

Oracle® Tuxedo

Installing the Oracle Tuxedo System

12c Release 1 (12.1.1)

June 2012

ORACLE®

Installing the Oracle Tuxedo System, 12c Release 1 (12.1.1)

Copyright © 1996, 2012, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

1. Preparing to Install the Oracle Tuxedo System

Oracle Tuxedo Web Distribution	1-1
Oracle Tuxedo Software Components	1-2
Tuxedo .NET Workstation Client	1-4
LLE and SSL Levels of Encryption	1-4
Hardware and Software Prerequisites	1-4
System Requirements.	1-5
Temporary Storage Space Requirements	1-5
Interprocess Communication Resources Configuration	1-6
Oracle Installation Program	1-7
Previous Tuxedo Versions	1-7
Installation Methods.	1-8
Cancelling Installation	1-8
Install Sets	1-9
Oracle Home Directory	1-10
Choosing a Oracle Home Directory.	1-11
Understanding the Functions of the Oracle Home Directory	1-11
Creating More Than One Oracle Home Directory.	1-12
Installation Road Map	1-12

2. Installing Oracle Tuxedo Using GUI-Mode Installation

What Is GUI-Mode Installation?	2-2
--------------------------------------	-----

Before You Start.	2-2
Starting GUI-Mode Installation on a Windows System.	2-2
Starting GUI-Mode Installation on a UNIX System	2-3
Running GUI-Mode Installation	2-4
Assigning File Ownership on a UNIX System	2-10
Understanding the Oracle Tuxedo Shortcuts on a Windows System.	2-11
Understanding the Oracle Administration Program on a Windows System	2-12
Reviewing the Windows Server Registry Content.	2-13
Developer Key	2-13
Environment Key	2-13
Security Key	2-13
What Do I Do Next?.	2-14

3. Installing Oracle Tuxedo on UNIX Systems Using Console-Mode Installation

What Is Console-Mode Installation?	3-2
Before You Start.	3-2
Starting Console-Mode Installation	3-2
Running Console-Mode Installation	3-3
What Do I Do Next?.	3-10

4. Installing Oracle Tuxedo Using Silent Installation

What Is Silent Installation?	4-2
Before You Start.	4-2
Using Silent Installation: Main Steps.	4-2
Creating a Template File	4-3
Invoking the Silent Installation Process on a Windows System	4-6
Invoking the Silent Installation Process on a UNIX System	4-7

Windows Template File	4-9
UNIX Template File	4-11
What Do I Do Next?	4-12

5. Upgrading the Oracle Tuxedo System to 12c Release 1 (12.1.1)

Preparing Your Machine for an Upgrade	5-1
Selecting an Upgrade Procedure	5-2
Backing Up Files	5-3
Performing a Simple Upgrade	5-3
Performing Additional Upgrade Steps to Windows Server	5-4
Upgrading from Tuxedo 6.5 to Tuxedo 12c Release 1 (12.1.1)	5-5
Performing a Hot Upgrade	5-6
Pre-Upgrade Notes	5-6
Hot-Upgrade Procedure	5-7
Hot-Upgrade Example	5-8
Rebuilding an Application	5-9
Installing Oracle Jolt 12c Release 1 (12.1.1) with Oracle WebLogic Server	5-10

6. Performing Post-Installation Tasks

Understanding the Oracle Tuxedo Directory Structure	6-2
Understanding the Oracle Tuxedo Architecture	6-6
UBBCONFIG File	6-7
MASTER Machine	6-7
TUXCONFIG File	6-7
TUXCONFIG Environment Variable	6-8
TUXDIR Environment Variable	6-8
Using the Registry to Further Restrict Access on a Windows System	6-8

Setting Up Your Environment	6-9
Setting Environment Variables on a Windows System.	6-11
Setting Environment Variables on a UNIX System	6-12
Starting the tlisten Process	6-13
tlisten Invocation.	6-14
tlisten Password.	6-14
Editing a UBBCONFIG File	6-15
Using the TYPE Parameter in UBBCONFIG	6-18
Checking IPC Requirements	6-18
Creating the Universal Device List and the Transaction Log	6-19
Defining the TLOG.	6-20
Creating the UDL and UDL Entries for TLOG	6-20
Running simpapp to Verify Your Installation	6-21
Running simpapp to Verify the Oracle Tuxedo ATMI Software Installation	6-22
Running simpapp to Verify the Oracle Tuxedo ATMI Software Installation on a Windows System	6-22
Running simpapp to Verify the Oracle Tuxedo ATMI Software Installation on a UNIX System	6-25
Running simpapp to Verify the Oracle Tuxedo CORBA C++ Software Installation . .	6-27
Running simpapp to Verify the Oracle Tuxedo CORBA C++ Software Installation on a Windows System.	6-28
Running simpapp to Verify the Oracle Tuxedo CORBA C++ Software Installation on a UNIX System	6-29
Running buildtms for Oracle Tuxedo Applications That Use XA Resource Managers	6-31
Uninstalling Oracle Tuxedo	6-31
Reinstalling Oracle Tuxedo	6-32

A. Oracle Tuxedo 12c Release 1 (12.1.1) Platform Data Sheets

Supported Platforms	A-2
Platform Support Policy	A-2
Platform End of Life Expectancy Table	A-3
Tunable Parameters.	A-4
Supported Platform Data Sheets	A-4
IBM AIX 6.1 (64-bit) on IBM PowerPC.	A-4
Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for IBM AIX 6.1 (64-bit).	A-5
Hardware Requirements for IBM AIX 6.1 (64-bit)	A-5
Software Requirements for IBM AIX 6.1 (64-bit).	A-5
Network Requirements for IBM AIX 6.1 (64-bit)	A-7
Disk Space Requirements for IBM AIX 6.1 (64-bit).	A-7
Mounting and Unmounting the CD for AIX 6.1 (64-bit).	A-8
Tunable Parameters for IBM AIX 6.1 (64-bit).	A-8
IBM AIX 7.1 (64-bit) on IBM PowerPC.	A-8
Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for IBM AIX 7.1 (64-bit).	A-9
Hardware Requirements for IBM AIX 7.1 (64-bit)	A-9
Software Requirements for IBM AIX 7.1 (64-bit).	A-9
Network Requirements for IBM AIX 7.1 (64-bit)	A-11
Disk Space Requirements for IBM AIX 7.1 (64-bit).	A-11
Mounting and Unmounting the CD for AIX 7.1 (64-bit).	A-11
Tunable Parameters for IBM AIX 7.1 (64-bit).	A-12
Novell SUSE Linux Enterprise Server 11 (64-bit) on x86-64.	A-12
Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Novell SUSE Linux Enterprise Server 11 (64-bit)	A-12

Hardware Requirements for Novell SUSE Linux Enterprise Server 11 (64-bit) . .	A-13
Software Requirements for Novell SUSE Linux Enterprise Server 11 (64-bit) . .	A-13
Network Requirements for Novell SUSE Linux Enterprise Server 11 (64-bit) . .	A-14
Disk Space Requirements for Novell SUSE Linux Enterprise Server 11 (64-bit)	A-14
Mounting and Unmounting the CD for Novell SUSE Linux Enterprise Server 11 (64-bit)	A-14
Tunable Parameters for Novell SUSE Linux Enterprise Server 11 (64-bit) .	A-14
Oracle Enterprise Linux 5.6 (64-bit) on x86-64	A-15
Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Enterprise Linux 5.6 (64-bit)	A-15
Hardware Requirements for Oracle Enterprise Linux 5.6 (64-bit)	A-15
Software Requirements for Oracle Enterprise Linux 5.6 (64-bit)	A-15
Network Requirements for Oracle Enterprise Linux 5.6 (64-bit)	A-17
Disk Space Requirements for Oracle Enterprise Linux 5.6 (64-bit)	A-17
Mounting and Unmounting the CD for Oracle Enterprise Linux 5.6 (64-bit)	A-17
Tunable Parameters for Oracle Enterprise Linux 5.6 (64-bit)	A-17
Oracle Linux 5.6 (64-bit) on Exalogic 2.0	A-17
Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Linux 5.6 (64-bit)	A-17
Hardware Requirements for Oracle Linux 5.6 (64-bit)	A-17
Software Requirements for Oracle Linux 5.6 (64-bit)	A-18
Network Requirements for Oracle Linux 5.6 (64-bit)	A-19
Disk Space Requirements for Oracle Linux 5.6 (64-bit)	A-19
Mounting and Unmounting the CD for Oracle Linux 5.6 (64-bit)	A-19

Tunable Parameters for Oracle Linux 5.6 (64-bit)	A-19
Oracle Solaris 10 (64-bit) on x86-64	A-19
Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Solaris 10 (64-bit) on x86-64	A-19
Hardware Requirements for Oracle Solaris10 (64-bit) on x86-64	A-20
Software Requirements for Oracle Solaris10 (64-bit)	A-20
Network Requirements for Oracle Solaris10 (64-bit) on x86-64	A-22
Disk Space Requirements for Oracle Solaris10 (64-bit) on x86-64	A-22
Mounting and Unmounting the CD for Oracle Solaris10 (64-bit) on x86-64	A-22
Tunable Parameters for Oracle Solaris10 (64-bit) on x86-64	A-22
Oracle Solaris 10 (64-bit) on SPARC	A-22
Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Solaris 10 (64-bit)	A-22
Hardware Requirements for Oracle Solaris 10 (64-bit) on SPARC	A-23
Software Requirements for Oracle Solaris 10 (64-bit) on SPARC	A-23
Network Requirements for Oracle Solaris 10 (64-bit) on SPARC	A-25
Disk Space Requirements for Oracle Solaris 10 (64-bit) on SPARC	A-25
Mounting and Unmounting the CD for Oracle Solaris 10 (64-bit) on SPARC	A-25
Tunable Parameters for Oracle Solaris 10 (64-bit) on SPARC	A-25
Oracle Solaris 11 (64-bit) on x86-64	A-25
Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Solaris 11 (64-bit) on x86-64	A-25
Hardware Requirements for Oracle Solaris11 (64-bit) on x86-64	A-26
Software Requirements for Oracle Solaris11 (64-bit)	A-26
Network Requirements for Oracle Solaris11 (64-bit) on x86-64	A-28
Disk Space Requirements for Oracle Solaris11 (64-bit) on x86-64	A-28
Mounting and Unmounting the CD for Oracle Solaris11 (64-bit) on x86-64	A-28
Tunable Parameters for Oracle Solaris11 (64-bit) on x86-64	A-28

Oracle Solaris 11 (64-bit) on SPARC	A-28
Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Solaris 11 (64-bit)	A-28
Hardware Requirements for Oracle Solaris 11 (64-bit) on SPARC.	A-29
Software Requirements for Oracle Solaris 11 (64-bit) on SPARC	A-29
Network Requirements for Oracle Solaris 11 (64-bit) on SPARC.	A-30
Disk Space Requirements for Oracle Solaris 11 (64-bit) on SPARC	A-31
Mounting and Unmounting the CD for Oracle Solaris 11 (64-bit) on SPARC.	A-31
Tunable Parameters for Oracle Solaris 11 (64-bit) on SPARC	A-31
Red Hat Enterprise Linux 5.6 (64-bit) on x86-64.	A-31
Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Red Hat Enterprise Linux 5.6 (64-bit)	A-31
Hardware Requirements for Red Hat Enterprise Linux 5.6 (64-bit)	A-32
Software Requirements for Red Hat Enterprise Linux 5.6 (64-bit).	A-32
Network Requirements for Red Hat Enterprise Linux 5.6 (64-bit).	A-33
Disk Space Requirements for Red Hat Enterprise Linux 5.6 (64-bit).	A-33
Mounting and Unmounting the CD for Red Hat Enterprise Linux 5.6 (64-bit) . . A-33	
Tunable Parameters for Red Hat Enterprise Linux 5.6 (64-bit).	A-33
Platforms Supporting Threads	A-33

1. Starting the Oracle Tuxedo Administration Console

What is the Oracle Tuxedo Administration Console?	B-1
Administration Console File Tree	B-2
Server Requirements	B-3
Browser Requirements	B-4
Hardware Requirements.	B-5
Setting Up Your Environment for the Oracle Tuxedo Administration Console	B-6

Starting tuxwsvr.	B-6
Starting wlisten	B-7
Starting the Oracle Tuxedo Administration Console	B-8
Limitation(s)	B-9
Exiting the Oracle Tuxedo Administration Console.	B-9

B. Oracle Jolt 12*c* Release 1 (12.1.1) Overview and Installation Information

About Oracle Jolt	C-1
Oracle Jolt Components	C-3
Jolt Server Listener	C-4
Jolt Server Handler	C-5
Jolt Repository Server	C-5
Jolt Repository	C-5
Jolt Internet Relay	C-5
Jolt Class Library.	C-6
JoltBeans	C-8
Oracle Jolt Client Personalities.	C-8
Oracle Jolt Capabilities.	C-9
Oracle Jolt Client Support.	C-10
Oracle Jolt Client Requirements	C-11
Oracle Jolt Client Class Library	C-11
Oracle Jolt Release Interoperability	C-12
Oracle Jolt Pre-Installation Checklist	C-12
Oracle Jolt Documentation	C-12

D. File and Database Management and Disk Space Allocation

Introduction.	D-1
-----------------------	-----

How the Oracle Tuxedo System Manages Files	D-1
Arranging for Raw Disk Space	D-2
How the Oracle Tuxedo Filesystem Is Organized	D-2
Space for Queue Spaces	D-4
Space for Application Servers	D-4

E. IPC Resource Configuration on a UNIX System

Parameter Sets Controlling IPC Resources	E-1
Shared Memory	E-2
Semaphores	E-3
Message Queues and Messages	E-4
What Is Application Deadlock?	E-5
Performance Implications of Blocking Conditions.....	E-5
Tunable Message Parameters	E-5
Other Kernel Tunables	E-7

Preparing to Install the Oracle Tuxedo System

The following sections provide information that you need to know before installing the Oracle Tuxedo 12c Release 1 (12.1.1) product software:

- [Oracle Tuxedo Web Distribution](#)
- [Oracle Tuxedo Software Components](#)
- [LLE and SSL Levels of Encryption](#)
- [Hardware and Software Prerequisites](#)
- [Interprocess Communication Resources Configuration](#)
- [Oracle Installation Program](#)
- [Oracle Home Directory](#)
- [Installation Road Map](#)

Oracle Tuxedo Web Distribution

An evaluation copy of Oracle Tuxedo is available for download from the Oracle corporate Web site at <http://www.oracle.com/technology/software/index.html>.

Platform-specific installer files for the Oracle Tuxedo product software are available for download from the Oracle corporate Web site.

Oracle Tuxedo Software Components

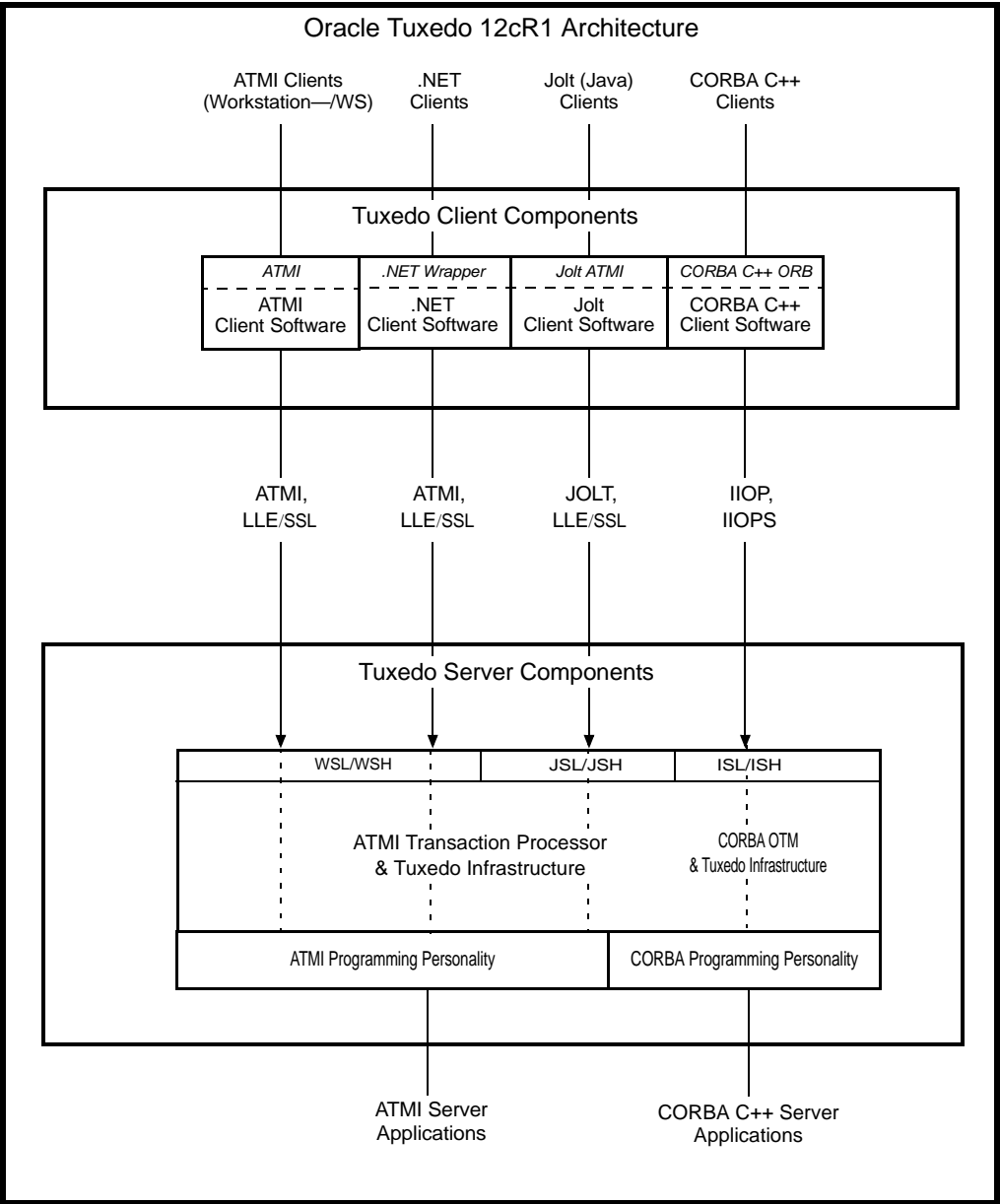
Included with the Oracle Tuxedo 12c Release 1 (12.1.1) software distribution is the Oracle Jolt product software, which enables Java clients to call Oracle Tuxedo services, and the Oracle SNMP Agent product software, which enables Oracle Tuxedo applications to be managed from a centralized enterprise management console.

The Oracle Tuxedo 12c Release 1 (12.1.1) distribution contains the following components:

- Server components:
 - Oracle Application-to-Transaction Monitor Interface (ATMI) server software
 - Oracle Common Object Request Broker Architecture (CORBA) C++ server software
 - Oracle Jolt 12c Release 1 (12.1.1) server software
 - Oracle SNMP Agent 12c Release 1 (12.1.1) software
 - Oracle Tuxedo Administration Console software
 - Link-Level Encryption (LLE) and Secure Sockets Layer (SSL) encryption software
- Client components:
 - Oracle ATMI Workstation (/WS) client software
 - Oracle CORBA C++ client software—includes the C++ client Object Request Broker (ORB) and the environmental objects
 - Oracle Jolt 12c Release 1 (12.1.1) client software
 - Tuxedo.NET Workstation client software
 - LLE and SSL encryption software

[Figure 1-1](#) illustrates a high-level view of the Oracle Tuxedo client and server components.

Figure 1-1 Oracle Tuxedo Client and Server Components



For descriptions of terms such as *remote client*, *native client*, *LLE*, and *SSL*, see “Client and Server Components” in [Oracle Tuxedo Product Overview](#).

Tuxedo .NET Workstation Client

The Tuxedo .NET Workstation Client requires:

- the ATMI Client to be automatically installed when you select the .NET Client install set.
- Microsoft’s .NET Framework 3.5 installed on your system. If .NET Framework is not detected, you can continue the .NET Client installation and install .NET Framework later. For more information on the Tuxedo .NET Workstation Client, see [Creating Tuxedo .NET Workstation Client Applications](#).

LLE and SSL Levels of Encryption

LLE and SSL are available with two levels of encryption: 56-bit and 128/256-bit.

Hardware and Software Prerequisites

The Oracle Tuxedo software must be installed on each server machine that will participate in a Oracle Tuxedo application, also known as a Tuxedo domain. A Tuxedo application is a business software program, built upon the Tuxedo system, that is defined and controlled by a single configuration file known as the `UBBCONFIG` file. The Tuxedo configuration file is described in reference page [UBBCONFIG\(5\)](#) in *Oracle Tuxedo File Formats, Data Descriptions, MIBs, and System Processes Reference*.

A Tuxedo application consists of many Tuxedo system processes, one or more application client processes, one or more application server processes, and one or more computer machines connected over a network. In a multi-machine Tuxedo application running different releases of the Oracle Tuxedo software, the *master machine*—designated via the `MASTER` parameter in the `RESOURCES` section of the `UBBCONFIG` file—must run the highest release of the Oracle Tuxedo software in the application. For more information about Tuxedo applications, see “Domains” in [Oracle Tuxedo Product Overview](#).

Note: Oracle advises against trying to share the Oracle Tuxedo system executables across remote filesystems; this practice has proven to be unreliable in the past.

System Requirements

The system requirements for Oracle Tuxedo 12c Release 1 (12.1.1) are given in [Table 1-1](#).

Table 1-1 The System Requirements for Oracle Tuxedo 12c Release 1 (12.1.1)

Component	Requirement
Platform*	Any platform identified in “Oracle Tuxedo 12c Release 1 (12.1.1) Platform Data Sheets” on page A-1.
Hard disk drive	As stated in the data sheet for the target platform in “Oracle Tuxedo 12c Release 1 (12.1.1) Platform Data Sheets” on page A-1. For additional information about disk space requirements, see “File and Database Management and Disk Space Allocation” on page D-1.
Memory	As stated in the data sheet for the target platform in “Oracle Tuxedo 12c Release 1 (12.1.1) Platform Data Sheets” on page A-1.
* Microsoft Windows XP systems support only the Oracle Tuxedo client components.	

For Oracle Jolt installation preparatory information, see [“Oracle Jolt 12c Release 1 \(12.1.1\) Overview and Installation Information”](#) on page B-1. No installation preparation is required for Oracle SNMP Agent.

Note: Before installing Oracle Tuxedo software on a UNIX system, Oracle recommends that you repartition your hard disk device in accordance to [“File and Database Management and Disk Space Allocation”](#) on page D-1.

Temporary Storage Space Requirements

The Oracle Installation program uses a temporary directory in which it extracts the files from the archive that are needed to install Oracle Tuxedo on the target system. During the installation process, your temporary directory must contain sufficient space to accommodate the compressed Java Runtime Environment (JRE) bundled with the installer and an uncompressed copy of the JRE that is expanded into the temporary directory. The installation program moves the JRE from the temporary directory to the *Oracle Home directory* at the end of the installation process. For information about the Oracle Home directory, see [“Oracle Home Directory”](#) on page 1-10.

The amount of temporary storage space needed depends upon the target platform, as stated in the data sheets in [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets”](#) on page A-1.

By default, the installation program uses the temporary directories shown in [Table 1-2](#).

Table 1-2 Temporary Directories

Platform	Directory
Windows	Directory referenced by the TMP system variable
UNIX	/tmp

To ensure there is adequate temporary space, you may want to allocate an alternate directory for use as a temporary directory for the installation. To do so, perform the appropriate step in [Table 1-3](#) before starting the Oracle Installation program.

Table 1-3 Perform This Step

On This Platform...	Perform This Step...
Windows	Set the TMP system variable to a directory of your choice.
UNIX	Enter the following command at the shell prompt: <code>export IATEMPDIR=tmpdirname</code> Replace <i>tmpdirname</i> with the name of a temporary directory of your choice.

Interprocess Communication Resources Configuration

Interprocess communication (IPC) is a capability supported by the Windows and UNIX operating systems that allows one process to communicate with another process. The processes can be running on the same computer or on different computers connected through a network.

On a Windows 2003 Server system, the Oracle Tuxedo system provides an IPC service called the Oracle ProcMGR (Process Manager), which facilitates interprocess communications. You use the Oracle ProcMGR service to adjust the IPC parameters and maximize the performance of a Oracle Tuxedo application.

On a UNIX system, you use methods native to the UNIX system to adjust the IPC parameters and maximize the performance of a Oracle Tuxedo application. Since most UNIX systems are shipped with default values that are too low for a Oracle Tuxedo application, you need to adjust the IPC parameters, using the methods given in [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform](#)

[Data Sheets](#)” on page A-1. For the recommended IPC values, see [“IPC Resource Configuration on a UNIX System”](#) on page E-1.

After installing the Oracle Tuxedo software and creating an application configuration file (UBBCONFIG file), you use the `tmloadcf(1)` command to calculate and print a list of the minimum IPC resources needed to support the application. If your Oracle Tuxedo application is distributed, the minimum IPC resources must be available on every machine participating in the application. For more information about calculating IPC resources using the `tmloadcf(1)` command, see [“Performing Post-Installation Tasks”](#) on page 6-1.

Note: Before installing Oracle Tuxedo software on a UNIX system, Oracle recommends that you adjust the IPC parameters on the target machine in accordance to [“IPC Resource Configuration on a UNIX System”](#) on page E-1.

Oracle Installation Program

The Oracle Tuxedo software is distributed as an installer file, which also contains a copy of the Oracle Installation program. The Oracle Installation program is the Oracle standard tool for installing the Oracle Tuxedo software on Windows or UNIX systems.

Note: On Windows platforms you must re-login to the system after Tuxedo 12c Release 1 (12.1.1) installation has completed.

Previous Tuxedo Versions

If a previous version of the Tuxedo software is detected during GUI mode and silent installation on a Windows system, the Tuxedo installer will automatically stop the following previous Tuxedo version ProcMGR and Listener services:

- Oracle ProcMGR V2.2
- Oracle ProcMGR V8.0
- Oracle ProcMGR V8.1
- Oracle ProcMGR V9.0
- Oracle ProcMGR V9.1
- Oracle ProcMGR V9.1with VS2005 v2
- Oracle ProcMGR V10.0 with VS2005
- Tuxedo 10.0 with VS2005 Listener on Port 3050

- Tuxedo 9.1 Listener on Port 3050
- Tuxedo 9.1with VS2005 Listener on Port 3050
- Tuxedo 9.0 Listener on Port 3050
- Tuxedo 8.1 Listener on Port 3050
- Tuxedo 8.0 Listener on Port 3050
- Tuxedo 7.1 Listener on Port 3050
- Tuxedo 6.5 Tuxedo IPC Helper

Note: During GUI installation, users will be prompted with the following message:
“Stopping Previous Version Tuxedo Services”

Listener services configured on ports other than 3050 must be stopped *manually*.

The current Tuxedo distribution requires these services to be stopped in order to run multiple Tuxedo versions on the same machine. The time required for stopping the ProcMGR services is minimal; however, the time required to stop the listener services may take a few minutes.

Installation Methods

The Oracle Installation program supports three installation methods. You can use any of these methods to install the Oracle Tuxedo product software:

- Graphical user interface (GUI) installation, described in [“Installing Oracle Tuxedo Using GUI-Mode Installation” on page 2-1.](#)
- Console Installation (UNIX systems only), described in [“Installing Oracle Tuxedo on UNIX Systems Using Console-Mode Installation” on page 3-1.](#)
- Silent installation, described in [“Installing Oracle Tuxedo Using Silent Installation” on page 4-1.](#)

Cancelling Installation

Clicking the “Cancel” or the “close” window button in GUI mode, or typing “quit” in console mode creates an incomplete Tuxedo 12c Release 1 (12.1.1) installation. You will have to re-install Tuxedo 12c Release 1 (12.1.1).

If you installed Tuxedo 12c Release 1 (12.1.1) in a previous Tuxedo product directory, and you want to return to your original configuration, you must re-install your previous Tuxedo version.

Install Sets

An install set is a bundle of product software components related by function. Oracle Tuxedo 12c Release 1 (12.1.1) offers the following seven install sets:

- *Full install set*—consists of all the Oracle Tuxedo 12c Release 1 (12.1.1) server and client components
- *Server install set*—consists of all the Oracle Tuxedo 12c Release 1 (12.1.1) server components:
 - Oracle Application-to-Transaction Monitor Interface (ATMI) server, native ATMI client, and ATMI Workstation (/WS) software
 - Oracle Common Object Request Broker Architecture (CORBA) C++ server
 - Oracle Jolt 12c Release 1 (12.1.1) server software including Oracle Jolt Internet Relay
 - Oracle SNMP Agent 12c Release 1 (12.1.1) software
 - Oracle Tuxedo Administration Console software
 - Link-Level Encryption (LLE) and Secure Sockets Layer (SSL) encryption software
- *Full client install set*—consists of all the Oracle Tuxedo 12c Release 1 (12.1.1) client components:
 - Oracle ATMI Workstation (/WS) client software
 - Oracle CORBA C++ client software (C++ client Object Request Broker, or ORB) including environmental objects
 - Oracle Jolt 12c Release 1 (12.1.1) client software
 - Oracle Tuxedo.NET Workstation client software
 - LLE and SSL encryption software
- *ATMI (/WS) client install set*—consists of the following Oracle Tuxedo 12c Release 1 (12.1.1) components:
 - Oracle ATMI (/WS) client software
 - LLE and SSL encryption software
- *CORBA client install set*—consists of the following Oracle Tuxedo 12c Release 1 (12.1.1) components:
 - Oracle CORBA C++ client software

- SSL encryption software
- *Jolt client install set*—consists of the following Oracle Tuxedo 12c Release 1 (12.1.1) components:
 - Oracle Jolt 12c Release 1 (12.1.1) client software
 - LLE and SSL encryption software
- *.NET client install set*—consists of the following Oracle Tuxedo 12c Release 1 (12.1.1) components and install sets: (Windows platforms only)
 - Tuxedo.NET Workstation client software
 - Oracle ATMI (/WS) client install set

In addition to selecting an install set during a Oracle Tuxedo 12c Release 1 (12.1.1) installation, a user can further customize the installation by selecting (adding) or deselecting (deleting) one or more software components from the install set. Customizing is only possible for GUI-mode and console-mode installation methods described in [“Installing Oracle Tuxedo Using GUI-Mode Installation” on page 2-1](#) and [“Installing Oracle Tuxedo on UNIX Systems Using Console-Mode Installation” on page 3-1](#); it is not possible for silent-mode installation method.

Oracle Home Directory

When you install Oracle Tuxedo, you are prompted to specify an Oracle Home directory. The Oracle Home directory is a repository for common files that are used by multiple Oracle products installed on the same machine. For this reason, the Oracle Home directory can be considered a “central support directory” for the Oracle products installed on your system.

The files in the Oracle Home directory are essential to ensuring that the Oracle software operates correctly on your system. These files perform the following types of functions:

- Facilitate checking of cross-product dependencies during installation
- Facilitate service pack installation

The structure of the Oracle Home directory created by the Oracle Tuxedo 12c Release 1 (12.1.1) installer program is shown in the following illustration.



