUSERKDYCSA` dynamically, but do not restart any TIBCO Object Service Broker region after doing so.
OSBZ-601	**Summary** This is a request for enhancing NEE error recovery to prevent looping on error recovery, which produces thousands of lines of messages. **Workaround** This problem was a side effect of another problem (1-9HA4GE), which has since been fixed. This problem should not occur any more given the fix of the original problem.
1-86U95X	**Summary** The TIBCO BusinessWorks OSB Plug-in connection resource contains an advanced tab that appears to allow additional TIBCO Object Service Broker session parameters. However, those parameters are ignored. **Workaround** Adjust the parameters on the EE side.

Key	Summary/Workaround
1-8W25AD	**Summary** The S6BDR014E EE main task TCB terminates prematurely due to the unrecoverable error. **Workaround** You must produce XML documents in a particular sequence according to the definition of the XML Documents/Field Maps as one of their interfacing applications processes XML in sequence model. The schema/.xsd information provided by the third party specifies an <xsd:sequence> element, indicating that the following elements and groups of elements must be presented in a particular sequence. Within TIBCO Object Service Broker, the XML Field Maps are built with the elements specified in the sequence requested and the correct elements belonging to groups within the field map. However, when the XML document is produced, it appears that TIBCO Object Service Broker places the individual elements directly after the root element name and then builds the group element name (and subsequent elements) in alphabetical order.
1-91K4GA	**Summary** DUPUSERID=N is not being properly enforced for Eclipse UI sessions. An Eclipse UI session and any spawned sessions can connect even if the same user is already logged in through TSO/NEE/CICS. **Workaround** None.
1-8Z3J29	**Summary** Untrappable CICS transaction abends can cause a TIBCO Object Service Broker user application to hang. In servicing a SYSCALL VSAMCLOSE, an EXEC CICS SYNCPOINT was issued, which abended ASP3 because of problems with a connected OEM database product. The ASP3 abend caused CICS to clean up the transaction and free up the user terminal; however, since it is not trappable by CICS HANDLE, SYSCALL never completed. The user application was designed to be single-threaded and, after the ASP3 abend, the user terminal was freed up but subsequent attempts to run the application failed because SYSCALL was still outstanding. **Workaround** None.
1-AQBRLU	**Summary** The TIBCO BusinessWorks OSB Plug-in installer and uninstaller on SPARC Solaris 10 might cause a core dump if invoked without the full path. **Workaround** Invoke the installer or uninstaller by its absolute path.

Key	Summary/Workaround
1-AIRPB1	**Summary** On z/OS, when an Eclipse UI is connected to an EE, the UI appears to "go to sleep" once you have opened the windows for browsing or editing table functions and the number of those windows exceeds that of the `INTERP` tasks. This is because each of the windows occupies one `INTERP` task and does not release until you close the window. **Workaround** Specify a sufficient value for `TASKEXECNUM` in the `EECONFIG` or `HRNIN` parameter.
OSBZ-890	**Summary** Separate statistics must be provided in `S6BTLADM` and SMF records when DOB is running with `SRBMODE=Y` or `Z`. **Workaround** None for the `S6BTLADM` utility, which shows whether `SRBMODE` processing is enabled on the INSTALLATION/CONFIGURATION screen (option J). If `SRBMODE=Y` or v, where reported, the `QUERY` TCB time includes any CPU time used when Query requests were running in SRB mode. Note that the CPU time is also accounted for in any SRB time field. CPU time is the time the Query request was active on a processor but does not include the relative speed of any individual processor.

OSBZ-894	**Summary** In defining a DB2 table, if you set a compound primary key's first field as a parm and if you do the selection like this:
	`get table where parm = x and non-key-field = Y`
	then TIBCO Object Service Broker passes only the parm as the selection criterion to DB2 and returns all the matching rows to the gateway, in which further selection occurs.
	
	Workaround Do the selection as follows:
	`get table(x) where non-key-field = y`
	That way, TIBCO Object Service Broker passes both fields to DB2 for selection, reducing the I/O and CPU usage.

OSBO-1373	**Summary** While installing a Service Gateway on Solaris, to enable the gateway within an existing TIBCO Object Service Broker installation rather than performing a stand-alone installation, be sure to set up the environment variable `HURON` and point it to the TIBCO Object Service Broker installation directory. That directory is usually `/opt/tibco/osb` or *~userid*`/tibco/osb`. **Workaround** None.

Key	Summary/Workaround
OSBO-1380	**Summary** On some Solaris 10 systems, the uninstallers for gateways and plug-ins leave behind empty directories, including the one that originally contained the uninstaller itself. **Workaround** Delete the directory that contains the uninstaller after uninstalling the product. Otherwise, a subsequent installation of the same product creates another such directory, appends a `2` to the directory name, and places the new uninstaller there. Run the uninstaller from a directory other than the one in the installation directory for TIBCO Object Service Broker and refer to the directory's full path. TIBCO recommends that you specify `/tmp` as the current working directory.