



THE ENTERPRISE MIDDLEWARE SOLUTION

# BEA Jolt

## Release Notes

BEA Jolt 1.1 Release  
Document Edition 1.1b  
September 1997

## Copyright

Copyright © 1996, 1997 BEA Systems, Inc. All Rights Reserved.

## Restricted Rights Legend

This software and documentation is subject to and made available only pursuant to the terms of the BEA Systems License Agreement and may be used or copied only in accordance with the terms of that agreement. It is against the law to copy the software except as specifically allowed in the agreement. This document may not, in whole or in part, be copied photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from BEA Systems, Inc.

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the BEA Systems License Agreement and in subparagraph (c)(1) of the Commercial Computer Software-Restricted Rights Clause at FAR 52.227-19; subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013, subparagraph (d) of the Commercial Computer Software--Licensing clause at NASA FAR supplement 16-52.227-86; or their equivalent.

Information in this document is subject to change without notice and does not represent a commitment on the part of BEA Systems. THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, BEA Systems DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE SOFTWARE OR WRITTEN MATERIAL IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

## Trademarks or Service Marks

BEA, BEA Connect, BEA Jolt, Distributed Application Framework, and Enterprise Middleware Solutions are trademarks of and are developed and licensed by BEA Systems, Inc., Sunnyvale, California. TUXEDO is a registered trademark of Novell, Inc., exclusively licensed to BEA Systems, Inc.

All other company names may be trademarks of the respective companies with which they are associated.

### Jolt Release Notes

Document Edition	Part Number	Date	Software Version
1.0	801-001104-001	November 1996	Jolt Release 1.0
1.0.1	801-001104-02	February 1997	Jolt Release 1.0.1
1.1b	801-001104-03	September 1997	Jolt Release 1.1

---

# Contents

## Read Me First!

BEA TUXEDO 6.1 and 6.2 Users.....	1
Windows NT Installation .....	1
Jolt 1.1 General.....	1

## Release Notes

About this Jolt Release .....	3
New Features in Release 1.1 .....	3
Encryption .....	3
Internet Relay .....	4
Event Subscriptions.....	4
International Character Conversion Support.....	5
Repository Bulk Loader .....	5
New Features to Support TUXEDO 6.3.....	5
Upgrading to BEA Jolt 1.1 .....	6
Before You Start.....	6
Upgrading TUXEDO and Jolt.....	7
Removing Jolt 1.0 Files and Directories .....	7
File Ownership .....	8
Upgrading to Jolt 1.1 .....	8
Supported Platforms .....	9
Jolt Server.....	9
Jolt Relay .....	10
Jolt Client Classes .....	10
Miscellaneous Issues .....	11
Known Limitations .....	14
Known Platform Networking Limitations.....	18

---

Software Problems Fixed in Jolt 1.1 .....	19
Contacting BEA Technical Support .....	20

# Read Me First!

The following items can affect your BEA Jolt 1.1 software installation. Before installing Jolt 1.1 software, please note these limitations and read the *BEA Jolt Release Notes*.

## BEA TUXEDO 6.1 and 6.2 Users

If you are using the BEA TUXEDO 6.1 or 6.2 RTK, you need the `libgp.a` file. For more information, see “Miscellaneous Issues” in the *BEA Jolt Release Notes*.

## Windows NT Installation

The following requirements or limitations affect your Windows NT installation:

- ◆ The Jolt 1.1 installation overwrites an existing `jrepository` file. See the CR# 010018 entry in “Known Limitations” in the *BEA Jolt Release Notes*.
- ◆ For your Windows NT installation of Jolt 1.1 software, you may need to contact the BEA Customer Support Center and request a software patch to resolve #CR 1c97-18801 (the TUXEDO Windows NT rolling patch). See the 1c97-18801 entry in “Known Limitations” and “Contacting BEA Technical Support” in the *BEA Jolt Release Notes*.

## Jolt 1.1 General

You need to correctly set up the CLOPT clause in the `UBBCONFIG` file. To get command-line options, the Jolt Server Listener (JSL) parses the CLOPT options that occur in the string `CLOPT="-- optionletters"`. If the CLOPT line contains any illegal option, the JSL stops parsing the CLOPT clause completely when the illegal option is first encountered. The only legal CLOPT options are: `-H`, `-D`, `-M`, `-T`, `-d`, `-m`, `-n`, `-t`, `-I`, `-w`, `-x`, `-c`, and `-z`. For more information, see “Miscellaneous Issues” in the *BEA Jolt Release Notes*.