Choosing a Oracle Home Directory

During the installation of Oracle Tuxedo 12c Release 1 (12.1.1), you are prompted to choose an existing Oracle Home directory or specify a path for a new Oracle Home directory. If you choose to create a new directory, the Oracle Tuxedo installer program automatically creates the directory for you.

Upon choosing a Oracle Home directory, you are prompted to choose a Oracle product directory for your Oracle Tuxedo installation. You may choose to create your Oracle product directory under the Oracle Home directory, but there is no requirement to do so.

Understanding the Functions of the Oracle Home Directory

The files and directories in an Oracle Home directory created by the Oracle Tuxedo installer are described in [Table 1-4](#).

Table 1-4 Description

Component	Description
registry.xml file	An XML registry file that contains a persistent record of all Oracle products installed on the target system for this Oracle Home directory. The registry contains product-related information, such as version level, service pack level, and product installation directory.
logs directory	A directory containing a Oracle Home location file and a history file of installation and uninstallation for this Oracle Home directory. For more information on these files, see the “Understanding the Oracle Tuxedo Shortcuts on a Windows System” on page 2-11.

Creating More Than One Oracle Home Directory

Although it is possible to create more than one Oracle Home directory, Oracle recommends that you avoid doing so. In almost all situations, a single Oracle Home directory is sufficient. There may be circumstances, however, in which you prefer to maintain separate development and production environments, each containing a separate product stack. With two directories, you can update your development environment (in an Oracle Home directory) without modifying the production environment until you are ready to do so.

Installation Road Map

You are now ready to begin your installation. To install Oracle Tuxedo 12c Release 1 (12.1.1), see one of the following sections:

- [“Installing Oracle Tuxedo Using GUI-Mode Installation” on page 2-1](#)
- [“Installing Oracle Tuxedo on UNIX Systems Using Console-Mode Installation” on page 3-1](#)
- [“Installing Oracle Tuxedo Using Silent Installation” on page 4-1](#)

If you are upgrading from Oracle WebLogic Enterprise or Oracle Tuxedo 9.1 or earlier software, see [“Upgrading the Oracle Tuxedo System to 12c Release 1 \(12.1.1\)” on page 5-1](#).

If you want to uninstall your Oracle Tuxedo software, see [“Performing Post-Installation Tasks” on page 6-1](#).

Installing Oracle Tuxedo Using GUI-Mode Installation

The following sections describe how to install Oracle Tuxedo using graphical user interface (GUI) mode installation on both Windows and UNIX systems:

- [What Is GUI-Mode Installation?](#)
- [Before You Start](#)
- [Starting GUI-Mode Installation on a Windows System](#)
- [Starting GUI-Mode Installation on a UNIX System](#)
- [Running GUI-Mode Installation](#)
- [Assigning File Ownership on a UNIX System](#)
- [Understanding the Oracle Tuxedo Shortcuts on a Windows System](#)
- [Understanding the Oracle Administration Program on a Windows System](#)
- [Reviewing the Windows Server Registry Content](#)
- [What Do I Do Next?](#)

What Is GUI-Mode Installation?

The graphical user interface mode installation is the graphics-based method of executing the Oracle Installation program. It can be run on both Windows and UNIX system.

To run GUI-mode installation, the console attached to the machine on which you are installing the software must support a Java-based GUI. All consoles for Windows systems support Java-based GUIs, but not all consoles for UNIX systems do.

Note: To install Oracle Tuxedo on a UNIX system with a non-graphics console, see [“Installing Oracle Tuxedo on UNIX Systems Using Console-Mode Installation”](#) on page 3-1.

Before You Start

If you are upgrading from Oracle WebLogic Enterprise or Oracle Tuxedo 9.1 or earlier software, see [“Upgrading the Oracle Tuxedo System to 12c Release 1 \(12.1.1\)”](#) on page 5-1 and follow the instructions given there. Then return here to continue your Oracle Tuxedo installation.

On a Windows machine that has the Oracle WebLogic Enterprise software or an older version of the Oracle Tuxedo software installed, a window named “Previous Version Detected!” will display at the very beginning of the Oracle Tuxedo installation. This window contains a message that encourages you to exit the installation program and remove the older software before installing Oracle Tuxedo 12c Release 1 (12.1.1).

If you choose to remove the older software, refer to [“Upgrading the Oracle Tuxedo System to 12c Release 1 \(12.1.1\)”](#) on page 5-1, follow the instructions to remove the older software from your machine, and then restart the installation.

If you choose *not* to remove the older software, install the Oracle Tuxedo 12c Release 1 (12.1.1) software to a different product directory than the product directory where the Oracle WebLogic Enterprise software or the older version of the Oracle Tuxedo software is installed. See also [“Previous Tuxedo Versions”](#) on page 1-7

Starting GUI-Mode Installation on a Windows System

To start the GUI-mode installation process on a Windows system, follow these steps:

1. Select a Windows system that meets the hardware and software requirements described in [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets”](#) on page A-1.

The Windows XP systems support only Oracle Tuxedo client components.

2. Log in to the Windows system as the Administrator or as a member of the Administrator group.

You need administrative privileges to install Oracle Tuxedo server components on a Windows system. If you are going to install only Oracle Tuxedo client components, you do not need administrative privileges.

3. Ensure that you have enough free space for the Oracle Tuxedo installation.

For disk space requirements, see “[Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets](#)” on page A-1. For important disk space allocation information, see “[File and Database Management and Disk Space Allocation](#)” on page D-1.

4. Install Oracle Tuxedo by downloading from the Oracle Web site:
 - a. Go to <http://www.oracle.com/technology/software/index.html> and download the Oracle Tuxedo 12c Release 1 (12.1.1) installation file specific to your platform.
 - b. Go to the directory where you downloaded the Oracle Tuxedo installer and double-click the installer filename to start the installation.
5. Proceed to “[Running GUI-Mode Installation](#)” on page 2-4.
6. Re-login to the system after Tuxedo 12c Release 1 (12.1.1) installation has completed.

Starting GUI-Mode Installation on a UNIX System

To start the GUI-mode installation process on a UNIX system, follow these steps:

1. Select a UNIX system that meets the hardware and software requirements described in “[Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets](#)” on page A-1
2. Log in to the UNIX system as the Oracle Tuxedo administrator.
3. Ensure that you have enough free space for the Oracle Tuxedo installation.

For disk space requirements, see “[Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets](#)” on page A-1. For important disk space allocation information, see “[File and Database Management and Disk Space Allocation](#)” on page D-1

4. Install Oracle Tuxedo by downloading from the Oracle Web site:
 - a. Go to <http://www.oracle.com/technology/software/index.html> and download the Oracle Tuxedo 12c Release 1 (12.1.1) installation file specific to your platform.

- b. Go to the directory where you downloaded the installer and invoke the installation procedure by entering the following command:

```
prompt> sh filename.bin
```

where *filename* is the name of the Oracle Tuxedo installer file.

Note: GUI mode is the default for Tuxedo 12c Release 1 (12.1.1) installation. If a GUI interface is not available on your Unix platform, you will receive an error message and the installation process is aborted.

If a GUI interface is not available on your Unix platform, you must use silent or console-mode installation. For more information, see [What Is Silent Installation?](#) and [What Is Console-Mode Installation?](#)

5. Proceed to [“Running GUI-Mode Installation” on page 2-4](#).

Running GUI-Mode Installation

The Oracle Tuxedo installer program is currently available in English and Japanese. By checking the value of a certain language-related environment variable on the target platform, the installer program automatically determines (1) the language displayed during the installation and (2) the language-specific message catalog installed during the installation. If the language-related environment variable is not set or is set to a value unknown to the installer program, both the language displayed and the message catalog installed default to English.

The Oracle Tuxedo installer program prompts you to enter specific information about your system and configuration. For instructions on responding to the prompts during installation, see [Table 2-1](#).

Table 2-1 Instructions

In This Window . . .	Perform the Following Action . . .
Introduction	Click Next to proceed with the installation. You may cancel the installation at any time by clicking Cancel.
Choose Install Set (.NET Client Install set is not available on UNIX platforms)	<p>Select the install set that you want installed on your system. The following seven choices are available:</p> <ul style="list-style-type: none"> • Full Install (the default)—all Tuxedo server and client software components • Server Install—Tuxedo server software components only • Full Client Install—Tuxedo client software components only • Jolt Client Install—Jolt client software components only • ATMI (/WS) Client Install—Tuxedo ATMI client software components only • CORBA Client Install—Tuxedo CORBA client software components only • .NET Client Install—Tuxedo .NET client software and ATMI client components only. <p>The .NET Client Install set installs the Tuxedo Workstation Client on your system. For more information on the Tuxedo .Net Workstation Client, see Creating Tuxedo .NET Workstation Client Applications in <i>Using the Tuxedo .NET Workstation Client</i></p> <ul style="list-style-type: none"> • Custom Install—select specific Tuxedo server and client software components. The following table entry provides a summary of options for the Custom Install. <p>For a detailed list of software components for each install set, see “Install Sets” on page 1-9.</p> <p>Note: A Windows XP system cannot be used as a Oracle Tuxedo server machine. On a Windows XP system, select Full Client Install, Jolt Client Install, ATMI Client Install, or CORBA Client Install from the installation menu.</p> <p>Select an install set and then click Next to launch the Choose Oracle Home Directory window—your selection includes the appropriate encryption software for Link-Level Encryption (LLE) and/or Secure Sockets Layer (SSL). Or select an install set and then click Customize to launch the Customize Install window.</p>

Table 2-1 Instructions

In This Window . . .	Perform the Following Action . . .
Customize Install	<p>Select (add) or deselect (clear) one or more software components from the selected install set, or choose one of the other five install sets or Custom Set from the drop-down list menu and customize its software components. For a description of the JRLY component, see “Jolt Internet Relay” on page B-5.</p> <p>Observe the following software component mappings:</p> <ul style="list-style-type: none"> • <i>Server</i>—contains ATMI server software; CORBA C++ server software; Oracle Jolt server software; Oracle SNMP Agent software, and Oracle Tuxedo Administration Console software • <i>ATMI Client</i>—contains Oracle ATMI Workstation (/WS) client software • <i>CORBA Client</i>—contains Oracle CORBA C++ client software (C++ client ORB) including environmental objects • <i>Jolt JRLY</i>—contains Oracle Jolt Relay software • <i>Jolt Client</i>—contains Oracle Jolt client software • <i>.NET Client</i>—contains Oracle .Net client software <p>After selecting or deselecting one or more software components from the selected install set, click Next to continue with the installation. The appropriate encryption software for LLE and/or SSL is automatically included.</p>
Choose Oracle Home Directory	<p>Specify the Oracle Home directory that will serve as the central support directory for all Oracle products installed on the target system. If you already have a Oracle Home directory on your system, you can select that directory (recommended) or create a new Oracle Home directory. If you choose to create a new directory, the Oracle Tuxedo installer program automatically creates the directory for you. For details about the Oracle Home directory, see “Oracle Home Directory” on page 1-10.</p> <p>Choose a Oracle Home directory and then click Next to continue with the installation.</p>

Table 2-1 Instructions

In This Window . . .	Perform the Following Action . . .
Choose Product Directory	<p>Specify the directory in which you want to install the Oracle Tuxedo software. You may select the default product directory (<code>tuxedo12cR1_VS2005</code> Windows or <code>tuxedo12cR1</code> UNIX) or create a new product directory. If you choose to create a new directory, the Oracle Tuxedo installer automatically creates the directory for you.</p> <p>WARNING: Running a previous Tuxedo version in conjunction with Tuxedo 12c Release 1 (12.1.1) on a Windows system requires that the product directories for both versions <i>do not have the same name</i>.</p> <p>Choose a product directory and then click Install to start the Oracle Tuxedo software installation.</p>
Install Samples	<p>Click the check box to install samples.</p> <p>Click Next to proceed with the installation.</p>
Confirm .Net Client Installation (Windows Platforms only)	<p>Click Next to proceed with the installation.</p> <p>Will only display if .NET Client set is selected and .Net Framework is not installed on your system.</p>
Choose Shortcut Folder (Windows Platforms only)	<p>Specify where you want to place Tuxedo 12c Release 1 (12.1.1) shortcut icons.</p>
Pre-Installation Summary	<p>Review your installation selections. Clicking Previous allows you to modify installation selections. Otherwise click Install to continue.</p>
Installing Tuxedo. . .	<p>No user input is required here. The installation program is installing Oracle Tuxedo in the user-specified product directory.</p> <p>Note: It is normal for the installation progress bar to stop for several minutes, especially at the end. The Oracle Tuxedo installer is still working when this occurs.</p>

Table 2-1 Instructions

In This Window . . .	Perform the Following Action . . .
Configure tlisten Service	<p>This window appears only if you have installed the Server Install set.</p> <p>Set the tlisten port value.</p> <p>Note: For Windows platforms, the default value is 3050. There is no default value for UNIX platforms.</p> <p>Enter a <code>tlisten</code> password of your choice. Your password must be a string of alphanumeric characters in clear-text format that is no more than 80 characters in length. You use this password to log in to the Oracle Tuxedo Administration Console.</p> <p>Note: Machines on the same domain must have the same tlisten password defined.</p> <p>Oracle Tuxedo uses the <code>tlisten</code> password to protect the local machine from administrative requests and operations that are <i>not</i> authorized. Whenever administrative communications arrive on the local machine through <code>tlisten(1)</code> or <code>wlisten(1)</code> gateway processes, Oracle Tuxedo authenticates them by means of the <code>tlisten</code> password.</p> <p>Enter and re-enter a <code>tlisten</code> password and then click Next to continue.</p>
Tlisten Encryption Method	<p>Specify the tlisten encryption method: LLE or SSL.</p> <p>The default is LLE.</p>
SSL Parameter Input	<p>If the SSL encryption method is chosen, you must enter Security Principal name, location, and password.</p>
Choose Encryption Bit	<p>Specify tlisten minimum and maximum encryption bit. The default minimum value is 0. The default maximum value is 256.</p>

Table 2-1 Instructions

In This Window . . .	Perform the Following Action . . .
SSL Installation Choice	<p>This window appears only if you have installed Server, ATMI and/or CORBA Client install set components.</p> <p>Oracle Tuxedo provides a Lightweight Directory Access Protocol (LDAP) based certificate retrieval mechanism that supports certificate-based authentication when using SSL. The Oracle Tuxedo certificate retrieval mechanism has been certified for use with the LDAP Directory server included with iPlanet Directory Server.</p> <p>Choose whether or not you want to install SSL support. If you intend to use SSL encryption in your application, select Yes and then click Next to launch the LDAP Configuration window. If you do <i>not</i> intend to use SSL encryption in your application, select No.</p>
LDAP Configuration for SSL Support	<p>Enter the following LDAP configuration information and then click Next to continue:</p> <ul style="list-style-type: none">• LDAP Service Name: Fully qualified domain name of the LDAP server (for example, <code>pcwiz.mydomain.com</code>)• LDAP PortID: Port number through which the local machine communicates with the LDAP server (for example, 389)• LDAP BaseObject: Distinguished name of the base object for search in the LDAP server (for example, <code>o=beasys.com</code>)• LDAP Filter File Location: (<code>tux_prod_dir/udataobj/security/bea_ldap_filter.dat</code> by default, where <code>tux_prod_dir</code> represents the product directory in which you installed the Oracle Tuxedo software)
Install Complete	Click Done to exit the installation program.

Congratulations! Your installation of the Oracle Tuxedo software is complete!

Assigning File Ownership on a UNIX System

If you installed the Oracle Tuxedo software on a UNIX system, Oracle strongly recommends that you create a separate user account for the Oracle Tuxedo administrator and give ownership of the

Oracle Tuxedo files to that account. To change ownership of the Oracle Tuxedo software on a UNIX machine, enter the following command:

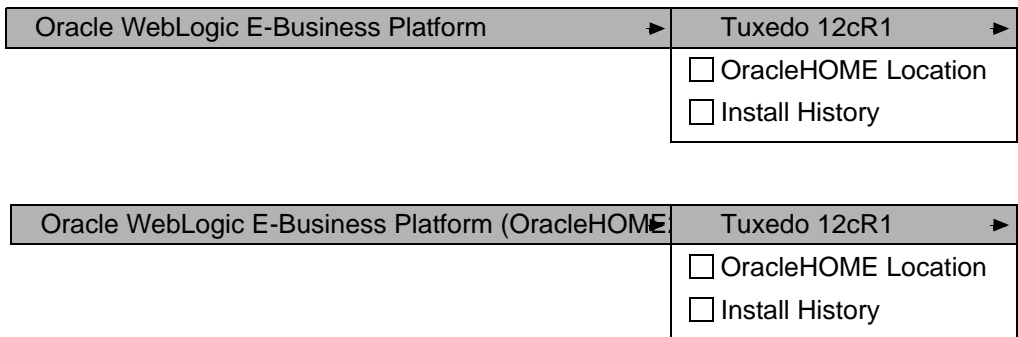
```
chown -R adm_account full_path_of_tux_prod_dir
chgrp -R adm_group full_path_of_tux_prod_dir
```

Replace *adm_account* with the Oracle Tuxedo administrator account, replace *adm_group* with the Oracle Tuxedo administrator group ID, and replace *full_path_of_tux_prod_dir* with the full pathname of the product directory in which you installed the Oracle Tuxedo software.

Understanding the Oracle Tuxedo Shortcuts on a Windows System

When installing the Oracle Tuxedo software on a Windows system, the installer program automatically uses Oracle WebLogic E-Business Platform as the parent folder for Oracle Tuxedo. As additional Oracle Home directories are added to the system, the installer program continues to create new Oracle WebLogic E-Business Platform folders using the convention shown in [Figure 2-1](#).

Figure 2-1 Tracking Multiple Oracle Home Directories on the Same Windows System



Each Oracle Home directory has an associated Oracle WebLogic E-Business Platform folder. Each Oracle WebLogic E-Business Platform folder contains one or more Oracle product folders (see note) and two files: OracleHOME Location and Install History.

Note: Because a Oracle Home directory may be the home directory for a Oracle Tuxedo installation, a Oracle WebLogic Server installation, or other Oracle product installations,

an Oracle WebLogic E-Business Platform folder may contain multiple Oracle product folders.

For a given Oracle Home directory and Oracle WebLogic E-Business Platform folder, the ORACLEHOME file contains the pathname of the Oracle Home directory, and the View History file contains a history of installation and uninstallation for the Oracle Home directory. Double-clicking the text icon associated with either file displays its content. The ORACLEHOME and history files contain information extracted from the `logs/log.txt` file located in the Oracle Home directory.

The Tuxedo 12c Release 1 (12.1.1) folder (Start → Programs → Oracle WebLogic E-Business Platform → Tuxedo 12c Release 1 (12.1.1)) contains the following shortcuts:

- *Uninstall Tuxedo*—choosing uninstall enables you to uninstall the Oracle Tuxedo 12c Release 1 (12.1.1) software. For more information about uninstalling Oracle Tuxedo, see [“Uninstalling Oracle Tuxedo” on page 6-31](#).

Understanding the Oracle Administration Program on a Windows System

In addition to the Oracle Tuxedo Administration Console, the Oracle Tuxedo 12c Release 1 (12.1.1) software for Windows provides an Oracle Administration program and two Windows services (Tlisten and Oracle procMGR) for configuring the Oracle Tuxedo system on a Windows 2003 Server system. Only if the installation included Oracle Tuxedo server components will these additional administration tools be installed on the Windows 2003 Server system.

After you have installed or upgraded the Oracle Tuxedo software on your Windows 2003 Server system, you can use the Oracle Administration control panel (Start → Settings → Control Panel → Oracle Tuxedo 12c Release 1 (12.1.1) with VS2008 Administration) to perform the following tasks:

- Access other machines on the network on which the Oracle Tuxedo system is installed
- Set and modify environment variables for the Oracle Tuxedo system
- Direct Oracle Tuxedo system messages to the Windows 2003 Server Event Log
- Configure Oracle Tuxedo `tlisten(1)` processes to start automatically
- Tune interprocess communication (IPC) resources to maximize Oracle Tuxedo system performance

For instructions on performing these tasks, see “[Configuring Oracle Tuxedo ATMI for Windows Server 2003](#)” in *Using Oracle Tuxedo ATMI on Windows*.

Reviewing the Windows Server Registry Content

The Windows Server Registry is the repository for all hardware, software, and application configuration settings for the Windows system. During the Oracle Tuxedo installation, the installer program writes general installation information as well as IPC Resource values to the Registry. The structure of the Registry relevant to Oracle Tuxedo is as follows.

```
HKEY_LOCAL_MACHINE\Software\ORACLE\Tuxedo\<version>\...
```

- Developer
- Environment
- Security

To view this structure, choose Start → Run to launch the Run dialog box, enter `regEdit32`, and click OK to launch the Registry Editor window.

Developer Key

The Developer key stores product information, including the major and minor version numbers of the release, and user and company names.

Environment Key

The Environment key stores the locations referenced by the Oracle Tuxedo environment variables set on your Windows system. It also stores other values such as IPC resource settings.

Security Key

The Security key holds the access permissions for Oracle Tuxedo processes and services. The following permissions are mandatory:

- Any user who runs `tlisten(1)` must have read access permissions.
- The account under which the Oracle ProcMGR service is running must have read access permissions.

Oracle recommends that the Administrator have full control permissions.

What Do I Do Next?

To configure your Oracle Tuxedo software, verify that your software is installed correctly, or to uninstall Oracle Tuxedo software, see [“Performing Post-Installation Tasks” on page 6-1](#).

Installing Oracle Tuxedo on UNIX Systems Using Console-Mode

Installation

The following sections describe how to install Oracle Tuxedo using console-mode installation:

- [What Is Console-Mode Installation?](#)
- [Before You Start](#)
- [Starting Console-Mode Installation](#)
- [Running Console-Mode Installation](#)
- [What Do I Do Next?](#)

What Is Console-Mode Installation?

Console-mode installation is the text-based method of executing the Oracle Installation program. It can be run only on UNIX systems and is intended for UNIX systems with non-graphics consoles. Console-mode installation offers the same capabilities as graphics-based installation

Before You Start

If you are upgrading from Oracle WebLogic Enterprise or Oracle Tuxedo 9.1 or earlier software, see [“Upgrading the Oracle Tuxedo System to 12c Release 1 \(12.1.1\)” on page 5-1](#) and follow the instructions given there. Then return here to continue your Oracle Tuxedo installation.

Starting Console-Mode Installation

To start the console-mode installation process, follow these steps:

1. Select a UNIX system that meets the hardware and software requirements described in [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#).
2. Log in to the UNIX system as the Oracle Tuxedo administrator.
3. Ensure that you have enough free space for the Oracle Tuxedo installation.

For disk space requirements, see [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#). For important disk space allocation information, see [“File and Database Management and Disk Space Allocation” on page D-1](#).

4. Install Oracle Tuxedo by downloading from the Oracle Web site:
 - a. Go to <http://www.oracle.com/technology/software/index.html> and download the Oracle Tuxedo 11g Release 1 (11.1.1.2.0) installation file specific to your platform.
 - b. Go to the directory where you downloaded the installer and invoke the installation procedure by entering the following command:

```
prompt> sh filename.bin -i console
```

where *filename* is the name of the Oracle Tuxedo installer file.

5. Proceed to [“Running Console-Mode Installation” on page 3-3](#).

Running Console-Mode Installation

To complete the console-mode installation process, which is currently available in English and Japanese, respond to the prompts in each section with one of the following actions.

- Enter the number associated with your choice or press <Enter> to accept the default.
- Enter `back` or `previous` at the prompt to review or change your selection.
- Enter `quit` in response to any prompt to exit the installation process.

In This Section . . .	Perform the Following Action . . .
Choose Locale	<p>Choose a language by entering the number associated with that language. Your choice determines both the language displayed during the installation and the language-specific message catalog installed during the installation.</p> <pre> ===== Choose Locale... ->1- English CHOOSE LOCALE BY NUMBER: ===== Launching console-mode installation on a system that utilizes a Japanese language environment adds a second choice, 2- Japanese, at the CHOOSE LOCALE BY NUMBER prompt.</pre>
Introduction	Read the Introduction and press Enter to continue with the installation process.

Choose Install Set

Select the install set that you want installed on your system. The following choices appear:

=====

Choose Install Set

Please Choose the Install Set to be installed by this installer.

- >1- Full Install
- 2- Server Install
- 3- Full Client Install
- 4- Jolt Client Install
- 5- ATMI Client Install
- 6- CORBA Client Install

7- Customize ...

ENTER THE NUMBER FOR THE INSTALL SET, OR PRESS <ENTER>
TO ACCEPT THE DEFAULT:

=====

For a list of software component for each install set, see [“Install Sets” on page 1-9](#).

Your selection includes the appropriate encryption software for Link-Level Encryption (LLE) and/or Secure Sockets Layer (SSL).

Custom Install

Select custom features.

```

=====
Choose Product Features
-----
Enter a comma-separated list of numbers representing the
features you want to select or deselect. To view a
feature's description, enter '?<number>'. Press <RETURN>
when you are done.

    1-[X] Server
    2-[X] ATMI Client
    3-[X] CORBA Client
    4-[X] Jolt JRLY
    5-[X] Jolt Client

```

```

SELECT FEATURES FOR CUSTOM INSTALLATION:
=====

```

When you first see the Custom Install screen, all Tuxedo components are selected. Entering a number or series of numbers deselects the components.

Note: Entering 1,2,3,4,5 and pressing <RETURN> deselects all the Tuxedo install components.

If you press <RETURN> and enter 'back' to return to the Custom Install screen, all 'X's are removed. You must enter a number to select a component.

Need To Select a Component

If you did not select a component, this screen appears.

```

=====
WARNING: Product Feature Selection
-----
You must select a product feature,
otherwise installation will not continue
->1- Re-Select
    2- Quit
=====

```

Note: Selecting '1- Re-Select' returns you to the Custom Install screen and all 'X's are removed. You must enter a number to select a component.

Choose Oracle Home
Directory

Specify the Oracle Home directory that will serve as the central support directory for all Oracle products installed on the target system. If you already have a Oracle Home directory on your system, you can select that directory (recommended) or create a new Oracle Home directory. If you choose to create a new directory, the Oracle Tuxedo installer program will automatically create the directory for you. For details about the Oracle Home directory, see [“Oracle Home Directory” on page 1-10](#).

Enter 1 to create a new Oracle Home directory or, if you already have a Oracle Home directory on your system, enter 2 to select an existing Oracle Home directory. Be sure to use the full pathname when specifying a new Oracle Home directory.

For example:

```
=====
Choose Oracle Home
-----
    1- Create a New Oracle Home
    2- Use Existing Oracle Home
Enter a number: 2
    1- /home3/oracle
    2- /home2/oraclehome
Existing Oracle Home: 1
=====
```

In this example, we enter 2 to display the Oracle Home directories that have already been created on this system. At the Existing Oracle Home prompt, we enter 1 to specify /home3/Oracle as the Oracle Home directory for this installation. Make sure that you enter the number associated with the Oracle Home directory instead of the directory name.

If a previous Tuxedo installation is in the existing directory you selected, the following screen appears:

```
=====
A previous installation of Tuxedo 12cR1
has been detected at /home3/Oracle.
The following options are available:
    1- Continue to install over /home3/oracle
    2- Cancel to select another Oracle Home Directory:

CHOOSE A NUMBER:
=====
```

Choose Product Directory

Specify the directory in which you want to install the Oracle Tuxedo software. You may select the default product directory (`tuxedo12cR1`) or create a new product directory. If you choose to create a new directory, the Oracle Tuxedo installer will automatically create the directory for you.

The initial default installs the product directory under the Oracle Home directory you specified in the previous section. To accept this default, enter 2 at the prompt.

The choices displayed in this section are as follows:

```
=====
Choose Product Directory
-----
    1- Modify Current Selection
    (/home3/oracle/tuxedo12cR1)
    2- Use Current Selection (/home3/oracle/tuxedo12cR1)
Enter a number:
=====
```

- Enter 1 to select an alternate directory. The following text appears:

Specify Product Installation Directory:

Specify a product directory using its full pathname. For example:

`/home3/mytux12cR1.`

When you press Enter, your modified selection appears as the default.

```
    1- Modify Current Selection (/home3/mytux12cR1)
    2- Use Current Selection (/home3/mytux12cR1)
Enter a number:
```

- Enter 2 to accept the current selection.

If you enter 2 at the initial prompt, you accept the default product directory (`/home3/oracle/tuxedo12cR1` in this example).

Install Samples

Enter "Y" to install sample
Enter "N" not to install samples

Pre-Installation
Summary

Review your installation selections. Enter back or previous to modify installation selection. Otherwise press <ENTER> to continue.

=====

Pre-Installation Summary

Please Review the Following Before Continuing:

Product Name:

Tuxedo 12cR1

Install Folder:

/home3/oracle/tuxedo12cR1

Link Folder:

/home3/oracle

Disk Space Information (for Installation target):

Required: 303,288,858 bytes

Available: 19,094,890,496 bytes

PRESS <ENTER> TO CONTINUE:

=====

Ready To Install

The choices displayed in this section are as follows:

=====

Ready To Install

InstallAnywhere is now ready to install Tuxedo 12cR1 onto your system at the following location:

/home3/oracle/tuxedo12cR1

PRESS <ENTER> TO INSTALL:

=====

Installing . . .

No user input is required here. The installation program is installing Oracle Tuxedo in the user-specified product directory.

It is normal for the installation progress bar to stop for a fairly long time, especially at the end. The Oracle Tuxedo installer is still working when this occurs.

Configure tlisten Service

Enter a `tlisten` password of your choice. Your password must be a string of alphanumeric characters in clear-text format that is no more than 80 characters in length. You use this password to log in to the Oracle Tuxedo Administration Console.

Oracle Tuxedo uses the `tlisten` password to protect the local machine from administrative requests and operations that are *not* authorized. Whenever administrative communications arrive on the local machine through `tlisten(1)` or `wlisten(1)` gateway processes, Oracle Tuxedo authenticates them by means of the `tlisten` password.

```
=====
Configure tlisten Service
-----
Password:
Verify Password:
Good Password! Press "Enter" to continue.
=====
```

SSL Installation Choice

```
=====
SSL Installation Choice
-----
Would you like to install SSL Support?
->1- Yes
   2- No
ENTER THE NUMBER FOR YOUR CHOICE, SELECTION, OR PRESS
<ENTER> TO ACCEPT THE DEFAULT:
=====
```

Enter Your LDAP
Settings for SSL
Support

This section appears only if you have installed CORBA server and/or client software components. Oracle Tuxedo provides a Lightweight Directory Access Protocol (LDAP) based certificate retrieval mechanism that supports certificate-based authentication when using SSL. The Oracle Tuxedo certificate retrieval mechanism has been certified for use with the LDAP Directory server included with iPlanet Directory Server.

```
=====
Enter Your LDAP Settings for SSL Support
-----
LDAP Service Name: a
LDAP PortID: b
LDAP BaseObject: c
LDAP Filter File Location:
Press <ENTER> to accept the default,
/home3/oracle/tuxedo12cR1/
  udataobj/security/bea_ldap_filter.dat
or enter your own:
=====
```

Installation Complete

```
=====
Installation Complete
-----
Congratulations.Tuxedo 12cR1 has been successfully
installed to:
    /home3/oracle/tuxedo12cR1
PRESS <ENTER> TO EXIT THE INSTALLER.
=====v
```

Congratulations! Your installation of the Oracle Tuxedo software is complete!

