BEA TUXEDO®
RELEASE NOTES

VERSION 9.0

Document Revised: February 08, 2006
Part Number: 801-001002-010

Copyright

Restricted Rights Legend

Trademarks or Service Marks

Contents

BEA Tuxedo 9.0 Release Notes

A. Software and Documentation Problems Fixed in Tuxedo 9.0

BEA Tuxedo 9.0 Release Notes

BEA Tuxedo Release 9.0
Date: November 2007

Table 1 Revision History

Revision Date	Summary of Change
November 2007	• Updated "Software Supported by Each Platform" in this document for post-GA platform data information. Added following post-GA platforms: Red Hat Linux Enterprise Server 4.0 (64-bit) on Itanium, Sun Microsystems Solaris 10 (32-bit) on SPARC, Sun Microsystems Solaris 10 (64-bit) on SPARC. Updated "Appendix A, BEA Tuxedo 9.0 Platform Data Sheets," of Installing the BEA Tuxedo System. The revised document can be found at the following URL: http://edocs.bea.com/tuxedo/tux90/install/inspds.htm

Table 1 Revision History

Revision Date	Summary of Change
February 2006	• Added CR112951 to "Product Constraints" as Constraint number 15. • Updated "Software Supported by Each Platform" in this document for post-GA platform data information. Added following post-GA platforms: HP-UX v11.23 (32-bit) on Itanium. Updated "Appendix A, BEA Tuxedo 9.0 Platform Data Sheets," of Installing the BEA Tuxedo System. The revised document can be found at the following URL: http://edocs.bea.com/tuxedo/tux90/install/inspds.htm
November 2005	• Updated "Software Supported by Each Platform" in this document for post-GA platform data information. Added following post-GA platforms: Novell SUSE Linux Enterprise Server 9 (64-bit). Updated "Appendix A, BEA Tuxedo 9.0 Platform Data Sheets," of Installing the BEA Tuxedo System. The revised document can be found at the following URL: http://edocs.bea.com/tuxedo/tux90/install/inspds.htm
October 2005	• Added CR238486, CR242065, and CR244891 to "Product Constraints" as Constraint number 12, 13, and 14 respectively. • Added CR236565, CR241622, CR244880, and CR244894 to "Known Problems in BEA Tuxedo 9.0."

Table 1 Revision History

Revision Date	Summary of Change
September 2005	• Updated reference of WebLogic Server 8.1 ORB to WebLogic Server 9.0 ORB in "Deprecated Features in BEA Tuxedo 8.1 (Not Supported in Tuxedo 9.0)." • Updated "Software Supported by Each Platform" in this document for post-GA platform data information. Added following post-GA platforms: HP 11i v2 64-bit, IBM AIX 5.3 64-bit, Solaris 9 64-bit, and Novell SUSE Linux Enterprise Server 9 (32-bit). Updated "Appendix A, BEA Tuxedo 9.0 Platform Data Sheets," of Installing the BEA Tuxedo System. The revised document can be found at the following URL: http://edocs.bea.com/tuxedo/tux90/install/inspds.htm • Added CR238601 to "Known Problems in BEA Tuxedo 9.0." • Added CR240505 to "Product Constraints" as Constraint number 11.
August 2005	Added CR236719 to "Product Constraints" as Constraint number 10.
July 2005	Added CR228589 to "Known Problems in BEA Tuxedo 9.0."
June 2005	Initial Release

This document contains release notes for the BEA Tuxedo 9.0 release, including ATMI, CORBA, BEA Jolt, and SNMP Agent.

Release Notes Topics

This document includes the following topics:

About This BEA Tuxedo Release

BEA Tuxedo software provides businesses and organizations that depend on mission-critical applications with the flexibility of two proven programming interfaces: an Application-to-Transaction Monitor Interface (ATMI) and a Common Object Request Broker Architecture (CORBA) interface. Both interfaces use the BEA Tuxedo infrastructure, which has demonstrated, through years of use in large, transaction-based, production systems, that it is powerful, robust, scalable, manageable, and reliable.

This topic includes the following sections:

What's New and Improved

BEA Tuxedo Release 9.0 includes the following new features and enhancements:

- Service Metadata Repository

  The Tuxedo Service Metadata Repository is a respository similar to the Jolt repository. It is designed to process interactive queries by developers and administrators during application development or modification.

- XML to and from FML/FML32

  To better enable integration of XML into BEA products, XML conversion to and from FML/FML32 in a Tuxedo environment is provided.

- XML parser upgrade

  Tuxedo 9.0 will deliver the Xerces parser 2.5.0 as part of the installation package. This upgraded parser will provide support for XML parser validation as well as continued XML support comparable to that of Tuxedo 8.1.

- Domain gateway performance enhancement

  This feature improves the Tuxedo domain GWTDOMAIN performance by introducing multithreaded execution capability.

- Domain gateway session connection policy

  This feature allows configuring a specific TDOMAIN session from a local GWTDOMAIN gateway to a remote GWTDOMAIN gateway. Connection policy and other QoS attributes can be specified for individual sessions. Establishing a connection from a local gateway to a remote one can be limited by removing the session definition.

- Domain gateway connection events

  Tuxedo 9.0 can report predefined events for Domain Gateway connections. Event reports are generated for connection success, failure and dropped connections.

- IIOP client failover

  This feature provides a transparent mechanism for a CORBA remote client to automatically connect to an alternative ISL and then retry the request in the event of failure.

- Service level blocktime

  This feature provides applications with greater control over blocktime, providing capabilities to specify blocktimes for individual services, calls made by a particular Tuxedo context, or for one particular blocking call.

- Kerberos authentication

  Tuxedo 9.0 provides a Kerberos security plug-in and a authentication server, KAUTHSVR, to support Kerberos authentication for Tuxedo native client and Tuxedo ACL function.

- Cert-C PKI Plug-in

  Tuxedo 9.0 provides a Cert-C PKI plug-in which allows users to sign, seal, and envelope Tuxedo typed message buffers by using the public key encryption algorithm.

- .NET Wrapper for Tuxedo Workstation Client

  The Tuxedo workstation client .NET wrapper provides customers with access to the Tuxedo system using the .NET environment. It is implemented as a set of APIs and development utilities for developers.

  The current release supports only the workstation *client programming*. Native client and server-side programming are not yet supported.

  The feature is developed and distributed as a stand-alone component of Tuxedo 9.0 and should be installed after the Tuxedo installation.

  Note: The .NET wrapper feature is not part of BEA Tuxedo 9.0 release for general availability (GA). Full testing of this feature will be complete after the release. Please check BEA dev2dev Tuxedo product Web site for the .Net wrapper code at the following URL.

  http://dev2dev.bea.com/tuxedo/

- WLS CORBA Clustering Support

  Tuxedo CORBA C++ client supports failover to Weblogic clustering servers and also supports load balancing.

- Custom Enhancements:
  - TMTRACE enhancement: allows user-level tracing
  - Servopt MIN=0 update: disables server booting when MIN=0 is specified
  - ULOG time stamp: allows ULOG time stamping in milliseconds
  - ULOG rotation file: allows log file size limitation
  - `tmadmin(1)` update to `pclt` reporting to include IP address: outputs client IP address when client name is not specified
  - MBSTRING support in VIEWs and JOLT: allows MBSTRING capabilities in VIEW32 buffers and JOLT
  - Support of sanity scans less than 5 seconds
  - Warning message provided if MAXACCESSERS or MAXSERVERS are missing
  - User-controlled ability to stop CMDTUX:4754 Messages

For a discussion of each these features, see the What's New link on the BEA Tuxedo online documentation.

Installation Upgrade Considerations

Before installing the product, be sure to review the product issues in "Known Problems in BEA Tuxedo 9.0" on page 31. For complete information on upgrading to BEA Tuxedo 9.0, see "Upgrading the BEA Tuxedo System to Release 9.0" in *Installing the BEA Tuxedo System*.

Hot Upgrade From Tuxedo 8.x to Tuxedo 9.0

In order to perform a hot upgrade from BEA Tuxedo 8.x to BEA Tuxedo 9.0, you must be running Tuxedo 8.0 rolling patch 22 at a minimum.

Adding Memory Capacity When Upgrading an Existing Application

The intent of this section is to provide an approximation of the proportional increased memory consumption that an application may have in Tuxedo 9.0, and to aid in further detailed analysis by the application developer or administrator.

Due to the newly added features in Tuxedo 9.0, the addition of new code and data into underlying Tuxedo shared libraries (e.g., `libengine.so`), some of these libraries are much larger than in Tuxedo 8.1. The size increase of shared libraries cause running Tuxedo system and application processes depending on those libraries to consume more memory than before.

The amount of additional memory consumed by a given Tuxedo 9.0 application compared with its Tuxedo 8.1 counterpart depends on many factors, such as the number of application processes, the amount of private data that application logic has, and so on. Consequently, it is difficult to accurately calculate the impact using a simple formula. However, it is important for an application developer or administrator to understand the implications from a system capacity planning standpoint.

A Tuxedo executable is defined as an executable binary file built using Tuxedo libraries and stored on the file system, and can be either a Tuxedo system server executable or application server executable. Generally, the memory consumed by one of an executable's running processes can be divided into two parts:

- the memory shared across all processes of that executable, and
- the memory allocated especially for that process itself and used only by that process.

For each Tuxedo process, memory is further separated into the memory consumed by Tuxedo libraries and the memory consumed by the executable's logic itself. Assume the following:

A = memory shared across all processes in Tuxedo 9.0 executables

a = memory shared across all processes in Tuxedo 8.1 executables

B = memory used only by the process itself in Tuxedo 9.0 executables

b = memory used only by the process itself in Tuxedo 8.1 executables

X = memory used by the application logic (same for Tuxedo 8.1 and 9.0)

N = number of active processes

And also assuming that A>=a, B>=b and X>=0, note the following observations:

1. A, B, a, b and X are constant. The proportion gradually changes from (A+B+X)/(a+b+X)-1 to (B+X)/(b+X)-1 when N increases from 1 to a rather great number, for example 100.
2. A, B, a, b and N are constant. The proportion is greatest when X is 0 and will decrease when X increases. In other words, the more application logic in the executable, the less the proportion.

To further illustrate, using the Tuxedo `simpapp` sample as an example tested on Solaris, when N of 1 and 100 was used, test results showed the corresponding proportion to be 68 percent and 36 percent respectively. That is, there are respectively 68 percent and 36 percent increase in memory consumption of `simpserv` processes comparing Tuxedo 9.0 with Tuxedo 8.1 when 1 and 100 `simpserv` processes are activated.

Since the `simpapp` sample is not likely a representative of real Tuxedo applications (meaning that X is almost 0 in `simpapp` servers), the average proportional increases are likely to be lower. In practice, there are likely many different executables starting their own processes in a Tuxedo application, so the ultimate proportion will be an weighted average of all the proportions coming from different executables.

It is approximated that, in most cases, the proportion will be in the 20 percent range. It is strongly recommended that application developers and administrators conduct tests and analysis unique to their application prior to upgrading to Tuxedo 9.0. Note that if the proportional increase is significant for an existing application, and sufficient physical memory cannot be added to the existing machines to accommodate this increase, there may be additional performance impact caused by overhead of possible additional memory paging activities at the operating system level. These performance implications are best determined by consulting with specific operating system vendor experts.

Unsupported Code Samples and Tools Web Page

BEA Tuxedo customers can download unsupported code samples and developer tools from the BEA Systems, Inc. dev2dev online site. Start on the following Web page:

http://dev2dev.bea.com/index.jsp

If you do not already have a BEA dev2dev login, links are provided with instructions for free membership.

BEA Tuxedo Software Components

The BEA Tuxedo software consists of the following components:

- BEA Tuxedo ATMI software

  The ATMI software enables you to build scalable ATMI applications using either of two programming languages: C or COBOL. This software includes the following components:

  – BEA Tuxedo ATMI servers

  – BEA Tuxedo /WS clients

  – BEA Tuxedo Native clients

  – BEA Tuxedo infrastructure

- BEA Tuxedo CORBA software

  The CORBA software enables you to build scalable CORBA applications in the C++ programming language. This software includes the following components:

  – CORBA C++ servers

  – C++ client and server Object Request Broker (ORB)

  – BEA Tuxedo object infrastructure

- BEA Tuxedo Administration Console

- BEA Jolt 9.0 software

  BEA Jolt is a Java-based interface to the BEA Tuxedo system that extends the functionality of existing BEA Tuxedo applications to include intranet- and Internet-wide availability.

- BEA SNMP Agent software

BEA SNMP Agent for BEA Tuxedo and BEA WebLogic Enterprise is a Simple Network Management Protocol (SNMP) agent that enables BEA Tuxedo and BEA WebLogic Enterprise applications to be managed from an Enterprise Management Console.

- BEA Tuxedo 56-bit or 128-bit Encryption Package software that provides Secure Sockets Layer (SSL) and Link-Level Encryption (LLE) for BEA Tuxedo applications. This software is included in the BEA Tuxedo 9.0 distribution and is enabled or disabled depending on which license is used.

Software Component Licensing Requirements

For BEA Tuxedo 9.0, all software components are included on the product CD-ROMs. A single license is issued when you purchase the product that enables the components that you want to use.

Licensing is used to enable the product components as follows:

- Basic license

  This license enables the following components:

  – The ATMI and CORBA programming environments (clients and servers)

  – Secure Sockets Layer (SSL), Link-Level-Encryption (LLE), and Public Key Interface (PKI) plug-ins

  – 56-bit encryption

- 128-bit encryption license

  This license enables 128-bit encryption as well as the components enabled by the basic license.

Note: It will not be possible for a customer to use 128-bit encryption for data messages without obtaining a 128-bit encryption license from BEA. However, 128-bit encryption can be used for BEA Tuxedo Administration Console messages without obtaining a 128-bit license.