*READ ME FIRST!*

---

# Release Notes

## About this Jolt Release

Welcome to BEA Jolt, Version 1.1. This version of Jolt contains new features that add additional security, reliability and flexibility to your Jolt client application development. These *Release Notes* provide a brief overview of each new feature and contain essential information to get you up and running with BEA Jolt software.

**Note:** The *BEA Jolt Release Notes* present changes or updates to the standard documentation provided with your Jolt 1.1 product release. In all cases, these *Release Notes* contain the most up-to-date information for the Jolt 1.1 product.

Before you install BEA Jolt 1.1 software, read this document in conjunction with the installation section in the *BEA Jolt User's Guide*. You can browse the HTML version of the *BEA Jolt User's Guide* on your distribution medium and via the BEA Systems Web site at <http://www.beasys.com/products/jolt/index.htm>.

## New Features in Release 1.1

The following sections describe the new features added for the BEA Jolt 1.1 software release.

### Encryption

When you enable encryption support, Jolt 1.1 uses a combination of DES and RC4 encryption algorithms to encrypt all data that is transmitted between a Jolt client and a Jolt server. DES (56-bit) encryption is used for key exchange and authentication, and

RC4 is used for all subsequent data exchanges. Jolt 1.1 features two versions of encryption support: one allows a 128-bit key for data encryption, the other allows a 40-bit key.

The 128-bit version generates a 128-bit RC4 session encryption key at logon time, and this session key is transmitted over the network in a message protected by 56-bit DES encryption. The temporary session key is then used to encrypt the session's data. The overall security level achieved is equivalent to 56-bit DES encryption, even though the data encryption is using a 128-bit key.

These two encryption versions of Jolt 1.1 (40-bit and 56/128-bit) are on separate CDs. All references in the *BEA Jolt User's Guide* to 128-bit encryption refer to the 56/128-bit capability as described in this paragraph.

**Note:** The 56/128-bit version of Jolt 1.1 cannot be used outside the United States without proper approval from the United States government.

The default for Jolt 1.1 is no encryption. The encryption feature can be turned on and off administratively using a Jolt Server Listener (JSL) command-line option in the `UBBCONFIG` file.

## Internet Relay

The Jolt Internet Relay removes the need for the Jolt Servers to be installed on the same machine as the Web Server software. The Jolt Relay consists of two components, Jolt Relay (JRLY) and Jolt Relay Adapter (JRAD), that work as a pair and route client requests to the appropriate Jolt Server. The Jolt Relay is an optional component and is transparent to both the Jolt client and the Jolt Servers. The Jolt Server can concurrently have clients connected via a Jolt Relay and clients connected directly to the Server.

## Event Subscriptions

Jolt 1.1 clients can subscribe to TUXEDO events and to unsolicited notifications from TUXEDO servers or clients. Jolt clients cannot post events or generate unsolicited notifications, they can only receive them.

A new API class, `JoltUserEvent`, has been added to support events and unsolicited notifications. The `JoltSession` class has been enhanced to provide an `onReply` method, which is invoked when events or notifications are received.



## International Character Conversion Support

Jolt 1.1 allows for the use of strings containing 16-bit characters as parameters for TUXEDO services. The character conversion routine can be customized by application developers, using the dynamic library `libjconv`. The Jolt default version of `libjconv` supports 8-bit Latin-1 characters. To use 16-bit characters, you need a version of TUXEDO that supports 16-bit characters.

## Repository Bulk Loader

The Jolt repository contains service definitions that you can add using the GUI-based Repository Editor. However, loading large numbers of service definitions can be time-consuming. Therefore, Jolt 1.1 also includes a repository bulk loader. The Jolt 1.1 bulk loader utility takes its input from a text file and bulk loads multiple service definitions into the Jolt repository.

