# Oracle® Tuxedo

Administration Console Online Help

10*g* Release 3 (10.3)

January 2009

ORACLE®

Tuxedo Administration Console Online Help, 10*g* Release 3 (10.3)

# Contents

## 1. Oracle Administration Console Product Overview

## 2. A Tour of the Main Window

## 3. Administration Tasks

# 4.  Creating and Configuring Core Administrative Resources

## 5. Creating and Configuring Application Queues

# 6. Creating and Configuring Workstation Listeners and Handlers

# 7. Creating and Configuring Access Control Lists

# 8. Creating and Configuring Tuxedo CORBA Resources

# A. Architecture and Implementation

# Oracle Administration Console Product Overview

The Oracle Administration Console supports high performance client-server applications. It is a powerful, intuitive solution that lets you configure and manage your company's Oracle Tuxedo applications.

Based on Java and Web technology, the Oracle Administration Console lets you operate your Oracle Tuxedo domains from virtually anywhere—even from home, given security authorization.

This topic includes the following sections:

- Administration Through the Web

- System Requirements

- Installation

- Setting Up Your Environment for the Oracle Administration Console

- Starting the Oracle Administration Console

- Exiting the Oracle Administration Console

- Oracle Administration Console Tutorial

- <a name="P_GENERAL"></a>

# Administration Through the Web

The Oracle Administration Console is a Java-based applet that you can download into your Internet browser and use to remotely manage Oracle Tuxedo system applications.

The Oracle Administration Console simplifies many of the system administration tasks required for managing multiple-tier systems. It lets you monitor system events, manage system resources, create and configure administration objects, and view system statistics.

As system administrator, your tasks include planning, implementing, setting up, and maintaining the operation of an application built on the Oracle Tuxedo system. Your application might be as simple as a single client and server operating entirely on one machine, or it might be more complex due to one of the following circumstances:

- Clients are being moved off the main machine (as they are, for example, when the Oracle Tuxedo system workstation feature is used).

- Service requests are being sent to a mainframe computer.

- Multiple machines are being networked together to make a multiprocessor application.

## Where to Get More Information

If you are new to the Oracle Tuxedo systems, make sure you are familiar with the concepts discussed in the *Introducing the Oracle Tuxedo System*, *Setting Up an Oracle Tuxedo Application*, *Administering an Oracle Tuxedo Application at Run Time*, and *Using the Oracle Tuxedo Domains Component*, which are on the online documentation CD, before using the Oracle Administration Console.

To work with the Oracle Tuxedo system resources provided through the Oracle Administration Console, you should become familiar with the Oracle Tuxedo system online documentation.

# System Requirements

This section lists the hardware, operating system resources, and browser you must provide to support the Oracle Administration Console.

## Hardware Requirements

To run the Oracle Administration Console, you need a color display device that meets the following requirements:

- Resolution—800 by 600 pixels is required; 1024 by 768 pixels or more is recommended.

- Colors—256 colors or more is recommended.

- CPU and memory—for reasonable performance, the machine on which the Web browser executes must have sufficient CPU power and memory to run a sophisticated graphical application. Your machine should not be running the lowest class of CPU available with the minimum amount of memory required for the operating system.

- Network—for reasonable performance, a throughput rate of at least 28.8 Kbps.

## Operating System Requirements

Given the broad outlines of an application design, you must verify the availability of operating system resources needed to support your application. Operating system resources include:

- UNIX system shared resources (IPC)—for control of the maximum message size and maximum queue length

- Resources governed by kernel parameters

For additional information about IPC parameters and other system tunables, see *Installing the Oracle Tuxedo System*.

## Browser Requirements

Although any Java capable Web browser might be able to run the Administration Console, at this time Oracle supports only the following browsers:

- Netscape Navigator 4.7 or higher

- Microsoft Internet Explorer 5.0 or higher

# Installation

The Oracle Administration Console is automatically installed when you install the Oracle Tuxedo software. For more information, see *Installing the Oracle Tuxedo System*.

# Setting Up Your Environment for the Oracle Administration Console

Make sure that the TUXDIR environment variable is set to the directory in which you installed Oracle Tuxedo software. For UNIX systems, if you installed Oracle Tuxedo in the default

directory, set `TUXDIR` to `/usr/local/TUXDIR`. Don't forget to export `TUXDIR` if you are using a UNIX system. For Windows 2000 systems, if you installed Oracle Tuxedo in the default directory, it is automatically set to `TUXDIR=c:\TUX`*n* (where *n* represents the release of Oracle Tuxedo being installed).