What Do I Do Next?

To prepare for the post-installation tasks, see [“Assigning File Ownership on a UNIX System” on page 2-10](#). To configure your Oracle Tuxedo software, verify that your software is installed correctly, or to uninstall Oracle Tuxedo software, see [“Performing Post-Installation Tasks” on page 6-1](#).

Installing Oracle Tuxedo Using Silent Installation

The following sections describe how to install Oracle Tuxedo using silent installation on both Windows and UNIX systems:

- [What Is Silent Installation?](#)
- [Before You Start](#)
- [Using Silent Installation: Main Steps](#)
- [Creating a Template File](#)
- [Invoking the Silent Installation Process on a Windows System](#)
- [Invoking the Silent Installation Process on a UNIX System](#)
- [Windows Template File](#)
- [UNIX Template File](#)
- [What Do I Do Next?](#)

What Is Silent Installation?

Silent installation reads the settings for your configuration from a text file that you create prior to beginning the installation. Manual intervention is not necessary during the installation process. Silent installation works on both Windows and UNIX systems.

Silent installation is a way of setting installation configurations only once and then using those configurations to duplicate the installation on many machines.

Note: Using silent installation implies your consent to the License Agreement. You neither see a copy of the Software License Agreement nor have any means to accept the terms of the agreement.

Before You Start

If you are upgrading from Oracle WebLogic Enterprise or Oracle Tuxedo 9.1 or earlier software, see [“Upgrading the Oracle Tuxedo System to 12c Release 1 \(12.1.1\)” on page 5-1](#) and follow the instructions given there. Then return here to continue your Oracle Tuxedo installation.

If you are planning to use Oracle Tuxedo 12c Release 1 (12.1.1) in conjunction with a previous Tuxedo version on a Windows system, see [“Oracle Installation Program” on page 1-7](#).

Using Silent Installation: Main Steps

The silent installation process has two primary steps:

1. Create a template file that contains the configuration settings, such as Oracle Home directory, product directory, and install set appropriate for your installation.

For the detailed procedure, see [“Creating a Template File” on page 4-3](#). Two sample template files are provided in [“Windows Template File” on page 4-9](#) and [“UNIX Template File” on page 4-11](#).

2. Invoke the installation process using the values specified in the template files.

For the detailed procedure, see [“Invoking the Silent Installation Process on a Windows System” on page 4-6](#) and [“Invoking the Silent Installation Process on a UNIX System” on page 4-7](#).

Creating a Template File

To create a template file for use in the silent installation process, follow these steps:

1. Display the template file specific to your platform. The following templates are available in:
 - Windows template file content at [“Windows Template File” on page 4-9](#).
 - UNIX template file content at [“UNIX Template File” on page 4-11](#).
2. Copy the contents of the template file and save it as a text file named `installer.properties`.
3. In the `installer.properties` file, modify the values for the keywords shown in the following table to create your desired configuration.

The `installer.properties` file is shown in [Table 4-1](#)

Table 4-1 The `installer.properties` File

For This Keyword . . .	Enter the Following Value . . .
INSTALLER_UI= (UNIX and Windows)	The mode of installation. The default is <code>silent</code> ; do not modify this value.
ORACLEHOME= (UNIX and Windows)	The full pathname of the Oracle Home directory of your choice. For details about the Oracle Home directory, see “Oracle Home Directory” on page 1-10 .
USER_INSTALL_DIR= (UNIX and Windows)	The full pathname of the product directory of your choice. WARNING: Running a previous Tuxedo version in conjunction with Tuxedo 12c Release 1 (12.1.1) on a Windows system requires that the product directories for both versions <i>do not have the same name</i> .
INSTALL_SAMPLES	“yes” installs Tuxedo 12cR1 samples. “no” does not install samples.
LDAP_HOSTNAME= (UNIX & Windows)	The fully qualified domain name of the Lightweight Directory Access Protocol (LDAP) server of your choice; for example, <code>pcwiz.mydomain.com</code> . (See Note at end of table.)

Table 4-1 The installer.properties File (Continued)

For This Keyword . . .	Enter the Following Value . . .
LDAP_PORTID= (UNIX & Windows)	The port number through which the local machine communicates with the LDAP server; for example, 389.* (See Note at end of table.)
TLISTEN_PORT (Windows)	The port number through which the local machine communicates with the TLISTEN server; for example, 1357.
LDAP_BASE_OBJECT= (Unix & Windows)	The distinguished name of the base object for search in the LDAP server; for example, o=beasys.com. (See Note at end of table.)
LDAP_FILTER_FILE= (Unix & Windows)	The full pathname of the LDAP filter file on your machine; for example, C:\oracle\tuxedo12cR1\udataobj\security\ bea_ldap_filter.dat. (See Note at end of table.)

Table 4-1 The installer.properties File (Continued)

For This Keyword . . .	Enter the Following Value . . .
TLISTEN_PASSWORD= (Unix & Windows)	A <code>tlisten</code> password of your choice. The password must be a string of alphanumeric characters in clear-text format that is no more than 80 characters in length. You will be prompted to enter this password when logging in to the Oracle Tuxedo Administration Console.
CHOSEN_INSTALL_SET=	<p>The chosen installation set. You must set this value to one of the following options:</p> <p>Notes: The valid options are case sensitive and should be coded exactly as defined below for proper installation. Use of invalid options will result in improper installation.</p> <p>Spaces should not be included after the install set name. If there are spaces after the install set name, the installation process uses the default mode <code>Full Install</code>.</p> <ul style="list-style-type: none"> • <code>Full</code> for a full installation • <code>Server</code> for server-only installation • <code>Client</code> for full client installation • <code>ATMI</code> for ATMI-client-only installation, • <code>CORBA</code> for CORBA-client-only installation • <code>Jolt</code> for Jolt-client-only installation • <code>DNET</code> for Tuxedo .NET Workstation client only installation (Windows platforms only.) <p>No customizing of install sets is possible with silent-mode installation. For a description of install sets, see “Install Sets” on page 1-9.</p>

Notes: Keywords: `LDAP_HOSTNAME`, `LDAP_PORTID`, `LDAP_BASE_OBJECT`, `LDAP_FILTER_FILE` are optional.

Include these keywords and their values only if you want to use Secure Sockets Layer (SSL) encryption. Otherwise, comment these lines out by preceding them with hash marks (#).

Keywords: `TLISTEN_PORT`, and `TLISTEN_PASSWORD` are also optional

Include these keywords and their values only if you select `Server` (server-only installation).

In the template files, comment lines are preceded by hash marks (#). The installer program does not read any line beginning with a hash mark.

Invoking the Silent Installation Process on a Windows System

Installing the software using silent installation takes the same amount of time as a standard installation. During silent installation, an initial installation program window appears briefly, indicating that the installation has started.

To invoke the silent installation process on a Windows system, follow these steps:

1. Select a Windows system that meets the hardware and software requirements described in [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#).

The Windows XP system supports only Oracle Tuxedo client components.

2. Log in to the Windows system as the Administrator or as a member of the Administrator group.

You need administrative privileges to install Oracle Tuxedo server components on a Windows system. If you are going to install only Oracle Tuxedo client components, you do not need administrative privileges.

3. Ensure that you have enough free space for the Oracle Tuxedo installation.

For disk space requirements, see [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#). For important disk space allocation information, see [“File and Database Management and Disk Space Allocation” on page D-1](#).

4. If installing Oracle Tuxedo by downloading from the Oracle Web site:

- a. Open a command-line shell.
- b. Go to <http://www.oracle.com/technology/software/index.html> and download the Oracle Tuxedo 12c Release 1 (12.1.1) installation file specific to your platform.
- c. Go to the directory where you downloaded the installer and invoke the installation procedure by entering the following command:

```
prompt> filename.exe -f path\installer.properties
```

where *filename* is the name of the Oracle Tuxedo installer file, and *path* is the full or relative pathname of the `installer.properties` file.

Note: If a file named `installer.properties` is present in the same directory as the installer (for example, `tuxedo81_win.exe`), it will be supplied automatically in silent mode. However, if `installer.properties` is in a different directory, you may use the `-f` command line option to specify the properties file. For example, `filename.exe -f path\installer.properties`

5. To verify that the software installed successfully, see [“Performing Post-Installation Tasks” on page 6-1](#).
6. Re-login to the system after Tuxedo 12c Release 1 (12.1.1) installation has completed.

Invoking the Silent Installation Process on a UNIX System

Installing the software using silent installation takes the same amount of time as a standard installation. During silent installation, a startup message appears followed by an “Installing . . .” message, indicating that the installation has started.

To invoke the silent installation process on a UNIX system, follow these steps:

1. Select a UNIX system that meets the hardware and software requirements described in [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#).
2. Log in to the UNIX system as the Oracle Tuxedo administrator.
3. Ensure that you have enough free space for the Oracle Tuxedo installation.

For disk space requirements, see [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#). For important disk space allocation information, see [“File and Database Management and Disk Space Allocation” on page D-1](#).

4. Install Oracle Tuxedo by downloading from the Oracle Web site:
 - a. Go to <http://www.oracle.com/technology/software/index.html> and download the Oracle Tuxedo 12c Release 1 (12.1.1) installation file specific to your platform.
 - b. Go to the directory where you downloaded the installer and invoke the installation procedure by entering the following command:

```
prompt> sh filename.bin -f path/installer.properties
```

where *filename* is the name of the Oracle Tuxedo installer file, and *path* is the full or relative pathname of the `installer.properties` file.

Note: If a file named `installer.properties` is present in the same directory as the installer (for example, `tuxedo81_sol.bin`), it will be supplied automatically in silent mode. However, if `installer.properties` is in a different directory, you may use the `-f` command line option to specify the properties file. For example, `filename.exe -f path/installer.properties`

5. To verify that the software installed successfully, see [“Performing Post-Installation Tasks” on page 6-1](#).

Windows Template File

This sample Windows template file (`installer.properties`) in [Listing 4-1](#) applies to a silent installation of Oracle Tuxedo 12c Release 1 (12.1.1).

Listing 4-1 Windows `installer.properties` Template File

```

INSTALLER_UI=silent
### Locale ###
USER_LOCALE=en
### Oracle Home Directory ###
### NOTE: backslashes must be escaped. ###
ORACLEHOME=c:\\oracle\\tuxedo12cR1
### Product Installation Directory ###
USER_INSTALL_DIR=c:\\oracle\\tuxedo12cR1
### LDAP Service Name ###
LDAP_HOSTNAME=pcwiz.mydomain.com
### LDAP PortID ###
LDAP_PORTID=389
### LDAP BaseObject ###
LDAP_BASE_OBJECT="o=beasys.com"
### LDAP Filter File Location ###
LDAP_FILTER_FILE=c:\\oracle\\tuxedo12cR1\\udataobj\\security\\
bea_ldap_filter.dat
### Default Listen Port, User Can Run ###
### Oracle Tuxedo 12cR1 Administration in ###
### Control Panel to Change to Another Port ###
TLISTEN_PORT=1357
### Tlisten Password ###
TLISTEN_PASSWORD=luckluck
### TLISTEN ENCRYPTION###
ENCRYPT_CHOICE_BOOLEAN_2=0
ENCRYPT_CHOICE_BOOLEAN_1=1
SSL_PNAME=test1
SSL_PLOC=test1.pem
SSL_PPASS=passwd
min_crypt_bits_1=0

```

```

min_crypt_bits_2=1
min_crypt_bits_3=0
min_crypt_bits_4=0
max_crypt_bits_1=0
max_crypt_bits_2=0
max_crypt_bits_3=1
max_crypt_bits_4=0
### Valid Install Sets, select one:###
### CHOSEN_INSTALL_SET=Full###
### CHOSEN_INSTALL_SET=Server###
### CHOSEN_INSTALL_SET=Client###
### CHOSEN_INSTALL_SET=ATMI###
### CHOSEN_INSTALL_SET=CORBA###
### CHOSEN_INSTALL_SET=Jolt###
### CHOSEN_INSTALL_SET=DNET###
### NOTE: Spaces should not be included after the install set name. ###
### If there are spaces after the install set name, the installation ###
### process uses the default mode Full Install. ###
CHOSEN_INSTALL_SET=Full Install
### Install Samples Y/N###
INSTALL_SAMPLES=Yes

```

Note: ENCRYPT_CHOICE_BOOLEAN_1=1: "1" means tlisten supports SSL.

ENCRYPT_CHOICE_BOOLEAN_2=1: "1" means tlisten supports LLE.

0 means SSL/LLE is not supported.

If one of the encryption methods (SSL or LLE) is set to 1, the other must be set to 0.

min_crypt_bits_1: "1" means min encryption bit 0 is set, "0" not set
min_crypt_bits_2: "1" means min encryption bit 56 is set, "0" not set
min_crypt_bits_3: "1" means min encryption bit 128 is set, "0" not set
min_crypt_bits_4: "1" means min encryption bit 256 is set, "0" not set
One encryption method must be set to 1; the other must be set to 0.

max_crypt_bits_1: "1" means max encryption bit 0 is set, "0" not set
max_crypt_bits_2: "1" means max encryption bit 56 is set, "0" not set
max_crypt_bits_3: "1" means max encryption bit 128 is set, "0" not set
max_crypt_bits_4: "1" means max encryption bit 256 is set, "0" not set
One encryption method must be set to 1; the other must be set to 0.

```

SSL_PNAME = SECURITY PRINCIPAL NAME

SSL_PLOC = SECURITY PRINCIPAL LOCATION

SSL_PPASS = SECURITY PRINCIPAL PASSWORD

```

UNIX Template File

This sample UNIX template file (`installer.properties`) in [Listing 4-2](#) applies to a silent installation of Oracle Tuxedo 12c Release 1 (12.1.1).

Listing 4-2 UNIX `installer.properties` Template File

```

INSTALLER_UI=silent
### Locale ###
USER_LOCALE=en
### Installation Mode ###
### Input New Install ###
INSTALL_MODE=New Install
### Oracle Home Directory ###
ORACLEHOME=/home/user/oracle/tuxedo12cR1
### Product Installation Directory ###
USER_INSTALL_DIR=/home/user/oracle/tuxedo12cR1
### LDAP Service Name ###
LDAP_HOSTNAME=pcwiz.mydomain.com
### LDAP PortID ###
LDAP_PORTID=389
### LDAP BaseObject ###
LDAP_BASE_OBJECT="o=beasys.com"
### LDAP Filter File Location ###
LDAP_FILTER_FILE=/home/user/oracle/tuxedo12cR1/udataobj/security/
bea_ldap_filter.dat
### Tlisten Password ###
TLISTEN_PASSWORD=luckluck
### Valid Install Sets, select one:###
### CHOSEN_INSTALL_SET=Full###
### CHOSEN_INSTALL_SET=Server###
### CHOSEN_INSTALL_SET=Client###

```

```
### CHOSEN_INSTALL_SET=ATMI###
### CHOSEN_INSTALL_SET=CORBA###
### CHOSEN_INSTALL_SET=Jolt###
### NOTE: Spaces should not be included after the install set name. ###
### If there are spaces after the install set name, the installation ###
### process uses the default mode Full Install. ###
CHOSEN_INSTALL_SET=Full Install
CHOSEN_INSTALL_SET=Full Install
### Install Samples Y/N###
INSTALL_SAMPLES=Yes
```

What Do I Do Next?

To prepare for the post-installation tasks, read the following sections:

- [“Assigning File Ownership on a UNIX System”](#) on page 2-10
- [“Understanding the Oracle Tuxedo Shortcuts on a Windows System”](#) on page 2-11
- [“Understanding the Oracle Administration Program on a Windows System”](#) on page 2-12
- [“Reviewing the Windows Server Registry Content”](#) on page 2-13

To configure your Oracle Tuxedo software, verify that your software is installed correctly, or to uninstall Oracle Tuxedo software, see [“Performing Post-Installation Tasks”](#) on page 6-1.

Upgrading the Oracle Tuxedo System to 12c Release 1 (12.1.1)

The following sections provide procedures for upgrading your Oracle Tuxedo or Oracle WebLogic Enterprise application to Oracle Tuxedo 12c Release 1 (12.1.1) using simple and hot upgrade procedures:

- [Preparing Your Machine for an Upgrade](#)
- [Selecting an Upgrade Procedure](#)
- [Backing Up Files](#)
- [Performing a Simple Upgrade](#)
- [Performing Additional Upgrade Steps to Windows Server](#)
- [Upgrading from Tuxedo 6.5 to Tuxedo 12c Release 1 \(12.1.1\)](#)
- [Performing a Hot Upgrade](#)
- [Rebuilding an Application](#)
- [Installing Oracle Jolt 12c Release 1 \(12.1.1\) with Oracle WebLogic Server](#)

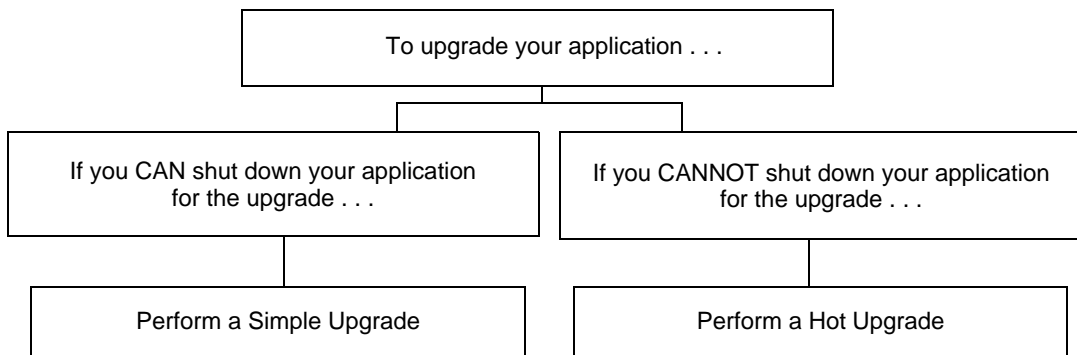
Preparing Your Machine for an Upgrade

Before you install the Oracle Tuxedo 12c Release 1 (12.1.1) software on a machine, make sure that the required version of the operating system and the compiler have been installed. For

operating system and compiler requirements for each supported platform, see [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#).

Selecting an Upgrade Procedure

If you are installing Oracle Tuxedo 12c Release 1 (12.1.1) on a platform that contains an earlier release of Oracle Tuxedo software or the Oracle WebLogic Enterprise software, be sure to read the following procedures carefully.



If you can shut down your application for the upgrade, then shut down your application and perform a simple upgrade. For instructions on performing a simple upgrade, see [“Performing a Simple Upgrade” on page 5-3](#).

If you cannot shut down your application for the upgrade, you can perform a hot upgrade for any of the following software products to Oracle Tuxedo 12c Release 1 (12.1.1) without having to shut down your enterprise:

- Versions 8.1, 9.0, 9.1, 10.0, 10.3, 11.1.1.1.0, 11.1.1.2.0, and 11.1.1.3.0 of Oracle Tuxedo
- Version 5.1 of WebLogic Enterprise

You can add the Oracle Tuxedo 12c Release 1 (12.1.1) software to existing Oracle Tuxedo or Oracle WebLogic Enterprise domains without shutting down the existing applications, and you can add new Oracle Tuxedo 12c Release 1 (12.1.1) application servers to those domains without shutting down the existing applications. For instructions on performing a hot upgrade, see [“Performing a Hot Upgrade” on page 5-6](#).

Backing Up Files

There are certain files that you need to back up prior to the installation and then restore after the installation is complete. Any modifications that you made to these files will be overwritten when the new Oracle Tuxedo software is installed.

To avoid having to modify these files again, follow these steps:

1. Back up the `RM` file to a temporary location. The `RM` file contains database vendor-specific settings that are used by commands such as `buildtms` and `buildXAJ5`. It is located in the `tux_prod_dir/udataobj` directory, where `tux_prod_dir` is the directory in which you installed the Oracle Tuxedo or Oracle WebLogic Enterprise software.
2. Back up the `tuxwsvr.ini` and `webgui.ini` files to a temporary location. These files contain customized initialization settings for the Oracle Tuxedo Administration Console server and GUI. The `tuxwsvr.ini` file is located in the `tux_prod_dir/udataobj` directory, and the `webgui.ini` file is located in the `tux_prod_dir/udataobj/webgui` directory.
3. Back up the `CATNAMES`, `jrepository`, and `jrly.config` files. The `CATNAMES` file is located in the `tux_prod_dir/locale` directory. The `jrepository` and `jrly.config` files are located in the `tux_prod_dir/udataobj/jolt` directory structure.
4. Move any files that you or your coworkers added to the `tux_prod_dir` directory to a temporary location.
5. After the installation is complete, restore these files to their original locations.

Performing a Simple Upgrade

A simple upgrade consists of the following basic steps:

1. If you already have the `DMTLOG`, make sure that no pending transaction log is in it.
2. Shut down the application.
3. Back up files as described in [“Backing Up Files” on page 5-3](#).
4. Update the text version of the configuration file on the `MASTER` machine by running the `tmunloadcf(1)` command.
5. Back up the old binary version of the configuration file.
6. If upgrading a Oracle Tuxedo 9.1 machine, uninstall the Oracle Tuxedo system software as described in [“Uninstalling Oracle Tuxedo” on page 6-31](#).

7. If upgrading a Windows machine running Oracle WebLogic Enterprise or Oracle Tuxedo 10.0 or earlier software, perform the additional steps specified in [“Performing Additional Upgrade Steps to Windows Server” on page 5-4](#).
8. Install the Oracle Tuxedo 12c Release 1 (12.1.1) software on at least the MASTER and backup MASTER machines.
9. Reload the text version of the configuration file by running `tmloadcf(1)` on the MASTER machine.
10. If necessary, recompile and relink your application programs.
11. If you already have the DMTLOG, remove it.
12. Reboot the application.
13. As time permits, shut down and upgrade the other machines in your configuration.

Performing Additional Upgrade Steps to Windows Server

For a Windows machine running Oracle WebLogic Enterprise or Oracle Tuxedo 12c Release 1 (12.1.1) or earlier software, perform the following additional steps before installing the Oracle Tuxedo 12c Release 1 (12.1.1) software to ensure that you are starting with a clean system.

1. Stop the Tuxedo TListen and Oracle `procMGR` services. From the Start menu, click Start → All Programs → Administrative Tools → Services. The Services window appears.
Note: The Oracle Tuxedo `procMGR` service is called IPC Helper in Oracle Tuxedo 6.5 or Oracle WebLogic Enterprise 5.1.
2. Scroll to the entry for the TListen service, select it, and then click the Stop button. The Status value should change from Started to a blank entry. Then scroll to the Oracle `procMGR` (or IPC Helper) service, select it, and click the Stop button. In some cases, you may see an error; however, the service’s Status value should change from Started to a blank entry. Click the Close button.
3. Run the Windows Add/Remove program to remove the prior Oracle Tuxedo or Oracle WebLogic Enterprise software version. From the Start menu, choose Start → Settings → Control Panel → Add/Remove Programs. The Add/Remove Programs Properties window appears.
4. Scroll to the entry for the prior Oracle Tuxedo or Oracle WebLogic Enterprise software, select it, and click the Add/Remove button.

Note: If you have both the C++ and Java components of WebLogic Enterprise 5.1 installed on your machine, you must remove the Java software *before* removing the C++ software.

5. In response to the prompt, confirm that you want to uninstall the software. After the program finishes, the Remove Programs From Your Computer window appears.
6. If the uninstall program was not able to remove all directories (usually because the files were added after the original installation), you can click the Details button to find out which directories remain on your system. If the files in the directories contain changes that you made, such as a modified sample file, move the modified files to a temporary location.
7. Reboot your system after the uninstall completes.

Upgrading from Tuxedo 6.5 to Tuxedo 12c Release 1 (12.1.1)

If you are upgrading from Tuxedo 6.5 to Tuxedo 12c Release 1 (12.1.1), please note the following:

- In order to allow a Tuxedo 12c Release 1 (12.1.1) server to interoperate with Tuxedo 6.5 software, the Tuxedo 12c Release 1 (12.1.1) server `servopts -t` option must be specified. The server may be a workstation listener (WSL) process (where the `servopts -t` option allows interoperability for all of its workstation handler—WSH—processes), a domain gateway (GWTDOMAIN) process, or a system or application server process. For more information, see [GWTDOMAIN\(5\)](#) in the *Oracle Tuxedo File Formats, Data Descriptions, MIBs, And System Processes Reference*.

- In order to allow a domain gateway (GWTDOMAIN) in Tuxedo 12c Release 1 (12.1.1) to interoperate with a domain gateway running on Tuxedo 6.5, you must upgrade to Tuxedo 6.5 patch level 446.

If you have not upgraded to patch level 446, then you must set the `TM_GWT_OLDSECHECK` environment variable of the domain gateway in this Tuxedo 12c Release 1 (12.1.1). For more Oracle Tuxedo environment variable information, see [tuxenv\(5\)](#) in the *Oracle Tuxedo File Formats, Data Descriptions, MIBs, And System Processes Reference*.

- If your Tuxedo 6.5 COBOL program calls `TPINITIALIZE()`, you must set the `TM_CBL_IGNORE_CONTEXT` environment variable to "Y" when you run the COBOL application. For more Oracle Tuxedo environment variable information, see [tuxenv\(5\)](#) in the *Oracle Tuxedo File Formats, Data Descriptions, MIBs, And System Processes Reference*.

Performing a Hot Upgrade

Before performing a hot upgrade, back up files as described in [“Backing Up Files” on page 5-3](#).

During the hot upgrade, you must shut down the MASTER and backup MASTER machines *serially*. The remainder of the configuration is unaffected.

Pre-Upgrade Notes

Once you have finished migrating the MASTER to Oracle Tuxedo 12c Release 1 (12.1.1) (via a hot upgrade), you will not be able to migrate it back to the acting backup until you have upgraded the backup to Oracle Tuxedo 12c Release 1 (12.1.1). In other words, migrating the MASTER is an irreversible procedure; once you have completed it, you cannot change the MASTER back to the release of the Oracle Tuxedo software on which it was running previously.

When you run `tmadmin(1)` on a Oracle Tuxedo 12c Release 1 (12.1.1) machine with an old-release MASTER, the `UPGRADE` file shown in [“Hot-Upgrade Example” on page 5-8](#) must reside in the current directory. To make sure that the files are in the current directory, you might, for example, run `tmadmin` from the Tuxedo application directory: `%APPDIR%` (Windows)/ `$APPDIR` (UNIX) directory.

During migration, keep in mind the following guidelines regarding the Oracle Tuxedo data storage areas:

- You will be instructed to remove the old `TUXCONFIG` file. The `TUXCONFIG` file is propagated automatically when the backup machine is run; therefore, you do not need to do anything.
- Create the `TLOG` when you load the software. If you already have a `TLOG`, remove it.
- Do not do anything with application queue spaces and queues.
- If you have a Domains configuration (`BDMCONFIG`) file, leave it as is.

To minimize downtime, Oracle recommends installing Oracle Tuxedo 12c Release 1 (12.1.1) on the MASTER machine and backup MASTER, and rebuilding all clients and servers in the directory on each machine that is defined by `%APPDIR%/ $APPDIR`, in parallel with the directory in which the old-release clients and servers were created. In other words, both an old-release application and a Oracle Tuxedo 12c Release 1 (12.1.1) application should be available on each machine. *This advice is not a requirement, but a recommendation.*

The Oracle Tuxedo software supports device list path entries of up to 256 bytes in length (in previous releases the limit was 64 bytes in length.) Queue Spaces, `TLOG` files, and other

configuration files originally created with an earlier version of Oracle Tuxedo are understood and supported with this release. However, such device lists originally created with Oracle Tuxedo 8.0 or earlier are restricted to 64-byte path names. Therefore, if the administrator wants to add a new extent using a path name greater than 64 bytes in length to a queue space in the future and such queue space is empty when the upgrade is performed, the administrator may want to create the queue space device list and the queue space at the time of the hot upgrade. Many applications need to preserve queue spaces across an upgrade and will not be able to recreate them. Furthermore, most applications will not need to add queue space extent in the future or will be able to do so using path names no longer than the ones they are currently using. *This queue space recreation is completely optional and mentioned for completeness.*

If you already have the DMTLOG, make sure that no pending transaction log is in it before shutting down the application, and keep in mind that removing the DMTLOG (if it exists) after the application is shut down.

Hot-Upgrade Procedure

This section provides an overview of the tasks for a “hot upgrade.”

1. Make sure your environment is set up as follows:
 - The configuration must have a backup MASTER machine.
 - The MODEL option in the configuration must be set to MP.
 - The MIGRATE option in the configuration must be set.
2. If upgrading a Oracle Tuxedo 9.1 machine, uninstall the Oracle Tuxedo system software as specified in [“Uninstalling Oracle Tuxedo” on page 6-31](#).
3. If upgrading a Windows machine running Oracle WebLogic Enterprise or Oracle Tuxedo 9.1 or earlier software, perform the additional steps specified in [“Performing Additional Upgrade Steps to Windows Server” on page 5-4](#).
4. Shut down the backup MASTER machine.
5. Install the Oracle Tuxedo 12c Release 1 (12.1.1) software on the backup MASTER machine and reboot the backup machine.
6. Migrate the MASTER machine running the old release to the backup MASTER machine running Oracle Tuxedo 12c Release 1 (12.1.1).
7. Shut down the machine that is now acting as the backup MASTER.

8. Install Oracle Tuxedo 12c Release 1 (12.1.1) on the old MASTER.
 9. Using Oracle Tuxedo 12c Release 1 (12.1.1), reboot the machine that you shut down in step 7.
- For more information about hot upgrades, refer to the following section, [“Hot-Upgrade Example.”](#)

Hot-Upgrade Example

The following example describes a sample scenario for a hot upgrade and the sequence of steps to perform the hot upgrade.

Suppose you have the following on a UNIX platform:

- A machine called MACH1 that is configured as the MASTER and that runs an old release
 - A corresponding application directory called \$APPPDIR1
 - A corresponding configuration file called \$TUXCONFIG1
- A machine called MACH2 that is configured as the backup
 - A corresponding application directory called \$APPPDIR2
 - A corresponding configuration file called \$TUXCONFIG2

Refer to the following instruction sequence in [Listing 5-1](#) to perform a hot upgrade for your application.