## New Features to Support TUXEDO 6.3

Jolt 1.1 works with TUXEDO 6.1, 6.2, and 6.3. All features of Jolt 1.1 are constant across all these releases of TUXEDO, except where new features have been added for TUXEDO 6.3. The new features of TUXEDO 6.3 supported by Jolt 1.1 are:

- ◆ Machine IP addresses and ports can be specified using Internet style syntax. The new supported style is `//hostname:port`, (e.g., `//myserver:3000`). The TUXEDO 6.1 hexadecimal notation is also supported (i.e., `0x0002PPPPNNNNNNNN`).
- ◆ Acknowledged notifications are supported. Refer to the TUXEDO 6.3 documentation for details of acknowledged notifications. See the “Known Limitations” section.
- ◆ Support for Network Translators using the `-H` option to JSL.

# Upgrading to BEA Jolt 1.1

The *BEA Jolt User's Guide* contains complete Jolt 1.1 installation instructions. This section highlights steps required to upgrade an existing installation of Jolt 1.0 or 1.0.1 to Jolt 1.1.

## Before You Start

Before you start any upgrades, make a backup of any existing Jolt installation that you have on your system. Table 1 lists the files/directories where Jolt files are typically installed. If you have a previous Jolt installation at these locations, you should do a backup.

**Table 1 Jolt Directories for Previous Jolt Releases**

Items	Default Location	Contains
JSL, JSH, JREPSVR	\$TUXDIR/bin	Jolt Server and Jolt Repository server executables.
libjnw1	\$TUXDIR/lib on UNIX systems \$TUXDIR/bin on Windows NT systems	Dynamic library for the Jolt server. The suffix of the library is either .so, .a, .sl or .dll, depending on the OS platform.
jrepository	\$TUXDIR/misc	Default Jolt Repository file. This file is pointed to by the -P option to JREPSVR. <b>This file may contain application service definitions. Do not delete it.</b>
Jolt Class Library [classes/bea/jolt atm/*.class]	\$TUXDIR/classes	Jolt client class library. The directory classes contain bea/jolt and bea/atm.
Jolt UNIX online documentation files Jolt NT online documentation files	\$TUXDIR/jolt/doc \$TUXDIR/docs	Online HTML documentation.
jrep.f16	\$TUXDIR/udataobj	FML field tables used in Jolt software.

## Upgrading TUXEDO and Jolt

Jolt 1.1 is available for TUXEDO versions 6.1, 6.2, and 6.3. If you are **upgrading both TUXEDO and Jolt**: upgrade TUXEDO first; then, upgrade Jolt. For information about upgrading TUXEDO, refer to your BEA TUXEDO documentation.

## Removing Jolt 1.0 Files and Directories

The Jolt 1.1 release uses different default installation locations for some Jolt components than previous Jolt versions. Table 2 shows these changes.

**Table 2 Default Component Locations**

Jolt Component	Default Location for Prior Jolt Version	Jolt 1.1 Default Location
Jolt Class Library	\$TUXDIR/classes	\$TUXDIR/udataobj/jolt/client/classes
Jolt Online Documentation	\$TUXDIR/joltdoc (UNIX) \$TUXDIR/docs (NT)	\$TUXDIR/udataobj/jolt/doc \$TUXDIR/udataobj/jolt/docs
Jolt Repository Template file (jrepository)	\$TUXDIR/misc \$TUXDIR/misc	\$TUXDIR/udataobj (UNIX) \$TUXDIR/udataobj/jolt/repository (NT)

After you have upgraded from a previous Jolt version to Jolt 1.1, locate and remove the Jolt directories shown in Table 2, Default Component Locations. The installation program on Windows NT systems should do this automatically, but the installation program on UNIX platforms will leave the original directories intact. The locations shown in the table are the default locations for these files—the actual location is dependent on the options selected when you installed your previous version of Jolt (E.A., 1.0, 1.0.1).

**Note:** Leaving these directories intact does not necessarily impact the performance of Jolt 1.1. However, it could be confusing for application administrators to have access to redundant files. If these files are accidentally used in place of the Jolt 1.1 files, you may receive an error (e.g, using Jolt 1.0 client classes instead of Jolt 1.1 client classes will render the client unable to logon).

## File Ownership

When installing Jolt 1.1 on UNIX platforms, the owner and group attributes of the installed files are set to the user ID and group ID of the current user (i.e., the user you are logged in as when you run the installation program). Since the default installation location for files is in `$TUXDIR`, make sure that the current user has permission to write into this directory.