To run the Oracle Administration Console, you need to set up two servers:

- `tuxwsvr`—a Web server provided with the Oracle Tuxedo system software. You are not required to use this server and can use your own commercial Web server.

- `wlisten`—a server required to administer the Oracle Administration Console. It must be run on the master machine in a multi-machine configuration. This machine must be able to support a Java capable browser.

## Starting tuxwsvr

To start `tuxwsvr` on a UNIX system, enter the following command:

```
tuxwsvr -l //machine:port -i $TUXDIR/udataobj/tuxwsvr.ini
```

To start `tuxwsvr` on a Windows 2000 system, enter the following command at a command prompt:

```
tuxwsvr -l //machine:port -i %TUXDIR%\udataobj\tuxwsvr.ini
```

The Oracle Tuxedo installation procedure creates the `tuxwsvr.ini` file. Usually, you do not need to edit this file—except under certain circumstances. 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, refer to the `tuxwsvr(1)` reference page in the *BEA Tuxedo Command Reference*.

## Starting wlisten

Before starting `wlisten`, check the `webgui.ini` file to verify that the default values assigned to the parameters during installation are appropriate; otherwise, make the appropriate changes. For example, on a machine called "popeye," the default port assigned to `wlisten` is 4003. To run `wlisten` with port 6060, edit the `NADDR` parameter as follows:

```
NADDR=//popeye:6060
```

Depending on your system, `webgui.ini` is located in the following locations:

Windows 2000 systems: `%TUXDIR%\udataobj\webgui\`

UNIX systems: `$TUXDIR/udataobj/webgui/`

For details about other parameters in the `webgui.ini` file, refer to the `wlisten(1)` reference page in the *BEA Tuxedo Command Reference*.

To start the `wlisten` process, enter the following command:

```
wlisten
```

# Configuring the Administration Console

Some features of the Administration Console, such as the level of encryption used for communication between the Administration Console applet and the Administration Console server, are configured by settings in the `webgui.ini` file. For more information, refer to Appendix A, "Configuring the Oracle Tuxedo Administration Console".

# Starting the Oracle Administration Console

To start the Oracle Administration Console, follow these steps:

1. Start a Web browser.

2. Enter the following URL (uniform resource locator):

   ```
   http://<machine_name>:<port>/webguitop.html
   ```

   Use of this URL depends on the following assumptions:

   – You are using `tuxwsvr` with the `tuxwsvr.ini` file

   – The `webgui.ini` file is in the default location, `TUXDIR/udataobj/webgui`

   **Note:** If you are using a commercial browser on the default port (8080), you can use something like the following URL:

   ```
   http://gumby:8080/webguitop.html
   ```

The Oracle Administration Console entry page is displayed, including warranty and license notices.

Figure 1-1 shows the Administration Console splash screen, including the button that starts the Administration Console, which is located at the bottom of the entry page.

**Figure 1-1  Oracle Administration Console Entry Page**
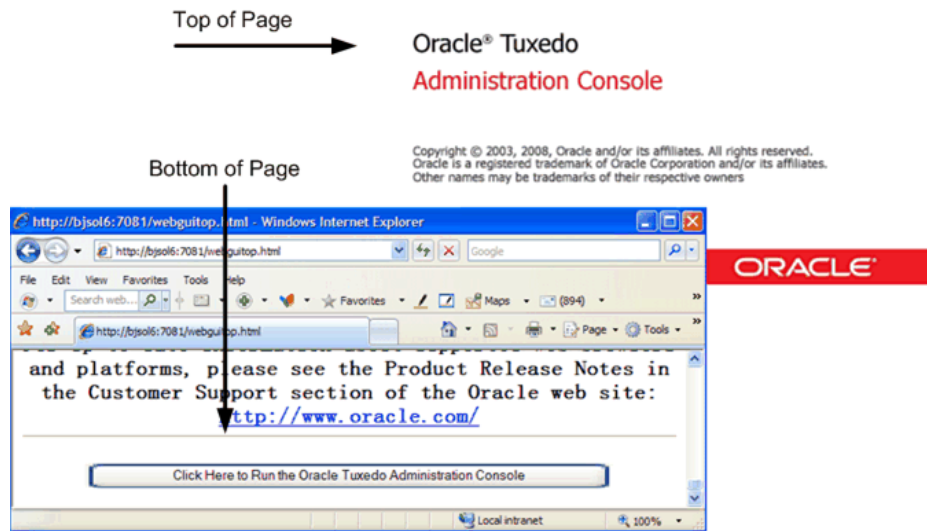


3. To start the Oracle Administration Console, scroll down to the bottom of the page and click the Click Here to Run the Oracle Administration Console button.