- Jolt license

  This license enables the BEA Jolt software as well as the components enabled by the basic license.

- Full license

  This license enables all product components.

BEA Tuxedo 9.0 requires that all customers upgrade their licenses to a new format; previous licenses will not work. Customers with existing support contracts may use one of the following methods to obtain a license upgrade:

- If you have a previous version of a BEA Tuxedo license, you may update your license from the BEA Support site using an automatic form, or by contacting a BEA representative.
- If you have a BEA WebLogic Enterprise license, you must contact your BEA representative to upgrade your license to BEA Tuxedo 9.0.

Supported Platforms

BEA Tuxedo software runs on the platforms listed in the following sections. BEA has certified these platforms for development and production use with the BEA Tuxedo release 9.0 product. BEA can provide customer support only for these platforms. Note that although BEA has attempted to implement the BEA Tuxedo software in a manner that conforms to industry-standards, it is not feasible for BEA to certify its use with all third-party databases, ORBs, and other products.

Additional software ports and certifications may continue after the initial release of BEA Tuxedo 9.0. For information regarding subsequent ports and certifications, please refer to the Platform Support information on the BEA web site at the following link:

`http://www.bea.com/products/tuxedo/platforms.shtml`

Platform information is maintained under the "Requirements" option under the Tuxedo product page.

Note: More detailed platform information is maintained on the secured eSupport portal under "Product News and EOL Updates." A customer eSupport password login is required. The eSupport link is:

`http://support.bea.com`

BEA Tuxedo Server Platforms

The BEA Tuxedo server components run on the following platforms:

- HP-UX v11.23 (32-bit) on Itanium
- HP-UX 11.23 (32-bit) on PA-RISC
- HP-UX 11i v2 (64-bit) on PA-RISC
- IBM AIX 5.3 (32-bit) on IBM PowerPC
- IBM AIX 5.3 (64-bit) on IBM PowerPC
- Microsoft Windows 2003 Server (32-bit) on Intel
- Novell SUSE Linux Enterprise Server 9 (32-bit) on Pentium
- Novell SUSE Linux Enterprise Server 9 (64-bit) on Itanium
- Red Hat Linux Enterprise Server 4.0 (64-bit) on Itanium
- Sun Microsystems Solaris 9 (32-bit) on SPARC
- Sun Microsystems Solaris 9 (64-bit) for SPARC
- Sun Microsystems Solaris 10 (32-bit) on SPARC
- Sun Microsystems Solaris 10 (64-bit) on SPARC

BEA Tuxedo Client Platforms

The BEA Tuxedo client software runs on the following platforms:

- BEA Tuxedo Native CORBA C++ clients:
  All server platforms listed in the previous section.
- BEA Tuxedo remote (IIOP) CORBA C++ clients:
  All server platforms listed in the previous section, plus Microsoft Windows XP.
- BEA Tuxedo /WS clients:
  All server platforms, plus Microsoft Windows XP.
- BEA Tuxedo Native clients:
  All server platforms listed in the previous section.

BEA Tuxedo Administration Console Platforms

The BEA Tuxedo Administration Console software runs on all the platforms listed in the section "BEA Tuxedo Server Platforms" on page 12.

Note: While the Administration Console software cannot be installed on Microsoft Windows XP systems, you can use the Web browser on your Microsoft Windows XP system to access and use the Administration Console software on any BEA Tuxedo server system that is accessible over your network.

BEA Security Service Platforms

The BEA Tuxedo Security Service (56-bit or 128-bit) runs on the following platforms:

- HP-UX v11.23 (32-bit) on Itanium
- HP-UX version 11.23 (32-bit) on PA-RISC
- HP-UX 11i v2 (64-bit) on PA-RISC
- IBM AIX 5.3 (32-bit) on IBM PowerPC
- IBM AIX 5.3 (64-bit) on IBM PowerPC
- Microsoft Windows 2003 Server (32-bit) on Intel
- Microsoft Windows XP (Client only)
- Novell SUSE Linux Enterprise Server 9 (32-bit) on Pentium
- Novell SUSE Linux Enterprise Server 9 (64-bit) on Itanium
- Red Hat Linux Enterprise Server 4.0 (64-bit) on Itanium
- Sun Microsystems Solaris 9 (32-bit) on SPARC
- Sun Microsystems Solaris 9 (64-bit) for SPARC
- Sun Microsystems Solaris 10 (32-bit) on SPARC
- Sun Microsystems Solaris 10 (64-bit) on SPARC

BEA Tuxedo 9.0 offers two type encryption services: secure sockets layer (SSL) and link-level encryption (LLE). The SSL and LLE encryption software is included on the BEA Tuxedo 9.0 distribution and is an integral part of the installation procedure. You also configure the SSL software during the installation.

Before you can use BEA Tuxedo Security Service software on any of the platforms listed above, you must first install it according to one of the following options:

- Option 1: On the Tuxedo server platforms, install at least one of the following BEA Tuxedo 9.0 server components:
  - Full Install
  - Server Install
- Option 2: On all platforms, install at least one of the following BEA Tuxedo 9.0 client components:
  - Client Install (All BEA Tuxedo client components; this is recommended)
  - Customized Install
    - CORBA C++ client
    - BEA Tuxedo /WS client
    - BEA Jolt client

Software Environment

The following sections list the software that can run on each platform supported by the BEA Tuxedo software.

Software Supported by Each Platform

Table 2 lists the software supported by each platform that can run the BEA Tuxedo software.

Table 2 Software Supported by Each Platform

Platform	Java 2 SDK and JRE[a]	C/C++ and COBOL Compilers	Clients	Servers
HP-UX V11.23 (32-bit) on PA-RISC	Java 2 SDK (or JRE) 1.4.x	HP C/ANSI C B.11.23.08 with patches PHSS_32152, PHSS_32513 applied; HP aC++ A.03.63; Server Express 4.0sp1 (COBOL) from Micro Focus, ACUCOBOL-GT 6.1 or later from Acucorp, or other compatible COBOL compiler; required only for BEA Tuxedo development environment.	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI
HP-UX v11.23 (32-bit) on Itanium	Java 2 SDK (or JRE) 1.5.0_0_1	HP aC++/ANSI C B3910B A.06.06 with patches (PHSS_34046, PHSS_34047, PHSS_34048)	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI

Table 2 Software Supported by Each Platform (Continued)

Platform	Java 2 SDK and JRE[a]	C/C++ and COBOL Compilers	Clients	Servers
HP-UX 11i v2 (64-bit) on PA-RISC	Java 2 SDK (or JRE) 1.5.0	HP ANSI C B.11.23.08 (Bundle B3901BA) with patches (PHSS_32511 ANSI C compiler B.11.11.12 cumulative patch (PHSS_32513 +O4/PBO Compiler B.11.11.12 cumulative patch) HP aCC C.03.62 (Bundle B3913DB) with patch (PHSS_32511 HP aCC Compiler (A.03.63))	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI
IBM AIX 5.3 (32-bit) on IBM PowerPC	Java 2 SDK (or JRE) 1.4.x	C for AIX 6.0; VisualAge C++ 6.0; Java: 1.4.1; Server Express 4.0.sp1 (COBOL) from Micro Focus, ACUCOBOL-GT 6.1 or later from Acucorp, or other compatible COBOL compiler; required only for BEA Tuxedo development environment	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI
IBM AIX 5.3 (64-bit) on IBM PowerPC	Java 2 SDK (or JRE) 1.4.2	Visual Age C/C++ 7.0	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI

Table 2 Software Supported by Each Platform (Continued)

Platform	Java 2 SDK and JRE[a]	C/C++ and COBOL Compilers	Clients	Servers
Microsoft Windows 2003 Server on Intel	Java 2 SDK (or JRE) 1.4.x	VC.net 2003 Professional; required for full (development) install, but not for server-only or client-only installations; Net Express 4.0.00 (COBOL) from Micro Focus, ACUCOBOL-GT 6.1 or later from Acucorp, NetCOBOL version 7.x or later from Fujitsu, or compatible COBOL compiler.	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI
Microsoft Windows XP (Client only)	Java 2 SDK (or JRE) 1.4.x	VC.net 2003 Professional; no additional compiler software is required on client-only systems.	CORBA C++ over IIOP; Tuxedo /WS	None
Novell SUSE Linux Enterprise Server 9 (32-bit) on Pentium	Java 2 SDK (or JRE) 1.5.0_4	gcc/g++ 3.3.3	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI
Novell SUSE Linux Enterprise Server 9 (64-bit) on Itanium	Java 2 SDK (or JRE) 1.4.2_08	gcc/g++ 3.3.3	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI
Red Hat Linux Enterprise Server 4.0 (64-bit) on Itanium	Java 2 SDK (or JRE) 1.4.2_08	gcc version 3.4.3 20041212 (Red Hat 3.4.3-9.EL4) + compat-libstdc++-33-3.2.3-47.3.ia64.rpm	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI

Table 2 Software Supported by Each Platform (Continued)

Platform	Java 2 SDK and JRE[a]	C/C++ and COBOL Compilers	Clients	Servers
Sun Microsystems Solaris 9 (32-bit) on SPARC	Java 2 SDK (or JRE) 1.4.x	SUN ONE Studio 8 or later compatible; Server Express 4.0.0 (COBOL) from Micro Focus, ACUCOBOL-GT 6.1 or later from Acucorp, NetCOBOL version 7.x or later from Fujitsu, or other compatible COBOL compiler; required only for BEA Tuxedo development environment	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI
Sun Microsystems Solaris 9 (64-bit) on SPARC	Java 2 SDK (or JRE) 1.5.0_3	Sun Studio 10	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI
Sun Microsystems Solaris 10 (32-bit) on SPARC	Java 2 SDK (or JRE) 1.5.0_01	Sun Studio 10	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI
Sun Microsystems Solaris 10 (64-bit) on SPARC	Java 2 SDK (or JRE) 1.5.0_01	Sun Studio 10	CORBA C++ over IIOP; CORBA C++ Native; Tuxedo /WS	CORBA C++; ATMI

a. The Java 2 JRE is needed for run-time environment.

Note: There are sample COBCC files for the NetCOBOL compiler by Fujitsu on the BEA dev2dev site. Please check BEA dev2dev Tuxedo product Web site for sample COBCC files at the following URL.

http://dev2dev.bea.com/tuxedo/

Database Support

BEA Tuxedo ATMI and CORBA C++ applications support the XA standard. This facilitates inter-operation with any XA-compliant software system including database management systems.

Security Related Software Supported

The following security software is supported on all BEA Tuxedo platforms:

- SSL Certificate authorities
  - Verisign
- To support certificate-based authentication when using SSL, BEA Tuxedo provides an LDAP-based certificate retrieval mechanism. This retrieval mechanism has been certified for use with the LDAP Directory server that is included with the Netscape Enterprise Server

Table 3 lists the BEA Tuxedo clients and connections that support SSL when the SSL Certificate software is installed and the clients and connections that are not supported.

Table 3 BEA Tuxedo Support for SSL 3.0

SSL is supported for...	SSL is not supported for...
BEA Tuxedo CORBA C++ and IIOP clients.	BEA Tuxedo ATMI /WS client connections to the BEA Tuxedo 9.0 Workstation Listener/Handler (WSL/WSH).[a] BEA Jolt client connections to BEA Jolt Listener/Handler (JSL/JSH).

a. BEA Tuxedo 9.0 56-bit or 128-bit encryption is available for link-level encryption of these connections and also for link-level encryption of connections between machines and domains.

BEA Tuxedo End-of-Life Information

BEA Systems, Inc. periodically finds it necessary to discontinue support for certain older products to ensure the highest level of quality and support for our customers going forward. BEA has a policy of providing advanced notification to our customers so migration strategies and plans can be made.

To access end-of-life (EOL) information for the BEA Tuxedo product, access the BEA eSupport web site at the following link.

http://support.bea.com

Log in to the Support site or register to get a login ID to access EOL information. After logging in, click the Product News and EOL Information link in the left navigation area of the Support page.

Deprecated Features in BEA Tuxedo 8.1 (Not Supported in Tuxedo 9.0)

When a product feature is deprecated, it is identified as a feature that will not be supported and may be removed in the next release of the product. The following features were deprecated in Tuxedo 8.1 and will no longer be supported in this release of Tuxedo.

- WebLogic Enterprise Connectivity (WLEC)
- ActiveX Client Support
- TOP END Gateway Support
- TxRPC (DCE) support

  The Tuxedo CORBA programming interface supports an Interface Definition Language based on the DCE IDL, so moving to the Tuxedo CORBA interface is an option for customers currently using TxRPC. Another option is to code your applications using ATMI.

- BEA CORBA Java Client and BEA Tuxedo CORBA Java client ORB

  All BEA Tuxedo CORBA Java client and BEA Tuxedo CORBA Java client ORB text references, associated code samples, etc. should only be used:

  - to help implement/run third party Java ORB libraries, and
  - for programmer reference only.

  Technical support for third party CORBA Java ORBs should be provided by their respective vendors. BEA Tuxedo does not provide any technical support or documentation for third party CORBA Java ORBs.

  Customers are encouraged to migrate to WebLogic Server 9.0 ORB or Sun Java ORB.

- Jolt ASP Connectivity

Online Documentation

The BEA Tuxedo product documentation is available from the following locations:

- Go directly to the BEA Tuxedo e-docs product documentation page at `http://edocs.bea.com/tuxedo/tux90/index.htm`
- On the BEA Tuxedo Documentation CD-ROM. The documentation CD-ROM includes Web-browsable HTML and easy-to-print Adobe Acrobat PDF documentation for this product.