## Upgrading to Jolt 1.1

When you have completed the pre-upgrade tasks, follow these steps to install Jolt 1.1:

1. Shut down the Jolt Servers if they are running:  
`tmshutdown -g <JOLT-SERVER-GROUP>`  
or `tmshutdown` to shut down the complete application.
2. Make a backup of your Jolt Repository file (`jrepository`).
3. Remove files from previous Jolt installations, as described in Table 2.
4. Mount the Jolt 1.1 CD.
5. On Windows NT platforms, change to the appropriate directory depending on your hardware/OS version and TUXEDO version, then run `setup.exe`.
6. On UNIX platforms, change to the `unix` directory on the CD and run `install.sh`.
7. Restore your original Jolt Repository file (the one you backed up in step 2).  
**Note:** You may need to modify the `UBBCONFIG` file to match the new `jrepository` path.
8. Restart the Jolt 1.1 Servers:  
`tmboot -g <JOLT-SERVER-GROUP>`  
or `tmboot` to boot the entire application.

9. Use the Jolt 1.1 client classes. Jolt 1.0 clients cannot work with Jolt 1.1 servers and vice-versa.

**Note:** If you are using the `getOccurrenceCount()` method, you need to recompile.

10. Test the environment using the Jolt Repository Editor. You should be able to access the repository and read the definitions of all your original services.

## Supported Platforms

Jolt 1.1 consists of several components that are available on various platforms. For the purpose of platform coverage the components are divided into:

- ◆ Jolt Server (JSL, JSH, JREPSVR)
- ◆ Jolt Relay (JRLY and JRAD)
- ◆ Jolt client class library (Java classes)

**Note:** The JSL, JSH, JREPSVR, and JRAD components of Jolt 1.1 are dependent on TUXEDO. In the following list of platforms, these components are only available on those platforms where the corresponding version of TUXEDO is available (e.g., JSL for TUXEDO 6.3 is not available on SPARC/Solaris 2.4, since TUXEDO 6.3 is not available on SPARC/Solaris 2.4).

## Jolt Server

The Jolt Server consists of the Jolt Server Listener (JSL), the Jolt Server Handler (JSH), the Jolt Repository Server (JREPSVR) and associated libraries (`libjnw1` and `libjconv`).

The Jolt Server is available on the following platforms:

- ◆ Solaris 2.4, 2.5, 2.5.1 on SPARC / UltraSPARC as appropriate
- ◆ HP-UX 10.10, 10.20 on HP 9000 series 800
- ◆ AIX 4.1.4, 4.2 on IBM RS/6000

- ◆ Digital UNIX 3.2, 4.0 on DEC Alpha
- ◆ DYNIX/ptx 4.2 on Sequent i386
- ◆ SGI IRIX 6.2 on Silicon Graphics IP22
- ◆ UnixWare 2.1 on Intel
- ◆ Windows NT 4.0 on DEC Alpha
- ◆ Windows NT 3.51 and 4.0 on Intel

## Jolt Relay

The Jolt Relay consists of the Jolt Relay front-end (JRLY) and back-end (JRAD).

The JRAD is dependent on TUXEDO and is available on the same platforms as the Jolt Server (see the previous list).

The JRLY is not dependent on TUXEDO and is available on the following platforms:

- ◆ Solaris 2.5 on SPARC
- ◆ HP-UX 10.10, 10.20 on HP 9000 series
- ◆ Windows NT 4.0 on Intel (requires Service Pack Level 3)

## Jolt Client Classes

The Jolt client classes are compiled using the `javac` compiler from Java Developer's Kit (JDK) 1.0.2. These classes are available on all platforms that support JDK 1.0.2. The following considerations apply when using the Jolt class library:

- ◆ Jolt client classes do not work on HP-UX platforms that use the HP Virtual Machine for Java Releases 1.0.3 and 1.0.2.
- ◆ With the JDK 1.0.2 socket implementation, when a Jolt client cannot successfully open a socket connection to the server, the Jolt client hangs.

# Miscellaneous Issues

When installing and using the Jolt 1.1 release, you may encounter some of the following issues that can make Jolt or a Jolt component inoperable.

Table 3 describes these issues and provides a workaround.