Listing 5-1 Typical Hot Upgrade

```
MACH1> tmsshutdown -B MACH2 -l MACH2
MACH1> Run tmconfig to reset the APPDIR, TUXCONFIG, TUXDIR for MACH2
MACH2> Kill tlisten process
MACH2> Load Tuxedo 12cR1 if it is not already loaded
MACH2> cd $APPPDIR2
MACH2> rm $TUXCONFIG2
MACH2> >UPGRADE      # Indicator that upgrade being done
MACH2> Reset PATH, LD_LIBRARY_PATH, TUXCONFIG, APPDIR
MACH2> Rebuild clients and servers, create TLOG
MACH2> Start Tuxedo 12cR1 tlisten
MACH1> >UPGRADE      # Indicator that upgrade being done
MACH1> tmboot -B MACH2 -l MACH2
MACH2> tmadmin
      master
      Y
```

```

psr
q
MACH2> tmsshutdown -B MACH1 -l MACH1
MACH2> Run tmconfig to reset the APPDIR, TUXCONFIG, TUXDIR for MACH1
MACH1> Kill tlisten process
MACH1> Load Tuxedo 12cR1 if it is not already loaded
MACH1> cd $APPDIR1
MACH1> rm $TUXCONFIG1
MACH1> Reset PATH, LD_LIBRARY_PATH, TUXCONFIG, APPDIR
MACH1> Rebuild clients and servers, create TLOG
MACH1> Start Tuxedo 12cR1 tlisten
MACH1> rm UPGRADE # Remove indicator that upgrade being done
MACH2> rm UPGRADE # Remove indicator that upgrade being done
MACH2> tmboot -B MACH1 -l MACH1

```

Note: If you are performing the upgrade illustrated in [Listing 5-1](#) from a Windows platform, be sure to stop the Oracle Tuxedo `procMGR` (called IPC Helper in Oracle Tuxedo 6.5 or Oracle WebLogic Enterprise 5.1) service after stopping the `tlisten` process and to start the Oracle Tuxedo `procMGR` service *before* starting the `tlisten` process.

Rebuilding an Application

Now that you have successfully installed the Oracle Tuxedo software, you need to rebuild all application clients and servers that you want to execute on a system upgraded to Oracle Tuxedo 12c Release 1 (12.1.1)

Notes: The rebuild is not necessary when you upgrade from Oracle Tuxedo 11.1.1.1.0 to 12.1.1.

The TMS server must be rebuilt when upgrading a previous Oracle Tuxedo version to Oracle Tuxedo 12c Release 1 (12.1.1).

You should also remove any old `TUXCONFIG` files and reload them using the newly installed `tmloadcf(1)`.

If your Oracle Tuxedo system applications are distributed, the `MASTER` and backup `MASTER` machines must run the highest release of the Oracle Tuxedo software of any machines in the configuration. Other machines, including Workstation clients, may continue to use executables from an earlier release. For example, even if your `MASTER` and backup `MASTER` machines are running Oracle Tuxedo 12c Release 1 (12.1.1), Oracle Tuxedo 6.5 or later may still be used on other machines.

There is one important exception to the “highest release” rule. If you wish to install Oracle Tuxedo 12c Release 1 (12.1.1) only on Workstation clients, and you do not want to upgrade any of your Oracle Tuxedo system server machines to Oracle Tuxedo 12c Release 1 (12.1.1), you may do so. In other words, with any release level beginning with Oracle Tuxedo 6.5, Workstation clients may be intermixed freely in a configuration in which the MASTER machines is running Oracle Tuxedo 6.5 or later.

Installing Oracle Jolt 12c Release 1 (12.1.1) with Oracle WebLogic Server

This section provides supplementary installation instructions for customers using Oracle Jolt 12c Release 1 (12.1.1) with Oracle WebLogic Server 6.1, 7.0, or 8.1 and Oracle Tuxedo 6.5, 7.1, 8.0, 8.1, 9.0, 9.1, or 10.0. You can install the Jolt client personality software known as “Oracle Jolt for Oracle WebLogic Server” on an Oracle WebLogic Server 6.1, 7.0, or 8.1 machine by copying the Jolt client personality software from a Oracle Tuxedo 12c Release 1 (12.1.1) machine to the target Oracle WebLogic Server machine. Of course, if the Oracle Tuxedo 12c Release 1 (12.1.1) machine and the target Oracle WebLogic Server machine are the same machine, you need only include the Jolt client library class files in your CLASSPATH variable setting.

The “Oracle Jolt for Oracle WebLogic Server” client software consists of three JAR files: `jolt.jar`, `joltjse.jar`, and `joltwls.jar`. Include these files in your CLASSPATH variable setting.

For instructions on configuring the Jolt connection pool on Oracle WebLogic Server, see [Using Oracle Jolt with Oracle WebLogic Server](#).

Performing Post-Installation Tasks

The following sections describe the tasks you perform after installing Oracle Tuxedo:

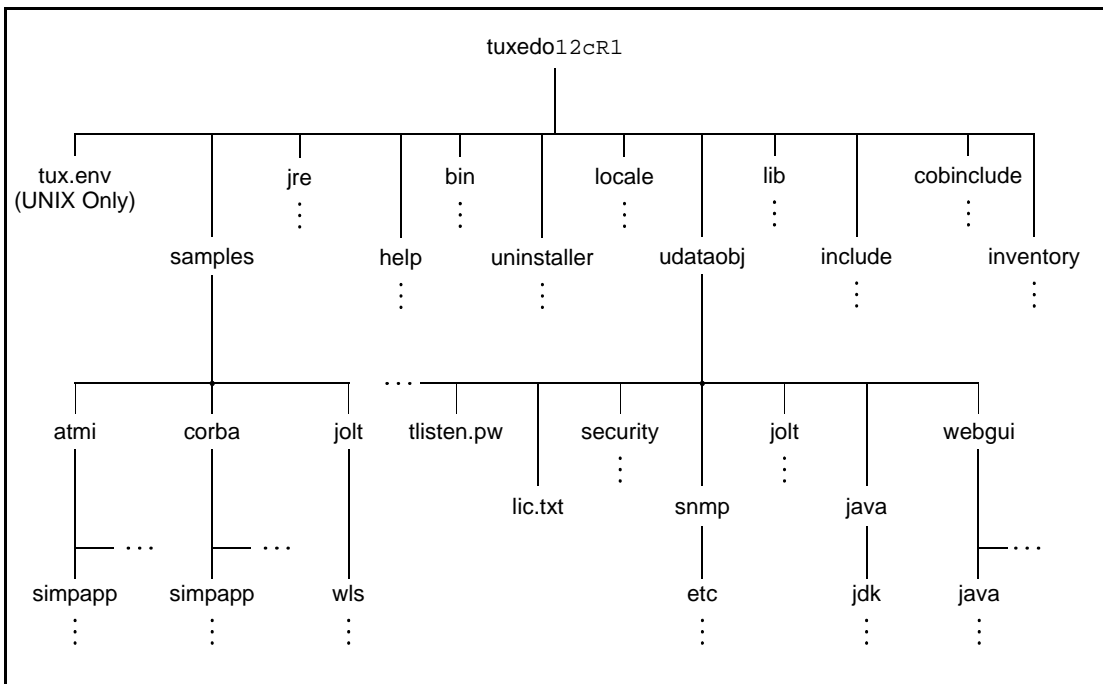
- [Understanding the Oracle Tuxedo Directory Structure](#)
- [Understanding the Oracle Tuxedo Architecture](#)
- [Using the Registry to Further Restrict Access on a Windows System](#)
- [Setting Up Your Environment](#)
- [Starting the tlisten Process](#)
- [Editing a UBBCONFIG File](#)
- [Using the TYPE Parameter in UBBCONFIG](#)
- [Checking IPC Requirements](#)
- [Creating the Universal Device List and the Transaction Log](#)
- [Running simpapp to Verify Your Installation](#)
- [Running buildtms for Oracle Tuxedo Applications That Use XA Resource Managers](#)
- [Uninstalling Oracle Tuxedo](#)
- [Reinstalling Oracle Tuxedo](#)

Understanding the Oracle Tuxedo Directory Structure

During the Oracle Tuxedo software installation, the installer program creates the following directory structure for a *full installation*. A full installation contains all the Oracle Tuxedo server and client software components plus the Link-Level Encryption (LLE) and Secure Sockets Layer (SSL) encryption software packages.

Oracle Tuxedo 12c Release 1 (12.1.1) Directory Structure is shown in [Figure 6-1](#).

Figure 6-1 Oracle Tuxedo 12c Release 1 (12.1.1) Directory Structure



The product directory shown here, `tuxedo12c` Release 1 (12.1.1), is the default for Oracle Tuxedo 12c Release 1 (12.1.1). The default name can be changed during installation.

The top-level directories and files of the Oracle Tuxedo directory structure are briefly described in [Table 6-1](#).

Table 6-1 Top-level Directories and Files of the Oracle Tuxedo Directory Structure

Directory Name	Description
<p><code>samples</code></p> <p>Note: Samples are not installed unless they are selected during Tuxedo 12cR1 installation.</p>	<p>Contains sample code and resources designed to help you learn how to develop your own applications using Oracle Tuxedo. The samples directory contains the following subdirectories:</p> <ul style="list-style-type: none"> • <code>atmi</code> A collection of simple applications that demonstrate the many features of the Oracle Tuxedo Application-to-Transaction Monitor Interface (ATMI) server software. • <code>corba</code> A collection of simple applications that demonstrate the many features of the Oracle Tuxedo Common Object Request Broker Architecture (CORBA) C++ server software. • <code>jolt</code> A collection of simple applications that demonstrate the many features of the Oracle Jolt server software.
<code>help</code>	Contains online help files for the Oracle Tuxedo Administration Console.
<code>jre directory</code>	<p>A directory containing the 1.5x version of the Java Runtime Environment (JRE). JRE 1.5.x provides the Java Virtual Machine, or JVM, required by the Oracle Tuxedo installation program. This version of the JRE is included in the Oracle Tuxedo 12c Release 1 (12.1.1) distribution. It is automatically installed in the Oracle Home directory when you install Oracle Tuxedo.</p> <p>Note: The JRE cannot be used for development. To develop Oracle Tuxedo Java client applications, you must ensure that the Java Development Kit (JDK) is installed on your system.</p>

Table 6-1 Top-level Directories and Files of the Oracle Tuxedo Directory Structure

Directory Name	Description
<code>bin</code>	Contains Tuxedo, Jolt, and SNMP Agent executable programs.
<code>uninstaller</code>	Contains code required to uninstall the Oracle Tuxedo software.
<code>locale</code>	Contains subdirectories to support the localization of system messages. C subdirectory contains message catalogs for the default locale (U.S. English).
<code>cobinclude</code>	Contains <code>.cbl</code> file entries for use in COBOL programs.

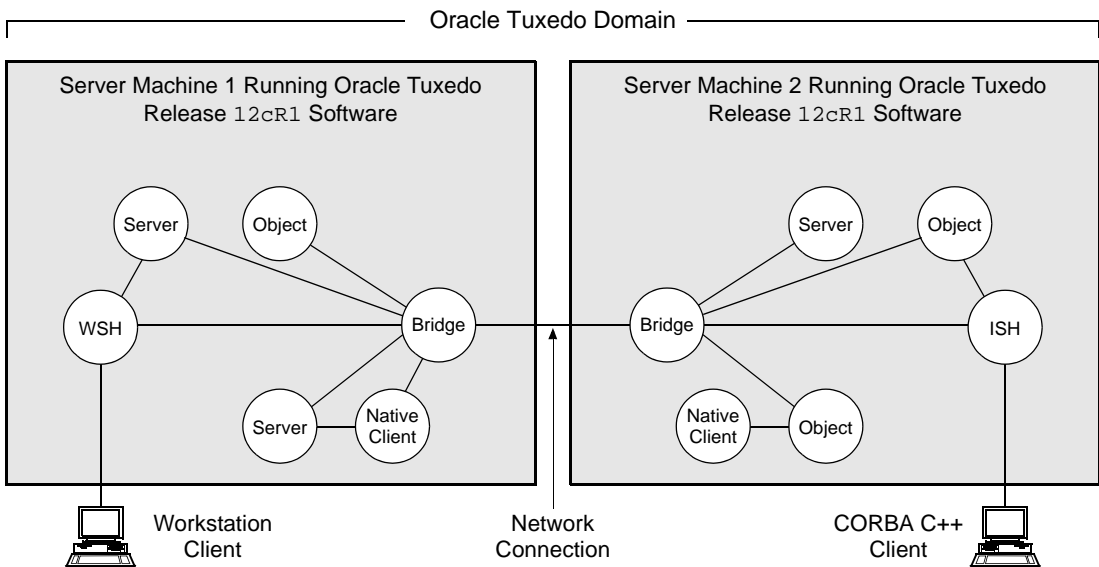
Table 6-1 Top-level Directories and Files of the Oracle Tuxedo Directory Structure

Directory Name	Description
lib	Contains compiled object files, including dynamic shared libraries (for platforms on which Oracle Tuxedo uses dynamic shared libraries) and other object files needed to build Oracle Tuxedo clients and servers.
include	Contains C and C++ language header files, as well as OMG IDL files. Includes subdirectories <code>unicode</code> and <code>xercesc</code> ; may include subdirectory <code>rpc</code> , depending on the platform.
udataobj	<p>Contains other directories and files required by Oracle Tuxedo. The <code>udataobj</code> directory contains the following subdirectories and files:</p> <ul style="list-style-type: none"> • <code>security</code> Contains the default Lightweight Directory Access Protocol (LDAP) filter file (<code>bea_ldap_filter.dat</code>) and LLE-related and SSL-related encryption files. • <code>jolt</code> Contains the files for the Oracle Jolt software components that you selected to install. • <code>snmp</code> Contains the <code>etc</code> directory, which in turn contains the SNMP Agent configuration files, MIB files, and other miscellaneous files. • <code>java</code> Contains the classes and Java archive files needed to run Java applications. • <code>webgui</code> Contains the Java and image files for the Oracle Tuxedo Administration Console. • <code>tlisten.pw</code> (file) Contains the <code>tlisten</code> administrative password that you entered during the installation.
tux.env (file)	UNIX systems only: Contains Oracle Tuxedo environment variables for UNIX installations and serves as a model for setting those variables.

Understanding the Oracle Tuxedo Architecture

Figure 6-2 shows an Oracle Tuxedo *domain*, which is the basis of the Oracle Tuxedo architecture.

Figure 6-2 Simplified View of Oracle Tuxedo Architecture



A Tuxedo domain, also known as a Tuxedo application, is a business software program, built upon the Tuxedo system, that is defined and controlled by a single configuration file—the `UBBCONFIG` file. A Tuxedo domain consists of many system processes, one or more application client processes, one or more application server processes, and one or more computer machines connected over a network.

Other important terms and concepts that you need to know about Oracle Tuxedo before performing post-installation checks are briefly described in the following sections:

- [UBBCONFIG File](#)
- [MASTER Machine](#)
- [TUXCONFIG File](#)
- [TUXCONFIG Environment Variable](#)

- **TUXDIR Environment Variable**

There is no need to fully understand these terms now; rather, use the sections as a reference. As you come upon these terms during the post-installation procedures, refer back to these sections to understand exactly what these terms mean.

UBBCONFIG File

Each Tuxedo domain is controlled by a configuration file in which installation-dependent parameters are defined. The text version of the configuration file is referred to as UBBCONFIG, although the configuration file may have any name, as long as the content of the file conforms to the format described in reference page [UBBCONFIG\(5\)](#) in *OracleTuxedo File Formats, Data Descriptions, MIBs, and System Processes Reference*.

The UBBCONFIG file for a Tuxedo domain contains all the information necessary to boot the application, such as lists of its resources, machines, groups, servers, available services, and so on. It consists of nine sections, five of which are required for all configurations: RESOURCES, MACHINES, GROUPS, SERVERS, and SERVICES.

MASTER Machine

The MASTER machine, or MASTER node, for a Tuxedo domain contains the domain's UBBCONFIG file, and is designated as the MASTER machine in the RESOURCES section of the UBBCONFIG file. Starting, stopping, and administering a Tuxedo domain is done through the MASTER machine.

In a multi-machine Tuxedo domain running different releases of the Tuxedo system software, the MASTER machine must run the highest release of the Tuxedo system software in the domain.

TUXCONFIG File

The TUXCONFIG file is a binary version of the UBBCONFIG file. It is created by running the `tmloadcf(1)` command, which parses UBBCONFIG and loads the binary TUXCONFIG file to the location referenced by the TUXCONFIG environment variable. As with UBBCONFIG, the TUXCONFIG file may be given any name.

The MASTER machine for a Tuxedo domain contains the master copy of the TUXCONFIG file. Copies of the TUXCONFIG file are propagated to all other server machines—referred to as non-MASTER machines—in a Tuxedo domain whenever the Tuxedo system is booted on the MASTER machine.

TUXCONFIG Environment Variable

The TUXCONFIG environment variable defines the location on the MASTER machine where the `tmloadcf(1)` command loads the binary TUXCONFIG file. It must be set to an absolute pathname ending with the device or system file where TUXCONFIG is to be loaded.

The TUXCONFIG pathname value is designated in the MACHINES section of the UBBCONFIG file. It is specified for the MASTER machine *and* for every other server machine in the Tuxedo domain. When copies of the binary TUXCONFIG file are propagated to non-MASTER machines during system boot, the copies are stored on the non-MASTER machines according to the TUXCONFIG pathname values.

TUXDIR Environment Variable

The TUXDIR environment variable defines the product installation directory of the Oracle Tuxedo software on the MASTER machine. It must be set to an absolute pathname ending with the name of the product installation directory.

The TUXDIR pathname value is designated in the MACHINES section of the UBBCONFIG file. It is specified for the MASTER machine *and* for every other server machine in the Tuxedo domain.

Using the Registry to Further Restrict Access on a Windows System

Oracle Tuxedo-provided client programs are run directly by users with the users' own permissions. In addition, users running native clients (that is, clients running on the same machine on which the server program is running) have access to the UBBCONFIG file and interprocess communication (IPC) mechanisms such as the *bulletin board* (a reserved piece of shared memory in which parameters governing the application and statistics about the application are stored).

To gain access to the Oracle Tuxedo functionality, native clients join a Oracle Tuxedo application using the identity of the application administrator (`tpsysadm`). However, because `tpsysadm` is a trusted user, this setting causes the Oracle Tuxedo system to bypass the user authentication process.

To prevent this lapse in security on your Windows 2003 server machine, follow these steps:

1. Choose Start → Run to launch the Run dialog box, enter `regedt32`, and click OK to launch the Registry Editor window.

2. Select `HKEY_LOCAL_MACHINE` → `Software` → `Oracle Systems` → `Tuxedo` → `10.3.0.0_VS2005`.
3. Select `SECURITY` → `Permissions`.
4. Disable control for `Everyone` and allow access only to users with administrative privileges.

Setting Up Your Environment

You need to set several environment variables before using Oracle Tuxedo to build and run Oracle Tuxedo applications. [Table 6-2](#), [Table 6-3](#), [Table 6-4](#), [Table 6-5](#) list and define many of those environment variables.

Table 6-2 Oracle Tuxedo Core Environment Variables

Environment Variable	Description
TUXDIR	Absolute pathname of the product directory in which you installed the Oracle Tuxedo software on this machine. TUXDIR must be set on both server and client-only machines.
APPDIR	Absolute pathname of the application directory in which application and administrative servers will be booted on this server machine. APPDIR may be set to more than one application directory.
TUXCONFIG	Absolute pathname of the device or system file where the binary TUXCONFIG file is found on this server machine. The TUXCONFIG file is created by running the <code>tmloadcf(1)</code> command on the <code>UBBCONFIG</code> configuration file.
WEBJAVADIR	Absolute pathname of the Java and image files for the Oracle Tuxedo Administration Console on this server machine.

Table 6-3 Oracle Tuxedo Client-Only Environment Variables

Environment Variable	Description
WSENVFILE	Tuxedo ATMI Workstation (/WS) client: Name of the file in which all environment variables are set for this workstation. There is no default for this variable.
TOBJADDR	Tuxedo remote CORBA client: Address of the Tuxedo server machine's listener; must match exactly (including case) the host and port specified for the server machine in the UBBCONFIG file.

Table 6-4 COBOL Environment Variables

Environment Variable	Description
COBCPY	Directories that contain a set of the COBOL COPY files to be used by the compiler.
COBOPT	Arguments that you may want to use on the compile command line.

Table 6-5 Java Environment Variables

Environment Variable	Description
JAVA_HOME	Absolute pathname of the Java Development Kit (JDK) 1.5.x installation directory on this machine; needed to build and run Java applications on this machine.*
JDKDIR	Set to JAVA_HOME value.
CLASSPATH	Absolute pathnames to classes and Java archive files on this machine; needed to run Java applications on this machine.

* The Oracle Tuxedo 12c Release 1 (12.1.1) distribution does not include a JDK.

Setting Environment Variables on a Windows System

On a Windows 2003 server machine, you need to set the following environment variables shown in [Table 6-6](#) to set up your environment:

Table 6-6 Environment Variables

<code>set TUXDIR=pathname_of_Oracle_Tuxedo_product_directory</code>
<code>set APPDIR=pathname_of_Oracle_Tuxedo_application_directory</code>
<code>set TUXCONFIG=pathname_of_TUXCONFIG_file</code>
<code>set WEBJAVADIR=%TUXDIR%\udataobj\webgui\java</code>
<code>set PATH=%APPDIR%;%TUXDIR%\bin;\bin;%PATH%</code>

Examples of TUXDIR, APPDIR, and TUXCONFIG are:

```
TUXDIR=C:\Oracle\tuxedo12cR1
APPDIR=C:\home\me\simpapp
TUXCONFIG=%APPDIR%\tuxconfig
```

The TUXDIR, APPDIR, and TUXCONFIG environment variables must match the values of the TUXDIR, APPDIR, and TUXCONFIG parameters in the MACHINES section of the UBBCONFIG file. As an alternative to setting environment variables from a command-line shell, use the Environment page of the Oracle Administration program, described in [“Setting and Modifying Environment Variables”](#) in *Using Oracle Tuxedo ATMI on Windows*.

Windows 2003 Server accesses the required dynamically loadable library files through its PATH variable setting. Specifically, Windows 2000 searches for dynamically loadable library files in the following order:

1. The directory from which the Oracle Tuxedo application was loaded
2. The current directory
3. The Windows system directory (for example, C:\WINDOWS\System32)
4. The Windows directory (for example, C:\WINDOWS)
5. The directories listed in the PATH environment variable

For more information about setting environment variables, see “Setting and Modifying Environment Variables” in *Using Oracle Tuxedo ATMI on Windows*.

Setting Environment Variables on a UNIX System

On a UNIX server machine, set and export the following environment variables shown in [Table 6-7](#) to set up your environment:

Table 6-7 Environment Variables

<code>TUXDIR=pathname_of_Oracle_Tuxedo_product_directory</code>
<code>APPDIR=pathname_of_Oracle_Tuxedo_application_directory</code>
<code>TUXCONFIG=pathname_of_TUXCONFIG_file</code>
<code>WEBJAVADIR=\$TUXDIR/udataobj/webgui/java</code>
<code>PATH=\$APPDIR:\$TUXDIR/bin:/bin:\$PATH</code>
<code>LD_LIBRARY_PATH=\$APPDIR:\$TUXDIR/lib:/lib:/usr/lib:\$LD_LIBRARY_PATH</code>
Note: For HP-UX systems only, use <code>SHLIB_PATH</code> instead of <code>LD_LIBRARY_PATH</code> . For AIX systems only, <code>LIBPATH</code> instead of <code>LD_LIBRARY_PATH</code> .
<code>export TUXDIR APPDIR TUXCONFIG WEBJAVADIR PATH LD_LIBRARY_PATH</code>

Examples of `TUXDIR`, `APPDIR`, and `TUXCONFIG` are:

```
TUXDIR=/home/Oracle/tuxedo12cR1
APPDIR=/home/me/simpapp
TUXCONFIG=$APPDIR/tuxconfig
```

The `TUXDIR`, `APPDIR`, and `TUXCONFIG` environment variables must match the values of the `TUXDIR`, `APPDIR`, and `TUXCONFIG` parameters in the `MACHINES` section of the `UBBCONFIG` file. A Bourne shell script named `tux.env`, located in the Oracle Tuxedo product directory, serves as a model for setting these and other environment variables on a UNIX system.

Starting the tlisten Process

You, as the application administrator, must start a `tlisten` process on each machine of a networked Oracle Tuxedo application before the application is booted. The `tlisten` process enables you and the Oracle Tuxedo software running on the MASTER machine to start, shut down, and administer Oracle Tuxedo processes running on the non-MASTER machines. For example, `tmboot(1)` can start Oracle Tuxedo system servers on the non-MASTER machines. Generally, one `tlisten` process is required for each Oracle Tuxedo application running on a server machine.

In addition to the installer program starting a `tlisten` process on port 3050 during the installation of Oracle Tuxedo, a `tlisten` process may be started as shown in [tlisten Process Table 6-8](#).

Table 6-8 tlisten Process

On This Machine . . .	By This Administrator . . .	Using This Method . . .
Windows 2003 server	Oracle Tuxedo application administrator	Listener page of the Oracle Administration program; for details, see “Configuring tlisten Processes to Start Automatically” in <i>Using Oracle Tuxedo ATMI on Windows</i>
		Manually starting a <code>tlisten</code> process from a command-line shell
UNIX server	UNIX system administrator	As part of a UNIX initialization (boot) script
	Oracle Tuxedo application administrator	As a <code>cron</code> job
		Manually starting a <code>tlisten</code> process from a command-line shell

tlisten Invocation

In all cases, the same basic syntax is used to invoke `tlisten`.

```
%TUXDIR%\bin\tlisten -l nlsaddr [-u appuid] (Windows)
$TUXDIR/bin/tlisten [-d devname] -l nlsaddr [-u appuid] (UNIX)
```

The `-l` option is required. The argument to `-l` must match the value of the `NLSADDR` parameter in the `NETWORK` section of the `UBBCONFIG` file. For information on determining the value of `NLSADDR`, see reference page [UBBCONFIG\(5\)](#) in *OracleTuxedo File Formats, Data Descriptions, MIBs, and System Processes Reference*.

The value of `devname` is the device name of the network provider; for example, `Starlan`. If the `tlisten` process is operating with `Sockets`, the `-d` option is not needed.

The value of `appuid` is the user identifier (UID), or login name, of the Oracle Tuxedo application administrator. It must match the value of the `UID` parameter in the `RESOURCES` section of the `UBBCONFIG` file.

Note: To obtain the UID on a UNIX system, run the `id` command.

On a UNIX machine, use the `-u appuid` option when the command is part of an installation script run by user `root` to run the `tlisten` process with the effective UID of the owner of the Oracle Tuxedo software installation on this machine. If `tlisten` is started by the Oracle Tuxedo application administrator, either as a `cron` job or manually, the `-u` option is unnecessary since the job is already owned by the correct account.

For more information about the `tlisten` command, see reference page [tlisten\(1\)](#) in *Oracle Tuxedo Command Reference*. For details on starting the `tlisten` process on a Windows 2003 server machine, see “[Configuring tlisten Processes to Start Automatically](#)” in *Using Oracle Tuxedo ATMI on Windows*.

tlisten Password

Oracle Tuxedo uses the administrative password that you specified during the installation to protect the machine on which Oracle Tuxedo is installed from administrative requests and operations (such as `tmboot(1)`) that are not authorized. Whenever administrative communications arrive on this machine through `tlisten(1)` or `wlisten(1)` gateway processes, Oracle Tuxedo authenticates them by means of the password.

A `tlisten` password must be a string of alphanumeric characters in clear-text format. It may contain no more than 80 characters.

A common password is required for two machines in a Oracle Tuxedo application to communicate successfully. For this reason, you must use the same password whenever you install Oracle Tuxedo on multiple machines for a single application. If during the Oracle Tuxedo installation process you use a different password for one machine, you must add that password using the `tlistpwd(1)$TUXDIR` utility to the `tlisten.pw` file on each machine that communicates with eachother.

For these reasons, you may have more than one administrative password in your `tlisten.pw` file. A single password file may contain no more than 20 passwords, with one password per line.

Note: The `tlisten.pw` file is system-encrypted. To add or change passwords, you must use the `tlistpwd(1)` utility.

Editing a UBBCONFIG File

Each Oracle Tuxedo application is controlled by a configuration file in which installation-dependent parameters are defined. In the Oracle Tuxedo documentation, this file is referred to as UBBCONFIG, but you can give your configuration file any name you like as long as the content of the file conforms to the format described in reference page [UBBCONFIG\(5\)](#) in *OracleTuxedo File Formats, Data Descriptions, MIBs, and System Processes Reference*. Typical configuration filenames begin with the string `ubb`, followed by a mnemonic string, such as `simple` in the filename `ubbsimple`.

As an example, consider `ubbsimple`, the UBBCONFIG file for the rudimentary ATMI-based `simpapp` application delivered with the Oracle Tuxedo installation. On a Windows system, this application is found in the directory `%TUXDIR%\samples\atmi\simpapp`; on a UNIX system, it is found in the directory `$TUXDIR/samples/atmi/simpapp`.

[Listing 6-1](#) shows an example `ubbsimple` file. The example has been modified from the `ubbsimple` file delivered on a Windows or UNIX system to include example pathname values for both Windows and UNIX systems.

Listing 6-1 ubbsimple for the ATMI-based simpapp Application

```
#ident "@(#)apps:simpapp/ubbsimple    $Revision: 1.3 $

#Skeleton UBBCONFIG file for the Tuxedo Simple Application.
#Replace the <bracketed> items with the appropriate values.

*RESOURCES
IPCKEY          <Replace with a valid IPC Key>
```

```

#Example:
#IPCKEY          123456

DOMAINID        simpapp
MASTER          simple
MAXACCESSERS     10
MAXSERVERS       5
MAXSERVICES      10
MODEL            SHM
LDBAL            N

*MACHINES
DEFAULT:
APPDIR="<<Replace with the current directory pathname>"
TUXCONFIG="<<Replace with your TUXCONFIG Pathname>"
TUXDIR="<<Directory where Tuxedo is installed>"

#Windows
#Example:
#               APPDIR="C:\home\me\simpapp"
#               TUXCONFIG="C:\home\me\simpapp\tuxconfig"
#               TUXDIR="C:\oracle\tuxedo12cR1"

#UNIX
#Example:
#               APPDIR="/home/me/simpapp"
#               TUXCONFIG="/home/me/simpapp/tuxconfig"
#               TUXDIR="/home/oracle/tuxedo12cR1"

<Machine-name> LMID=simple

#Example:
#beatux         LMID=simple

*GROUPS
GROUP1          LMID=simple GRPNO=1 OPENINFO=NONE

*SERVERS
DEFAULT:
CLOPT="-A"
simplserv       SRVGRP=GROUP1 SRVID=1

*SERVICES
TOUPPER

```

In the configuration file for your application, you must replace the strings enclosed in angle brackets with values specific to your application. [Listing 6-9](#) provides a sample of the parameters that must be defined in every configuration file.

Table 6-9 Parameter Sample

This Parameter . . .	Specifies . . .
IPCKEY	A numeric key that identifies the shared memory segment where the structures used by your application are located. The value must be greater than 32,768 and less than 262,143.
<i>machine_name</i>	The node name of the machine. To obtain the node name on a Windows 2003 Server system, see your system administrator. To obtain the node name on a UNIX system, run the <code>uname -n</code> command.
<code>APPDIR = string</code>	<p>A list of one or more directories in which application and administrative servers will be booted on this machine.</p> <p>For Windows, the value of <i>string</i> is the absolute pathname of one directory, optionally followed by a semicolon-separated list of pathnames for other directories on the machine being defined.</p> <p>For UNIX, the value of <i>string</i> is the absolute pathname of one directory, optionally followed by a colon-separated list of pathnames for other directories on the machine being defined.</p>
<code>TUXCONFIG = string</code>	The absolute pathname of the device or system file where the binary TUXCONFIG file is to be created on this machine. The TUXCONFIG file is created by running the <code>tmloadcf (1)</code> command on the UBBCONFIG file.
<code>TUXDIR = string</code>	The absolute pathname of the product directory of the Oracle Tuxedo software on this machine.