Accessing the Documentation CD-ROM on Microsoft Windows Systems

To access the online documentation on CD-ROM, proceed as follows:

1. Insert the BEA Tuxedo Documentation CD-ROM into the drive.
2. Using Windows Explorer, click `index.htm` in the following directory of the Online Documentation CD-ROM:

   `docs\tuxedo\tux90\index.htm`

   The documentation home page is displayed in your browser.

Accessing the Documentation CD-ROM on UNIX Systems

To access the online documentation, proceed as follows:

1. Insert the BEA Tuxedo Documentation CD-ROM into the drive.
2. Mount the CD-ROM. This step might not be required for Solaris systems.

Note: For mounting instructions for the supported UNIX platforms, refer to Appendix A of *Installing the BEA Tuxedo System*.

3. Start the Internet Explorer browser and set the browser to `/mnt/docs/tuxedo/tux90/index.htm` (where `mnt` is the CD-ROM mount point) and press Enter.

The documentation home page is displayed in your browser.

Accessing the Java API Documentation

Documentation for the BEA Jolt 9.0 is automatically installed on each machine on which the BEA Tuxedo software has been installed. This is in addition to the Java API documentation available on the online documentation CD-ROM.

The BEA Tuxedo API documentation is installed in the following location. `TUXDIR` represents the top-level directory where BEA Tuxedo is installed:

On Windows Systems:

For BEA Jolt: `%TUXDIR%\udataobj\jolt\doc\index.html`

On UNIX Systems:

For BEA Jolt: `$TUXDIR/udataobj/jolt/doc/index.html`

Using a Microsoft Internet Explorer browser, open the `index.htm` start page in that directory.

Copying the Product Documentation to Your System

Although it is not necessary to copy the online documentation to your system, you can do so. Depending on the speed of your computer, you may want to copy the content of the CD-ROM to a local drive for better response time.

To make the content of the CD-ROM available on the network, put the CD-ROM in a server's CD-ROM reader and designate it as shared. This option is an alternative to taking up approximately 250 MB on the server's hard drive.

Another option is to copy the contents of the Documentation CD-ROM to a Web server on your corporate intranet.

Copying the Product Documentation to Microsoft Windows Systems

To copy the content of the online documentation CD-ROM to your system, proceed as follows:

1. Insert the online documentation CD-ROM into the CD-ROM reader.
2. Using Windows Explorer, double-click the CD-ROM drive icon. Windows Explorer displays a `doc` folder at the root of the CD-ROM.
3. In the top directory, select the `doc` folder and copy the content of the CD-ROM using Edit—>Copy on the Windows Explorer menu bar, or press Ctrl+c.

4. Paste the copy on your local drive (for example, your C: drive) using Edit—>Paste on the Windows Explorer menu bar, or press Ctrl+v.

Note: The content of the CD-ROM can also be copied by using the drag-and-drop feature of Windows Explorer.

Copying the Product Documentation to UNIX Systems

To copy the content of the Online Documentation CD-ROM to your system, proceed as follows:

1. Mount the CD-ROM. This step might not be required for Solaris systems.

Note: For mounting instructions for the supported UNIX platforms, see *Installing the BEA Tuxedo System*.

2. Change directory to the target directory where you want to place the files. For example: `cd /mydirectory/docs`.

3. Copy the entire CD-ROM content using a recursive copy command. For example: `cp -r /mnt/cdrom/*.`

Printing from a Web Browser

You can print a copy of the online documents, one file at a time, from your Web browser. Before you print, make sure that the chapter or appendix you want to print is displayed and *selected* in your browser. We recommend the Adobe Acrobat PDF format as a better format for printing hard copies of the BEA Tuxedo documentation, instead of printing HTML files from the browser.

Printing Adobe Acrobat PDF Files

The BEA Tuxedo documentation also includes Adobe Acrobat PDF files of all the online documents. You can use the Adobe Acrobat Reader to print all or a portion of each document, as follows:

1. On the online documentation home page, click the PDF Files button.

2. Scroll to the entry for the document you want to print.

Product Constraints

Table 4 describes product constraints for BEA Tuxedo 9.0 and provides information about recommended workarounds.

Table 4 Product Constraints

1.	**Tuxedo XDR encode/decode functions cannot correctly encode integer data types larger than 32-bit**	
	Description	The XDR encode/decode functions that Tuxedo uses to pass data between different machine types currently cannot encode 64-bit integer data types containing values which would not fit in a 32-bit datatype. So if the VIEW/VIEW32 long type data is out of 32-bit range, it will be truncated during XDR encode/decode.
	Platforms	AIX, HP, and Solaris.
	Workaround	None.
2.	**Reserved Repository IDs.**	
	Description	Use of `Repository` as an interface repository ID in a `#pragma` ID OMG IDL directive, or use of `pk_` as the beginning part of such an ID, causes a conflict with IDs used internally in the BEA Tuxedo system. Their use results in undefined behavior.
	Platforms	All.
	Workaround	If you use the `#pragma` ID OMG IDL directive, do not use `Repository` or any identifier that begins with `pk_` as the repository identifier.
3.	**Restrictions to using the Dynamic Invocation Interface (DII).**	
	Description	When you use DII, you may encounter problems if you do not observe certain restrictions.
	Platforms	All.

Table 4 Product Constraints (Continued)

	Workarounds	Adhere to the following restrictions: • If you use `CORBA::Request::set_return_type()`, you must set the location where you want the ORB to place the result for the specified request. This can be done by using `CORBA::Any::replace()` (after setting the return type) on the Any reference returned by `CORBA::Request::return_value()`. • The CORBA 2.2 specification appears to imply that when using DII the user need not specify the return type or any `out` only arguments; or, if specified, the user need not specify the location in which to store the result or `out` values. In any case, the BEA Tuxedo software requires that you specify the return type, any `out` only arguments, and the location in which to store the result or `out` values. • When using `CORBA::ORB::send_multiple_requests_deferred()`, if any of the individual requests results in an error (that is, throws a locally detected exception), the exception is propagated to the user and any requests in the sequence after the said request will not be sent (that is, the behavior is as if the `CORBA::INV_TERM_ON_ERR` flag had been specified). The workaround is to not use this function; instead, call `CORBA::Request::send_deferred()` on each request in the sequence with appropriate `try/catch` statements around each call.
4.	**Remote BEA Tuxedo client logon/authenticate binary userdata cannot contain NULL's.**	
	Description	With an AuthType level of `Tobj::TOBJ_APPAUTH`, remote BEA Tuxedo client binary user_data passed into the logon/authenticate API cannot contain NULL's.
	Platforms	All.
	Workaround	It is possible for binary user_data to contain legitimate NULL's. One solution is to use the native mode rather than the IIOP mode. The native mode does not have the problem.
5.	**Client invocation times out after 60 seconds by default.**	
	Description	The default behavior for a client application invoking a BEA Tuxedo IIOP Server Listener/Handler or server application is to time out if 60 seconds elapse without a response. The client application receives a `NO_RESPONSE`, `NO_RESOURCES`, or `COMM_FAILURE` exception.
	Platforms	All.
	Workaround	The default timeout can be changed for the application by adjusting `SCANUNIT` and `BLOCKTIME` in the `UBBCONFIG` file for the application, or by decreasing the load on the server.
6.	**Known problem with fonts when running the University sample applications.**	

Table 4 Product Constraints (Continued)

	Description	When running the University sample applications, the availability and size of fonts varies from platform to platform and even from machine to machine, depending on the installation. As a result, text sizes may appear too large or too small on some platforms.
	Platforms	All.
	Workaround	None.
7.	**Transactions with deferred, synchronous requests experience problems.**	
	Description	When using DII, if you initiate deferred synchronous requests in the context of a transaction, the transaction does not complete successfully.
	Platforms	All.
	Workaround	Wait until you have received responses for deferred synchronous requests before you commit the transaction. Otherwise, results are unpredictable.
8.	**C++ reserved words in IDL for operation names.**	
	Description	Do not use C++ reserved words as operation names in IDL files. Using C++ reserved words as operation names might cause the IDL compiler to fail.
	Platforms	All.
	Workaround	Change an operation name in the IDL file to a name that is not a C++ reserved word.
9.	**Mapping of CORBA and FML data types for applications that allow interoperability between BEA Tuxedo ATMI and BEA Tuxedo CORBA.**	

Table 4 Product Constraints (Continued)

Description	BEA Tuxedo applications use FML field buffers that support a limited number of data types as described in the *Programming BEA Tuxedo ATMI Applications Using FML*. If you want BEA Tuxedo and BEA Tuxedo CORBA applications to interoperate, you must handle the mapping between the corresponding data types carefully. For example, a `CORBA::Long` is not a long for all platforms. Hence, there needs to be a conversion when reading from or writing to FML buffers after and prior to doing tpcalls from BEA Tuxedo applications. If a BEA Tuxedo application were to make a BEA Tuxedo call passing a `CORBA::long` as an FML long, a straight forward use would be as follows: Fadd32(fmlbuf, LongFmlFld, 0, (char*) &WLE_CORBA_long, 0); which you would issue prior to doing a tpcall such as: tpcall((char*)SomeTuxService, (char*) fmlbuf, (char**) &fmlbuf, sizeof(fmlbuf), 0) Such a call will succeed on most platforms, but will fail on Compaq Tru64 UNIX. **Note:** Any applications that use the BEA Tuxedo Notification Service to allow interoperability between BEA Tuxedo events and BEA Simple or CosNotification Structured Events would be affected by this problem.
Platforms	All.

Table 4 Product Constraints (Continued)

Workaround

For BEA Tuxedo and BEA Tuxedo CORBA applications to interoperate, you must change the Fadd32 statement from:

```
Fadd32(fmlbuf, LongFmlFld, 0, (char*) &WLE_CORBA_long, 0);
```

to:

```
long FML_long = WLE_CORBA_long;
Fadd32(fmlbuf, LongFmlFld, 0, (char*) &FML_long, 0);
```

A reverse conversion would likewise be required to receive a `long` value returned from a `tpcall` and used as a `CORBA::Long`.

The University Wrapper sample application shows an example of this, where the `ACCOUNT_NO` is a `long` in the BEA Tuxedo Billing application wrapped by the BEA Tuxedo `billw_server`. The BEA Tuxedo `univw_server` and `billw_server` use a `CORBA::Long` for the `AccountNumber` and handle the conversion appropriately as follows:

```
CORBA::Double Teller_i::get_balance(
BillingW::AccountNumber account)
   {
    long account_l = account;
           :
    call_tux(m_tuxbuf, "CURRBALANCE");
           :
    }
```

10. **The C++ IDL compiler does not generate correct codes for _copy_value() method if defined valuetype graph contains cycles.**

Description

The CORBA OMG specification is vague about how to avoid infinite loops when copying a valuetype graph that contains cycles. For example:

```
//IDL
   valuetype A;
   valuetype B {
       public A foo;
   };
   valuetype A {
       public B bar;
   };
```

If the _copy_value() function of valuetype A is invoked, it will call _copy_value() function of valuetype B, while _copy_value() function of valuetype B will call _copy_value() function of valuetype A, then infinite loops are in the _copy_value() function when invoked.

Platforms

All.

Table 4 Product Constraints (Continued)

	Workaround	Override the _copy_value() function and make it so it can deal with the loops in a manner specific to the valuetype involved or avoid defining a valuetype graph that contains cycles.
11.	**The C++ IDL compiler does not support using C++ keyword as an identifier for valuetype.**	
	Description	The generated code by the C++ IDL compiler for valuetype that includes C++ keywords as an identifier is incorrect.
	Platforms	All.
	Workaround	Avoid using C++ keywords as the identifier for valuetype.
12.	**The pure virtual accessor and modifier functions for valuetype private state members are not protected.**	

	Description	Tuxedo IDL compiler does not comply with CORBA specification when compiling valuetype with private state members. CORBA specification requires that the private members' pure virtual accessor and modifier functions should be protected, but the generated pure virtual accessor and modifier functions by Tuxedo IDL compiler are public. For examples:
		`//IDL`
		`module A {`
		`  valuetype V{`
		`  ......`
		`  private V m_v;`
		`  ......`
		`  };`
		`};`
		
		`// The generated code:`
		`namespace A{`
		`  class V : public virtual ::CORBA::ValueBase`
		`  {`
		`  ...`
		`  public:`
		`  // accessor and modifier for m_v`
		`  virtual ::A::V * m_v() const = 0;`
		`  virtual void m_v(::A::V *) = 0;`
		`  }; // class V`
		`}`

	Platforms	All.

Table 4 Product Constraints (Continued)

	Workaround	None.
13.	**CORBA Object-By-Value semantics are not supported in "InProc" invokes.**	

	Description	If the CORBA server acting as a client invokes an IOR for an object that is activated in the same server process, then this is considered an "InProc" invokes and the method is invoked directly with no GIOP message generated.
		The Object-By-Value semantics are not supported in "InProc" invokes scenario. If valuetype/valuebox is passed as an argument or returned as a return value in "InProc" invokes scenario, the CORBA::INTERNAL exception will be raised.
		For example:
		`valuetype A {`
		`  ...`
		`};`
		`interface foo {`
		`  void op1(in A arg1);`
		`};`
		`interface bar {`
		`  void op2(in foo arg2);`
		`};`
		If the op2() method of "bar" object invokes op1() method of "foo" object passed in, and both "foo" and "bar" objects are deployed in the same server process, a CORBA::INTERNAL exception will be thrown.