**Table 3 Jolt Troubleshooting Issues**

<b>1. Jolt Repository utilities require connection-retained servers.</b>	
<b>Problem</b>	If you configure a JSL with the <code>-c RECONNECT</code> option, the Jolt Repository Editor and the Jolt Repository Bulk Loader cannot connect to the JSL because both utilities require a RETAINED connection.
<b>Platform</b>	All platforms.
<b>Workaround</b>	Do not configure the JSL with <code>-c RECONNECT</code> when using the Jolt Repository Editor or the Jolt Repository Bulk Loader.
<b>2. Using connectionless clients.</b>	
<b>Problem</b>	If a connectionless client is in a disconnected state (i.e., the network connection has been closed), and the client is terminated abnormally (e.g., the process is killed), the Jolt Server is not aware that the client is dead.
<b>Platform</b>	All platforms.
<b>Workaround</b>	When using connectionless clients, use the <code>-T</code> option with the Jolt Servers. The <code>-T</code> option specifies the maximum length of time a client can be inactive before the JSH automatically terminates the client session.
<b>3. Configuring “slow” clients.</b>	
<b>Problem</b>	During logon, once the JSL has assigned a JSH to a Jolt client, the JSL allows the client a specific time (60 seconds default) to contact the JSH. In slow environments (e.g., when using slow networks), 10 seconds might not be adequate and clients might not be allowed to complete their logon.
<b>Platform</b>	All platforms.
<b>Workaround</b>	When operating in environments where clients could take more than 10 seconds to contact a JSH, increase the default timeout by using the <code>-I</code> option on the JSL.

**Table 3 Jolt Troubleshooting Issues**

<b>4. Unable to read the Jolt installation program from the CD.</b>	
<b>Problem</b>	When mounting the Jolt CD on certain platforms, some operating system specific command-line options are required. If these are not specified, you cannot read the Jolt CD properly.
<b>Platform</b>	Digital UNIX, HP-UX
<b>Workaround</b>	<p>Use the following command-line options when mounting the Jolt CD on the following platforms:</p> <ul style="list-style-type: none"> <li>◆ Digital UNIX: Use the option <code>-o noversion</code>. If you do not use this option, all files and directory names display in uppercase and the file names end with “;1”.</li> <li>◆ HP-UX: Before mounting the CD, start the appropriate daemons as shown in the following example: <pre>% nohup pfs_mountd &amp;  % nohup pfsd &amp;</pre> <p>Mount the CD with the <code>pfs_mount</code> command:</p> <pre>% pfs_mount -o xlat=unix /dev/dsk/dev_file /mount_point</pre> </li> </ul>
<b>5. Entering incorrect options in the CLOPT clause of the ubbconfig file can prevent JSL from starting.</b>	
<b>Problem</b>	<p>The Jolt Server Listener (JSL) parses the CLOPT options that occur in the string <code>CLOPT="-- optionletters"</code> to get command-line options. The following problems can occur if you enter an incorrect option in the CLOPT line or if it already contains an incorrect option:</p> <ul style="list-style-type: none"> <li>◆ When the JSL first encounters an incorrect option, it stops parsing the CLOPT clause and uses the options it encountered up to the incorrect one.</li> <li>◆ If the incorrect option occurs before the <code>-n</code> option, the JSL will not parse the <code>-n</code> option and may not start.</li> <li>◆ The JSL may be started successfully with some of the built-in default options. For example, if the JSL encounters an incorrect option before the <code>-z</code> option (the encryption option), the JSL will be forced to use the default value (NO ENCRYPTION). This may not be ideal.</li> </ul>



**Table 3 Jolt Troubleshooting Issues**

<b>Platform</b>	All platforms
<b>Workaround</b>	Make sure that you have constructed CLOPT correctly. The only legal options are: -H, -D, -M, -T, -d, -m, -n, -t, -I, -w, -x, -c, and -Z. Check the TUXEDO user log for any warning messages on the JSL startup.
<b>6.</b>	<b>The libgp.a file is required for BEA TUXEDO 6.1 RTK.</b>
<b>Problem</b>	The <code>libgp.a</code> file is required for proper installation of BEA Jolt 1.1 software with the BEA TUXEDO 6.1 RTK.
<b>Platform</b>	All platforms.
<b>Workaround</b>	Before installing Jolt 1.1, either obtain the <code>libgp.a</code> file from BEA Technical Support or create a dummy <code>libgp.a</code> file in the <code>\$TUXDIR</code> directory: <code>\$TUXDIR/libgp.a</code> After you install Jolt 1.1, remove the dummy file.
<b>7.</b>	<b>The <code>uninst.exe</code> on NT only uninstalls the last component.</b>
<b>Problem</b>	The <code>uninst.exe</code> on Windows NT platforms only uninstalls the last component.
<b>Platform</b>	Windows NT
<b>Workaround</b>	This is an Windows NT specific limitation. You may need to perform a manual uninstall of Jolt components on Windows NT systems.