You must define APPDIR, TUXCONFIG, and TUXDIR for every machine in your Oracle Tuxedo application. If you need to look up other parameters when editing your UBBCONFIG file, see reference page [UBBCONFIG\(5\)](#) in *OracleTuxedo File Formats, Data Descriptions, MIBs, and System Processes Reference*.

You must edit your UBBCONFIG file before running `tmloadcf (1)` to verify the IPC requirements in the section that follows. If you run `tmloadcf` without first editing the UBBCONFIG file, the command will fail with syntax errors.

Using the TYPE Parameter in UBBCONFIG

The `TYPE` parameter in the `MACHINES` section of a configuration file specifies the invocation of the External Data Representation (XDR) encode/decode routines when messages are passed between unlike machines. The term “unlike” applies even to machines of the same type if the compiler on each machine is different. In such a case, give each machine a unique `TYPE` string to force every message to go through the encode/decode routines.

Checking IPC Requirements

The Oracle Tuxedo system uses interprocess communications (IPC) resources heavily. On many systems, the default values for the parameters that control the size and quantity of the various IPC resources are below the minimums needed to run even a modest Oracle Tuxedo application. Therefore, you may need to reset some parameters. After editing your `UBBCONFIG` file, you should determine whether you have enough IPC resources for your application.

To perform this task, enter the following `tmloadcf(1)` command, specifying your edited `UBBCONFIG` file as input:

```
tmloadcf -c UBBCONFIG
```

With the `-c` option, the `tmloadcf` program prints a list of the minimum IPC resources required for your application, but does not create or update the `TUXCONFIG` file.

[Listing 6-2](#) is an output report based on the values in `ubbsimple`.

Listing 6-2 Output Produced by `tmloadcf -c`

Ipc sizing (minimum /T values only)...							
Fixed Minimums Per Processor							
SHMMIN: 1							
SHMALL: 1							
SEMMAP: SEMMNI							
Variable Minimums Per Processor							
	SEMUME ,		A				SHMMAX
	SEMMNU ,		*				*
Node	SEMMNS	SEMMSL	SEMMSL	SEMMNI	MSGMNI	MSGMAP	SHMSEG
-----	-----	-----	-----	-----	-----	-----	-----
sftuxe	17	5	12	A + 1	13	26	75K
where 1 <= A <= 8.							

The number of expected application clients per processor should be added to each MSGMNI value.

The output report identifies IPC resources by their traditional UNIX names. To map the traditional names to the names specific to a UNIX platform, see the data sheet for that platform in [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#). To map the traditional names to the names specific to the Windows 2003 Server platform, see the table titled [“IPC Resource Name Mappings Between Windows and UNIX Systems”](#) in *Using Oracle Tuxedo ATMI on Windows*.

The example output report indicates that to run `simpapp`, your system must have `SEMUME`, `SEMMNU`, and `SEMMNS` set to no less than 17. `SEMMSL` must be at least 5, and `SEMMNI` and `SEMMAP` at least 4 (assuming the value of `A` is 3). `MSGMNI` must be at least 13, and `MSGMAP` at least 26. Finally, the product of `SHMMAX` and `SHMSEG` must be at least 75K bytes.

The IPC values are application-dependent, and the numbers in this example reflect a very small configuration. If other client or server applications that use IPC resources are running on the same system with a Oracle Tuxedo application, then the requirements of both applications must be satisfied. Keep in mind also that every machine participating in an application must have sufficient IPC resources available.

If the current IPC resources are inadequate, you must increase the values of the associated IPC parameters. For instructions on changing the current IPC values for a Windows 2003 Server system, see [“Configuring IPC Resources to Maximize System Performance”](#) in *Using Oracle Tuxedo ATMI on Windows*. For instructions on changing the current IPC values for a UNIX system, see the data sheet for your platform in [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#).

Creating the Universal Device List and the Transaction Log

You create the Universal Device List (UDL) and define a UDL entry for the global transaction log (TLOG) on each machine in your application that will use global transactions. The TLOG is a log file in which information about transactions is kept until the transaction is completed.

Defining the TLOG

Before creating the UDL and defining UDL entries for TLOG, you must set the following parameters in the `MACHINES` section of the `UBBCONFIG` file for each machine in your application that will use global transactions.

[Table 6-10](#) lists the parameters.

Table 6-10 Parameter

This Parameter . . .	Specifies . . .
<code>TLOGDEVICE = string</code>	The Oracle Tuxedo filesystem containing the distributed transaction processing (DTP) TLOG for this machine. If not specified, it is assumed that this machine has no TLOG.
<code>TLOGOFFSET = offset</code>	The numeric offset in pages (from the beginning of the device) to the start of the Oracle Tuxedo filesystem containing the DTP transaction log for this machine. The default is 0.
<code>TLOGNAME = string</code>	The name of the DTP transaction log for this machine. If not specified, the default is “TLOG.”
<code>TLOGSIZE = size</code>	The numeric size, in pages, of the DTP transaction log for this machine. If not specified, the default is 100 pages.

Because the TLOG seldom needs to be larger than 100 blocks (pages) and because disk partitions are always substantially larger than that, it may make sense to use the same device for both the `TUXCONFIG` file and the TLOG. If so, the pathname of the device needs to be specified by both the `TUXCONFIG` and the `FSCONFIG` environment variables.

Creating the UDL and UDL Entries for TLOG

You must manually create a UDL entry for the `TLOGDEVICE` on each machine where a TLOG is needed. You may create these entries either before or after you have loaded `TUXCONFIG`, but you must create these entries before booting the application.

To access the create device list command, `crdl`, you invoke `tmadmin -c` with the application inactive. The `-c` option invokes `tmadmin` in configuration mode.

To create the UDL and a UDL entry for TLOG on each machine in your application that will use global transactions, follow these steps:

1. Log in as the application administrator on the MASTER machine.
2. Enter the following command:

```
tmadmin -c
crdl -z config -b blocks
```

Here `-z config` specifies the full pathname of the device on which the UDL should be created (that is, where the TLOG will reside), and `-b blocks` specifies the number of blocks to be allocated on the device. The value of `config` should match the value of the `TLOGDEVICE` parameter in the `MACHINES` section of the `UBBCONFIG` file. The blocks must be larger than the value of `TLOGSIZE`. If `-z` is not specified, the value of `config` defaults to the value of the `FSCONFIG` environment variable.

3. Log in as the application administrator on each remaining non-MASTER machine that will use global transactions and repeat step 2.

If the `TLOGDEVICE` is mirrored between two machines, step 3 is not required on the paired machine. To be recoverable, the TLOG should reside on a device that can be mirrored.

Running simpapp to Verify Your Installation

One of the ways to verify that your Oracle Tuxedo software is installed correctly is to run one or more of the sample applications included with the installation. The sample applications demonstrate the capabilities of the ATMI and CORBA clients, and the ATMI and CORBA C++ servers.

The following sections provide procedures for verifying both the ATMI and CORBA C++ parts of your Oracle Tuxedo installation:

- [Running simpapp to Verify the Oracle Tuxedo ATMI Software Installation](#)
- [Running simpapp to Verify the Oracle Tuxedo CORBA C++ Software Installation](#)

The `simpapp` application is a non-distributed application, meaning that it runs on a single machine. It is designed so that it can be up and running within minutes after the Oracle Tuxedo software is installed.

The `simpapp` application offers a single service called `TOUPPER`, which converts strings from lowercase to uppercase. The client is invoked with a single argument: a lowercase string to be converted to uppercase. The server returns the converted string to the client, and the client prints the converted string.

For example, the invocation

```
simpcl "hello world"
```

results in the output

```
Returned string is: HELLO WORLD
```

Two versions of `simpapp` exist: an ATMI version and a CORBA version. The ATMI version consists of an ATMI server, an ATMI client, and a `UBBCONFIG` file. The CORBA version consists of a CORBA C++ server, a CORBA C++ client.

Running simpapp to Verify the Oracle Tuxedo ATMI Software Installation

To verify that you have successfully installed the Oracle Tuxedo ATMI software on your system, run the ATMI version of the `simpapp` application, which on a Windows system can be found in the directory `%TUXDIR%\samples\atmi\simpapp`, and on a UNIX system can be found in the directory `$TUXDIR/samples/atmi/simpapp`.

Note: Samples are not installed unless they are selected to be installed during Tuxedo 12cR1 installation.

The procedure presented in the following two sections is also provided in the `README` file in the `simpapp` directory, and in “[Tutorial for simpapp, a Simple C Application](#)” in *Tutorials for Developing Oracle Tuxedo ATMI Applications*.

Running simpapp to Verify the Oracle Tuxedo ATMI Software Installation on a Windows System

To configure and run the ATMI version of `simpapp` on a Windows system, follow these steps:

1. Log in to the target machine as the Administrator or as a member of the Administrator group and open a command-line shell.
2. Create a working directory for your sample application and change to it:

```
cd C:\home\me
mkdir atmi
cd atmi
```

3. Set the environment variables used by the Oracle Tuxedo system, as explained in “[Setting Environment Variables on a Windows System](#)” on page 6-11. Set `APPDIR` and `TUXCONFIG` as follows:

```
set APPDIR=C:\home\me\atmi
set TUXCONFIG=%APPDIR%\tuxconfig
```

Note: You do not have to set the `WEBJAVADIR` environment variable.

4. Copy the `simpapp` files to your working directory. You will need to edit one of them: the configuration file, `ubbsimple`. Check the permissions on all the files in your working directory and, if necessary, change the permissions to allow full access. For example:

```
copy %TUXDIR%\samples\atmi\simpapp\*. * *.*
attrib -R /S *.*
```

5. Compile the `simpapp` client and server programs by entering the following commands:

```
buildclient -o simpcl -f simpcl.c
buildserver -o simpserv -f simpserv.c -s TOUPPER
```

6. In the sample configuration file, `ubbsimple`, replace the strings shown in angle brackets with values appropriate to your Oracle Tuxedo system installation. Comments in `ubbsimple` explain how to customize the file. Set the following parameters in the `ubbsimple` file:

- Set `IPCKEY` to a valid IPC key. This value must be greater than 32,768 and less than 262,143.
- Set `APPDIR` to "`C:\home\me\atmi`".
- Set `TUXCONFIG` to the literal pathname corresponding to `$APPDIR/tuxconfig` (in our example, "`C:\home\me\atmi\tuxconfig`").
- Set `TUXDIR` to the absolute pathname of the product directory of the Oracle Tuxedo software on this machine (for example, "`C:\oracle\tuxedo12cR1`").
- Set `MACHINE-NAME` to the name of your system. `MACHINE-NAME` must be entered in uppercase letters on a Windows system.

Note: The `APPDIR`, `TUXCONFIG`, and `TUXDIR` parameter settings in the `ubbsimple` file must match the `APPDIR`, `TUXCONFIG`, and `TUXDIR` environment variable settings.

7. Create the binary version of your edited configuration file by invoking `tmloadcf(1)`, which produces a file named `tuxconfig`. This file, referenced by the `TUXCONFIG` environment variable, provides the Oracle Tuxedo system with a description of the application configuration at run time:

```
tmloadcf -y ubbsimple
```

8. Boot `simpapp` by typing the following command:

```
tmboot -y
```

If the boot succeeds, output similar to the following appears and you can proceed to step 10.

[Listing 6-3](#) shows the output produced by `tmboot -y`.

Listing 6-3 Output Produced by `tmboot -y`

```
Booting all admin and server processes in C:\home\me\atmi\tuxconfig
INFO: Oracle Tuxedo(r) System Release 10.3.0.0
INFO: Serial #: 000102-9125503751, Maxusers 25
Booting admin processes ...
exec BBL -A:
    process id=24180 ... Started.
Booting server processes ...
exec simpserv -A :
    process id=24181 ... Started.
2 processes started.
```

9. If the boot fails, examine the log named `ULOG.mmdyy` in your application directory (`%APPDIR%`, `C:\home\me\atmi`). The string `mmdyy` is a placeholder for the date (digits representing the current month, day, and year) that will make up the end of the filename. If you see a message such as the following, near the end of the log

can't create enough semaphores for BB

then the interprocess communication (IPC) resources configured in your operating system are not adequate for running `simpapp`.

To confirm this hypothesis, invoke the Oracle Tuxedo system command `tmloadcf(1)` and specify the name of your configuration file, as shown in the following example:

```
tmloadcf -c %APPDIR%\ubbsimple
```

If the current value of any IPC parameter configured in your operating system is less than a minimum (either variable or fixed) listed in the `tmloadcf` output, you must increase the value of that parameter. For instructions on determining and changing the current IPC values for your platform, see [“Configuring IPC Resources to Maximize System Performance”](#) in *Using Oracle Tuxedo ATMI on Windows*.

10. If the boot succeeded, you can invoke the client. For example, enter the following command:

```
simpcl "hello world"
```

The following is displayed:

```
Returned string is: HELLO WORLD
```

11. When you have finished, shut down `simpapp` with the following command:

```
tmshutdown -y
```

Running simpapp to Verify the Oracle Tuxedo ATMI Software Installation on a UNIX System

To configure and run the ATMI version of `simpapp` on a UNIX system, follow these steps:

1. Log in to the target machine as the Oracle Tuxedo application administrator and open a command-line shell.
2. Create a working directory for your sample application and change to it:

```
cd /home/me
mkdir atmi
cd atmi
```

3. Set and export the environment variables used by the Oracle Tuxedo system, as explained in [“Setting Environment Variables on a UNIX System” on page 6-12](#). Set `APPDIR` and `TUXCONFIG` as follows:

```
APPDIR=/home/me/atmi
TUXCONFIG=$APPDIR/tuxconfig
export APPDIR TUXCONFIG
```

Note: You do not have to set the `WEBJAVADIR` environment variable.

4. Copy the `simpapp` files to your working directory. You will need to edit one of them: the configuration file, `ubbsimple`. Make sure that the client and server files, `simpcl` and `simpserv`, are executable, and that the configuration file, `ubbsimple`, is writable. For example:

```
cp $TUXDIR/samples/atmi/simpapp/* .
chmod 755 simpserv simpcl
chmod 644 ubbsimple
```

5. Compile the `simpapp` client and server programs by entering the following commands:

```
buildclient -o simpcl -f simpcl.c
buildserver -o simpserv -f simpserv.c -s TOUPPER
```

6. In the sample configuration file, `ubbsimple`, replace the strings shown in angle brackets with values appropriate to your Oracle Tuxedo system installation. Comments in `ubbsimple` explain how to customize the file. Set the following parameters in the `ubbsimple` file:

- Set `IPCKEY` to a valid IPC key. This value must be greater than 32,768 and less than 262,143.
- Set `APPDIR` to `"/home/me/atmi"`.
- Set `TUXCONFIG` to the literal pathname corresponding to `$APPDIR/tuxconfig` (in our example, `"/home/me/atmi/tuxconfig"`).
- Set `TUXDIR` to the absolute pathname of the product directory of the Oracle Tuxedo software on this machine (for example, `"/home/oracle/tuxedo12cR1"`).
- Set `MACHINE-NAME` to the name of your system.

To determine the name of your system on a UNIX machine enter the command:

```
uname -n
```

Note: The `APPDIR`, `TUXCONFIG`, and `TUXDIR` parameter settings in the `ubbsimple` file must match the `APPDIR`, `TUXCONFIG`, and `TUXDIR` environment variable settings.

7. Create the binary version of your edited configuration file by invoking `tmloadcf(1)`, which produces a file named `tuxconfig`. This file, referenced by the `TUXCONFIG` environment variable, provides the Oracle Tuxedo system with a description of the application configuration at run time:

```
tmloadcf -y ubbsimple
```

8. Boot `simpapp` by typing the following command:

```
tmboot -y
```

If the boot succeeds, output similar to the following appears and you can proceed to step 10.

[Listing 6-4](#) shows the output produced by `tmboot -y`.

Listing 6-4 Output Produced by `tmboot -y`

```
Booting all admin and server processes in /home/me/atmi/tuxconfig
INFO: Oracle Tuxedo(r) System Release 10.3.0.0
INFO: Serial #: 000102-9125503751, Maxusers 25
Booting admin processes ...
exec BBL -A:
    process id=24180 ... Started.
Booting server processes ...
exec simpserv -A :
```



```
process id=24181 ... Started.
2 processes started.
```

9. If the boot fails, examine the log named `ULOG.mmmddyy` in your application directory (`$APPDIR`, `/home/me/atmi`). The string `mmmddyy` is a placeholder for the date (digits representing the current month, day, and year) that will make up the end of the filename. If you see a message such as the following, near the end of the log

```
can't create enough semaphores for BB
```

then the interprocess communication (IPC) resources configured in your operating system are not adequate for running `simpapp`.

To confirm this hypothesis, invoke the Oracle Tuxedo system command `tmloadcf(1)` and specify the name of your configuration file, as shown in the following example:

```
tmloadcf -c $APPDIR/ubbsimple
```

If the current value of any IPC parameter configured in your operating system is less than a minimum (either variable or fixed) listed in the `tmloadcf` output, you must increase the value of that parameter. For instructions on determining and changing the current IPC values for your platform, see the data sheet for your platform in [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#).

10. If the boot succeeded, you can invoke the client. For example, enter the following command:

```
simpcl "hello world"
```

The following is displayed:

```
Returned string is: HELLO WORLD
```

11. When you have finished, shut down `simpapp` with the following command:

```
tmshutdown -y
```

Running simpapp to Verify the Oracle Tuxedo CORBA C++ Software Installation

To verify that you have successfully installed the Oracle Tuxedo CORBA C++ software on your system, run the CORBA version of the `simpapp` application, which on a Windows system can be found in the directory `%TUXDIR%\samples\corba\simpapp`, and on a UNIX system can be found in the directory `$TUXDIR/samples/corba/simpapp`. The procedure presented in the following two sections is also provided in the `README` file in the `simpapp` directory, and in

[“Tutorial for simpapp, a Simple C Application”](#) in *Tutorials for Developing Oracle Tuxedo ATMI Applications*.

Running simpapp to Verify the Oracle Tuxedo CORBA C++ Software Installation on a Windows System

To configure and run the CORBA version of `simpapp` on a Windows system, follow these steps:

1. Log in to the target machine as the Administrator or as a member of the Administrator group and open a command-line shell.
2. Create a working directory for your sample application and change to it:

```
cd C:\home\me
mkdir corba
cd corba
```

3. Make sure that the product directory in which you installed the Oracle Tuxedo software is set in the `TUXDIR` environment variable. For example, if you installed the software in the `C:\oracle\tuxedo12cR1` directory, set `TUXCONFIG` as follows:

```
set TUXDIR=C:\oracle\tuxedo12cR1
```

4. Copy the `simpapp` files to your working directory and change the permissions on all files to allow full access. For example:

```
copy %TUXDIR%\samples\corba\simpapp\*. * *. *
attrib -R /S *. *
```

5. Ensure that `nmake` is in your path.
6. To run `simpapp` automatically, enter `runme`. The `simpapp` application runs and prints the following messages:

```
Testing simpapp
  cleaned up
  prepared
  built
  loaded ubb
  booted
  ran
  shutdown
  saved results
PASSED
```

7. To run the sample manually to observe the `simpapp` processes starting and stopping, follow these steps:
 - a. Enter `results\setenv`.
 - b. Enter `tmboot -y`. The application starts several processes.
 - c. Enter `simple_client`. The prompt `String?` appears.
 - d. Enter a word in lowercase letters. The application converts the word to uppercase and then to lowercase letters.
 - e. Enter `tmshutdown -y`. The application shuts down the processes.
8. To restore the directory to its original state, perform these steps:
 - a. `results\setenv`
 - b. `nmake -f makefile.nt clean`

Running simpapp to Verify the Oracle Tuxedo CORBA C++ Software Installation on a UNIX System

To configure and run the CORBA version of `simpapp` on a UNIX system, follow these steps:

1. Log in to the target machine as the Oracle Tuxedo application administrator and open a command-line shell.
2. Create a working directory for your sample application and change to it:


```
prompt> cd /home/me
prompt> mkdir corba
prompt> cd corba
```
3. Make sure that the product directory in which you installed the Oracle Tuxedo software is set in the `TUXDIR` environment variable. For example, if you installed the software in the `/home/oracle/tuxedo12cR1` directory, set and export `TUXCONFIG` as follows:


```
prompt> TUXDIR=/home/oracle/tuxedo12cR1
prompt> export TUXDIR
```
4. Copy the `simpapp` files to your working directory and change the permissions on all files to allow full access. For example:


```
prompt> cp $TUXDIR/samples/corba/simpapp/* .
prompt> chmod 777 *
```
5. Ensure that `make` is in your path.

6. To run `simpapp` automatically, enter `../runme.ksh`. The `simpapp` application runs and prints the following messages:

```
Testing simpapp
  cleaned up
  prepared
  built
  loaded ubb
  booted
  ran
  shutdown
  saved results
PASSED
```

7. To run `simpapp` manually to observe the processes starting and stopping, follow these steps:

- a. `prompt> ksh`

- b. `prompt> ../results/setenv.ksh`

- c. `prompt> tmboot -y`

The application starts several processes.

- d. `prompt> simple_client`

The prompt `String?` is displayed.

- e. `prompt> enter_a_word_in_lowercase_letters`

The application converts the word to uppercase and then to lowercase letters and displays the results.

- f. `prompt> tmshutdown -y`

- g. The application shuts down the processes.

8. To restore the directory to its original state, follow these steps:

- a. `prompt> ../results/setenv.ksh`

- b. `prompt> make -f makefile.mk clean`

Running buildtms for Oracle Tuxedo Applications That Use XA Resource Managers

For Oracle Tuxedo applications that use distributed transactions and XA-compliant resource managers, you must use the `buildtms` command to construct a transaction manager server load module. This requirement exists on Windows 2003 Server and UNIX systems. When the module has been created, it must reside in `%TUXDIR%\bin` on Windows 2003 Server systems, or `$TUXDIR/bin` on UNIX systems.

If you run the CORBA C++ University sample applications, each sample's makefile creates the TMS load module for you and calls it `tms_ora.exe`. Therefore, running `buildtms` as a separate step is necessary only if you do not plan to run any of these sample applications.

For information about the `buildtms` command with Oracle Tuxedo applications, see reference page `buildtms(1)` in *Oracle Tuxedo Command Reference*.

Uninstalling Oracle Tuxedo

Uninstalling Oracle Tuxedo 12c Release 1 (12.1.1), 10.0, 9.1, 9.0, 8.1 or 8.0 does not remove the Oracle Home directory associated with the installation but does remove all Oracle Tuxedo components installed by the installer program. The uninstallation also removes the product directory associated with the installation unless one of the following is true:

- The product directory contains customer-created configuration or application files—the uninstallation does not delete customer-created configuration or application files.
- The uninstall was invoked from within the product directory structure—specifically, within the `uninstaller` directory.

To uninstall Oracle Tuxedo 12c Release 1 (12.1.1), complete the procedures for the appropriate platform, provided in [Table 6-11](#). To uninstall Oracle Tuxedo 9.1 or 10.0, substitute “Tuxedo 9.1 or 10.0” for “Tuxedo 12c Release 1 (12.1.1)” in [Table 6-11](#).

Table 6-11 Uninstall Oracle Tuxedo

To Uninstall Oracle Tuxedo on This Platform	Perform the Following Procedure . . .
...	
Windows	<div>1. Shut down any Oracle Tuxedo servers that are running. For instructions on using the <code>tmshutdown</code> command to shut down Oracle Tuxedo applications, see reference page <code>tmshutdown(1)</code> in <i>Oracle Tuxedo Command Reference</i>.</div> <div>2. From the Windows Start menu, choose Start → Programs → Oracle WebLogic E-Business Platform → Tuxedo 12c Release 1 (12.1.1) → uninstall.</div> <div>The Oracle Installation program Uninstaller window appears.</div> <div>3. Click Uninstall to start the uninstall program.</div> <div>4. Click Exit in the Uninstall Complete window.</div>
UNIX	<div>1. Shut down any Oracle Tuxedo servers that are running. For instructions on using the <code>tmshutdown</code> command to shut down Oracle Tuxedo applications, see reference page <code>tmshutdown(1)</code> in <i>Oracle Tuxedo Command Reference</i>.</div> <div>2. Go to the <code>tux_prod_dir/uninstaller</code> directory, where <code>tux_prod_dir</code> represents the product directory in which you installed the Oracle Tuxedo software.</div> <div>3. Choose one of two methods for uninstalling the software:<div><div>– To use the GUI-mode installation program, go to step 4.</div><div>– To use the console-mode procedure, go to step 5.</div></div></div> <div>4. (GUI-mode method) Enter the <code>sh Uninstall_Tuxedo_12cR1</code> command at the prompt. In the Uninstaller window, click Uninstall to start the uninstall program, then click Exit in the Uninstall Complete window to complete the uninstallation.</div> <div>5. (Console-mode method) Enter the <code>sh Uninstall_Tuxedo_12cR1 -i console</code> command at the prompt. When the uninstall process is complete, press Enter to exit the uninstaller.</div>

Reinstalling Oracle Tuxedo

When you start the Oracle Tuxedo Installation program on system that already has a copy of Oracle Tuxedo installed, the installation program detects the existing Oracle Tuxedo home directory and asks if you want to:

- Copy the new installation directory over the existing one
- Create a new installation directory

[Table 6-12](#) shows the reinstallation screen.

Table 6-12 Reinstalling Oracle Tuxedo

Click ...	To ...
Continue	Close the warning window and continue with the installation. This option overwrites the previous installation.
Cancel	Return to the Choose Oracle Home Directory window. To continue installing the software using a different Oracle Home directory, select an existing Oracle Home directory that does not contain the release 12c Release 1 (12.1.1) software or create a new Oracle Home directory.
Exit	Exit the installation program and uninstall the previous installation. You can invoke the uninstall program as described in “Uninstalling Oracle Tuxedo” on page 6-31 , and reinstall the software as described in one of the following: <ul style="list-style-type: none">• “Installing Oracle Tuxedo Using GUI-Mode Installation” on page 2-1• “Installing Oracle Tuxedo on UNIX Systems Using Console-Mode Installation” on page 3-1• “Installing Oracle Tuxedo Using Silent Installation” on page 4-1

Oracle Tuxedo 12c Release 1 (12.1.1) Platform Data Sheets

The following sections provide platform-specific information for the platforms on which the Oracle Tuxedo 12c Release 1 (12.1.1) system software is supported:

- [Supported Platforms](#)
 - [Platform Support Policy](#)
- [Supported Platform Data Sheets](#)
 - [IBM AIX 6.1 \(64-bit\) on IBM PowerPC](#)
 - [IBM AIX 7.1 \(64-bit\) on IBM PowerPC](#)
 - [Novell SUSE Linux Enterprise Server 11 \(64-bit\) on x86-64](#)
 - [Oracle Enterprise Linux 5.6 \(64-bit\) on x86-64](#)
 - [Oracle Linux 5.6 \(64-bit\) on Exallogic 2.0](#)
 - [Oracle Solaris 10 \(64-bit\) on x86-64](#)
 - [Oracle Solaris 10 \(64-bit\) on SPARC](#)
 - [Oracle Solaris 11 \(64-bit\) on x86-64](#)
 - [Oracle Solaris 11 \(64-bit\) on SPARC](#)
 - [Red Hat Enterprise Linux 5.6 \(64-bit\) on x86-64](#)
- [Platforms Supporting Threads](#)

Supported Platforms

Vendor	Operating System	Release/Version
IBM	AIX	IBM AIX 6.1 (64-bit) on IBM PowerPC
IBM	AIX	IBM AIX 7.1 (64-bit) on IBM PowerPC
Novell	Linux	Novell SUSE Linux Enterprise Server 11 (64-bit) on x86-64
Oracle	Linux	Oracle Linux 5.6 (64-bit) on Exalogic 2.0 Oracle Enterprise Linux 5.6 (64-bit) on x86-64
	Solaris	Oracle Solaris 10 (64-bit) on x86-64 Oracle Solaris 10 (64-bit) on SPARC Oracle Solaris 11 (64-bit) on x86-64 Oracle Solaris 11 (64-bit) on SPARC
Red Hat	Linux	Red Hat Enterprise Linux 5.6 (64-bit) on x86-64

Note: Oracle Jolt 12c Release 1 (12.1.1) and Oracle SNMP Agent 12c Release 1 (12.1.1) are supported on the same platforms that are supported by Oracle Tuxedo 12c Release 1 (12.1.1).

A data sheet is provided for each platform. Each data sheet includes the following platform-specific information:

- A list of available Oracle Tuxedo install sets
- Hardware, software, network, and disk space requirements
- Instructions for mounting and unmounting the Oracle Tuxedo software DVD
- Tunable parameters

Platform Support Policy

The Oracle policy regarding discontinuance of operating system platform support for Oracle Tuxedo and related products is as follows:

- x86 or x86-64 denotes various CPUs based on x86 or x86-64 architecture.
- Oracle Tuxedo 32-bit binaries are supported on 64-bit operating systems, such as on Solaris, AIX and Windows. This support is based on OS vendor compatibility assurance that 32-bit applications can run on 64-bit OS without any changes. For more 32-bit application support on 64-bit OS information, please refer to respective OS vendor documentation.
- Oracle Tuxedo is considered supported in virtualization software environments provided by the operating system vendor (such as Solaris Containers and AIX6 WPARs). Even though Oracle has performed limited testing in these environments, the support is based on the OS vendor compatibility statement that software applications will run in a virtualization software environment without any changes.

Other virtualization software environments (such as OVM, VirtualBox), are also considered supported, provided that the virtualization software environment is supported by the operating system vendor and the support criteria specified in the operating system data sheets are met.

For more information, see [Supported Platform Data Sheets](#).

- The Oracle policy is that when the operating system provider discontinues supporting a specific release, Oracle support is also discontinued.
- Oracle will not be able to notify customers and users when support for a specific operating system release is discontinued. As a user of the operating system release, we expect you to find out from the operating system provider of discontinued support and plan accordingly.

Platform End of Life Expectancy Table

The following is the End-of Life expectancy table for Oracle Tuxedo 12c Release 1 (12.1.1) supported platforms.

Note: OS EOL dates may have changed since time of this publication. Check with vendor for current OS EOL dates.

Platforms	Release Date	OS EOL Date
IBM AIX 6.1 (64-bit) on IBM PowerPC	2012.06.29	TBD
IBM AIX 7.1 (64-bit) on IBM PowerPC	2012.06.29	TBD
Novell SUSE Linux Enterprise Server 11 (64-bit) on x86-64	2012.06.29	TBD

Oracle Enterprise Linux 5.6 (64-bit) on x86-64	2012.06.29	TBD
Oracle Linux 5.6 (64-bit) on Exalogsic 2.0	2012.06.29	TBD
Oracle Solaris 10 (64-bit) on x86-64	2012.06.29	TBD
Oracle Solaris 10 (64-bit) on SPARC	2012.06.29	TBD
Oracle Solaris 11 (64-bit) on x86-64	2012.06.29	TBD
Oracle Solaris 11 (64-bit) on SPARC	2012.06.29	TBD
Red Hat Enterprise Linux 5.6 (64-bit) on x86-64	2012.06.29	TBD

EOL information for all Oracle products can be found at:

<http://www.oracle.com/support/library/brochure/lifetime-support-middleware.pdf>.