	Platforms	All.
	Workaround	Avoid passing valuetype/valuebox in "InProc" invokes scenario, or deploy CORBA objects in different server processes.
14.	**Valuetypes that formed circular graphs will cause a CORBA server memory leak.**	

Table 4 Product Constraints (Continued)

	Description	CORBA specification allows that Valuetypes can be used to form arbitrary, potentially circular graphs. But the reference counts of circular Valuetypes may never drop to zero, this will cause server memory leak. CORBA specification is vague about how to resolve this problem in C++. BEA Tuxedo CORBA does not provide a mechanism to resolve such problems. This means if Valuetypes formed circular graphs, the CORBA server will have memory leak.
		For examples:
		`// IDL`
		`valuetype A;`
		`valuetype B;`
		
		`valuetype A {`
		`  public B foo;`
		`}`
		
		`valuetype B {`
		`  public A bar;`
		`}`
		Valuetype A and B formed a circular graph.

	Platforms	All.
	Workaround	Avoid defining Valuetypes that will form circular graphs.
15.	**Request-level interceptors do not support Object-by-Value syntax.**	
	Description	The marshalling stream is modified when request-level interceptors call `DataInputStream::read_Value()` to decode valuetypes. The Tuxedo ORB will not re-try and consequently fails to unmarshall the input stream because the stream no longer contains a properly formatted GIOP message.
	Platforms	All.
	Workaround	None.

Known Problems in BEA Tuxedo 9.0

The following sections describe known problems with the BEA Tuxedo software and include recommended workarounds. The problems are listed by the Change Request (CR) number. The CR number is provided to facilitate the tracking of these problems.

Contact your BEA Customer Support Center for assistance in the tracking of any unresolved problems. When contacting the BEA Customer Support Center, please refer to the CR number.

Table 5 lists known problems for BEA Tuxedo 9.0.

Table 5 Known Problems for Tuxedo 9.0

CR012652	**Sequences and arrays contained by a `CORBA::Any` cannot be passed remotely due to their lack of a repository ID.**	
	Problem	Sequences and arrays contained within a `CORBA::Any` cannot be passed remotely. These types do not have a repository ID in the syntax specified for transferring them across the wire. For a description of the Common Data Representation (CDR) Transfer Syntax, see Table 13-2 of *The Common Object Request Broker: Architecture and Specification*, Revision 2.2. The repository ID is used by the receiving ORB to unmarshal the type contained by the `Any`. Without the ID, the ORB cannot obtain sufficient information to unmarshal the data type. This restriction may be removed in a future release.
	Platforms	All.
	Application Type	CORBA C++.
	Workaround	Sequences and arrays can be used within other data types, such as structures, or they can be used directly as parameters.
CR012697	**When you use VIEWs, strings are null terminated and characters truncated.**	
	Problem	There is an inconsistency between BEA Jolt and native BEA Tuxedo clients. BEA Jolt does not know what a client and server agreed to exchange—a string or a null-terminated string.
	Platforms	All
	Workaround	Accommodate a null in the definition of the string.
CR014128	**ISL will not start if there is an underscore (_) in the IP name.**	
	Problem	The ISL fails to boot if the `-n` option has a hostname that contains an underscore. The error written in the user log (ULOG) is: `ISNAT_CAT:1242: ERROR: Bad Internet type of listening address provided: <node name>`
	Platform	Microsoft Windows.

Table 5 Known Problems for Tuxedo 9.0

	Workaround	Ensure that the ISL is configured correctly with host and port values.
CR016275	**Cannot have two data types for the same field name in input and output VIEWs.**	
	Problem	The member in a VIEW is qualified by the structure name, so the input view and output view may use the same name for a member, but with different data types. However, BEA Jolt does not support this feature and some existing BEA Tuxedo services cannot be accessed by BEA Jolt client. A typical service that uses this feature is a gateway; it translates one data type to another data type with the same name.
	Platforms	All

	Workaround	Use the `VIEW` name to qualify the field name. For example, INVIEW and OUTVIEW both have a field birthdate, but one is an integer and the other is a string.
		`svc.setlnt[“INVIEW.birthdate”,19980308];`
		`svc.call[null];`
		`String bdate =`
		`svc.getStringDef[“OUTVIEW.birthdate”,null]`

CR017633	**User’s `jrepository` file needs to be converted to use the services in JREPSVR.**	
	Problem	Because the JREPSVR in BEA Jolt 1.1 has been changed to use FML32 instead of FML16, BEA Jolt 1.1 users should update their BEA Jolt Repository file (`jrepository`) to use the services in JREPSVR. This buffer type update is recommended, but not required because the old FML16 buffer type can still work with the new JREPSVR.
	Platforms	All
	Workaround	BEA Jolt 1.1 user should update the BEA Jolt Repository file (`jrepository`) to use the services in JREPSVR.
CR017675	**Changing FML data types does not update JSH with changes.**	
	Problem	It does not appear that after changing the FML field data type from `int` to `string` the JSH gets updates.
	Platforms	All
	Workaround	After you change the FML field definitions, reboot JSL/JSH.
CR018177	**client_response() or target_response() method.**	

Table 5 Known Problems for Tuxedo 9.0

	Problem	Invocations on a `CORBA::Object` from a C++ interceptor in the `client_response() or target_response()` method is not supported. The results of such an operation are not predictable.
	Platforms	All.
	Workaround	None.
CR018211	**RLI administration restrictions.**	
	Problem	The RLI administrative functions of registering, unregistering, and changing interceptor order are not supported while the BEA Tuxedo application is running. The RLI administrative functions must only be done before issuing the `tmboot` command or after issuing the `tmshutdown` command. BEA Tuxedo application results are not predictable if interceptors are administered while the application is running.
	Platforms	All.
	Workaround	Do not register, unregister, or change the interceptor order while the BEA Tuxedo application is running.
CR018476	**Missing host:port on ISL -n option causes Application and TMNTS servers to crash.**	
	Problem	If you define an ISL without a `-n host:port` entry in a `UBBCONFIG` file, an application server that uses the notification server (TMNTS) and TMNTS crashes.
	Platforms	All.

Table 5 Known Problems for Tuxedo 9.0

Workaround	According to RFC 810, the document that specifies the syntax for IP hostnames, hostnames cannot contain an underscore (_) character. Therefore, a listening address such as `//my_computer.com:8000` is considered invalid and processes such as ISL and `tlisten` fail to start. This is a problem on Microsoft Windows NT systems because hostnames with underscore characters are often used. If you have a Windows NT system whose hostname contains an underscore, do one of the following: • Rename the machine so that its hostname does not contain an underscore, and change the corresponding name in the Domain Naming System (DNS) files or, if you are not using DNS, change the corresponding name in all the local HOST files. • Add a DNS alias (without an underscore) to the machine by adding a `CNAME` record. • Add an alias to the `%WINDIR%\System32\drivers\etc\hosts` file: `123.45.67.89 my_computer.com mycomputer.com` where the first entry is the IP address of the machine. Then perform the following procedure: 1. Reboot the machine. 2. In the `UBBCONFIG` file, use `my_computer.com` when specifying the `PMID` entry for this machine in the `MACHINES` section. 3. In all other places in the `UBBCONFIG` file, such as the `NETWORK` or `SERVERS` sections, or when starting listening processes, such as `tlisten` on the command line, use `mycomputer.com` as the hostname.
CR019447	**Signatures get dropped when floating point values are used.**

Table 5 Known Problems for Tuxedo 9.0

	Problem	When a process is determining whether to keep or disregard a previously attached signature on a message buffer, it recalculates the signature on that buffer. If the new signature does not match the old one, then the previously verified signature on the buffer is silently dropped by the system. The behavior of doubles and floats in this context may cause problems. Signatures are calculated on the encoded representation. For buffer types supplied by the BEA Tuxedo system, this calculation causes doubles and floats to be encoded as XDR doubles. XDR doubles may not have the same precision as the native doubles or floats on your machine. Therefore, when a buffer containing a floating-point number or double precision floating-point number is transported to a machine with different floating-point precision and decoded on that machine, the resulting value may differ from the original value. When the decoded number is then re-encoded from native format to XDR format, the encoding may be different. Therefore, the signature will not be verified and will be silently dropped, even if the application did not change the buffer at all.
	Platforms	All.
	Workaround	If you are using buffer types that are not supplied by the BEA Tuxedo system, replace the `_tmencdec` function in the types switch to use an encoded format with as much precision as the native format (everywhere).
CR019611	**`tpcall()` fails when it is invoked (with the `TPNOCHANGE` flag set and FML or FML32 buffers) by clients on pre-Release 7.1 nodes.**	
	Problem	In an MP configuration running multiple releases of the BEA Tuxedo system, a native client running on a release 6.5 (or earlier) node invokes `tpcall()` for a service on a release 7.1 node. If `tpcall()` sends an FML or FML32 buffer and the `TPNOCHANGE` flag is set, the call fails with `tperrno` set to `TPEOTYPE`. This problem does not occur for Workstation clients on pre-release 7.1 nodes.
	Platforms	All.
	Workaround	Do not set the `TPNOCHANGE` flag in `tpcall()` when pre-release 7.1 native clients are calling, with FML buffers, services on a release 7.1 node.
CR020175	**Automatic calls to `tpterm()` may not work for multithreaded and multicontexted clients.**	

Table 5 Known Problems for Tuxedo 9.0

	Problem	The BEA Tuxedo system automatically calls `tpterm()` if a client exits without calling it explicitly. In a multicontexted and multithreaded environment, however, release 7.1 and later systems cannot detect the number of contexts that are active when the client terminates; only a single call to `tpterm()` is made. As a result, only one context is shut down. Other limitations, especially when unsolicited thread notification is used, may also exist.
	Platforms	All.
	Workaround	Make sure that before a multicontexted client exits, it explicitly calls `tpterm()` for all active contexts.
CR020514	**There is a 1K limit on authentication tokens on `GWTDOMAIN` links.**	
	Problem	The amount of data that can be sent in authorization tokens over GWTDOMAIN links may not exceed 1024 bytes for security reasons.
	Platforms	All.
	Workaround	There is no workaround for a security provider that needs tokens larger than 1024. A security provider may be able to compress tokens.
CR025411	**SET operation fails to reset the state of the class from ACTIVE to OPEN.**	

	Problem	The `SET` operation is as follows:
		`TA_OPERATION SET`
		`TA_CLASS T_APPQSPACE`
		`TA_APPQSPACENAME MYQSPACE1`
		`TA_QMCONFIG D:\temp\tmp.1\files/QUE1`
		`TA_LMID L1`
		`TA_STATE OPE`
		The output does not show the `TA_STATE` as `OPEN`, instead it shows it as `ACTIVE` only.
		Note: This happens for `tpadmcall` only. The `SET` operation passes for `tpcall`, `tpacall` and `tpenque`.

	Platforms	Microsoft Windows 2003 Server.
	Workaround	Use `tpcall` or `tpacall` instead of `tpadmcall`.
CR031385	**Value of `TA_CURDISPATCHTHREADS` field in `TA_SERVER` class in not updated properly.**	

Table 5 Known Problems for Tuxedo 9.0

	Problem	The value of the `TA_CURDISPATCHTHREADS` field in the `TA_SERVER` class is not updated properly on Windows 2003 whenever there is a new thread started for the server.
	Platforms	Microsoft Windows 2003 Server.
	Workaround	None.
CR031816	**Requests on methods using multithreading appear to time out prematurely.**	
	Problem	In some timing situations involving multithreaded applications, a method can time out too early. A symptom of this problem is an error message similar to the following in the log file: `LIBTUX_CAT:669 "ERROR: Message operation failed because of the invalid message queue identifier."` Note that this error message can also result from insufficient configuration settings for options such as `MAXACCESSORS` and `MAXWSCLIENTS`. Verify these settings before increasing the timeout settings.
	Platforms	All.
	Workaround	Increase the `UBBCONFIG` file settings for `BLOCKTIME` and `SCANUNIT`. For example: `SCANUNIT 5, BLOCKTIME 6000`.
CR039550	**Memory leak using poll_next_response() for native client.**	
	Problem	When running a CORBA client in native mode, calls to `poll_next_response()` create a small memory leak. Because `poll_next_response()` is typically invoked iteratively, this may eventually result in a `NO_MEMORY` exception.
	Platforms	All.
	Workaround	One solution is to use IIOP mode rather than native mode. IIOP mode does not have the problem. Another is to insert a short pause in the iterative loop so that the leak does not expand too quickly. Even a 10 millisecond pause in the loop will greatly slow down the leaking. As large a pause as is acceptable is best.
CR042034	**ud32 crashes when `SYSTEM_ACCESS=PROTECTED` and with `-C tpsysadm`.**	
	Problem	The ud32 `-C tpsysadm` crashes when `SYSTEM_ACCESS` in the configuration file is set to `PROTECTED`.
	Platforms	All.

Table 5 Known Problems for Tuxedo 9.0

	Workaround	None.
CR042294	**ACL Security Failure on `tpconnect` does not return disconnect to OSI-TP Gateway.**	

	Problem	Whenever a `tpconnect (TPSENDONLY)` is issued by a client to a service for which the ACL Security fails (client user ID not found in a group for which the service is allowed), a `nw_disconnect` is not returned to the OSI TP 4.0 Gateway. The following error messages are output to the ULOG:
		`111324.DALNTSMP44!CONVSERV.240.310.0:`
		`LIBTUX_CAT:1515: WARN: Access control violation - user 2 on SITE1 tried to access SERVICE TOUPPER`
		`111324.DALNTSMP44!CONVSERV.240.310.0:`
		`LIBTUX_CAT:6187: WARN:`
		`AUDIT_POSTOP`
		`SECURITY FAILURE: who = , operation_name = CONNECT OPERATION,`
		`operation_target = TOUPPER`

	Platforms	All.
	Workaround	If it is a valid client, create the ACL entry.
CR045148	**Tlisten has a typo on its usage.**	

	Problem	The usage of tlisten shows as:
		`tlisten -l naddr [-d dev] [ -u {uid-# |uid-name}]`
		`[-z {0|40|56|128}]`
		`[-Z {0|40|56|128}]`
		`-l naddr` should be `-l nlsaddr` as it appears in the document. This typo misleads the user to enter in an incorrect input.