# Known Limitations

This section describes limitations in the current BEA Jolt 1.1 release and includes possible workarounds, where applicable.

Table 4 lists a CR (Change Request) number for each problem. Refer to this number to conveniently track the solution as the problems are resolved.

Please contact BEA Technical Support for assistance in tracking any unresolved problems. For contact information, see the section “Contacting BEA Technical Support.”

**Table 4 Jolt Change Requests**

<b>CR# 141</b>	<b>jrly might trim the last character of the last line in its configuration file.</b>	
	<b>Problem</b>	If the last line in the JRLY configuration file does not end with a newline character, the jrly program will ignore the last character of the last line. Depending on what is in the last line of the configuration file, this might result in incorrect configuration.
	<b>Platform</b>	All platforms.
	<b>Workaround</b>	Make sure that the last line in the jrly configuration file ends with a newline character (ASCII 10 decimal).
<b>CR# 010001</b>	<b>When upgrading from a previous Jolt release to Jolt 1.1 some directories are not removed.</b>	
	<b>Problem</b>	When upgrading from previous Jolt releases , the directories \$TUXDIR/classes and \$TUXDIR/docs from Jolt 1.0 are not removed. The new classes and documentation are installed in \$TUXDIR/udataobj/jolt.
	<b>Platform</b>	All platforms.
	<b>Workaround</b>	Remove these directories manually.

**Table 4 Jolt Change Requests**

<b>CR# 010003</b>	<b>Jolt Relay front-end (jrly) installed for Solaris 2.4 and 2.5.1.</b>	
	<b>Problem</b>	The Jolt Relay Front-End (JRLY) has not been certified on Solaris 2.4 and 2.5.1. However, this component installs when the All option is selected during installation.
	<b>Platform</b>	Solaris 2.4 and 2.5.1
	<b>Workaround</b>	Remove this component manually following installation.
<b>CR# 010008</b>	<b>JoltSession attributes TUXVERSION and JOLTVERSION are not supported.</b>	
	<b>Problem</b>	The attributes TUXVERSION and JOLTVERSION in the class JoltSessionAttributes are not supported in Jolt 1.1. They still exist for compatibility reasons.
	<b>Platform</b>	All platforms.
	<b>Workaround</b>	Do not reference the values of these attributes. The attributes will be removed in a future version of Jolt. Remove references to these attributes in your existing applications.
<b>CR# 010016</b>	<b>Existing Jolt-specific files in \$TUXDIR/udataobj are deleted during installation.</b>	
	<b>Problem</b>	The install.sh uses \$TUXDIR as the temporary directory when copying files from the Jolt 1.1 CD. Any Jolt files in \$TUXDIR/udataobj from any previous installations are automatically removed. The overwritten files are: jrep.fl6, jwsladmin.fl32, and the jolt directory.
	<b>Platform</b>	UNIX platforms
	<b>Workaround</b>	If you have a previous Jolt release installed and you want to preserve your existing Jolt files in \$TUXDIR/udataobj, copy these files manually before installing Jolt 1.1.
<b>CR# 010017</b>	<b>Installation program specifies TUXEDO 6.1 as option for AIX 4.2.</b>	
	<b>Problem</b>	TUXEDO 6.1 is not available on AIX4.2; however, the Jolt installation program provides this as an option.
	<b>Platform</b>	AIX 4.2
	<b>Workaround</b>	Do not select this option. A release of TUXEDO 6.1 is not available for AIX 4.2.