Tunable Parameters

You probably need to reconfigure the platform kernel before running Oracle Tuxedo software because the default values of some tunable parameters are too low.

To adjust the tunable parameters, follow these steps:

1. Determine whether the current values are adequate.

For instructions about determining whether the current tunable parameter values are adequate, see “[IPC Resource Configuration on a UNIX System](#)” on page E-1 and “[Checking IPC Requirements](#)” on page 6-18.

2. Reset the tunable parameters as necessary.

Supported Platform Data Sheets

IBM AIX 6.1 (64-bit) on IBM PowerPC

The following sections list Oracle Tuxedo 12c Release 1 (12.1.1) requirements for AIX 6.1 (64-bit) systems on IBM PowerPC.

Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for IBM AIX 6.1 (64-bit)

- Full install set
- Server install set
- Full client install set
- ATMI (/WS) client install set
- CORBA client install set
- Jolt client install set

For descriptions of Oracle Tuxedo install sets, see [“Install Sets” on page 1-9](#).

Hardware Requirements for IBM AIX 6.1 (64-bit)

- IBM PowerPC
- 1 GB of RAM minimum
- At least 5MB RAM for each Oracle Tuxedo system server

Software Requirements for IBM AIX 6.1 (64-bit)

Component	Requirement
OS Version (Patches)	IBM AIX 6.1 64-bit
C/C++ and COBOL compilers	C/C++: IBM XL C/C++ Enterprise Edition V10 or V11 for AIX or later compatible versions COBOL: Micro Focus 5.1
Java 2 JRE for the Java run-time environment	Tested with Java 2 JRE 1.6.0 and JRE 7
Java 2 Software Development Kit (SDK) for the Java development environment	Tested with JDK 1.6.0 and JRE 7

Component	Requirement
Non-Oracle CORBA Java clients	Tested with Java 2 SDK 1.6.0 (or higher) Interface Definition Language (IDL) ORB (run time)
Database for CORBA C++ applications	Tested with Oracle 11gR1
SSL certificate authorities	Verisign
Lightweight Directory Access Protocol (LDAP) directory servers	iPlanet Directory Server; needed to retrieve X.509v3 digital certificates for SSL

Additional Notes:

- If you use GAUTHSVER with JDK1.6 or later, please specify the following:
`JAVA_OPTS=-Djavax.xml.stream.XMLInputFactory=com.bea.xml.stream.MXParserFactory` in your environment before booting GAUTHSVR. If you use GAUTHSVER with Java 1.6, please add the 64-bit JRE library path to `LIBPATH`. The default library is located in “/usr/java6_64/jre/lib/ppc64”. Run `export LIBPATH=/usr/java6_64/jre/lib/ppc64:$LIBPATH` to set the correct `LIBPATH`.
- When using the Oracle MQ Adapter for Oracle Tuxedo in an AIX version 6.1 64-bit environment, if the `dlrmqlnk` command is not executed to remove the symbolic link to the 32-bit WebSphere MQ libraries, you must export the `TUXEDO_LIBPATH_PREPEND` environment variable and set its value to the 64-bit WebSphere MQ libraries path
- When using Oracle, Programmer/2000 Pro*C/C++ version 10.1.0.3.0 is required to build the Oracle Tuxedo University sample applications.
- ATMI users need a C, C++, or COBOL compiler.
- CORBA C++ users need a C++ compiler and linker.
- When developing CORBA C++ client/server applications, you must use `ifdef` directives to include standard C++ headers or old C headers, otherwise compiler will fail.

The applications use the `arco` "OBB_ANSI_CPP" macro to distinguish the two types of headers.

- If "OBB_ANSI_CPP" is defined, the standard C++ headers are included.
- If "OBB_ANSI_CPP" is undefined, the old C headers are included.

- For Oracle Tuxedo 56-bit, 128-bit, and 256-bit encryption:
 - LLE and SSL are available for Oracle Tuxedo ATMI client (/WS) connections to the Oracle Tuxedo Workstation Listener (WSL) or Workstation Handler (WSH).
LLE and SSL are available for Oracle Jolt client connections to the Oracle Tuxedo Jolt Server Listener (JSL) or Jolt Server Handler (JSH).
LLE and SSL support connections between machines and domains.
 - The Oracle Tuxedo Internet Inter-ORB Protocol (IIOP) Listener (ISL) and IIOP Handler (ISH) support SSL 3.0 for IIOP connections.
 - Oracle Tuxedo CORBA C++ clients support SSL 3.0. SSL connectivity between these clients, and the Oracle Tuxedo ISL/ISH has been certified.
- To support certificate-based authentication when using SSL, Oracle Tuxedo provides an LDAP-based certificate retrieval mechanism. This retrieval mechanism has been certified for use with the LDAP Directory server included with iPlanet Directory Server.
- For compiler compatibility:

Look for documentation from the compiler vendor that provides assurance that the version of compiler you want to use is compatible with the tested version of the compiler. The compatibility assurance must be provided for the following:

 - Source Code:

C/C++ source code that was compiled and linked using the tested version will compile with the newer version of compiler without requiring any changes.
 - Executables and libraries:

Libraries and executables built using the tested version will run with libraries and executables that are built with the newer version of compilers.

Network Requirements for IBM AIX 6.1 (64-bit)

- TCP/IP, using the Sockets network interface

Disk Space Requirements for IBM AIX 6.1 (64-bit)

For all install sets and components, 449,800KB is the minimum disk space requirement for installation on IBM AIX 6.1 (64-bit) IBM PowerPC systems. This requirement assumes installation of the default components for the selected install set.

Mounting and Unmounting the CD for AIX 6.1 (64-bit)

To mount a CD, examine the file `/etc/filesystems` to determine whether there is a standard place in which to mount a CD. If there is, enter the `mount` command and specify the directory named in the `/etc/filesystems` entry.

For example, to mount a CD when an entry in `/etc/filesystems` specifies `/cd` as the mount point, enter:

```
su
/usr/sbin/mount /cd
```

If `/etc/filesystems` does not contain a CD entry, enter:

```
su
mkdir /cd
/usr/sbin/mount -v cdrfs -r cd_device /cd
```

In the latter command line, `cd_device` represents the name of the CD device file, typically `/dev/cd0`.

Alternatively, you can use the System Management Interface Tool (SMIT) to perform the mount. To use SMIT, enter the following:

```
smit mount
```

To unmount the CD, enter the following command:

```
umount /cd
```

In this command line `cd` represents the mount point.

Tunable Parameters for IBM AIX 6.1 (64-bit)

No IPC configuration is required for AIX. To *check* the value of a kernel tuning parameter (maxuproc only) enter the following command: `lsattr -El sys0 -a maxuproc`.

To *change* the value of a kernel tuning parameter (maxuproc only) enter the following command: `chdev -l sys0 -a maxuproc=", , ,"` or can enter `smit chgsys` and select "maximum number of processes allowed per user."

IBM AIX 7.1 (64-bit) on IBM PowerPC

The following sections list Oracle Tuxedo 12c Release 1 (12.1.1) requirements for AIX 7.1 (64-bit) systems on IBM PowerPC.

Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for IBM AIX 7.1 (64-bit)

- Full install set
- Server install set
- Full client install set
- ATMI (/WS) client install set
- CORBA client install set
- Jolt client install set

For descriptions of Oracle Tuxedo install sets, see [“Install Sets” on page 1-9](#).

Hardware Requirements for IBM AIX 7.1 (64-bit)

- IBM PowerPC
- 1 GB of RAM minimum
- At least 5MB RAM for each Oracle Tuxedo system server

Software Requirements for IBM AIX 7.1 (64-bit)

Component	Requirement
OS Version (Patches)	IBM AIX 7.1 64-bit
C/C++ and COBOL compilers	C/C++: IBM XL C/C++ for AIX, V10 or V11 COBOL: Micro Focus 5.1, IBM COBOL for AIX V74.1, or other compatible COBOL compiler
Java 2 JRE for the Java run-time environment	Tested with Java 2 JRE 1.6.0 and JRE 1.7.0
Java 2 Software Development Kit (SDK) for the Java development environment	Tested with JDK 1.6.0 and JDK 1.7.0

Component	Requirement
Non-Oracle CORBA Java clients	Tested with Java 2 SDK 1.6.0 (or higher) Interface Definition Language (IDL) ORB (run time)
Database for CORBA C++ applications	Tested with Oracle Database 11gR2
SSL certificate authorities	Verisign
Lightweight Directory Access Protocol (LDAP) directory servers	iPlanet Directory Server; needed to retrieve X.509v3 digital certificates for SSL

Additional Notes:

- If you use GAUTHSVER with JDK1.6 or later, please specify the following:
`JAVA_OPTS=-Djavax.xml.stream.XMLInputFactory=com.bea.xml.stream.MXParserFactory` in your environment before booting GAUTHSVR. If you use GAUTHSVER with Java 1.6, please add the 64-bit JRE library path to `LIBPATH`. The default library is located in “/usr/java6_64/jre/lib/ppc64”. Run `export LIBPATH=/usr/java6_64/jre/lib/ppc64:$LIBPATH` to set the correct `LIBPATH`.
- When using Oracle, Programmer/2000 Pro*C/C++ version 10.1.0.3.0 is required to build the Oracle Tuxedo University sample applications.
- ATMI users need a C, C++, or COBOL compiler.
- CORBA C++ users need a C++ compiler and linker.
- When developing CORBA C++ client/server applications, you must use `ifdef` directives to include standard C++ headers or old C headers, otherwise compiler will fail.
The applications use the `arco` "OBB_ANSI_CPP" macro to distinguish the two types of headers.
 - If "OBB_ANSI_CPP" is defined, the standard C++ headers are included.
 - If "OBB_ANSI_CPP" is undefined, the old C headers are included.
- For Oracle Tuxedo 56-bit, 128-bit, and 256-bit encryption:
 - LLE and SSL are available for Oracle Tuxedo ATMI client (/WS) connections to the Oracle Tuxedo Workstation Listener (WSL) or Workstation Handler (WSH).

LLE and SSL are available for Oracle Jolt client connections to the Oracle Tuxedo Jolt Server Listener (JSL) or Jolt Server Handler (JSH).

LLE and SSL support connections between machines and domains.

- The Oracle Tuxedo Internet Inter-ORB Protocol (IIOP) Listener (ISL) and IIOP Handler (ISH) support SSL 3.0 for IIOP connections.
- Oracle Tuxedo CORBA C++ clients support SSL 3.0. SSL connectivity between these clients, and the Oracle Tuxedo ISL/ISH has been certified.
- To support certificate-based authentication when using SSL, Oracle Tuxedo provides an LDAP-based certificate retrieval mechanism. This retrieval mechanism has been certified for use with the LDAP Directory server included with iPlanet Directory Server.
- For compiler compatibility:

Look for documentation from the compiler vendor that provides assurance that the version of compiler you want to use is compatible with the tested version of the compiler. The compatibility assurance must be provided for the following:

- Source Code:

C/C++ source code that was compiled and linked using the tested version will compile with the newer version of compiler without requiring any changes.
- Executables and libraries:

Libraries and executables built using the tested version will run with libraries and executables that are built with the newer version of compilers.

Network Requirements for IBM AIX 7.1 (64-bit)

- TCP/IP, using the Sockets network interface

Disk Space Requirements for IBM AIX 7.1 (64-bit)

For all install sets and components, 430, 448KB is the minimum disk space requirement for installation on IBM AIX 7.1 (64-bit) IBM PowerPC systems. This requirement assumes installation of the default components for the selected install set.

Mounting and Unmounting the CD for AIX 7.1 (64-bit)

To mount a CD, examine the file `/etc/filesystems` to determine whether there is a standard place in which to mount a CD. If there is, enter the `mount` command and specify the directory named in the `/etc/filesystems` entry.

For example, to mount a CD when an entry in `/etc/filesystems` specifies `/cd` as the mount point, enter:

```
su
/usr/sbin/mount /cd
```

If `/etc/filesystems` does not contain a CD entry, enter:

```
su
mkdir /cd
/usr/sbin/mount -v cdrfs -r cd_device /cd
```

In the latter command line, `cd_device` represents the name of the CD device file, typically `/dev/cd0`.

Alternatively, you can use the System Management Interface Tool (SMIT) to perform the mount. To use SMIT, enter the following:

```
smit mount
```

To unmount the CD, enter the following command:

```
umount /cd
```

In this command line `cd` represents the mount point.

Tunable Parameters for IBM AIX 7.1 (64-bit)

No IPC configuration is required for AIX. To *check* the value of a kernel tuning parameter (`maxuproc` only) enter the following command: `lsattr -El sys0 -a maxuproc`.

To *change* the value of a kernel tuning parameter (`maxuproc` only) enter the following command: `chdev -l sys0 -a maxuproc=",,,,"` or can enter `smit chgsys` and select “maximum number of processes allowed per user.”

Novell SUSE Linux Enterprise Server 11 (64-bit) on x86-64

The following sections list Oracle Tuxedo 12c Release 1 (12.1.1) requirements for Novell SUSE Linux Enterprise Server 11 (64-bit) systems on x86-64

Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Novell SUSE Linux Enterprise Server 11 (64-bit)

- Full install set

- Server install set
- Full client install set
- ATMI client install set
- CORBA client install set
- Jolt client install set

For descriptions of Oracle Tuxedo install sets, see [“Install Sets” on page 1-9](#).

Hardware Requirements for Novell SUSE Linux Enterprise Server 11 (64-bit)

- x86-64
- 1 GB of RAM minimum
- At least 4MB RAM for each Oracle Tuxedo system server

Software Requirements for Novell SUSE Linux Enterprise Server 11 (64-bit)

Component	Requirement
OS Version (Patches)	Novell SUSE Linux Enterprise Server 11 x86-64 (64-bit)
C/C++ and COBOL compilers	C/C++: gcc/g++ 4.3.2 COBOL: Micro Focus 5.1, or other compatible COBOL compiler;
Java 2 JRE for the Java run-time environment	Tested with Java 2 Jrockit 1.6.0_31 and JRE 1.7.0_04
Java 2 Software Development Kit (SDK) for the Java development environment	Tested with Jrockit 1.6.0_31 and Jrockit1.7.0_04
Database for CORBA C++ applications	Tested with Oracle Database 11gR2
IBM WebSphere MQ for Oracle Tuxedo MQ Adapter	Tested with WebSphere MQ 7.0.1.1

Additional Notes

- The Kerberos Security Plug-in is not supported on this platform.
- ATMI users need a C, C++ or COBOL compiler.
- CORBA C++ users need a C++ compiler and linker.
- For compiler compatibility:

Look for documentation from the compiler vendor that provides assurance that the version of compiler you want to use is compatible with the tested version of the compiler. The compatibility assurance must be provided for the following:

- Source Code:

C/C++ source code that was compiled and linked using the tested version will compile with the newer version of compiler without requiring any changes.

- Executables and libraries:

Libraries and executables built using the tested version will run with libraries and executables that are built with the newer version of compilers.

Network Requirements for Novell SUSE Linux Enterprise Server 11 (64-bit)

- TCP/IP, using the Sockets network interface

Disk Space Requirements for Novell SUSE Linux Enterprise Server 11 (64-bit)

For all install sets and components, 192, 824 KB is the minimum disk space requirement for installation on Novell SUSE Linux Enterprise Server 11 (64-bit) x86-64 systems. This requirement assumes installation of the default components for the selected install set.

Mounting and Unmounting the CD for Novell SUSE Linux Enterprise Server 11 (64-bit)

Mount CD-ROM automatically.

Tunable Parameters for Novell SUSE Linux Enterprise Server 11 (64-bit)

See sysctl manpage.

Oracle Enterprise Linux 5.6 (64-bit) on x86-64

The following sections list Oracle Tuxedo 12c Release 1 (12.1.1) requirements for Oracle Enterprise Linux 6.1 (64-bit) on x86-64 systems.

Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Enterprise Linux 5.6 (64-bit)

- Full install set
- Server install set
- Full client install set
- ATMI client install set
- CORBA client install set
- Jolt client install set

For descriptions of Oracle Tuxedo install sets, see [“Install Sets” on page 1-9](#).

Hardware Requirements for Oracle Enterprise Linux 5.6 (64-bit)

- x86-64
- 1 GB of RAM minimum
- At least 4MB RAM for each Oracle Tuxedo system server

Software Requirements for Oracle Enterprise Linux 5.6 (64-bit)

Component	Requirement
OS Version (Patches)	Oracle Linux 5.6 (64-bit), kernel 2.6.18-238.el5xen
C/C++ and COBOL compilers	C/C++: gcc/g++ 4.4.4 COBOL: Micro Focus 5.1, or other compatible COBOL compiler.
Java 2 JRE for the Java run-time environment	Tested with Jrockit1.6.0_31 and JRE 1.7.0_04

Component	Requirement
Java 2 Software Development Kit (SDK) for the Java development environment	Tested with Jrockit1.6.0_29 and JDK 1.7.0_04
Database for CORBA C++ applications	Tested with Oracle Database 11gR2

Additional Notes

- The Direct Cross Node Communication Leveraging RDMA, Self-tuning Lock Mechanism, and SDP and Oracle Tuxedo SDP Support features are only supported on Exalogic. The TLOG Information To Oracle Database feature is supported on Exalogic and normal platforms.

For more information, see the [Oracle Tuxedo 12c Release 1 \(12.1.1\) Release Notes](#).

- The Kerberos Security Plug-in is not supported on this platform.
- ATMI users need a C, C++ or COBOL compiler.
- CORBA C++ users need a C++ compiler and linker.
- For COBOL-IT, you must create the softlink cob to the \$COBOL-IT_HOME/bin/cobmf utility.
- For compiler compatibility:

Look for documentation from the compiler vendor that provides assurance that the version of compiler you want to use is compatible with the tested version of the compiler. The compatibility assurance must be provided for the following:

- Source Code:

C/C++ source code that was compiled and linked using the tested version will compile with the newer version of compiler without requiring any changes.

- Executables and libraries:

Libraries and executables built using the tested version will run with libraries and executables that are built with the newer version of compilers.

Network Requirements for Oracle Enterprise Linux 5.6 (64-bit)

- TCP/IP, using the Sockets network interface

Disk Space Requirements for Oracle Enterprise Linux 5.6 (64-bit)

For all install sets and components, 193,464 KB is the minimum disk space requirement for installation on Oracle Enterprise Linux 5.6 (64-bit) systems. This requirement assumes installation of the default components for the selected install set.

Mounting and Unmounting the CD for Oracle Enterprise Linux 5.6 (64-bit)

Mount CD-ROM automatically.

Tunable Parameters for Oracle Enterprise Linux 5.6 (64-bit)

See `sysctl` manpage.

Oracle Linux 5.6 (64-bit) on Exallogic 2.0

The following sections list Oracle Tuxedo 12c Release 1 (12.1.1) requirements for Oracle Linux 5.6 (64-bit) on Exallogic 2.0 systems.

Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Linux 5.6 (64-bit)

- Full install set
- Server install set
- Full client install set
- ATMI client install set
- CORBA client install set
- Jolt client install set

For descriptions of Oracle Tuxedo install sets, see [“Install Sets” on page 1-9](#).

Hardware Requirements for Oracle Linux 5.6 (64-bit)

- x86-64

- 96 GB of RAM minimum
- At least 4MB RAM for each Oracle Tuxedo system server

Software Requirements for Oracle Linux 5.6 (64-bit)

Component	Requirement
OS Version (Patches)	Oracle Linux 5.6 (64-bit), kernel 2.6.32-200.21.1.el5uek
C/C++ and COBOL compilers	C/C++: gcc/g++ 4.1.2 20080704 COBOL: Micro Focus 5.1, or other compatible COBOL compiler.
Java 2 JRE for the Java run-time environment	Tested with JRockit 1.6 (1.6.0_20-20101014-1350-linux-x86_64) Tested with JRE 1.7 (1.7.0_04-b20)
Java 2 Software Development Kit (SDK) for the Java development environment	Tested with JRockit 1.6 (1.6.0_20-20101014-1350-linux-x86_64) Tested with JDK 1.7 (1.7.0_04-b20)
Database for CORBA C++ applications	Tested with Oracle Database 11.2.0.1.0

Additional Notes

- The Kerberos Security Plug-in is not supported on this platform.
- ATMI users need a C, C++ or COBOL compiler.
- CORBA C++ users need a C++ compiler and linker.
- For COBOL-IT, you must create the softlink `cob` to the `$COBOL-IT_HOME/bin/cobmf` utility .
- For compiler compatibility:

Look for documentation from the compiler vendor that provides assurance that the version of compiler you want to use is compatible with the tested version of the compiler. The compatibility assurance must be provided for the following:

- Source Code:

C/C++ source code that was compiled and linked using the tested version will compile with the newer version of compiler without requiring any changes.

- Executables and libraries:

Libraries and executables built using the tested version will run with libraries and executables that are built with the newer version of compilers.

Network Requirements for Oracle Linux 5.6 (64-bit)

- TCP/IP, using the Sockets network interface
- InfiniBand network, using 40Gb

Disk Space Requirements for Oracle Linux 5.6 (64-bit)

For all install sets and components, 198,319 KB is the minimum disk space requirement for installation on Oracle Linux 5.6 (64-bit) systems. This requirement assumes installation of the default components for the selected install set.

Mounting and Unmounting the CD for Oracle Linux 5.6 (64-bit)

Mount CD-ROM automatically.

Tunable Parameters for Oracle Linux 5.6 (64-bit)

See `sysctl` manpage.

Oracle Solaris 10 (64-bit) on x86-64

The following sections list Oracle Tuxedo 12c Release 1 (12.1.1) requirements for Oracle Solaris 10 (64-bit) systems on x86-64.

Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Solaris 10 (64-bit) on x86-64

- Full install set

- Server install set
- Full client install set
- ATMI client install set
- CORBA client install set
- Jolt client install set

For descriptions of Oracle Tuxedo install sets, see [“Install Sets” on page 1-9](#).

Hardware Requirements for Oracle Solaris10 (64-bit) on x86-64

- x86_64
- 1 GB of RAM minimum
- At least 7MB RAM for each Oracle Tuxedo system server

Software Requirements for Oracle Solaris10 (64-bit)

Component	Requirement
OS Version (Patches)	SunOS 5.10 (64-bit)
C/C++ and COBOL compilers	C/C++: Sun Studio 12 COBOL: Micro Focus 5.1, or other compatible COBOL compiler
Java 2 JRE for the Java run-time environment	Tested with Java 2 JRE 1.6.0_32 and JRE 1.7.0_04
Java 2 Software Development Kit (SDK) for the Java development environment	Tested with JDK 1.6.0_13 and JDK 1.7.0_04
Non Oracle CORBA Java clients	Tested with Java 2 SDK 1.6.0_13 Interface Definition Language (IDL) ORB (run time)
Database for CORBA C++ applications	Tested with Oracle Database 11gR2
SSL certificate authorities	Verisign

Component	Requirement
Lightweight Directory Access Protocol (LDAP) directory servers	iPlanet Directory Server; needed to retrieve X.509v3 digital certificates for SSL
IBM WebSphere MQ for Oracle Tuxedo MQ Adapter	Tested with WebSphere MQ 7.0.0.1

Additional Notes

- When Oracle is used, Programmer/2000 Pro*C/C++ version 10.1.0.3.0 is required to build the Oracle Tuxedo University sample applications.

- ATMI users need a C, C++, or COBOL compiler.

Note: To use Micro Focus 5.1, C/C++ compilers should use Sun Studio 12 or higher.

- CORBA C++ users need a C++ compiler and linker.
- For Oracle Tuxedo 56-bit, 128-bit, and 256-bit encryption:
 - LLE and SSL are available for Oracle Tuxedo ATMI client (/WS) connections to the Oracle Tuxedo Workstation Listener (WSL) or Workstation Handler (WSH).
LLE and SSL are available for Oracle Jolt client connections to the Oracle Tuxedo Jolt Server Listener (JSL) or Jolt Server Handler (JSH).
LLE and SSL support connections between machines and domains.
 - The Oracle Tuxedo Internet Inter-ORB Protocol (IIOP) Listener (ISL) and IIOP Handler (ISH) support SSL 3.0 for IIOP connections.
 - Oracle Tuxedo CORBA C++ client supports SSL 3.0. SSL connectivity between these clients, and the Oracle Tuxedo ISL/ISH has been certified.
- To support certificate-based authentication when using SSL, Oracle Tuxedo provides an LDAP-based certificate retrieval mechanism. This retrieval mechanism has been certified for use with the LDAP Directory server included with iPlanet Directory Server.
- For compiler compatibility:

Look for documentation from the compiler vendor that provides assurance that the version of compiler you want to use is compatible with the tested version of the compiler. The compatibility assurance must be provided for the following:

- Source Code:

C/C++ source code that was compiled and linked using the tested version will compile with the newer version of compiler without requiring any changes.

- Executables and libraries:

Libraries and executables built using the tested version will run with libraries and executables that are built with the newer version of compilers.

Network Requirements for Oracle Solaris10 (64-bit) on x86-64

TCP/IP using the Sockets network interface.

Disk Space Requirements for Oracle Solaris10 (64-bit) on x86-64

For all install sets and components, 435,192 KB is the minimum disk space requirement for installation on Oracle Solaris 10 (64-bit) x86-64 systems. This requirement assumes installation of the default components for the selected install set.

Mounting and Unmounting the CD for Oracle Solaris10 (64-bit) on x86-64

The Oracle Solaris Volume Management software automatically mounts CDs on `/cdrom/cdrom0/s0`.

It is not necessary to unmount CDs on Oracle Solaris systems. However, it is necessary to issue a command to open the CD reader. To open the CD reader, `cd` to `/` (root) and enter `eject`.

Tunable Parameters for Oracle Solaris10 (64-bit) on x86-64

Use the `mdb` tool to change the kernel parameters.

The tunable parameters currently set on your system are in the kernel configuration file located in the `/etc` directory (`/etc/system`).

Oracle Solaris 10 (64-bit) on SPARC

The following sections list Oracle Tuxedo 12c Release 1 (12.1.1) requirements for Oracle Solaris 10 (64-bit) systems on SPARC.

Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Solaris 10 (64-bit)

- Full install set

- Server install set
- Full client install set
- ATMI client install set
- CORBA client install set
- Jolt client install set

For descriptions of Oracle Tuxedo install sets, see [“Install Sets” on page 1-9](#).

Hardware Requirements for Oracle Solaris 10 (64-bit) on SPARC

- UltraSPARC
- 1 GB of RAM minimum
- At least 7MB RAM for each Oracle Tuxedo system server

Software Requirements for Oracle Solaris 10 (64-bit) on SPARC

Component	Requirement
OS Version (Patches)	SunOS 5.10 (64-bit)
C/C++ and COBOL compilers	C/C++: Sun Studio 12 COBOL: Micro Focus 5.1, or other compatible COBOL compiler
Java 2 JRE for the Java run-time environment	Tested with Java 2 JRE 1.6.0_32 and JRE 1.7.0_04
Java 2 Software Development Kit (SDK) for the Java development environment	Tested with JDK 1.6.0_32 and JRE 1.7.0_04
Non Oracle CORBA Java clients	Tested with Java 2 SDK 1.6.0_10 Interface Definition Language (IDL) ORB (run time)
Database for CORBA C++ applications	Tested with Oracle Database 10gR2
SSL certificate authorities	Verisign

Component	Requirement
Lightweight Directory Access Protocol (LDAP) directory servers	iPlanet Directory Server; needed to retrieve X.509v3 digital certificates for SSL
IBM WebSphere MQ for Oracle Tuxedo MQ Adapter	Tested with WebSphere MQ 7.0.0.1

Additional Notes

- When Oracle is used, Programmer/2000 Pro*C/C++ version 10.1.0.3.0 is required to build the Oracle Tuxedo University sample applications.

- ATMI users need a C, C++, or COBOL compiler.

Note: To use Micro Focus 5.1, C/C++ compilers should use Sun Studio 12 or higher.

- CORBA C++ users need a C++ compiler and linker.

- For Oracle Tuxedo 56-bit, 128-bit, and 256-bit encryption:

- LLE and SSL are available for Oracle Tuxedo ATMI client (/WS) connections to the Oracle Tuxedo Workstation Listener (WSL) or Workstation Handler (WSH).

LLE and SSL are available for Oracle Jolt client connections to the Oracle Tuxedo Jolt Server Listener (JSL) or Jolt Server Handler (JSH).

LLE and SSL support connections between machines and domains.

- The Oracle Tuxedo Internet Inter-ORB Protocol (IIOP) Listener (ISL) and IIOP Handler (ISH) support SSL 3.0 for IIOP connections.

- Oracle Tuxedo CORBA C++ client supports SSL 3.0. SSL connectivity between these clients, and the Oracle Tuxedo ISL/ISH has been certified.

- To support certificate-based authentication when using SSL, Oracle Tuxedo provides an LDAP-based certificate retrieval mechanism. This retrieval mechanism has been certified for use with the LDAP Directory server included with iPlanet Directory Server.

- For compiler compatibility:

Look for documentation from the compiler vendor that provides assurance that the version of compiler you want to use is compatible with the tested version of the compiler. The compatibility assurance must be provided for the following:

- Source Code:

C/C++ source code that was compiled and linked using the tested version will compile with the newer version of compiler without requiring any changes.

- Executables and libraries:

Libraries and executables built using the tested version will run with libraries and executables that are built with the newer version of compilers.

Network Requirements for Oracle Solaris 10 (64-bit) on SPARC

TCP/IP using the Sockets network interface.

Disk Space Requirements for Oracle Solaris 10 (64-bit) on SPARC

For all install sets and components, 447,686KB is the minimum disk space requirement for installation on Oracle Solaris 10 (64-bit) SPARC systems. This requirement assumes installation of the default components for the selected install set.

Mounting and Unmounting the CD for Oracle Solaris 10 (64-bit) on SPARC

The Oracle Solaris Volume Management software automatically mounts CDs on `/cdrom/cdrom0/s0`.

It is not necessary to unmount CDs on Oracle Solaris systems. However, it is necessary to issue a command to open the CD reader. To open the CD reader, `cd` to `/` (root) and enter `eject`.

Tunable Parameters for Oracle Solaris 10 (64-bit) on SPARC

Use the `mdb` tool to change the kernel parameters.

The tunable parameters currently set on your system are in the kernel configuration file located in the `/etc` directory (`/etc/system`).

Oracle Solaris 11 (64-bit) on x86-64

The following sections list Oracle Tuxedo 12c Release 1 (12.1.1) requirements for Oracle Solaris 11 (64-bit) systems on x86-64.

Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Solaris 11 (64-bit) on x86-64

- Full install set

- Server install set
- Full client install set
- ATMI client install set
- CORBA client install set
- Jolt client install set

For descriptions of Oracle Tuxedo install sets, see [“Install Sets” on page 1-9](#).