	Platforms	All.

	Workaround	Use this usage instead:
		`tlisten -l nlsaddr [-d dev] [ -u {uid-# | uid-name}]`
		`[-z {0|40|56|128}]`
		`[-Z {0|40|56|128}]`

CR045947	**NO_MEMORY exception on HP.**	

Table 5 Known Problems for Tuxedo 9.0

	Problem	Testing of the Active Object Map (AOM) limit for 100,000 active objects gets the `NO_MEMORY` exception that reports insufficient physical memory. The HP system, on which the test fail, had about 850 MB of physical memory. The Solaris system, which had 1.025 GB, worked okay for the test.
	Platforms	HP-UX
	Workaround	None
CR046254	**TA_DMCODEPAGE returns garbage.**	
	Problem	A DMIB test case tries to create a remote service entry using ud32; its TA_DMCODEPAGE was *not* passed in as part of the input. The operation succeeds without any issues, but in the return packet, the TA_DMCODEPAGE parameter is returned as garbage.
	Platforms	All.
	Workaround	Ignore the output for TA_DMCODEPAGE in the return packet.
CR050552	**SSL URL syntax parsing not working correctly.**	
	Problem	A URL address list containing randomized address groups followed by comma-separated lists of addresses is not being parsed correctly.
	Platforms	All.
	Workaround	Do not mix URL types in a list.
CR051108	**`Tlisten.pw` file not the same between BEA WebLogic Enterprise 5.1 and BEA Tuxedo 8.1 or 9.0.**	
	Problem	The creation of the tlisten.pw is different between BEA WebLogic Enterprise 5.1 and BEA Tuxedo 8.1 or later. The impact to the customer is in a mixed environment when there are BEA Tuxedo 8.1 or later masters and BEA WebLogic Enterprise 5.1 non-masters (potentially the same thing could be true for previous versions of BEA Tuxedo ATMI). When booting, the user will get a security violation in the `ULOG` when trying to connect the other system that has the previous version. **Note:** When viewing the files, the contents may look the same, but they are not, the file termination string is different by one character. The only way to ensure they are the same is to do a copy or file transfer.
	Platforms	All.

Table 5 Known Problems for Tuxedo 9.0

	Workaround	To resolve this problem or prevent it from happening, a single version of the `${TUXDIR}/udataobj/tlisten.pw` file should be copied to all systems in the domain(s). This file can either be taken from the BEA Tuxedo 8.1 or later home directory or the previous version's home directory.
CR071576	**tlisten.pw file not the same between WLE 5.1 and Tuxedo 8.1 or later**	
	Problem	In a mixed environment with Tuxedo 8.1 or later `MASTERS` and WLE 5.1 non-`MASTERS`, you will get a security violation in the `ULOG` when trying to connect the WLE 5.1 system. This same issue may be true for previous versions of Tuxedo-ATMI.
	Platforms	All.
	Workaround	Copy a single version of the `$TUXDIR/udataobj/tlisten.pw` file to all systems in the domain. You can use this file from the Tuxedo 8.1 or later installation or from the previous version's home directory. When viewing the files, the content may look the same, but they are not. The file termination string is different by 1 character. The only way to ensure the files are the same is to do a copy or file transfer.
CR071624	**FML function Fgetalloc and CFgetalloc gives GPF in Windows 2003.**	
	Problem	`Fgetalloc` gives GPF when proper FBFR`*`, `fieldid` is passed.
	Platforms	Windows 2003 Server
	Workaround	The problem can be fixed in one of the following ways in `fml_nt.mak`: • Use buildclient directly to compile from `.c` files into `.exe` files, setting `CFLAGS` (not `CCFLAGS`) to specify any extra compiler options. • Add the `/MD` flag to the `cl -c` line.
CR071960	**buildclient and buildserver fails with unresolved references.**	
	Problem	`buildclient` and `buildserver` fails with unresolved references.
	Platforms	All
	Workaround	When building a client or server with the XML library (`libtxml`), export CC= the C++ compiler before using the `buildclient` or `buildserver` command.

Table 5 Known Problems for Tuxedo 9.0

CR089272, CR217048	**Jolt 8.1 and 9.0 do not work with WLS 6.1 SP4 when security context is enabled.**	
	Problem	An exception is sent when a client request to WebLogic Server 6.1 (when jolt pool is configured) with Jolt 8.1 or 9.0 and the security context is enabled.
	Platforms	All.
	Workaround	There were changes made to the security feature in WebLogic Server 7.0. Due to that, Jolt had to be changed to work with the new security in WebLogic Server. The result is that if security context propagation is enabled, then Jolt 8.1 or 9.0 clients cannot be used with WebLogic Server releases earlier than 7.0.
CR092413	**Inconsistent number of retries to connect to remote domain.**	
	Problem	The number of retries to connect to remote domains are inconsistent when RETRY_INTERVAL value is different for different remote domains in a local domain.
	Platforms	All.
	Workaround	The retry message is only printed once in the ULOG even when MAXRETRY > 1 and multiple retries are attempted. Check the ULOG for the following: The ULOG will show "INFO: Stopped retrying domain" if the remote domain could not be reached. Otherwise, the ULOG will show "Connection established."
CR092441	**On a 64-bit platform running Tuxedo 6.5, tppost of a VIEW or VIEW32 to Tuxedo 8.1 or 9.0 sometimes fails with LIBTUX_CAT:1555.**	
	Problem	On a 64-bit platform running Tuxedo 6.5, `tppost()` of a VIEW or VIEW32 buffer to a Tuxedo 8.1 or later node sometimes fails with: `LIBTUX_CAT:1555: ERROR: Unsolicited message encoding/decoding failed (_tmencdec(TMDECODE) tperrno=12).`
	Platforms	All.

Table 5 Known Problems for Tuxedo 9.0

	Workaround	Depending on the specific set of fields contained in the VIEW or VIEW32 buffer, the Tuxedo 6.5 node may not be allocating sufficient space to decode the buffer. The problem either always happens or never happens for a specific VIEW or VIEW32. If this problem is occurring in your application, please contact BEA Customer Support to obtain a rolling patch for the Tuxedo 6.5 node.
CR092647	**tmboot fails in MP mode configuration between Tuxedo 8.1 or later and WLE 5.1.**	
	Problem	`tmboot` with Master as Tuxedo 8.1 or later and WLE 5.1 as non-Master fails with error message in ULOG.
	Platforms	All
	Workaround	Apply the WLE 5.1 RP60 or greater to resolve this issue.
CR092705	**tpcall from Tuxedo 8.1 or 9.0 to Tuxedo 7.1 fails with TPESYSTEM error in Domain configuration.**	
	Problem	If connection policy is `INCOMING_ONLY` in Tuxedo 8.1 or 9.0 and `ON_DEMAND` on Tuxedo 7.1, all the services are shown in suspend mode, and client call fails `TPENOENT` error.
	Platforms	All.
	Workaround	This is a problem with base Tuxedo 7.1 release. You must apply Tuxedo 7.1 RP 135 or later to resolve this issue.
CR092867	**All enqueue fails after Unprivileged user enqueues between WLE 5.1 and Tuxedo 8.1 or later, when security is set to MANDATORY_ACL.**	
	Problem	All enqueue fails after unprivileged user enqueues when security is `MANDATORY_ACL` and the configuration is MP mode. SHM mode works fine in Tuxedo 8.1 and later.
	Platforms	All.
	Workaround	Ensure that only authorized users access services during the limited period when the application is partially upgraded from WLE 5.1 to Tuxedo 8.1 or 9.0.
CR216274	**The performance of GWTDOMAIN will downgrade dramatically if the two domains located at one AIX machine.**	

Table 5 Known Problems for Tuxedo 9.0

	Problem	This issue has somewhat of a relationship with the ngale algorithm when the two end points of TCP located at one AIX box. Under such situation, the communication just involves system buffer transitions, and contest with IPC operations. Nagle algorithm results in performance downgrade dramatically.
	Platforms	AIX
	Workaround	When you turned off the ngale in TCP, the behavior will be normal. Tuxedo GPNET provides an environment variable called SETTCPNODELAY to control the nagle algorithm usage. Set SETTCPNODELAY to any value before GWTDOMAIN startup.
CR220004	**Interop: Server group cannot be migrated from Tuxedo 9.0 to Tuxedo 6.5 via MIB request.**	

	Problem	The server group cannot be migrated from Tuxedo 9.0 to Tuxedo 6.5 via MIB request. For example, ud32 accepts a MIB request input file as:
		`TA_OPERATION SET`
		`TA_CLASS T_GROUP`
		`TA_STATE MIGRATING`
		`TA_SRVGRP GROUP1`
		The reason is an operation code of interoperability was not handled well after Tuxedo 6.5.

	Platforms	All.

	Workaround	Use tmadmin to do the group migration. For example, with the following tmadmin commands:
		`>shutdown -R -g GROUP1`
		`>migg GROUP1`

CR222951	**Tuxedo 9.0 may consume more memory**	
	Problem	Due to the newly added features of Tuxedo 9.0, adding plenty of new code and data into underlying Tuxedo shared libraries (e.g. libengine.so), some of them now have much bigger size than they previously did in Tuxedo 8.1. The size increase of shared libraries cause running Tuxedo system and application processes depending on those libraries to consume more memory than before.
	Platforms	All.

Table 5 Known Problems for Tuxedo 9.0

	Workaround	Add memory capacity if upgrading an existing application. See Installation Upgrade Considerations section of Release Note.
CR223832	**cnsbind utility core dumps on AIX platform. Although it core dumps, it still can finish its work.**	
	Problem	This occurs when exiting and when enabling CORBA trace.
	Platforms	AIX.
	Workaround	Do not use CORBA trace.
CR218842, CR219402	**CORBA: limitation of C++ Compiler on AIX 5.3**	
	Problem	There is a limitation on CORBA code to be compiled with C++ compiler on AIX 5.3. Type CORBA::Any or structure containing an item of this type cannot be used to declare a global or static object. Otherwise, the execution will crash when exiting due to referring to already-destructed TypeCode.
	Platforms	AIX
	Workaround	Use these types to declare dynamic objects.
CR217411	**SNMP: SMUX agent does not start on Windows.**	
	Problem	Start `snmp_integrator.exe` as a service with start parameters `-p` and `-r`. Start `tux_snmpd.exe` as a service with parameters `-r`. It exits with error "Could not start the <agentname> service on local computer".
	Platforms	Windows
	Workaround	Configure the correct TUXDIR as the system environment before starting up the SNMP agent.
CR195891	**Outbound IIOP connection cannot end by releasing the reference to the remote object.**	
	Problem	After the outbound user limit (set by -o and -u parameters of ISL CLOPT) being reached, the server tries to release a user by assigning an empty pointer to the reference of the remote object. In 8.0RP292, assigning 0 to the reference can release a user. In Tuxedo 8.1 and 9.0 it cannot.
	Platforms	Windows

Table 5 Known Problems for Tuxedo 9.0

	Workaround	None
CR228589	**Jolt Runtime MBeans could not be accessed using the console.**	
	Problem	The Jolt Connection Pool monitoring pages in the Administration Console are unable to access the Jolt Runtime MBeans, and therefore Jolt connection pools cannot be monitored or reset using the Administration Console.
	Platforms	All
	Workaround	Access the Jolt Runtime MBeans using WebLogic Scripting Tool (WLST).
CR236565	**CORBA: The generated stub/skeleton C++ files for an array of valuetype cannot be compiled.**	

	Problem	The C++ IDL compiler does not support array of valuetype definition. For example,
		`//IDL valuetype A{ ......`
		`}`
		
		`typedef A A_array[10];`
		
		The generated C++ code for the above IDL file cannot pass compilation.

	Platforms	All.
	Workaround	None.
CR238601	**CORBA valuetype: stub code for value box of some underlying boxed types does not provide enough overloaded operators or accessor/modifer functions.**	

Table 5 Known Problems for Tuxedo 9.0

	Problem	IDL compiler does not generate enough overloaded operators or accessors/modifiers functions in the C++ stub code for valuebox classes of some underlying type. These operators or functions are convenient instruments to manipulate data/members of those underlying boxed type of the valuebox class. See the following list. • valuebox class of sequence types: overloaded subscriptor operators are not provided. • valuebox class of array types: overloaded subscriptor operators are not provided. • valuebox class of string types: overloaded subscriptor operators are not provided. • valuebox class of wstring types: overloaded subscriptor operators are not provided. • valuebox class of struct types: underlying struct member accessor/modifier functions are not provided. • valuebox class of union types: underlying union member accessor/modifier functions, union discriminant functions are not provided. • valuebox class of any types: overloaded insertion/extraction operators of any type are not provided.
	Platforms	All.
	Workaround	First get the value/instance of the underlying boxed type using accessor method `_value()` of the valuebox class. Then use relevant overloaded operators or accessor/modifier functions of the real boxed type.
CR241622	**CORBA valuetype: valuetype with array member cannot be constructed and a core dump results.**	

Table 5 Known Problems for Tuxedo 9.0

	Problem	If there are array type members defined in a valuetype in the IDL file, the generated C++ codes are not correct.
		For example,
		`//IDL`
		`typedef long LongArr[10];`
		`valuetype A {`
		`  ......`
		`  public LongArr arr_val;`
		`  ......`
		`};`
		
		The generated C++ file may cause a runtime problem.