**Table 4 Jolt Change Requests**

<b>CR# 010018</b>	<b>jrepository file is overwritten during installation.</b>	
	<b>Problem</b>	If Jolt 1.1 is installed over an existing Jolt installation, the existing Jolt repository file (jrepository) is overwritten without being backed up to jrepository.bak.
	<b>Platform</b>	Windows NT
	<b>Workaround</b>	Make a backup of the Jolt repository file jrepository before performing the installation.
<b>CR# 010020</b>	<b>Jolt 1.1 installation cannot overwrite existing installations of Jolt.</b>	
	<b>Problem</b>	For previous Jolt releases, the online documentation directory is created with read-only permission. If you attempt to install Jolt 1.1 documentation files to the same location the installation program fails with an error indicating that some files cannot be copied. The error message is OS dependent, but will indicate that the copy command failed, for example: cp: cannot create ..... Permission denied
	<b>Platform</b>	All UNIX platforms
	<b>Workaround</b>	If you have documentation for a previous Jolt release installed in \$TUXDIR/udataobj/jolt, delete this directory manually before installing Jolt 1.1.
<b>CR# 010021</b>	<b>jrepository file on UNIX platforms is installed in the \$TUXDIR/udataobj directory.</b>	
	<b>Problem</b>	The template Jolt repository file jrepository is installed in \$TUXDIR/udataobj instead of \$TUXDIR/udataobj/jolt/repository.
	<b>Platform</b>	All UNIX platforms
	<b>Workaround</b>	Create the \$TUXDIR/udataobj/jolt/repository directory and move the jrepository file to the created directory.

**Table 4 Jolt Change Requests**

<b>CR# 010032</b>	<b>When the Jolt server encounters an internal error, the client gets an exception with TPEJOLT.</b>	
	<b>Problem</b>	The client gets an exception with TPEJOLT and the <code>u.log</code> file has either a 1059, 1060 or 1535 error message, when the Jolt server has encountered an internal error.
	<b>Platform</b>	All server platforms
	<b>Workaround</b>	There is no workaround currently available. Contact the BEA Customer Support Center.
<b>CR# 010033</b>	<b>JSH crashes after an error message 1060 is logged.</b>	
	<b>Problem</b>	<p>JSH crashes after logging the following error message in the <code>u.log</code> file:</p> <pre>1060 ERROR: Received message with context associated with a different connection.</pre> <p>On certain platforms, a core file is generated.</p>
	<b>Platform</b>	<ul style="list-style-type: none"> <li>◆ Digital UNIX 3.2, 4.0 on DEC Alpha</li> <li>◆ DYNIX/ptx 4.2 on Sequent i386</li> <li>◆ Windows NT 3.51 and 4.0 on Intel-based and DEC Alpha</li> <li>◆ UnixWare 2.1 on Intel</li> </ul>
	<b>Workaround</b>	Contact the BEA Customer Support Center and request a software patch to resolve CR# 010033.
<b>CR# 010034</b>	<b>The client program may hang when calling <code>JoltSession.endSession()</code>.</b>	
	<b>Problem</b>	<p>This error only applies to application development.</p> <p>The client program may hang when calling <code>JoltSession.endSession()</code>.</p>
	<b>Platform</b>	Solaris 2.4 and 2.5 with JDK 1.0.2
	<b>Workaround</b>	<p>Don't call a <code>JoltSession.endSession()</code>. Allow the Java garbage collection to call the <code>Jolt endSession()</code>.</p> <p><b>Note:</b> In JDK 1.1, JavaSoft has fixed a socket bug that causes this error.</p>

**Table 4 Jolt Change Requests**

<b>lc97-018801</b>	<b>Jolt Servers (JSL and JSH) may not accept new connections after processing many clients and may require a software patch (lc97-18801).</b>	
	<b>Problem</b>	A single Jolt Server does not always recycle closed connections when accepting a new client connection. The number of new clients that a JSL can accept decreases over time, eventually reaching 0. When this occurs, Jolt clients cannot connect to the JSL. In the worst case, this may occur after 63 clients have been processed by the JSL.
	<b>Platform</b>	Windows NT (Jolt 1.1 for TUXEDO 6.1, 6.2, and 6.3).
	<b>Workaround</b>	<ol style="list-style-type: none"> <li>1. A software patch to fix the problem is available (see Action below).</li> <li>2. The following steps may help minimize the impact of this limitation: <ul style="list-style-type: none"> <li>◆ Do not use connectionless clients. Each time a Jolt client connects and disconnects from the JSL, network resources are lost in the JSH. This reduces the ability of the JSH to accept new client connections or to process reconnected clients.</li> <li>◆ Configure several JSL servers each supporting a smaller number (less than 63) of clients.</li> </ul> </li> </ol>
	<b>Action</b>	Contact the BEA Customer Support Center and request a software patch to resolve lc97-18801 (the TUXEDO Windows NT rolling patch).