Hardware Requirements for Oracle Solaris11 (64-bit) on x86-64

- x86_64
- 1 GB of RAM minimum
- At least 7MB RAM for each Oracle Tuxedo system server

Software Requirements for Oracle Solaris11 (64-bit)

Component	Requirement
OS Version (Patches)	SunOS 5.11 (64-bit)
C/C++ and COBOL compilers	C/C++: Sun Studio 12 COBOL: Micro Focus 5.1, or other compatible COBOL compiler
Java 2 JRE for the Java run-time environment	Tested with Java 2 JRE 1.6.0_32 and JRE 1.7.0_04
Java 2 Software Development Kit (SDK) for the Java development environment	Tested with JDK 1.6.0_32 and JRE 1.7.0_04
Non Oracle CORBA Java clients	Tested with Java 2 SDK 1.6.0_24 Interface Definition Language (IDL) ORB (run time)
Database for CORBA C++ applications	Tested with Oracle Database 11gR2
SSL certificate authorities	Verisign

Component	Requirement
Lightweight Directory Access Protocol (LDAP) directory servers	iPlanet Directory Server; needed to retrieve X.509v3 digital certificates for SSL
IBM WebSphere MQ for Oracle Tuxedo MQ Adapter	Tested with WebSphere MQ 7.0.1.1

Additional Notes

- When Oracle is used, Programmer/2000 Pro*C/C++ version 10.1.0.3.0 is required to build the Oracle Tuxedo University sample applications.
- ATMI users need a C, C++, or COBOL compiler.
Note: To use Micro Focus 5.1, C/C++ compilers should use Sun Studio 12 or higher.
- CORBA C++ users need a C++ compiler and linker.
- For Oracle Tuxedo 56-bit, 128-bit, and 256-bit encryption:
 - LLE and SSL are available for Oracle Tuxedo ATMI client (/WS) connections to the Oracle Tuxedo Workstation Listener (WSL) or Workstation Handler (WSH).
 LLE and SSL are available for Oracle Jolt client connections to the Oracle Tuxedo Jolt Server Listener (JSL) or Jolt Server Handler (JSH).
 LLE and SSL support connections between machines and domains.
 - The Oracle Tuxedo Internet Inter-ORB Protocol (IIOP) Listener (ISL) and IIOP Handler (ISH) support SSL 3.0 for IIOP connections.
 - Oracle Tuxedo CORBA C++ client supports SSL 3.0. SSL connectivity between these clients, and the Oracle Tuxedo ISL/ISH has been certified.
- To support certificate-based authentication when using SSL, Oracle Tuxedo provides an LDAP-based certificate retrieval mechanism. This retrieval mechanism has been certified for use with the LDAP Directory server included with iPlanet Directory Server.
- For compiler compatibility:
 Look for documentation from the compiler vendor that provides assurance that the version of compiler you want to use is compatible with the tested version of the compiler. The compatibility assurance must be provided for the following:

- Source Code:

C/C++ source code that was compiled and linked using the tested version will compile with the newer version of compiler without requiring any changes.

- Executables and libraries:

Libraries and executables built using the tested version will run with libraries and executables that are built with the newer version of compilers.

Network Requirements for Oracle Solaris11 (64-bit) on x86-64

TCP/IP using the Sockets network interface.

Disk Space Requirements for Oracle Solaris11 (64-bit) on x86-64

For all install sets and components, 230,052 KB is the minimum disk space requirement for installation on Oracle Solaris 11 (64-bit) x86-64 systems. This requirement assumes installation of the default components for the selected install set.

Mounting and Unmounting the CD for Oracle Solaris11 (64-bit) on x86-64

The Oracle Solaris Volume Management software automatically mounts CDs on `/cdrom/cdrom0/s0`.

It is not necessary to unmount CDs on Oracle Solaris systems. However, it is necessary to issue a command to open the CD reader. To open the CD reader, `cd` to `/` (root) and enter `eject`.

Tunable Parameters for Oracle Solaris11 (64-bit) on x86-64

Use the `mdb` tool to change the kernel parameters.

The tunable parameters currently set on your system are in the kernel configuration file located in the `/etc` directory (`/etc/system`).

Oracle Solaris 11 (64-bit) on SPARC

The following sections list Oracle Tuxedo 12c Release 1 (12.1.1) requirements for Oracle Solaris 11 (64-bit) systems on SPARC.

Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Oracle Solaris 11 (64-bit)

- Full install set

- Server install set
- Full client install set
- ATMI client install set
- CORBA client install set
- Jolt client install set

For descriptions of Oracle Tuxedo install sets, see [“Install Sets” on page 1-9](#).

Hardware Requirements for Oracle Solaris 11 (64-bit) on SPARC

- SPARC
- 1 GB of RAM minimum
- At least 6MB RAM for each Oracle Tuxedo system server

Software Requirements for Oracle Solaris 11 (64-bit) on SPARC

Component	Requirement
OS Version (Patches)	SunOS 5.11
C/C++ and COBOL compilers	C/C++: Sun Studio 12 COBOL: Micro Focus 5.1, or other compatible COBOL compiler
Java 2 JRE for the Java run-time environment	Tested with Java 2 JRE 1.6.0_32 and JRE 1.7.0_04
Java 2 Software Development Kit (SDK) for the Java development environment	Tested with JDK 1.6.0_32 and JRE 1.7.0_04
Database for CORBA C++ applications	Tested with Oracle Database 11gR2

Additional Notes

- When Oracle is used, Programmer/2000 Pro*C/C++ version 10.1.0.3.0 is required to build the Oracle Tuxedo University sample applications.

- ATMI users need a C, C++, or COBOL compiler.

Note: To use Micro Focus 5.1, C/C++ compilers should use Sun Studio 12 or higher.

- CORBA C++ users need a C++ compiler and linker.
- For Oracle Tuxedo 56-bit, 128-bit, and 256-bit encryption:
 - LLE and SSL are available for Oracle Tuxedo ATMI client (/WS) connections to the Oracle Tuxedo Workstation Listener (WSL) or Workstation Handler (WSH).
LLE and SSL are available for Oracle Jolt client connections to the Oracle Tuxedo Jolt Server Listener (JSL) or Jolt Server Handler (JSH).
LLE and SSL support connections between machines and domains.
 - The Oracle Tuxedo Internet Inter-ORB Protocol (IIOP) Listener (ISL) and IIOP Handler (ISH) support SSL 3.0 for IIOP connections.
 - Oracle Tuxedo CORBA C++ client supports SSL 3.0. SSL connectivity between these clients, and the Oracle Tuxedo ISL/ISH has been certified.
- To support certificate-based authentication when using SSL, Oracle Tuxedo provides an LDAP-based certificate retrieval mechanism. This retrieval mechanism has been certified for use with the LDAP Directory server included with iPlanet Directory Server.
- For compiler compatibility:

Look for documentation from the compiler vendor that provides assurance that the version of compiler you want to use is compatible with the tested version of the compiler. The compatibility assurance must be provided for the following:

- Source Code:
C/C++ source code that was compiled and linked using the tested version will compile with the newer version of compiler without requiring any changes.
- Executables and libraries:
Libraries and executables built using the tested version will run with libraries and executables that are built with the newer version of compilers.

Network Requirements for Oracle Solaris 11 (64-bit) on SPARC

TCP/IP using the Sockets network interface.

Disk Space Requirements for Oracle Solaris 11 (64-bit) on SPARC

For all install sets and components, 224,355 KB is the minimum disk space requirement for installation on Oracle Solaris 11 (64-bit) SPARC systems. This requirement assumes installation of the default components for the selected install set.

Mounting and Unmounting the CD for Oracle Solaris 11 (64-bit) on SPARC

The Oracle Solaris Volume Management software automatically mounts CDs on `/cdrom/cdrom0/s0`.

It is not necessary to unmount CDs on Oracle Solaris systems. However, it is necessary to issue a command to open the CD reader. To open the CD reader, `cd` to `/` (root) and enter `eject`.

Tunable Parameters for Oracle Solaris 11 (64-bit) on SPARC

Use the `mdb` tool to change the kernel parameters.

The tunable parameters currently set on your system are in the kernel configuration file located in the `/etc` directory (`/etc/system`).

Red Hat Enterprise Linux 5.6 (64-bit) on x86-64

The following sections list Oracle Tuxedo 12c Release 1 (12.1.1) requirements for Red Hat Enterprise Linux 5.6 (64-bit) on x86-64.

Available Oracle Tuxedo 12c Release 1 (12.1.1) Install Sets for Red Hat Enterprise Linux 5.6 (64-bit)

- Full install set
- Server install set
- Full client install set
- ATMI client install set
- CORBA client install set
- Jolt client install set

For descriptions of Oracle Tuxedo install sets, see [“Install Sets” on page 1-9](#).

Hardware Requirements for Red Hat Enterprise Linux 5.6 (64-bit)

- x86-64
- 1 GB of RAM minimum
- At least 4MB RAM for each Oracle Tuxedo system server

Software Requirements for Red Hat Enterprise Linux 5.6 (64-bit)

Component	Requirement
OS Version (Patches)	Red Hat Enterprise Linux 5.6 x86 64-bit
C/C++ and COBOL compilers	C/C++: gcc 4.1.2 COBOL: Micro Focus 5.1, or other compatible COBOL compiler
Java 2 JRE for the Java run-time environment	Tested with Jrockit 2 JRE 1.6.0_31 and JRE1.7.0_04
Java 2 Software Development Kit (SDK) for the Java development environment	Tested with Jrockit 1.6.0_31 and JDK1.7.0_04
Database for CORBA C++ applications	Tested with Oracle Database 11gR2
IBM WebSphere MQ for Oracle Tuxedo MQ Adapter	Tested with WebSphere MQ 7.0.0.1

Additional Notes

- The Kerberos Security Plug-in is not supported on this platform.
- ATMI users need a C, C++ or COBOL compiler.
- CORBA C++ users need a C++ compiler and linker.
- For compiler compatibility:

Look for documentation from the compiler vendor that provides assurance that the version of compiler you want to use is compatible with the tested version of the compiler. The compatibility assurance must be provided for the following:

- Source Code:

C/C++ source code that was compiled and linked using the tested version will compile with the newer version of compiler without requiring any changes.

- Executables and libraries:

Libraries and executables built using the tested version will run with libraries and executables that are built with the newer version of compilers.

Network Requirements for Red Hat Enterprise Linux 5.6 (64-bit)

- TCP/IP, using the Sockets network interface

Disk Space Requirements for Red Hat Enterprise Linux 5.6 (64-bit)

For all install sets and components, 200,840 KB is the minimum disk space requirement for installation on Red Hat Enterprise Linux Server Release 5.6 (64-bit) on x86-64 systems. This requirement assumes installation of the default components for the selected install set.

Mounting and Unmounting the CD for Red Hat Enterprise Linux 5.6 (64-bit)

Mount CD-ROM automatically.

Tunable Parameters for Red Hat Enterprise Linux 5.6 (64-bit)

See `sysctl` manpage.

Platforms Supporting Threads

The following platforms support threads:

- [IBM AIX 6.1 \(64-bit\) on IBM PowerPC](#)
- [IBM AIX 7.1 \(64-bit\) on IBM PowerPC](#)
- [Novell SUSE Linux Enterprise Server 11 \(64-bit\) on x86-64](#)
- [Oracle Linux 5.6 \(64-bit\) on Exalogic 2.0](#)
- [Oracle Enterprise Linux 5.6 \(64-bit\) on x86-64](#)

- Oracle Solaris 10 (64-bit) on x86-64
- Oracle Solaris 10 (64-bit) on SPARC
- Oracle Solaris 11 (64-bit) on x86-64
- Oracle Solaris 11 (64-bit) on SPARC
- Red Hat Enterprise Linux 5.6 (64-bit) on x86-64

Note: If threads are not supported on your platform, your application must either:
(1) exclude threads or (2) serialize threaded access through all Oracle Tuxedo system calls.

Starting the Oracle Tuxedo Administration Console

The following sections provide the system requirements for the Oracle Tuxedo Administration Console and explain how to start and exit the Console.

- [What is the Oracle Tuxedo Administration Console?](#)
- [Administration Console File Tree](#)
- [Server Requirements](#)
- [Browser Requirements](#)
- [Hardware Requirements](#)
- [Setting Up Your Environment for the Oracle Tuxedo Administration Console](#)
- [Starting the Oracle Tuxedo Administration Console](#)
- [Exiting the Oracle Tuxedo Administration Console](#)

What is the Oracle Tuxedo Administration Console?

The Oracle Tuxedo Administration Console is a graphical user interface that enables administrators to perform most administration and configuration tasks for Oracle Tuxedo

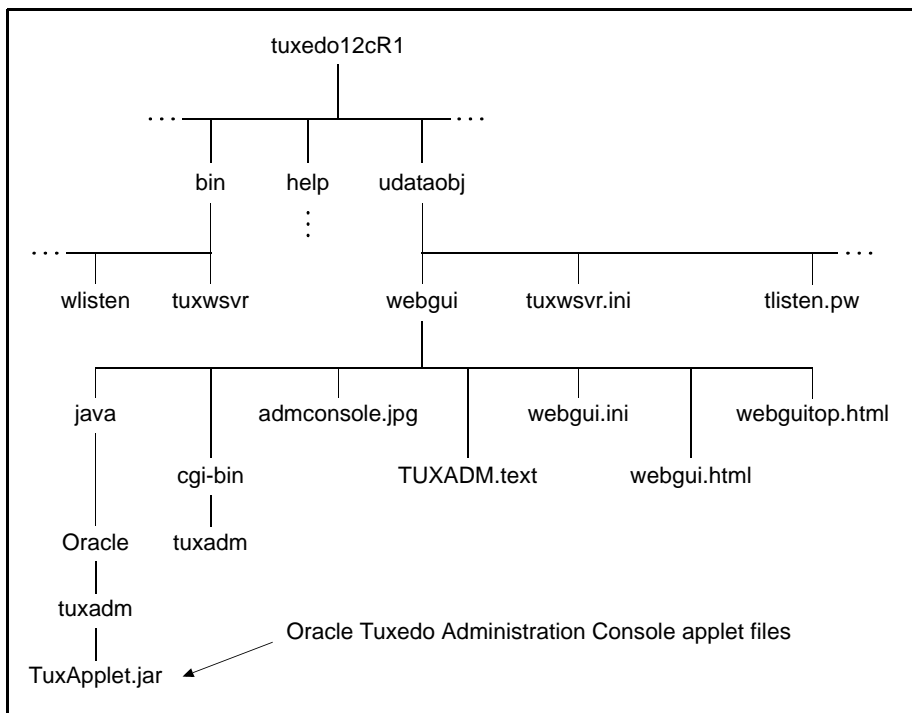
applications. It is implemented as a set of Java applets, which can run on most platforms that support a Java-capable Web browser.

The server-side components of the Oracle Tuxedo Administration Console reside on one of the server machines in a Oracle Tuxedo application. To use the Console, you must enter the URL of the server and download the Java applets.

Administration Console File Tree

During the Oracle Tuxedo software installation, the installer program places most of the directories and files for the Oracle Tuxedo Administration Console in the `webgui` directory, as shown in [Figure 1-1](#).

Figure 1-1 Oracle Tuxedo Administration Console File Tree



The installer program installs the following HTML files:

- An HTML template file named `webgui.html`, which is used by the common gateway interface (CGI) program `tuxadm` as the basis for many screens displayed during a Oracle Tuxedo Administration Console session.
- An HTML file named `webguitop.html`, which displays legal notices and warnings when the Oracle Tuxedo Administration Console is first brought up on the screen.
- The HTML documentation files for the Oracle Tuxedo Administration Console, which are installed in a top-level directory named `help`.

The installer program installs the class files for the Java applet in the `java` directory, and installs the `tuxadm` program in the `cgi-bin` directory. The installer assigns an alias pathname for `tuxadm`, which is used by Web clients to access `tuxadm`. The alias pathname is `\cgi-bin` on a Windows 2003 Server system and `/cgi-bin` on a UNIX system.

Server Requirements

The Oracle Tuxedo Administration Console server for Oracle Tuxedo 11g Release 1 (11.1.1.2.0) is supported on the following platforms:

- HP-UX 11i v2 (64-bit) on Itanium
- 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
- IBM AIX 6.1 (32-bit) on IBM PowerPC
- IBM AIX 6.1 (64-bit) on IBM PowerPC
- Microsoft Windows 2003 Server (32-bit) on x86 with MS Visual Studio 2005 Professional Edition
- Microsoft Windows XP on x86 (Client Only)
- Novell SUSE Linux Enterprise Server 10 (64-bit) on x86-64
- Oracle Enterprise Linux 5.0 (64-bit)
- Oracle Solaris 10 (32-bit) on SPARC
- Oracle Solaris 10 (64-bit) on x86-64
- Oracle Solaris 10 (64-bit) on SPARC

Browser Requirements

The encryption level for the Oracle Tuxedo Administration Console is set with the `ENCRYPTBITS` parameter in the `webgui.ini` file for the Console. In the following `webgui.ini` file, the encryption level is set to 56 bits.

```
# Web GUI initialization file.
# Created Sep 12, 2002 2:02:07 PM by Oracle software installation program.
#
TUXDIR=C:\oracle\tuxedo12cR1
INIFILE=C:\oracle\tuxedo12cR1\udataobj\webgui\webgui.ini
NADDR=//foo5:4003
DEVICE=/dev/tcp
CODEBASE=/java
DOCBASE=/doc
SNAPDIR=C:\oracle\tuxedo12cR1\udataobj\webgui\java\snapshot
SNAPBASE=/java/snapshot
ENCRYPTBITS=56
```

The `ENCRYPTBITS` parameter specifies the strength of encryption used in communication between the GUI applet and the Oracle Tuxedo Administration Console server. The `ENCRYPTBITS` parameter can be set to 0, 56, or 128. The default value is 128.

[Table 1-1](#) lists the platforms and browsers supported for Oracle Tuxedo 11g Release 1 (11.1.1.2.0) when the Oracle Tuxedo Administration Console is configured for 56-bit or 128-bit encryption.

Table 1-1 Platforms and Browsers

On this platform . . .	Oracle tested these browsers . . .	Using Java plug-in . . .
HP-UX Version 11.23 (32-bit) on PA-RISC		
IBM AIX 5.3 (32-bit) on IBM PowerPC		
Microsoft Windows 2008 Server (32-bit) on x86	Internet Explorer 6.0	1.4.x
Microsoft Windows XP on x86	Internet Explorer 6.0	1.4.x
Oracle Solaris 10(32-bit) on SPARC		

If the Oracle Tuxedo Administration Console is configured with no encryption (0-bit encryption), Internet Explorer 6.0 is supported on any Windows platform regardless of the Java plug-in version.

Hardware Requirements

To run the Oracle Tuxedo Administration Console, you need a color display device that meets the following requirements:

- Resolution: 800 by 600 or more is required; 1024 by 768 or more is recommended.
- Colors: A minimum of 256 colors is recommended.

Setting Up Your Environment for the Oracle Tuxedo Administration Console

To run the Oracle Tuxedo Administration Console, first set up your environment as described in [“Setting Up Your Environment” on page 6-9](#). Ensure that your TUXDIR, WEBJAVADIR, and PATH environment variables are set correctly. Then set up the following two server processes:

- `tuxwsvr(1)`
A Web server provided with the Oracle Tuxedo system software. You are not required to use this server; you may, if you prefer, use your own commercial Web server.
- `wlisten(1)`
A server required to administer the Oracle Tuxedo Administration Console. It must be run on the MASTER machine in a Tuxedo multi-machine configuration.

After starting the `tuxwsvr` and `wlisten` server processes, you can start the Oracle Tuxedo Administration Console to monitor the `tuxwsvr` server and the Oracle Tuxedo application.

Starting tuxwsvr

The `tuxwsvr` process is a Web server process that can be used to support the Oracle Tuxedo Administration Console GUI by customers who do not have a commercial Web server or a public-domain Web server on the machine on which the Oracle Tuxedo Web GUI processes are running. The `tuxwsvr` process places itself in the background when invoked unless otherwise specified, and continues running until the machine shuts down or the `tuxwsvr` process is killed using an operating system command.

[Table 1-2](#) lists the commands for starting `tuxwsvr`.

Table 1-2 Starting tuxsvr

To Start Tuxwsvr on This Platform . . .	Enter the Following Command . . .
Windows 2003 Server system	<code>tuxwsvr -l //machine:port -i %TUXDIR%\udataobj\tuxwsvr.ini</code>
UNIX system	<code>tuxwsvr -l //machine:port -i \$TUXDIR/udataobj/tuxwsvr.ini</code>

The Oracle Tuxedo installer program creates the `tuxwsvr.ini` file, an example of which is shown below:

```
# tuxwsvr initialization file.
# Created Sep 12, 2008 2:02:07 PM by Oracle software installation program.
#
CGI      /cgi-bin      C:\oracle\tuxedo12cR1\udataobj\webgui/cgi-bin
HTML     /java        C:\oracle\tuxedo12cR1\udataobj\webgui/java
HTML     /doc         C:\oracle\tuxedo12cR1\help
HTML     /            C:\oracle\tuxedo12cR1\udataobj\webgui
```

Usually, you do not need to edit this file, but under certain circumstances, you may want to do so. For example, you may decide to move your Java files to a non-default directory. In that case, you would need to edit the pathnames in the initialization file appropriately. For details, see reference page `tuxwsvr(1)` in *Oracle Tuxedo Command Reference*.

Starting wlisten

The `wlisten` process is a listener process that receives incoming connections from Web GUI applets and starts a Web GUI gateway process (`wgated`). All `wlisten` options are taken from an initialization file that is specified by the `-i` option. If the `-i` option is not given, then `%TUXDIR%\udataobj\webgui\webgui.ini` is used as the default initialization file on a Windows 2003 Server system, and `$TUXDIR/udataobj/webgui/webgui.ini` is used as the default initialization file on a UNIX system.

To start `wlisten`, follow these steps:

1. Check the `webgui.ini` file to make sure that the default values assigned to the parameters during installation are appropriate. If they are not, make the necessary changes.

For example, on a machine called `foo5`, the default port assigned to `wlisten` is 4003. To run `wlisten` with port 6060, edit the `NADDR` parameter as follows:

```
NADDR=//foo5:6060
```

For details about other parameters in the `webgui.ini` file, see reference page `wlisten(1)` in *Oracle Tuxedo Command Reference*.

2. Enter the following command:

```
wlisten
```

Starting the Oracle Tuxedo Administration Console

The `tuxadm` process is a CGI process used to initialize the Web GUI from a browser. As shown in the “Synopsis” section of reference page `tuxadm(1)`, this program can be used only as a location, or URL from a Web browser; normally it is not executed from a standard command-line prompt. Like other CGI programs, `tuxadm` uses the `QUERY_STRING` environment variable to parse its argument list.

To start the Oracle Tuxedo Administration Console, complete the following four-step procedure:

1. Start the browser.
2. Enter the following URL:

```
http://machine:port/webguitop.html
```

For example: `http://foo5:4003/webguitop.html`

Use of this URL depends on the following assumption: You are using `tuxwsvr` with the `tuxwsvr.ini` file. If you are using a commercial browser on the default port (8080), you can use a URL such as `http://ctomsn:8080/webguitop.html`.

The Oracle Tuxedo Administration Console entry page-displays.

3. To start the Oracle Tuxedo Administration Console, select the Click Here to Run the Oracle Tuxedo Administration Console prompt at the bottom of the screen. The Login window displays.
4. Enter your login name and password in the appropriate fields, and select LOGIN. The password must correspond to one of the entries in the system-encrypted `tlisten.pw` file.

The main window of the Oracle Tuxedo Administration Console displays.

[Table 1-3](#) contains instructions for accessing additional information about the Oracle Tuxedo Administration Console main window.

Table 1-3 Accessing Information About the Oracle Tuxedo Administration Console Main Window

If. . .	Then. . .
The main window is displayed and you want to start working with the GUI	See “Administration Console Tutorial” in Oracle Tuxedo Administration Console Online Help .

Table 1-3 Accessing Information About the Oracle Tuxedo Administration Console Main Window

If. . .	Then. . .
The main window is displayed and you want to read a description of it	See “A Tour of the Main Window” in Oracle Tuxedo Administration Console Online Help .
The main window is not displayed and the Connect Failed error message is displayed	<ol style="list-style-type: none"> 1. On the Administration Console <i>server</i> machine, verify that the <code>wlisten</code> process is running. On a UNIX machine, for example, enter the <code>ps</code> command. 2. If <code>wlisten</code> is not running, open the <code>webgui.ini</code> file and, in the line <code>NADDR=//foo5:4003</code>, replace the port number (4003) with a valid port number. 3. Enter <code>wlisten</code> again: <ul style="list-style-type: none"> • <code>\$ wlisten -i %TUXDIR%\udataobj\webgui\webgui.ini</code> (on Windows 2003 Server) • <code>\$ wlisten -i \$TUXDIR/udataobj/webgui/webgui.ini</code> (on UNIX) 4. Check that the <code>tuxwsvr</code> process is running at the port specified in the URL. 5. Verify the password. It must match one of the entries in the <code>tlisten.pw</code> file. 6. Return to step 1.

Limitation(s)

The Oracle Tuxedo Administration Console has not been updated to support any new features introduced after Oracle Tuxedo release 7.1.

Exiting the Oracle Tuxedo Administration Console

To exit the Oracle Tuxedo Administration Console, choose Domain → Exit from the menu bar. This menu option closes the current domain and exits the Oracle Tuxedo Administration Console applet.

Oracle Jolt 12*c* Release 1 (12.1.1)

Overview and Installation Information

The following sections provide an overview of Oracle Jolt 12*c* Release 1 (12.1.1) and present the prerequisites and preparatory information for installing the Oracle Jolt 12*c* Release 1 (12.1.1) software components:

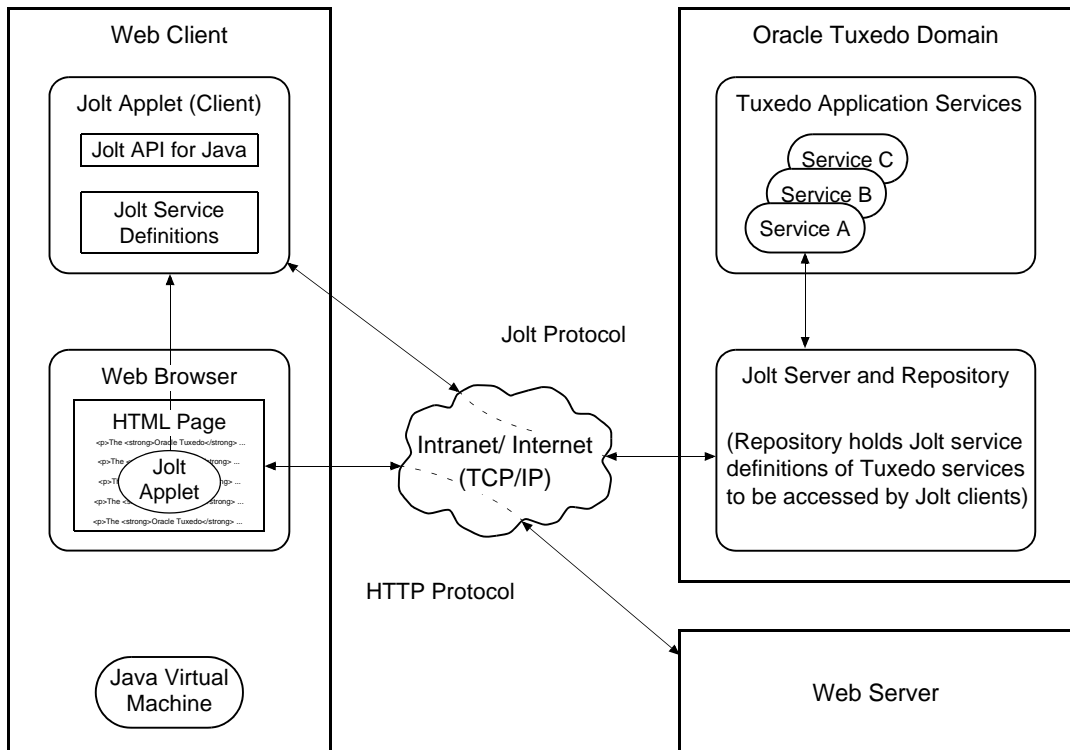
- [About Oracle Jolt](#)
- [Oracle Jolt Components](#)
- [Oracle Jolt Client Personalities](#)
- [Oracle Jolt Capabilities](#)
- [Oracle Jolt Client Support](#)
- [Oracle Jolt Release Interoperability](#)
- [Oracle Jolt Pre-Installation Checklist](#)
- [Oracle Jolt Documentation](#)

About Oracle Jolt

Oracle Jolt is a Java class library and API that enables remote Java clients to access existing Oracle Tuxedo ATMI services. As shown in the following [Figure C-1](#), Oracle Jolt extends the

functionality of existing Tuxedo ATMI applications to include intranet- and Internet-wide availability.

Figure C-1 Oracle Jolt Communication Architecture Using the Jolt Applet



Note: Web server and Jolt server must be running on the same computer machine.

The preceding figure shows one of five types of Jolt client personalities supported by Oracle Jolt, all of which are briefly described in [“Oracle Jolt Client Personalities”](#) on page B-8.

7

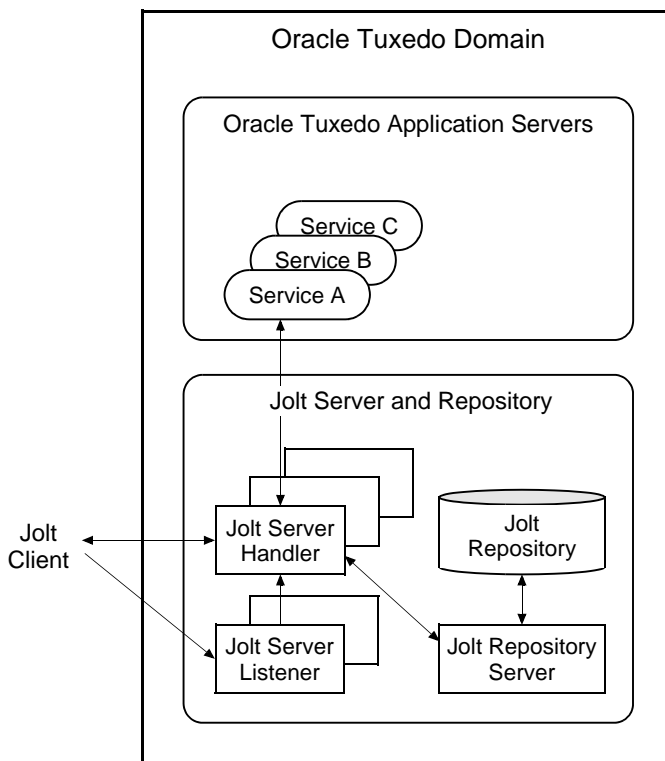
Oracle Jolt Components

Oracle Jolt consists of the following components for enabling secure, reliable access to servers inside corporate firewalls, and for creating Java-based client programs that access Tuxedo ATMI services:

- Jolt Server Listener
- Jolt Server Handler
- Jolt Repository Server
- Jolt Repository
- Jolt Internet Relay
- Jolt Class Library
- JoltBeans

As shown in the following [Figure C-2](#), the Jolt server implementation consists of one or more Jolt Server Handlers, one or more Jolt Server Listeners, and one and only one Jolt Repository Server, all running on the same Oracle Tuxedo server machine.