	Platforms	All.
	Workaround	None.
CR244880	**Valuetype with underlying non-basic typed sequence members is not supported.**	

	Problem	Valuetype definition with sequence members only support the following basic underlying types: signed/unsigned short, signed/unsigned long, octet, float, double, signed/unsigned long long, long double.
		For example,
		`//IDL`
		`valuetype item {`
		`  ......`
		`};`
		`typedef sequence<item, 10> ItemSeq;`
		`typedef sequence<short, 5> ShortSeq;`
		`valuetype V {`
		`  public ShortSeq s_val; // support`
		`  public ItemSeq i_val; // Do not support`
		`};`

	Platforms	All.
	Workaround	None.
CR244894	**Implementation of valuetype custom marshaling does not process wide chars correctly.**	

Table 5 Known Problems for Tuxedo 9.0

	Problem	CORBA specification provides custom marshaling to override the default marshaling/unmarshaling model of valuetype. Tuxedo CORBA ORB does not process wide chars correctly when custom marshaling. The `read_wchar_array()` method of `CORBA::DataInputStream` and the `write_wchar_array()` method of `CORBA::DataOutputStream` do not process wide chars properly.
	Platforms	All.
	Workaround	None.

Documentation Addenda

This topic includes the following sections:

- Documentation Addenda for BEA Tuxedo 9.0
- Documentation CD-ROM Search Applet Limitations

Documentation Addenda for BEA Tuxedo 9.0

Table 6 lists documentation addenda.

Table 6 BEA Tuxedo 9.0 Documentation Addenda

CR019784	**Some Windows NT and UNIX syntax errors in BEA Administration Console Online Help.**	
	Problem	The syntax for pathnames (specifically the use of slashes and backslashes) on UNIX and Windows systems is reversed in various sections of the Online Help for the BEA Administration Console.
	Workaround	Pathnames on UNIX systems are specified with slashes. Pathnames on Windows systems are specified with backslashes.
NA	**The Run Simpapp step is incorrect in the BEA Administration Console Online Help.**	
	Problem	The procedure provided for "Step 6: Run Simpapp" is incorrect in the Online Help for the BEA Administration Console.

Table 6 BEA Tuxedo 9.0 Documentation Addenda (Continued)

	Workaround	Modify this procedure as follows:
		1. Change the item number 2 in step 6 to read as follows:
		From the command shell that you used in step 1, set and export the `TUXCONFIG` environment variable as follows:
		`$ TUXCONFIG=your_simp_dir/tuxconfig`
		`$ export TUXCONFIG`
		2. Ignore item 5 in step 6.

CR092416	**MBSTRING is treated the same as CARRAY. Use memcpy with MBSTRING instead of strcpy.**	
	Problem	Error occurs if using `strcpy` instead of `memcpy` when you copy the string data after `tpalloc("MBSTRING")`.
	Workaround	Use `memcpy` with MBSTRING.
CR094270	**GW_KEEPALIVE env does not work in Tuxedo 8.1 or 9.0.**	
	Problem	Documentation specifies enabling TCP Keepalive using the GW_KEEPALIVE environment variable. For BEA Tuxedo 8.1 or later, TCP Keepalive is enabled through the DMCONFIG file.
	Workaround	If you are using TCP Keepalive with Tuxedo 7.1 and need to migrate to Tuxedo 8.1 or later, configure this feature using DMCONFIG. For information refer to the *File Formats, Data Descriptions, MIBs, and System Processes Reference* for how to configure DMCONFIG file to use TCP Keepalive in Tuxedo 8.1 or later.

Multithreaded CORBA C++ Client Considerations

Table 7 lists a multithreaded CORBA C++ client consideration.

Table 7 Multithreaded CORBA C++ Client Considerations

NA	Multithreaded CORBA C++ clients are supported.
	BEA Tuxedo provides support for multithreaded CORBA client applications, for both the thread-per-request and the thread-per-object concurrency strategies. Build a multithreaded CORBA client as you would any CORBA client application. Whether the client application functions as a multithreaded client application depends on the environment in which it is run. Note the following considerations for multithreaded CORBA client applications running in the BEA Tuxedo environment: • Multithreaded CORBA client applications using IIOP are supported. • Multithreaded native CORBA client applications are not supported. • Multithreaded joint client/servers are not supported. • A multithreaded CORBA client application is limited to a singe Bootstrap object. • A multithreaded CORBA client application is limited to a single logon to the BEA Tuxedo domain. • CORBA client applications that use static invocation can use multiple threads. • CORBA client applications that use the dynamic invocation interface (DII) cannot be multithreaded.

Documentation CD-ROM Search Applet Limitations

The BEA Tuxedo 9.0 documentation CD-ROM includes a standalone Java search applet to help you find topics. This section describes current limitations with the search applet.

UNC Pathnames

The Java search applet on the BEA Tuxedo 9.0 Online Documentation CD-ROM uses precompiled search databases of topics. You must adhere to one of the following options to use the Documentation CD-ROM search feature:

- Use the CD-ROM on a local CD-ROM reader.
- Copy the content of the CD-ROM to a local drive on your system.

- Map a network drive to a remote, shared device that contains the CD-ROM or a copy of the CD-ROMs contents; in your browser, use the network drive to find and open the `index.htm` file in the `\doc\tux90\` directory.
- Copy the CD-ROMs content to a Web server on your corporate intranet. Make sure that `index.htm` is the default filename used by the Web server software. The `\doc\tuxedo\tux90\` directory of the BEA Tuxedo 9.0 Documentation CD-ROM contains an `index.htm` file; it is the documentation home page. If your Web server software does not allow you to use a file named `index.htm`, make a copy of `index.htm` and rename the copy to the default filename you must use, such as `default.htm`; keep both the `index.htm` file and the copied file in the same directory.

You cannot use the search applet if you accessed the CD-ROM or a copy of its content through a Universal Naming Convention (UNC) path. For example, UNC paths are used by the Windows NT Network Neighborhood. The search applet will not interpret the relative paths to the matched, target `*.htm` pages because the UNC path is added to the beginning of each link. Use one of the four recommended methods described in the preceding list.

You can use UNC paths, such as accessing the documentation CD-ROM on a Network Neighborhood system's shared CD-ROM device, for all other relative hyperlinks on the CD-ROM. Only the search applet's results list is affected by this UNC limitation.

CLASSPATH Environment Variable

On some UNIX platforms, you may encounter a browser error when you open the Online Documentation CD-ROMs Search page. For example:

```
Unable to start a java applet: Can't find 'java40.jar' in your CLASSPATH.
Read the release notes and install java40.jar properly before restarting.
```

If the search applet does not work on your UNIX platform, try using the latest Netscape browser for the platform and add the Netscape Java ARchive (JAR) file to your `CLASSPATH` environment variable. The path to the JAR file is in the directory in which you installed Netscape.

For example:

```
CLASSPATH=mytools/netscape/communicator/program/java/classes/java40.jar
```

After you revise the `CLASSPATH`, exit Netscape and then restart Netscape in the updated environment. When you access the Search page, the search feature should work properly.

How to Obtain Patches

To obtain patches, access the BEA Customer Support page at `http://www.bea.com/support/` and open a New Case to request the patches. BEA customers can gain access to support information by accessing the Customer Support page and registering for a Web account.

You can also contact Customer Support by using the contact information provided on the BEA Tuxedo 9.0 Customer Support Card, which is included in the product box.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address
- Your machine type and authorization codes
- The name and version of the product you are using
- A description of the problem and the content of pertinent error messages

APPENDIX A

Software and Documentation Problems Fixed in Tuxedo 9.0

The following tables list the software and documentation problems that have been fixed in Release 9.0. Problems are listed by CR (Change Request) number.

Any software or documentation fixes that are made to Release 9.0 after these *BEA Tuxedo Release 9.0 Release Notes* are printed will be reported in updates to this section in the online version of these *Release Notes*. The online version is available through the BEA Tuxedo 9.0 Release Notes link at

`http://e-docs.bea.com`

Problems Fixed in This Release

Table A-1 lists problems fixed in BEA Tuxedo 9.0.

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR Number	Former Problem Description
CR016561	Documentation does not warn users to avoid long blocking actions in tpsvrinit(3c).
CR016909	txrpt cannot report service statistics before 100 days (CR091175)
CR017023	Unclear description of the `-p` option on the `servopts(5)` page.
CR018788	Documentation is incorrect on the effect of a call to tpenqueue() or tpdequeue() on the TPQCTL structure.
CR019383	Missing requirements for running EventBroker servers in an MP configuration.

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR025419	Aborted trans stay in BB".
CR031657	Problem initializing ANY data types with array data.
CR036566	TPNOBLOCK does not work with WSL -N.
CR041343	Jolt12: Jolt Session Pool slows WLS down when Tuxedo node/network failure occurs
CR041877	`tmboot` reports `"executable file not found"` error.
CR049498	After installing Tuxedo 7.1 on a machine that had previously run earlier versions of Tuxedo, the Tuxedo performance object does not appear in the object window of the Performance Monitor.
CR050166	C++ definitions in wrong order in IDL->C++.
CR050406	Generated value `type _copy_value()` routine not complete.
CR050516	Typecodes have incompatible type for value factories
CR050537	Multithreaded /WS client tpinit fails TPEPERM & userlog LIBTUX_CAT:6251
CR050593, CR051024, and CR050664	Any insertion for WStringValue not defined. Any insertion for ValueBase not defined. Cannot unmarshall sequence of Any's
CR050720	Tuxedo 6.4 : TPETIME on tpgetrply()/tpcall()/tpcommit() during transaction
CR050760	Tuxedo does not correctly recognize Null value types from WebLogic Server.
CR050841	Nested valuetypes cause IDL compiler core dumps.
CR053911	Change Request for printing service name for TPENOENT.
CR054994	`tmipcrm.exe` is missing on W2K
CR057981	Service advertised as AVAIL is premature during tpsvrinit
CR058313	TMFFNAME core after application servers died and MP node shutdown/reboot
CR065239	Create functionality to clean all stale messages
CR066833	Patch installation script does not work if it is /WS-only installation.
CR078694	/Q messages show up out of order in some transaction scenarios.

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR079773	Connection drop does not happen when APPKEEPALIVE set.
CR081929	Bridge process shutdown when a connection is made via portscan tool
CR082016	Tuxedo 8 client to Tuxedo 6.5 server has one thread that pends forever.
CR082856	Web GUI hangs when messages on the queue.
CR082934	Thread notification is not reliable.
CR083954	Tuxedo 8.0 Windows NT/2000> incomplete installation (missing tlisten/ProcMgrV8.0/TuxAdmin)
CR084750	C++ IDL compiler crashes on a struct containing a union
CR084953	Cannot run Tuxedo out of the box, customer need to make a link manually after installation
CR087140	Enhance tmadmin -v option to include patchlev, OS, and number of bits
CR087157	tx_open() not called by default tpsvrinit() in PROTECTED mode.
CR087314	/Q when tpdequeue with TPEDIAGNOSTIC >1 , LIBTUX_CAT: 6031/428 when shutdown.
CR088355	Workstation client dumps core when tpcall is made without explicit tpinit.
CR088823	Now legal for hostname to start with letter or digit (GP_CAT:1287)
CR088912	SSL cert chain attack possible with CAs without Basic Constraints
CR089061	Jolt 8.0 - Number of JSHs included in license calculation.
CR089214	Limitation supporting multiple Tobj_Bootstrap to the same domain is not clearly documented.
CR089403	Jolt 1.2.1 - jrly leak in single threaded applications
CR089423	tpcommit returns TPETIME when xa_commit returns XA_RETRY
CR089456	NO_RESOURCES exceptions if many ISH processes used
CR089476	WLE4.2 Java Clients hang if 1st logon() attempt fails
CR089573	Inaccessible remote domain causes incorrect output from dmunloadcf.
CR089738	MIB request fails with GETNEXT operation on NT
CR090096	Webgui cannot display stats for all the servers selected

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR090248	CORBA.INTERNAL error in transactional EJB to Tuxedo CORBA call.
CR090390	Request to propagate the fix for CR054218 from Tuxedo 7.1. TUX7.1 WS: Client may immediately get network timeout.
CR090530	/Q false deadlock detection.
CR090614	CORBA C++ client gets the IDL:omg.org/CORBA/BAD_TYPECODE:1.0 exception when using an Any containing a sequence of sequences
CR090661	WSNAT_CAT:1057:ERROR:Error processing message received from network.
CR090706	/WS client on Windows takes a long time in tuxedo network layer.
CR090732	Memory leak using tpacall (TPNOREPLY) with TPU_THREAD.
CR090942	Tuxedo/eLink SNA: DMADM/GWADM taking too long at boot time.
CR090948	CORBA callback failed with GP_CAT:1287
CR091184	Client does not receive TPETIME if TPU_THREAD is set in tpinit flags.
CR091307	Second tpinit hangs when NOTIFY=SIGNAL in UBBCONFIG and flags=TPMULTICONTEXTS.
CR091563	Callback invocation never timeout
CR091604	Multithreaded CORBA server intermittently gets CORBA::INTERNAL exception within transaction
CR091631	Native client failed to get unsolicited message with tpinit (flags=TPU_THREAD).
CR091632	tpexport() dumps a core file when using embedded FML
CR091673	BBL may go to sleep, causing slow response times to .TMIB requests
CR091686	Unsolicited message fails with Tuxedo 8 patch 109, Jolt 8 client
CR091757	Maximum action table size reached
CR091803	Addition of elements in the DM_MIB is limited.
CR091816	(Domain) - two-phase commit with TUX65 gives LIBGWT_CAT:1072 when CMPLIMIT=0.
CR091832	/WS client goes out of handles if tpcall() returns TPENOENT