## Known Platform Networking Limitations

For Windows NT systems, there is a TUXEDO system limit of 64 network sockets per process. When configuring a JSL server on Windows NT systems, there is a limit of 63 for the `-x` multiplexing value.

When you configure a JRAD server on Windows NT systems as part of a Jolt Internet Relay, there is also a limit of 64 active network connections per JRAD server. One connection is used to communicate with the JRLY process, and a maximum of 63 active network connections are available for carrying client communications to the corresponding JSL and JSH processes.

# Software Problems Fixed in Jolt 1.1

In the Jolt 1.1 release, we have fixed software problem that occurred in the Jolt 1.0 and Jolt 1.0.1 releases. Table 5 lists these fixes.

**Table 5 Resolved Jolt Problems**

ID	Problem	Comments
1.	Jolt STRING parameters cannot be greater than 64K in length.	STRINGS can now be up to 2Gb in length.  <b>Note:</b> The ability of an application to use large data sizes depends on hardware and OS capabilities.
2.	On Windows NT systems, JREPSVR reader server(s) cannot detect updates to the repository made by the JREPSVR writer server.	Fixed.
3.	The JREPSVR writer leaves temporary file lying around after the repository has been changed.	Fixed.
4.	The Jolt Repository does not support different input and output buffer types.	Fixed.
5.	The <code>JoltRemoteService.getOccurence()</code> method is misspelled.	Name changed to <code>getOccurrenceCount()</code> .
6.	Jolt Client cannot read messages greater than ~40K.	The client can now read messages of any size.
7.	The JREPSVR cannot read service definitions that are greater than 1K in length.	Service definitions can be up to 16K in length.
8.	In a VIEW buffer, if a STRING field is not NULL-terminated, the following field is concatenated to the STRING field.	STRING fields are now truncated to the size of the field (as specified in the VIEW definition file).
9.	Jolt clients cannot use an application password with a length greater than 8 characters.	Fixed.
10.	The JSH requires FLDTBLDIR to explicitly include \$TUXDIR/udataobj.	\$TUXDIR/udataobj is included automatically by JSH.
11.	User Password cannot contain a NULL character	Fixed.

**Table 5 Resolved Jolt Problems**

12.	Jolt clients could get RECVMTIMEOUT errors when invoking long-running transactions.	When running a transaction, the RECVMTIMEOUT value, for service calls related to the current transaction, is automatically set to be greater than the current transaction timeout.
13.	The Jolt Repository Editor does not support input or output data values for CARRAY fields.	CARRAY fields can now be read and written using hexadecimal data format (e.g., 0a8fde51).

## Contacting BEA Technical Support

If you have any questions about this version of BEA Jolt, or if you have problems installing and running Jolt, contact BEA Technical Support at one of the following telephone numbers or use the email addresses:

### **North America Support Center**

Sunnyvale, CA USA  
M-F 5:00 am - 6:00 pm Pacific Time  
1-888-232-7878  
1-408-743-4070  
1-408-743-4071 fax  
email: [support@beasys.com](mailto:support@beasys.com)

### **European Support Center**

Paris, France  
M-F 9:00 am - 6:00 pm GMT +1  
+33-1-41-45-7090  
+33-1-41-45-7009 fax  
email: [support@beasys.fr](mailto:support@beasys.fr)

### **Asia Pacific Support Center**

Brisbane, Australia  
M-F 8:30 am - 5:30 pm  
+61-7-3255-0506  
+61-7-3255-0441 fax  
email: [support@beasys.com.au](mailto:support@beasys.com.au)

### **Japan Support Center**

Yokohama, Japan  
M-F 9:00 am - 6:00 pm



+81-4-5224-1250

+81-4-5224-1251 fax

email: [support@beasys.co.jp](mailto:support@beasys.co.jp)