Figure C-2 Oracle Jolt Server Implementation



A Jolt server listens for network connections from Jolt clients, translates Jolt messages, multiplexes multiple Jolt clients into a single process, and submits and retrieves requests to and from a Tuxedo ATMI application. As with all Tuxedo system executables, the Jolt server components reside in the *tux_prod_dir/bin* directory, where *tux_prod_dir* represents the directory in which the Oracle Tuxedo 12c Release 1 (12.1.1) distribution is installed.

Jolt Server Listener

A Jolt Server Listener (JSL) is a listening process, running on the Tuxedo server, that accepts connection requests from Jolt clients and assigns connections to a Jolt Server Handler also running on the Tuxedo server. It also manages the pool of Jolt Server Handler processes, starting them in response to load demands.

Jolt Server Handler

A Jolt Server Handler (JSH) is a gateway process, running on the Tuxedo server, that handles communications between Jolt clients and the Tuxedo ATMI server application. A JSH process resides within the administrative domain of the application and is registered in the local Tuxedo bulletin board as a client.

Each JSH process can manage multiple Jolt clients. A JSH multiplexes all requests and replies with a particular Jolt client over a single connection.

Jolt Repository Server

The Jolt Repository Server (JREPSVR), running on the Tuxedo server, retrieves Jolt service definitions from the Jolt Repository and returns the service definitions to the Jolt Server Handler. The Jolt Repository Server also provides user support for updating or adding Jolt service definitions to the Jolt Repository.

Jolt Repository

The Jolt Repository, located on the Tuxedo server, is a central repository that contains definitions of Tuxedo ATMI services. These Jolt repository definitions are used by Jolt at run time to access Tuxedo services. You can export services to a Jolt client application or unexport services by hiding the definitions from the Jolt client. Using the Repository Editor, you can test new and existing Tuxedo services independently of the client applications.

Jolt Internet Relay

Jolt Internet Relay routes messages from a Jolt client to a Jolt Server Listener (JSL) or Jolt Server Handler (JSH). It eliminates the need for the JSL, JSH, and Tuxedo application to run on the same machine as the Web server. The Jolt Internet Relay consists of the following components:

- Jolt Relay (JRLY)

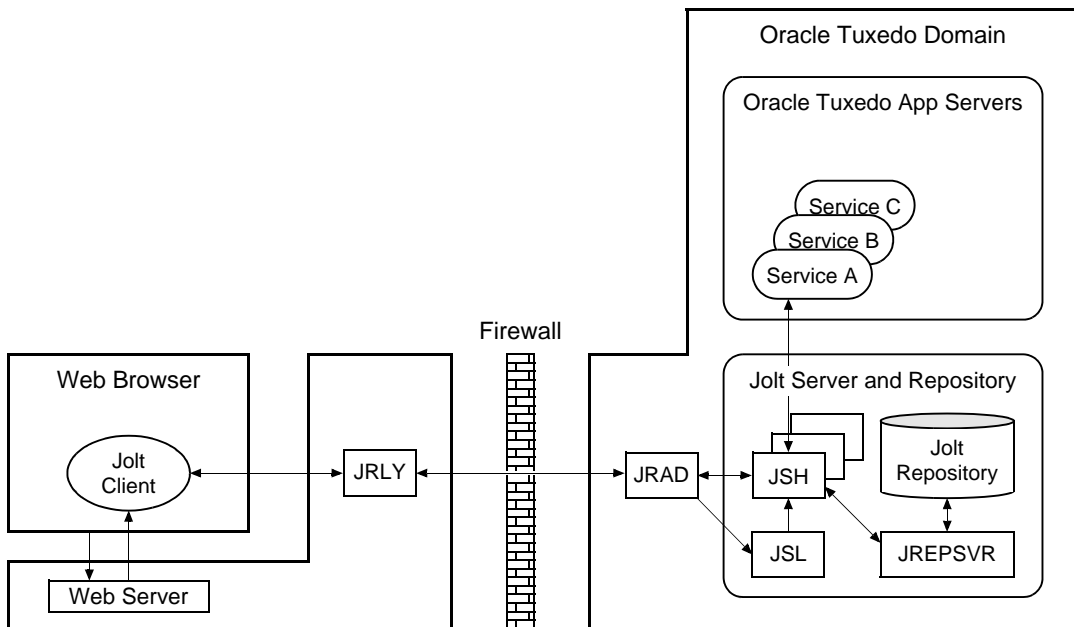
A standalone program that routes Jolt messages from Jolt clients to the JSL or JSH via the Jolt Relay Adapter (JRAD). Jolt Relay is not a Tuxedo system server or client process, and it is not dependent on the Oracle Tuxedo system software version.

- Jolt Relay Adapter (JRAD)

A Tuxedo system server that operates as a back-end relay for Jolt Relay. The JRAD may or may not be located on the same Tuxedo host machine and server group to which the JSL server is connected.

Figure C-3 illustrates the Jolt Internet Relay connection path.

Figure C-3 Oracle Jolt Internet Relay Connection Path



Note: Web server software and JRLY process must be running on the same computer machine.

A Jolt server can connect directly to intranet Jolt clients and can connect indirectly to Internet Jolt clients through the Jolt Internet Relay, all at the same time. Jolt Internet Relay is transparent to Jolt servers and Jolt clients.

The JRLY executable (`jrlly`) resides in the `tux_prod_dir/udataobj/jolt/relay` directory, whereas the JRAD executable resides in the `tux_prod_dir/bin` directory.

Jolt Class Library

The Jolt class library consists of Java class files that implement the Jolt API. These classes enable Java clients to invoke Oracle Tuxedo ATMI services. The Jolt class library provides functions to set, retrieve, manage, and invoke communication attributes, notifications, network connections, transactions, and services.

The Jolt class library files reside in the *tux_prod_dir/udataobj/jolt* directory and are stored in the following JAR files:

- jolt.jar
- joltadmin.jar
- joltatm.jar
- JoltBeanDev.jar
- JoltBeanRt.jar
- JoltBeanDevAwt.jar
- JoltBeanRtAwt.jar
- JoltBeanDevSwing.jar
- JoltBeanRtSwing.jar
- JoltBeanDevSwing11.jar
- JoltBeanRtSwing11.jar
- joltjse.jar
- joltwls.jar

To view the content of a Jolt JAR file, ensure that the path to the JDK 1.3 (or higher) software is included at the front of your `PATH` variable, go to the *tux_prod_dir/udataobj/jolt* directory, and enter the following command:

```
prompt> jar -tvf filename.jar
```

For example:

```
prompt> jar -tvf jolt.jar
0 Thu Aug 08 07:19:02 EDT 2002 META-INF/
68 Thu Aug 08 07:19:02 EDT 2002 META-INF/MANIFEST.MF
547 Thu Aug 08 07:19:00 EDT 2002
    bea/jolt/ApplicationException.class
741 Thu Aug 08 07:19:00 EDT 2002 bea/jolt/BData.class
951 Thu Aug 08 07:19:00 EDT 2002 bea/jolt/ByteArrayUtil.class
.
.
.
```

JoltBeans

JoltBeans provides a JavaBeans-compliant interface to Oracle Jolt. JoltBeans are Bean components that can be used in JavaBeans-enabled integrated development environments (IDEs) to construct Jolt clients.

JoltBeans consists of two sets of Java Beans: JoltBeans toolkit (a JavaBeans-compliant interface to Oracle Jolt that includes the JoltServiceBean, JoltSessionBean, and JoltUserEventBean) and Jolt GUI beans, which consist of Jolt-aware Abstract Window Toolkit (AWT) and Swing-based beans. The separation of Oracle Jolt into these components permits the transactional and Internet components of client/server applications to be implemented separately with the security and scalability required for large-scale Internet and intranet services.

Oracle Jolt Client Personalities

In addition to using Oracle Jolt to build client applets and applications that remotely invoke existing and new Tuxedo applications, Java programmers can use Oracle Jolt to build HTTP servlets to perform server-side Java tasks in response to HTTP requests. This latter type of Jolt connectivity enables simple Web clients to access Tuxedo application services through any Web application server that supports generic servlets.

Oracle Jolt supports the following types of Java client personalities:

- Jolt applet—a downloadable Java interactive program, running in a Web browser environment, that uses the Jolt classes to invoke Tuxedo services from within a Web page. Accessing Tuxedo ATMI services in this manner requires the downloading and installation of Jolt class package `jolt.jar` (and possibly other Jolt class packages *except* `joltjse.jar`, and `joltwls.jar`) on the machine running the Jolt applet.
- Jolt application—a standalone Java client, running on the Java Virtual Machine and operating system of a client machine, that uses the Jolt classes to invoke Tuxedo services. Accessing Tuxedo ATMI services in this manner requires the installation of Jolt class package `jolt.jar` (and possibly `joltadmin.jar`) on the machine running the Jolt application.
- JSE Connectivity for Oracle Tuxedo—a Jolt HTTP servlet, running in a Java Web application server environment (for example, Oracle WebLogic Server), through which simple Web-browser clients can invoke Tuxedo ATMI services. Accessing Tuxedo ATMI services in this manner requires the installation of Jolt class packages `jolt.jar` and `joltjse.jar` on the machine running the Web application server.

A Jolt HTTP servlet uses Jolt session pool classes to invoke Tuxedo services on behalf of simple browser clients. Thus, the servlet handles all Jolt transactions on the Web server, which enables simple browser clients to invoke Oracle Tuxedo services without directly connecting to the Jolt server and Oracle Tuxedo.

- WebLogic Connectivity for Oracle Tuxedo—a customized version of Jolt JSE Connectivity for the Oracle WebLogic Server. Accessing Tuxedo ATMI services in this manner requires the installation of Jolt class packages `jolt.jar`, `joltjse.jar`, and `joltwls.jar` on the machine running Oracle WebLogic Server.

The Jolt client personality “WebLogic Connectivity for Oracle Tuxedo” is also known as “Oracle Jolt for Oracle WebLogic Server.”

Oracle Jolt Capabilities

Running as a Java applet or as a standalone Java client application, Oracle Jolt supports the following capabilities:

- Transaction context propagation
- Access to asynchronous Oracle Tuxedo event notifications
- Typed buffer support

Oracle Jolt automatically converts Java messages to native Oracle Tuxedo data types and buffers, and converts Oracle Tuxedo data types and buffers back to Java messages.

- Jolt session pooling

Oracle Jolt supports the construction and use of Jolt session (connection) pools to increase efficiency, availability, and reliability.

- Jolt session pooling reset

Oracle Jolt support an option to reset a Jolt session pool without stopping the Jolt client, in the event of session pool failure. For example, if the Tuxedo server crashes or the Jolt Server Handler shuts down, the Jolt session pool may be reset without stopping the Jolt client.

To accommodate the Oracle Jolt 12c Release 1 (12.1.1) server (JSL, JSH, JREPSVR) and Jolt Internet Relay (JRLY, JRAD) components, your environment must provide 2 MB of disk space. For Oracle Jolt 12c Release 1 (12.1.1) system requirements, including supported platforms, see [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#).

Oracle Jolt Client Support

As stated in “[Oracle Jolt Client Personalities](#)” on [page B-8](#), Oracle Jolt 12c Release 1 (12.1.1) supports the following four client types:

- Jolt applet
- Jolt application
- JSE Connectivity for Oracle Tuxedo
- WebLogic Connectivity for Oracle Tuxedo

[Figure C-1](#) lists the requirements for the Jolt 12c Release 1 (12.1.1) applet and application client types.

Table C-1 Requirements for Oracle Jolt 12c Release 1 (12.1.1) Applet and Application Client Types

This client type . . .	Is supported on . . .
Java applet running in a browser	Microsoft Internet Explorer (IE) 6.0 or higher
Jolt application—standalone application— running in a desktop environment	Java Development Kit (JDK) 1.4.x

[Figure C-2](#) lists the requirements for the remaining three Jolt client types, all of which are implemented by HTML-based Jolt client classes running inside a Web server. The Web servers certified with Oracle Jolt 12c Release 1 (12.1.1) are listed in the table.

Table C-2 Requirements for Web-Server based Oracle Jolt 12c Release 1 (12.1.1) Client Types

Vendor	Web Server Version	OS Version	Oracle Jolt Client Personality	Comments
Any vendor	Java Servlet Engine (Any version)	Any OS running JDK 1.4.x	JSE Connectivity for Oracle Tuxedo	For a description, see the first note after this table.
Oracle Systems, Inc.	Oracle WebLogic Enterprise 5.1 or Oracle WebLogic Server 6.0 or higher	Any	WebLogic Connectivity for Oracle Tuxedo (also known as “Oracle Jolt for Oracle WebLogic Server”)	For example installation instructions, see “Installing Oracle Jolt 12c Release 1 (12.1.1) with Oracle WebLogic Server” on page 5-10.

Note: JSE Connectivity for Oracle Tuxedo is the name of the Jolt Web application server that simplifies the handling of servlets in a Oracle Tuxedo application environment. JSE is short for *Java Servlet Engine*.

Oracle Jolt Client Requirements

Oracle Jolt has the following client requirements:

- 574 KB of disk space for client classes
- 1364 KB of disk space for client API documentation
- 190 KB of disk space for client examples
- Java Development Kit (JDK) 1.5.x is certified for Oracle Jolt 12c Release 1 (12.1.1) application development (<http://java.sun.com/j2se>)
- Java-enabled browser (Internet Explorer) or Java Virtual Machine (JVM)

Oracle Jolt Client Class Library

Various implementations of Java tend to show minor differences in characteristics. Oracle Jolt 12c Release 1 (12.1.1) is based on JDK 1.5.0_0_9.

The Oracle Jolt class library is compatible with the browsers and JDK versions shown in [Figure C-3](#).

Table C-3 Oracle Jolt Class Library Compatibility

Vendor	Browser Version	Java Virtual Machine (JVM)	OS Version
Microsoft	Internet Explorer 6.0	JDK 1.5.0_0_9	Windows 2003 Server, Windows XP Professional

Oracle Jolt Release Interoperability

An Oracle Jolt 12c Release 1 (12.1.1) client can interoperate with a Oracle Jolt 1.2, 1.2.1, 8.0, 8.1, 9.0, 9.1, or 10.0 server implementation, and the client is able to use the new features available with Oracle Jolt 12c Release 1 (12.1.1).

An Oracle Jolt 1.2, 1.2.1, 8.0, 8.1, 9.0, 9.1, 10.0 client can interoperate with an Oracle Jolt 12c Release 1 (12.1.1) server implementation, but only the Oracle Jolt 1.2, 1.2.1, 8.0, 8.1, 9.0, 9.1, or 10.0 functionality is available to the Oracle Jolt client, even though new functionality is added to the Oracle Jolt server-side components when they are upgraded to Oracle Jolt 12c Release 1 (12.1.1).

Oracle Jolt Pre-Installation Checklist

On Windows systems, Oracle Jolt 12c Release 1 (12.1.1) automatically installs two Microsoft dynamic link libraries (DLLs), `MSVCRT.DLL` and `MSVCRP71.DLL`.

Oracle Jolt Documentation

For more information about Oracle Jolt, see the following documentation:

- [Using Oracle Jolt](#)
- [Using Oracle Jolt with Oracle WebLogic Server](#)

File and Database Management and Disk Space Allocation

The following sections describe Oracle Tuxedo file and database management and provide guidelines for allocating disk space for a Oracle Tuxedo application:

- [Introduction](#)
- [How the Oracle Tuxedo System Manages Files](#)
- [Arranging for Raw Disk Space](#)
- [How the Oracle Tuxedo Filesystem Is Organized](#)
- [Space for Queue Spaces](#)
- [Space for Application Servers](#)

Introduction

For the most part, the following discussions apply to both Windows and UNIX systems except for the guidelines to repartition hard disk devices. On a Windows system, input/output (I/O) is buffered by default, but Oracle Tuxedo sets a certain system-level flag to change the default. Thus, all I/O for Oracle Tuxedo processes is *unbuffered*, meaning that you do not need to make any special disk-space arrangements on a Windows system.

How the Oracle Tuxedo System Manages Files

The Oracle Tuxedo system provides a facility called the Disk Management Interface (DMI), which manages logical files within a single disk device or set of devices. The DMI performs tasks

such as storing binary configuration tables and the transaction log. You can use it to create, initialize, or destroy entries in the Oracle Tuxedo filesystem. To access the DMI, use the `tmadmin(1)` administrative commands described in the [Oracle Tuxedo Command Reference](#).

There are two ways that the logical files managed by the DMI can be stored physically:

- Stored on an operating-system (OS) filesystem
- Stored on disk space, set aside for the Oracle Tuxedo system, that is outside the control of all OS filesystems

Oracle Tuxedo files reside on device special files in the designated space and are managed directly by the DMI disk management software. The DMI supports the notion of a Oracle Tuxedo filesystem distinct from any OS filesystem.

Space outside the OS filesystem is usually referred to as *raw disk space*. Not only is I/O faster when done by system calls reading directly from and writing directly to device special files on raw disks, but a physical `write()` occurs right away. When using an OS filesystem, Oracle Tuxedo cannot predict or control the precise moment at which a `write()` is done. When using raw disk space, however, Oracle Tuxedo has accurate control of the write operation, which is particularly important for entries in the Oracle Tuxedo transaction log. Also, when multiple users are accessing the system, being able to control the write operation is important for assuring database consistency.

Arranging for Raw Disk Space

If you decide to use raw disk space for your Oracle Tuxedo application, and you are using a UNIX system, you may find that the hard disk devices on your system are fully allocated to filesystems such as `/` (root) and `/usr`. If that is the case, you must repartition your hard disk device in order to set aside some partitions for use as non-OS filesystems. For repartitioning instructions, refer to the system administration documentation for your platform.

How the Oracle Tuxedo Filesystem Is Organized

A Oracle Tuxedo filesystem has a Volume Table of Contents (VTOC), which lists the files residing on the devices named in the Universal Device List (UDL). The UDL contains information about the location of the physical storage space for Oracle Tuxedo system tables.

In a Oracle Tuxedo application, all system files might be stored together on the same raw disk slice or OS filesystem. While it is possible to use regular OS filesystem files for the configuration tables, we strongly recommend that you store the transaction log, TLOG, on a raw disk device.

Because the TLOG seldom needs to be larger than 100 blocks (51200 bytes assuming 512-byte blocks), and because disk partitions are always substantially larger than 100 blocks, it may make sense to use the same device for both the configuration files and the TLOG.

[Listing D-1](#) shows a sample VTOC and UDL diagram for a bankapp (sample application) configuration on a single machine.

Listing D-1 VTOC and UDL Diagram

Output based on setting FSCONFIG=\$TUXCONFIG, and invoking tmadmin:

No bulletin board exists. Entering boot mode.

```
> livtoc
```

Volume Table of Contents on /usr2/bank/tuxconfig:

```
0: VTOC: Device 0 Offset 0 Pages 7
1: UDL: Device 0 Offset 7 Pages 28
2: _RESOURCE_SECT: Device 0 Offset 35 Pages 6
3: _MACHINES_SECT: Device 0 Offset 41 Pages 40
4: _GROUPS_SECT: Device 0 Offset 141 Pages 100
5: _SERVERS_SECT: Device 0 Offset 241 Pages 150
6: _SERVICES_SECT: Device 0 Offset 391 Pages 60
7: _ROUTING_SECT: Device 0 Offset 451 Pages 100
8: _NETWORK_SECT: Device 0 Offset 551 Pages 20
9: _MIBPERMS_SECT: Device 0 Offset 571 Pages 2
10: _NETGROUPS_SECT: Device 0 Offset 573 Pages 2
11: _INTERFACES_SECT: Device 0 Offset 575 Pages 10
```

```
# If the TLOG is stored on the same device, there will be an
# entry something like:
```

```
12: TLOG: Device 0 Offset 585 Pages 100
```

The Oracle Tuxedo application administrator must make sure raw disk slices are available as needed on each node participating in an application. [Table D-1](#) lists the size of each element in the Oracle Tuxedo filesystem.

Table D-1 Size of Oracle Tuxedo System Tables

Entity	512-Byte Pages (Blocks)
VTOC	7
TUXCONFIG	550
TLOG	100 (default)
UDL	28
TOTAL	685

The amount of space required for the `TUXCONFIG` file must be larger if there are more entries in the configuration file, `UBBCONFIG`, than in the `bankapp` sample application. The administrator is encouraged to allocate additional space for dynamic reconfiguration and growth of the application. The default block size assumed by the `crdl` subcommand of `tmadmin` is 1000 blocks (512000 bytes assuming 512-byte blocks), which should be adequate for the initial installation.

Space for Queue Spaces

If your Oracle Tuxedo application uses `/Q` for store-and-forward queue management, your queue space can be listed in the same UDL as the one used to store the `TUXCONFIG` file and the `TLOG`, and managed by the Oracle Tuxedo `VTOC`.

Space for Application Servers

As you are calculating your space requirements for the Oracle Tuxedo system, you should also consider the requirements of the servers that perform the work of the application. These requirements are specified by the application; they are unrelated to the requirements for the Oracle Tuxedo system itself (unless otherwise specified).

IPC Resource Configuration on a UNIX System

The following sections describe the interprocess communication (IPC) parameters on a UNIX system and provide guidelines for configuring them:

- [Parameter Sets Controlling IPC Resources](#)
- [Shared Memory](#)
- [Semaphores](#)
- [Message Queues and Messages](#)
- [Other Kernel Tunables](#)

Parameter Sets Controlling IPC Resources

On a UNIX system, the Oracle Tuxedo system uses the IPC resources provided by the UNIX operating system, which are controlled by the tunable parameters listed in [Table E-1](#).

Table E-1 Tunable Parameters

Tunable Parameters Starting with This Prefix . . .	Control the . . .
SHM	Amount of shared memory
SEM	Number of semaphores
MSG	Size of message queues and messages

The settings for these parameters are application-dependent. Most UNIX systems are shipped with default values that are too low for a Oracle Tuxedo application.

Because the IPC parameters vary across different versions of the UNIX system, the descriptions provided in the following sections are generic. For the exact parameter names and defaults for each platform and for information on how to change parameter values, see [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#). If you change a parameter value, you will need to rebuild the kernel and reboot the operating system, using standard administrative tools. Consult your operating system administrator or the system administrator’s guide for your platform for details.

If your Oracle Tuxedo application is distributed, the minimum IPC resources must be available on every UNIX platform participating in the application.

Shared Memory

In the Oracle Tuxedo environment, shared memory is used for the bulletin board and the control table of the workstation listener (WSL) and the IIOP listener (ISL) processes. An application may also use shared memory for its own purposes.

The following shared memory parameters may need to be adjusted:

SHMMAX

Maximum size, in bytes, of a shared memory segment. This number represents the largest shared memory segment that can be allocated. A process can, however, attach to more than one segment of size SHMMAX.

SHMSEG

Maximum number of shared memory segments per process. For a given configuration, the maximum amount of shared memory to which a process can attach is the product (in bytes) of SHMMAX * SHMSEG. A value between 6 and 15 should be adequate.

SHMNNI

Maximum number of shared memory identifiers in the system. The Oracle Tuxedo system requires one identifier per bulletin board and an additional identifier for each workstation listener (WSL) and IIOP listener (ISL) that is running.

SHMMIN

Minimum size, in bytes, of shared memory segment. This parameter should always be set to 1.

Semaphores

Every process that participates in a Oracle Tuxedo application requires a *semaphore*. A semaphore is a hardware or software flag used to prevent processes from accessing the same shared memory space at the same time. When a process has control of a shared memory resource, all other processes are locked out of the shared memory resource until the process releases the resource.

When the Oracle Tuxedo application is booted, the underlying Oracle Tuxedo system checks the number of semaphores configured in the operating system. If the configured number is not high enough, the boot fails.

The following semaphore parameters may need to be adjusted:

SEMMNS

Maximum number of semaphores in the system. The minimum requirement for SEMMNS is $\text{MAXACCESSERS} - \text{MAXWSCLIENTS} + 13$

where MAXACCESSERS is the maximum number of Oracle Tuxedo system processes on a particular machine (including servers and native clients) and MAXWSCLIENTS is the maximum number of Oracle Tuxedo remote clients. Both of these parameters are specified in the UBBCONFIG file for the application. For more information about UBBCONFIG, see “Creating the Configuration File” in [Setting Up an Oracle Tuxedo Application](#) or UBBCONFIG(5) in the [File Formats, Data Descriptions, MIBs, and System Processes Reference](#).

SEMMNI

Maximum number of active semaphore sets.

SEMMSL

Maximum number of semaphores per semaphore set. SEMMNI and SEMMSL are commonly chosen so that their product equals SEMMNS. The Oracle Tuxedo system does not perform semaphore operations on semaphore sets; however, it attempts to allocate as many semaphores per semaphore set as possible.

SEMMAP

Size of the control map used to manage semaphore sets. SEMMAP should be equal to SEMMNI.

SEMMNU

Number of undo structures in the system. Because an undo structure is needed for each process that can access the bulletin board, SEMMNU must be at least as large as SEMMNS. (The UNIX operating system uses undo structures to unlock semaphores held by processes that die unexpectedly.)

SEMUME

Maximum number of undo entries per undo structure. The value 1 suffices.

Message Queues and Messages

The Oracle Tuxedo system uses UNIX system messages and message queues for client/server communication. Examples of such messages are service requests, service replies, conversational messages, unsolicited notification messages, administrative messages, and transaction control messages.

Every Multiple Servers, Single Queue (MSSQ) set of servers and every individual server has a message queue for receiving requests. Every client has its own queue for receiving replies. Servers that specify the `REPLYQ` parameter also get individual reply queues.

The adjustment of kernel message parameters is important to the proper tuning of an application. Inappropriate values can lead to an inability to boot, or to severe performance degradation.

Several message queue parameters are available to define various characteristics of the queue space, as indicated in [Table E-2](#).

Table E-2 Message Que Parameters

This Parameter . . .	Specifies the . . .
MSGTQL	Total number of outstanding messages that can be stored by the kernel
MSGMNB	Total number of bytes that can be stored on one queue
MSGMAX	Maximum size of an individual message
MSGSEG	Total number of message segments that can be outstanding at one time
MSGSSZ	Size of each segment

If the limit specified by any of these parameters is exceeded, then a *blocking condition* occurs. There is one exception to this rule: `MSGMAX`. Messages that exceed 75 percent of `MSGMNB`, or that are larger than `MSGMAX`, are placed in a UNIX file. A very small message containing the filename is then sent to the recipient. Because this mode of operation results in a severe reduction in performance, we strongly recommend that you avoid it.

What Is Application Deadlock?

An application deadlock can result if every process is blocked while trying to send a message. For example, when clients fill up the message space with requests, servers that are trying to send replies are blocked. Therefore, no server can read a message and a deadlock results. Occasionally, timeouts can break a deadlock, but no useful work will have been done.

A client that sends its requests with the `TPNOREPLY` flag is especially troublesome. This practice can fill either individual queues or the system message space, depending on the size of the messages. Such applications may have to implement their own flow control to limit the number of outstanding messages.

To summarize, if clients or servers are blocking on their send operations (requesting services or sending replies), there is potential for trouble. It is usually no problem, though, for a single server request queue to remain full, as long as there is space in the system for more messages on other queues.

Performance Implications of Blocking Conditions

There are performance implications to queue blocking conditions, both on the sending side and the receiving side. When waking up blocked processes, the UNIX operating system wakes up all the processes blocked on a particular event, even if only one can proceed. The other processes go back to sleep. This process scheduling overhead can be expensive.

For example, on an empty server request queue on which more than one server (MSSQ) resides, an arriving message wakes up all the idle (blocked) servers on that queue. In the case of a full server request queue, as each request is read by a server, the system wakes up all the blocked clients. Depending on the size of the messages, zero or more clients can place messages on the queue. The rest go back to sleep. Because there may be hundreds of clients in the system, the mass wakeup of all of these clients every time a service request is processed can severely degrade performance.

Tunable Message Parameters

A properly tuned system rarely fills its queues. Enough slack should be left in the queues to handle the natural variability of the message flow. No exact settings can be recommended. Tuning is very application dependent. The UNIX `ipcs(1)` command provides a snapshot of the queues so you can determine whether they are full. You can try setting the `TPNOBLOCK` flag when sending requests. If you do, clients can tell when queues are full, and they can slow down a bit. It might help to increase the scheduling priority of servers with full request queues.

The following message parameters may need to be adjusted:

MSGMNI

Number of unique message queue identifiers. Each process participating in a Oracle Tuxedo application on a particular machine typically needs at least one message queue. This number is reduced if MSSQ sets are used, which means that multiple server processes share a single queue. For transaction processing, count an additional queue per server group for transaction manager server (TMS) processes. Thus, the minimum requirement for MSGMNI can be determined by the following formula:

$$\begin{aligned} \text{MSGMNI} &= \text{MAXACCESSERS} + 7 \\ &+ (\text{number of servers with REPLYQ}) \\ &+ (\text{number of MSSQ sets}) \\ &- (\text{number of servers in MSSQ sets}) \end{aligned}$$

MSGMAX

Maximum message size in bytes. MSGMAX must be big enough to handle any Oracle Tuxedo application running on this machine.

MSGMNB

Maximum message queue length in bytes. This number must accommodate the total size of all messages that are on a queue and have not been taken off by the associated processes. The minimum value for MSGMNB is the value of MSGMAX. Messages longer than 75% of MSGMNB are sent to a file instead of a message queue—a situation that should be avoided because it severely degrades performance.

MSGMAP

Number of entries in the control map used to manage message segments. The value of MSGMAP should be the number of message segments (specified in MSGSEG).

MSGSSZ

Size, in bytes, of a message segment. A message can consist of several such segments. The value of MSGSSZ should be such that a multiple of MSGSSZ is equal to the size (including the Oracle Tuxedo system header) of the most commonly sent message. By dividing messages into segments in this way, you can avoid wasting space.

MSGSEG

Number of message segments in the system.

MSGTQL

Total number of outstanding messages that can be stored by the kernel. This is the maximum number of unread messages at any given time.

Other Kernel Tunables

Experience with the Oracle Tuxedo system has shown that some other UNIX system tunables may need to be set to higher values. These parameters are very application dependent and do not apply to all applications. [“Oracle Tuxedo 12c Release 1 \(12.1.1\) Platform Data Sheets” on page A-1](#) includes information on the defaults for each platform and instructions for changing them.

ULIMIT

Maximum file size. `ULIMIT` needs to be large enough so that you can install the Oracle Tuxedo system and build servers. We recommend 4 MB.

NOFILES

Maximum number of open files per process. A Oracle Tuxedo server requires a minimum of four file descriptors.

MAXUP

Maximum number of processes per non-superuser. The Oracle Tuxedo system processes—servers and administrative processes—run with the `UID` specified in the application’s `UBBCONFIG` file. `MAXUP` needs to be large enough to allow all of these processes to run.

NPROC

Maximum number of processes (system wide).

NREGION

Number of region table entries to allocate. Most processes have three regions: text, data, and stack. Additional regions are needed for each shared memory segment and each shared library (including text and data) that is attached. However, the region table entry for the text of a “shared text” program is shared by all processes executing that program. Each shared memory segment attached to one or more processes uses another region table entry.

NUMTIM

Maximum number of `STREAMS` modules that can be pushed by the Transport Layer Interface (TLI). A typical default value is 16; we recommend setting this parameter to at least 256.

NUMTRW

The number of TLI read/write structures to allocate in kernel data space. A typical default value is 16; we recommend setting this parameter to at least 256.