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR091903	Client stubs don't compile if IDL defines recursive constructed types between separate IDL files
CR091904	Win: empty wstring are not marshalled
CR091915	TRANTIME cannot be changed dynamically.
CR092010	FactoryFinder parser fails if `.ini` file not terminated with comment
CR092036	Server failed to shutdown with LIBTUX_CAT:6031 after enqueue/dequeue messages to /Q space
CR092356	BBL dumps a core if TMS queue full and TMSYSEVT booted
CR092357	Native client tpinit(flags=TPMULTICONTEXTS) leaks memory when application is not booted.
CR092389 (Duplicate CR080631)	WLE 5.1 - several ISL/ISHs shutdown and restarted may lead to client not reconnecting to any ISL
CR092416	Error occurs if using `strcpy` instead of `memcpy` when you copy the string data after `tpalloc("MBSTRING")`
CR092528	BRIDGE displays LIBTUX_CAT:5026 error
CR092679	Installation fails on HP-UX when LANG=ja_JP.eucJP.
CR092682	Installer does not work on Japanese Solaris box.
CR092719	Regression from CR036566 that WSL -N is no longer working with tpcall (TPNOBLOCK).
CR092751	Add timeout to tmshutdown used without -w or -k so that it doesn't hang
CR092753	Enhance SVCTIMEOUT to optionally send SIGTERM before SIGKILL
CR092761	NT/W2K: dmloadcf fails if user does not have write permission to the root directory
CR092856	Bad timing when cross-domain transaction is committed.
CR092948	For buildtuxedo to work on Windows 2000 Advanced Server, you must log out and then log in to the system.
CR093041	Need to support 256 byte directory names in -d option of viewc
CR093053	Windows 2000 AS :- AUTHSVR fails to boot without tpusr file

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR093216	Load balancing for transactions does not work with service caching.
CR093219	tpterm fails with LIBWSC_CAT:1477.
CR093434	University OBV sample does not build
CR093526	TMFFNAME -N failed with LIBMFF_CAT:1011 if restart and exactly 2 registered factories
CR093534	Database not in sync for slave TMFFNAME -N (especially in MP config)
CR093575	Strange behavior when dynamically adding a 3rd remote domain.
CR093601	Not able to use dmunloadcf o/p as i/p to dmloadcf for eLink OSITP
CR093661	Tux6.5rp380> ULOG overflowed with "INFO: Goes into sleep 10 second" on a busy system with a lot of TMIB calls
CR093665	File descriptors above 19 are not released in spawned servers
CR093721	Inconsistent behavior between Native & WS client when invalid TPMBENC is set
CR093728	tmboot timeout feature
CR093763	tmadmin/pclt displays inaccurate information
CR093777	Memory leak in native client when one of the applications is down.
CR093780	TMFFNAME doesn't produce error when local and remote factory identifiers clash
CR093804	Hot Upgrade from WLE5.1 (Tuxedo6.5) to Tuxedo8.1 fails with "CMDTUX_CAT:1590" error.
CR093943	tpcancel with invalid descriptor does not return TPEBADDESC
CR09400	tmloadcf done twice causes CMDTUX_CAT:1601 when TUXCONFIG length is 256
CR094212	No load balancing across domains with TMQFORWARD.
CR094243	Not able to execute service from local domain.
CR094342	AccessControlException when using a Jolt applet
CR094358	During silent install, there is an error in the files copied for CORBA if install option is invalid and silent install defaults to Full Install.
CR094400	If ISL not defined in UBB, CORBA native client gets much slower.

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR094467	Dynamic LoadLibrary()/FreeLibrary of libwsc.dll
CR094635	Request for a timeout option in restartsrv
CR094704	Win32: Cannot shutdown/release ORB, then do new ORB_init.
CR094840	Installation error due to potential files being corrupted.
CR095032	Factories not being exported when they should, and vice versa
CR095039	MAXPENDINGBYTES functionality in GWTDOMAIN
CR095042	University OBV sample failed on HPUX
CR095326	TA_BRTHREADS can't be set while instantiating a fresh T_DOMAIN class
CR095619	On NT/W2K, message catalogs LIBSEC_CAT and LIBGWTE_CAT are missing
CR096310	Restarted domain cannot re-establish connections
CR096338	Linux g++ & gcc compiler throws EJBHome_c.h:104
CR096366	Tuxedo 8.1 fails to cache .xsd file
CR096585	tmshutdown -k kill does not kill C++ server hanging in static constructor on Windows.
CR096709	Fix oracle_close not to check for the RM string
CR096779	TUX8.0 rp135 CORBA interface cannot be suspended.
CR097357	T_SERVERCTXT class member TA_CURRSERVICE not updated when server crashes or killed.
CR097496	tpgetrply with TPNOFLAGS does not receive BBL timeout with WSL -N.
CR097498	Added ability to turn off quick checking of encoding names
CR097504	WSL -t interoperate with Tux65 /WS client causing stdout:ERROR:Couldn't find with a dummy tpusr file.
CR097518	Internationalization (i18n) performance improvements
CR098116	Incorrect table size in LIBGW_CAT:5205
CR098456	CORBA client crashes unmarshalling recursive valuetype
CR098533	Add newly translated catalog files, and fix make files

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR098915	Remote factory (wrongly) returned when RNAME matches
CR099452	Fix AIX COBOL server linking (add -CC before -qstaticinline)
CR100444	Workstation runtime error, cannot find libtux.dll.
CR100608	/Q deadlock under heavy load/transactions in multi-CPU machine.
CR100679	Threaded CORBA client dumps core at resolve_initial_references() on HP
CR100805	Tuxedo Admin applet does not appear in control panel.
CR101022	`tpacall` with flag `TPNOREPLY` yields invalid service increment.
CR101122	Missing `CMDTUX.text` file in `locale/ja_JP.eucJP` directory
CR101140	TMNTS core due to the Fboolco32() crashed at invalid filter expression.
CR101251	`jrly` executables in `bin` and `udataobj/jolt/relay` should match
CR101290	Commit in one-phase optimization performed by TMS as opposed to server.
CR101321	Performance counters use incorrect registry key/Add button disabled.
CR101769	When GW_MAXPENDINGBYTES_FILE used, LIBGWT_CAT:1027 appears under heavy conversational client load
CR101963	Memory leak in WSH when WSTYPE was specified.
CR102710	SSL implementation subject to vulnerability attack.
CR102793	tmadmin dumps a core in printclient in MP mode(interoperability between Tux 8.0 with Tux 6.5)
CR102884	Tuxedo 7.1RP242> wrong initial status of service (via remote domain) when it is dynamically added via TMIB calls.
CR103268	WSL -N option does not work with TPU_THREAD.
CR103276	Unaligned access messages on Tru64 5.1.
CR103372	Jolt 8 client-only installation, applying RP does not copy/update the patchlev and it will copy over some unwanted CORBA sample
CR103492	Inappropriate memory growth when using large and small message buffers

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR103546	Enable MBSTRING to be self-describing if sendlen set to zero
CR103666	ACLs are blocking the call to custom Authentication Service
CR103980	With JOLTI18N encoding on, buffer was always truncated to 256 bytes
CR104043	Include -mt in buildclient/buildserver compiler flags on Solaris
CR104062	ATMI calls fail after network timeout.
CR104396	Fboolev does not evaluate regular expressions correctly for string
CR104496	Tuxedo 8.1 cannot boot slave node by using thread bridge: brget_node OUTOFBOUNDS
CR104498	non-JOLTI18N standalone java client should not be forced to include jolti18n.jar file into their $CLASSPATH
CR104957	User event/message is missing or lost when it is going through the /Q (evctl.flags=TPEVQUEUE) when slave TMUSREVT -S is used
CR105003	Unaligned access (SIGBUS) in unmarshal routine for valuetype
CR105011	Interoperability issue of tmadmin commands
CR105139	Map TPEOS to CORBA::TRANSIENT (used to be CORBA::INTERNAL)
CR105323	AUTHSVC gets TA_OCCURS 0 from TMIB T_CLIENT request when one user logs on.
CR105485	Nested value type's cause hang on mutex when _remove_ref delete's object
CR105493	When application password is not provided, the connection pool cannot be created.
CR105646	Cannot install Tux8.0 RP171 after performing a silent install
CR105796	/WS tpinit() fails when app. password is NULL (zero length passwd)
CR106043	Performance data alignment problem on Windows 2000.
CR106175	Fldid32 intermittently failed in a multi-threaded program with Fnmid_unload32.
CR106495	Extended original fix to Fnext32 and Flast32
CR106692	TUX6.5: Environment variable to set maximum transaction commit retry interval
CR106714	TUX6.5: Environment variable to treat xa_commit() XAER_RMERR as XA_RETRY
CR106986	Need silent install for patches.

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR107062	IDL C++ compiler aborts
CR107191	TUX6.5: Report TPEHEURISTIC/TPEHAZARD with event and ULOG message
CR107307	Fappend32 of ptr field cores when using buffer returned by Fget32
CR107377	GWTDOMAIN memory leak when using MAXPENDINGBYTES and large buffers.
CR107467	Jolt truncates wchar string
CR107718	dmunloadcf/tmboot does not support two different types of domains.
CR107767	tmshutdown in infinite loop when server queue removed prematurely
CR108088	Tux8.1GA>installer does not honor the USER_LOCALE setting with silence mode installation
CR108093	MBSTRING processing may go in an infinite loop
CR108475	tmshutdown can display wrong server Group and Id in Warning message
CR108761	Corrected code to return embedded pointers correctly from Fnext32
CR108788	Workstation client crashes on retry after WSH timeout when WSTYPE same as TYPE.
CR109039	Allow 256 byte subcommand arguments in tmadmin and dmadmin
CR109079	Reusing MBSTRING buffer, after conversion, with API arg sendlen==0 will fail
CR109327	`epifreg` cores on HP-UX B.11
CR109496	Handler established by Usignal() not called if TM_SVCTIMEOUT_SIGTERM selected
CR109795	Gateway network connection failover does not work if local domain is on the same host as primary remote domain
CR110337	Tuxedo 6.5: TMGDECIDED GTT entry deleted if commit timeout processed while TMS for all local groups is shutdown
CR110790	/Q deadlock under heavy load with blocking/TPQWAIT tpdequeues.
CR111157	Domain gateway socket connection leak
CR111240	Problems with IDL union types
CR111440	Temp files created during VTOC file is world writable

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR111458	WLS 8.1/Tux 8.1: Need more specific error message for WLS Jolt Connection Pool timeout
CR111571	Unsolicited messages fail in Tuxedo client under high load
CR111731	Security vulnerability in web-gui access.
CR111788	IIOP connectivity problems between Tuxedo 8.1 and WLS8.1
CR111825	Jolt repository editor gives error of "Fnext32() gets FIRSTFLDID"
CR111854	RP install/uninstall on Unix can leave GA-installed base files in altered state
CR112490	Cannot commit transaction if tpforward follows tpcall in AUTOTRAN service.
CR112599	CORBA C++ can't typesafe extract valuetype from CORBA::Any
CR112653	64 bit snmp_integrators dumps core on HP-UX 11.11 (64 bit)
CR112920	64 bit snmp trap daemon (snmptrapd) not generating traps on Solaris 2.8
CR112942	Jolt service calls catch timeouts prematurely.
CR113588	Multithreaded /WS client fails in Fldid when WSENVFILE is set.
CR120430	tpcommit fails with CMDTUX_CAT:428 and bogus group
CR120678	tmshutdown option to hold BB lock before dispatching SIGKILL
CR120784	snmp build warns
CR120784	snmp build warns on Linux
CR120879	Tux 8.1: Japanese message catalogs missing on HP-UX
CR120879	Japanese message catalogs missing on HP-UX (revisited)
CR120997	tpforward in transaction makes receiving server fail with LIBTUX 6314
CR121032	valuetype continuation chunks not handled correctly
CR121278	/WS tpinit fails with LIBTUX_CAT:6031
CR121651	valuetype related memory leaks (boxed valuetypes, >200 OBV's in same message, sequence of OBV's)
CR121770	Domain: memory allocation error LIBGW_CAT:1090

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR121981	Fboolco(32) broken by rolling patch
CR122410	Tuxedo 8.1: Silent install on Unix crashes
CR122591	Link errors when using macro UDEFERSIGS on HP-UX
CR122707	dmloadcf fails with CMDGW_CAT:3730 for valid domain configuration.
CR122740	'cnsls remote' hang without response when reply message has LOCATION_FORWARD from the remote IOR (Orbix)
CR122759	Tuxedo 7.1 RP272 (or GA) - tlisten leak memory for every tmboot/tmshutdown(MP) cycle
CR123601	Some CORBA samples of Tuxedo 8.1 won't work on Linux
CR123967	Tuxedo 7.1, 8.0, 8.1 with latest RP> Tuxedo un-installation did not properly un-install all files and registry settings on Windows platform
CR124060	Another memory leak after CR121651
CR124661	MP load balancing does not work as documented with service caching enabled
CR124718	Autotran service does not try xa_open after XAER_RMFAIL is returned in previous invocation.
CR125016	Some Tuxedo processes could crash during shutdown on Windows platform.
CR125435	Corba Exception due to marshalling failures between Tuxedo and WLS
CR125651	CORBA ISH crashes during shutdown when authentication is used.
CR125671	xa_recover not retried when more than 100 transactions to be returned
CR125715	CMDGW_CAT:3629 when accessing T_DM_TRANSACTION MIB via ud32
CR126070	RP install rollback does not remove files that were not in base installation
CR126276	CORBA BAD_PARAM exceptions (Tuxedo msg size > GIOP size)
CR126436	Tuxedo 8.1: cannot unmarshal Object references from exceptions (OBV Patches)
CR126828	IIOP client ORB's lookup_value_factory() implementation is too sensitive to user errors that it will always result in hard crash instead of throwing catchable exceptions.
CR127362	GWTDOMAIN either crashes or consumes CPU under portscan attach.

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR127531	The Tuxedo 8.1 installers need additional links to recognize the Japanese SJIS character set
CR127531	The Tuxedo 8.1 installers need additional links to recognize the Japanese SJIS character set
CR127604	Tux8.1rp64 > RETRY_INTERVAL in *TDOMAIN section won't work unless non-zero MAXRETRY is defined.
CR127654	DBBL prints misleading duplicate LIBTUX_CAT:262 ULOG message during DBBL restart (or MP master migration)
CR127875	Tuxedo 8.1: Duplicate symbol names AIT_CFB, AIT_CBC, AIT_ECB in libengine.a
CR128115	LIBGWT_CAT:1008 - to obtain a meaningful error code
CR128423	Server core dumps when making a tpacall through BRIDGE under load
CR128729	Tuxedo 8.1: MP interoperability with Tuxedo 6.4 - BUFTYPE field feature not functioning
CR128777	Load balancing can degrade if internal wkqueued counter overflows to negative value
CR129226	Tuxedo81: idl compiler wrong scoping
CR129247	/WS client tpinit failed with wrong TPESYSTEM (should be TPENOENT) and LIBWSC_CAT:2004 when the MAXWSCLIENT is exceeded.
CR129884	tmloadcf error: CMDTUX_CAT:6321: ERROR: AUTOTRAN specified in an application with the NO_XA option.
CR130149	Tuxedo 6.5: Tuxedo web server (tuxwsvr) dies when a user closes the web page while the page is loading
CR130192	Tuxedo 8.1: Make Tuxedo SHM segments size configurable via environment variable TM_ENGINE_TMSHMSEGSZ
CR130470	TxRPC poor performance for large arrays.
CR131609	Add classes to support security with JDK 1.4 ORB
CR132250	CORBA ORB cannot allocate >64k blocks of memory (can't unmarshal lots of valuetypes)
CR132351	CORBA IIOP client will hang forever if it exceeds the WLS IIOP idle connection timeout
CR132544	Jolt Session Pool: add keepalive property
CR132546	tpbroadcast fails and LIBTUX_CAT:2020 is logged

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR132695	Jolt1.2 Printing service name for TPENOENT.
CR133309	Client timeout before partitioning a broken connection in MP mode.
CR133667	/WS conversational client gets TPEPROTO in tprecv
CR133966	Tuxedo 8.1 RP65> Tuxedo 8.1 RP installation will get license error if the Tuxedo 8.1 GA installation's USER_INSTALL_DIR/TUXDIR path has space
CR134160	Abnormal disconnection errors reported by BRIDGE
CR134200	Intermittent Q_CAT:1449: ERROR: Create queue - no more queue entries (max = 400)
CR134321	Cross-domain transactions generate LIBGWT_CAT:1202 messages
CR135060	Tuxedo 8.1 RP#34 and up - GWTDOMAIN core dumps in TRU64
CR135942	Failure with FML buffers across 64 bit machines in MP mode Tuxedo 8 Solaris and Alpha
CR136063	HP: cannot use XML DDR with 1 byte attribute name.
CR136188	Tuxedo 8.0RP259: /Domain "-t" interoperability option creates memory leak
CR136652	TMQFORWARD becomes very CPU intensive if destination service is unavailable
CR155069	Tuxedo 8.1 64-bit - CORBA C++ client core dumps when invoking on an EJB
CR161918	sigaction() with SA_SIGINFO causes app-defined signal handler to fail
CR168248	Memory leak in Fprint32 and Ffprint32 on Solaris 8
CR172163	Request to block INFO messages - "Reached 90% of TUXEDO System Binary Licensed User Count "
CR172369	Tuxedo 8.0: Queue space becomes insane if shutdown TMQFORWARD while in transaction
CR174368	Windows 2000 RP installer was hard-coded to need c:\temp in order to complete the ObjectBridge license manager installation
CR174406	urcode in TPQCTL cannot be set/retrieved correctly on platforms with 64-bit wide long data type
CR175777	Tuxedo 8.0/WS: LIBWSC_CAT:1512: ERROR: Unable to generate first diffie-hellman packet

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR175815	Uninitialized variable leads to ISH not processing any more requests
CR176229	"bea.jolt.SessionException: Cannot connect" thrown after a few iterations
CR176747	Tuxedo 8.0: numerous T_SERVER requests locking the BB for too long a time
CR177210	Tuxedo: solution for DoS attacks due to very large messages
CR177529	Appkey not transferred across 64 bit MP domains when using global acl policies
CR178188	Tuxedo 8.0 RP 262: qlist displays inaccurate information if ipcrm is not run.
CR178369	tmshutdown "-w <delay>" leads to LIBTUX 4055 for service TMS
CR179780	GWTDOMAIN interoperability option for issues related to CR127362 and CR134321.
CR179797	Tuxedo 7.1RP295> can't delete DM_REMOTE_SERVICES using DM_MIB dynamically
CR180212	GWTDOMAIN cored with SEGV under heavy load
CR180306	CORBA IIOP client: client randomly deadlocks/crash on AIX
CR181347	Tuxedo 8.1RP98>On Windows 2000 sp4 platform, Tuxedo Admin Tool cannot correctly create/start/stop new TListener processes
CR182332	Request timeout with domain DDR
CR183090	GWTDOMAIN interoperability problem with Tuxedo 6.4 server in MP mode (core/LIBTUX_CAT:6031/6185)
CR183332	Tuxedo 8.1: propagate qspace insane and qmadmin hang CRs
CR183518	Tuxedo 8.1 > Windows 2003 - RP 82 installer tries to copy the beadmin81.cpl file from %tuxdir%\bin to c:\winnt\system32
CR184051	/WSC client hangs on a tpacall with TPNOREPLY
CR185088	Call RNAME of exported service won't work on 64-bit platforms
CR185118	tpcall under transaction failed with TPESYSTEM across domain
CR185451	Tuxedo 8.1:RP107 installer does not chmod 555 $TUXDIR/udataobj/webgui/cgi-bin/tuxadm
CR185535	Tuxedo 8.0rp280>Unexpected interactions among SYSTEM_ACCESS, MAXOBJECTS, TMSYSEVT and ATMI simpserv server causing endless ULOG LIBTUX_CAT:480 errors.

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR185632	tmboot hang on GWTDOMAIN
CR185970	Tuxedo 8.1 RP89> On Windows 2000 fails to shutdown server when service reaches SVCTIMEOUT
CR186408	Compiler error while using the Macros UDEFERSIGS() and URESUMESIGS().
CR186408	Tuxedo 8.1 RP#107/AIX 5.1 - Compiler error while using the Macros UDEFERSIGS() and URESUMESIGS().
CR187751	Tuxedo 8.1: programs crash when owner is root and setuid bit is on
CR187887	Visual Age v6.04 compiler has changed its default name-space mangling schematics.
CR188300	Tuxedo 8.1: Problem with running Tuxedo in different locale
CR188485	Tuxedo 8.1 - ISH processes crash in a large, busy CORBA C++ application
CR188565	Tuxedo 8.0/Solaris 8: forbidden service in remote /Domain -> TPESYSTEM -> strange returned buffer
CR188788	JSL/JSH: Jolt session idle timeout
CR189122	Tuxedo 8.1 - dmunloadcf does not provide for quotation marks in DM_IMPORT section
CR189138	Tuxedo 8.1 - Interface timeout applied to Server::release()
CR189641	Tuxedo 8.0: msglen of /Q message increases when message is moved
CR190243	Cannot boot GWSNAX
CR190258	Tuxedo 6.5 /AIX 5.1 - FML: customer want to keep the evaluation behavior of fboolev as before RP424
CR190469	Tuxedo 8.1: buffer pre-process errors with VIEWS
CR190469	Tuxedo 8.1: buffer pre-process errors with VIEWS
CR190684	catopen return 0 for non-existing message catalogue on Linux
CR191177	License: Change GP_CAT:1561:ERROR to GP_CAT:1591:INFO
CR191322	Tuxedo 8.1: AIX 5.1 TMQUEUE processes disappear if another slave machine is pclean'd
CR191356	GWTDOMAIN problem reporting action table reached threshold.

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR191401	Tuxedo 8.0 RP287/292 - PROTECTED mode: LIBTUX_CAT:1199: ERROR: Cannot re-attach
CR191585	When stopping gateway group (GWSNAX and GWADM) sometimes shutdown has to be forced out.
CR191809	Add environment variable to enable sleep during spin-lock to get ticket.
CR192992	Tuxedo 8.0/WS/Windows: TPU_SIG or TPU_THREAD might cause hanging process
CR193068	Tuxedo 8.1 - /Q Raw Device creation fails on Linux
CR193706	Tuxedo 7.1 - WSH detects bad message from network, but allocates memory for message anyway.
CR193963	Tuxedo 7.1 : Significant delay on remote tpcall
CR194482	Tuxedo 8.1 - dmadmin core dump with Tuxedo 8.1 RP117
CR195584	Tuxedo 71RP317>Under load, WSH crashes unexpectedly with multi-threaded /WS clients abruptly terminated
CR195786	Tuxedo 8.0: GWTDOMAIN leaks memory when transactions are in flight and there is a disconnection
CR196624	Tuxedo 8.1RP130> multi-threaded Tux server leak memory when running in PROTECTED mode under load
CR197387	Tuxedo 8.1 - Chained CORBA C++ invokes cause a load balancing problem
CR197610	Tuxedo 8.0RP210 (or above) regression>unwanted TUXCONFIG reading was introduced causing 2-phase XA transactions performance in large/heavily-loaded Tux system
CR201058	Tuxedo 7.1 - DMADM memory leak for MIB request
CR201643	Tuxedo 8.1/Windows: sbbl.exe dies with Message Box
CR202165	Tuxedo 7.1: RegisterClass failure code is not reported correctly
CR202752	Tuxedo 8.0: GWTDOMAIN crash under load
CR203621	Tuxedo 7.1 - tcm_passthrough is set by default, causing unknown TCMs to be processed
CR204199	_tmcleanup being called twice for domain gateways

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR205632	Tuxedo 8.1 RP141 - Tuxedo server hangs forever when a transaction timeout occurs shortly around a new transaction branch is started
CR205869	Tuxedo 8.1: Simultaneous connections lead to time outs
CR208162	Tuxedo 8.0RP316>interoperability problem that FML32 buffer requests from Tuxedo 6.5 node goes across Bridge to Tuxedo 8.0 node will cause Tuxedo 8.0 ATMI server memory leak
CR208465	Jolt 8.0 - JSH reports unsupported Jolt opcode 25440
CR208945	Tuxedo 8.1 boolean function error - wrong result returned
CR209163	Tuxedo 7.1 - GWTDOMAIN crash
CR209335	`dmloadcf` core dumps while adding two `DOMAINID` to `DM_LOCAL_DOMAINS`
CR209929	Tuxedo 8.1 BBL dumps core while scanning GTT. (GTT full condition)
CR210050	Tuxedo 7.1 - Gateway attack causes crash
CR210185	Performance degradation in 'tprealloc()' when upgrading from Tuxedo 6.5 to Tuxedo 8.1.
CR210382	UNIX Tuxedo tuning issues regarding tmloadcf -c
CR210469	Fix large memory size increase for multithreaded servers in Tuxedo 8.1
CR210510	Tuxedo 8.1 - WTC cannot get response when VIEW type does not match/exist
CR210961	Tuxedo 8.1RP158> simultaneous tmadmin execution against MP configuration causes IPC queue resource leak
CR211024	Tuxedo 8.1 tmshutdown/restartsrv/cleanupsrv are stuck in BB.
CR211030	Support for sanity scans of <5 seconds.
CR211318	Tuxedo 8.1: non-copying form of insertion into Any declared but not defined in idl output
CR211936	Service call failed with `BRTHREAD=Y`
CR212187	presend the embedded FML32 buffer failed while `TPMBACONV=MBAUTOCONVERSION_ON` is set.
CR212219	Tuxedo 6.5: inconsistent behavior with TPNOTRAN in ROLLBACKONLY state
CR212437	Tuxedo 8.1 - no SDK license is needed when using "buildserver -r"

Table A-1 BEA Tuxedo Problems Fixed in This Release

CR212441	Tuxedo 7.1: Service becomes unadvertised when DM_IMPORT section deleted
CR212455	Tuxedo 8.1 - dmloadcf causing LIBGWT_CAT:1029 error
CR212777	Tuxedo 8.0: no response after TPFAIL
CR212966	Tuxedo 8.0 - /WS client returns incorrect xid.formatID
CR214212	Tuxedo 8.1 WSNAT_CAT:1043: ERROR: tpacall() call failed, tperrno = 0
CR214647	Tuxedo 8.1 : LIBTUX_CAT:4027 error when MAXINTERFACES is 0
CR216047	Interop with Tuxedo 6.5: failed to call Tuxedo 9.0 service from Tuxedo 6.5 client with view buffer type
CR216521	Tuxedo 8.1 RP152. Boolean issue - problem using FML_EVAL_ASBEFORE on AIX/HPUX
CR217617	Interoperability with Tuxedo 6.5: unsolicited message cannot be encode/decode between Tuxedo 9.0 and Tuxedo 6.5.
CR217632	When the `cnsbind` utility prompts for a password, it is echoed back to the user in clear text
CR218774	Tuxedo 8.1 - Q_CAT:1352: ERROR: encountered after adding extent to queue space
CR219914	Tuxedo 7.1 - GWADM crash in GMT+0 time zone
CR220189	Workstation version `tpconnect` with sendonly flat to WSH running a release prior to Tuxedo 9.0 will hang.
CR220575	Tuxedo 8.1 - Factory Based Routing does not work as expected in MP mode
CR220804	XML buffer cannot convert successfully in MP mode when master is Tuxedo 9.0 and another is Tuxedo 8.1, Tuxedo 8.0 or Tuxedo 7.1.
CR221596	Tuxedo 8.1: JRAD crash
CR222209	Tuxedo 8.1: load password entry from DMADM lead to memory access violation