   The Oracle Administration Console Login window is displayed.

4. Enter your login name and password in the appropriate fields, and click LOGIN. The password must be one of the entries in the `tlisten.pw` file, which is located in the `udataobj` directory where Oracle Tuxedo is installed.

The main Oracle Administration Console window is displayed.

Figure 1-2 illustrates the Administration Console main window that displays after you complete your login.

**Figure 1-2  Oracle Administration Console Main Window**
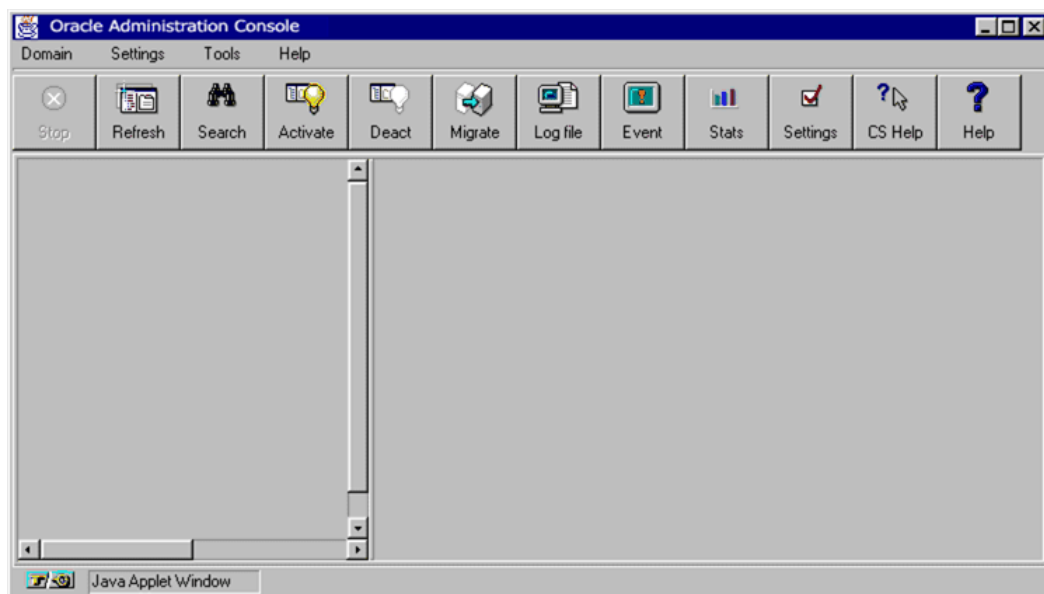


**Table 1-1  Accessing Information About the Oracle Administration Console Main Window**

| If . . . | Then . . . |
| --- | --- |
| The main window is displayed and you want to start working with the GUI. | Refer to the section "Oracle Administration Console Tutorial" on page 1-9. |

**Table 1-1  Accessing Information About the Oracle Administration Console Main Window (Continued)**

| If . . . | Then . . . |
|---|---|
| The main window is displayed and you want to read a description about it's features. | Refer to the section "A Tour of the Main Window" on page 2-1. |
| The main window does not display and the `Connect Failed` error message is displayed. | 1. Use the `ps` command to verify that the `wlisten` process is running.<br><br>2. If `wlisten` is not running, open the `webgui.ini` file, and in the line `NADDR=//<machine_name>:4003` replace the port number (4003) with a valid port number.<br><br>3. Enter `wlisten` again:<br>`$ wlisten -i`<br>`$TUXDIR/udataobj/webgui/webgui.ini`<br><br>4. Check that the `tuxwsvr` process is running at the port as described in the URL.<br><br>5. Verify the password. It must match one of the entries in the `tlisten.pw` file in the `$TUXDIR/udataobj` directory.<br><br>6. Return to step 1 above. |

You have successfully started the Oracle Administration Console. Now you can start setting up a single-machine domain environment for the `simpapp` sample application provided with your Oracle Tuxedo software.

# Exiting the Oracle Administration Console

To exit the Oracle Administration Console, select Domain—>Exit from the menu bar. This closes the current domain and exits the Oracle Administration Console applet.

# Oracle Administration Console Tutorial

This tutorial provides a step-by-step procedure for using the Oracle Administration Console to configure a single-machine domain for either a Oracle Tuxedo ATMI or CORBA environment.The tutorial includes the `simpapp` sample application, which is provided with your Oracle Tuxedo software.

**Note:** To learn more about the `simpapp` domain, refer to the appropriate `README` file in one of the following Oracle Tuxedo sample application directories:

- ATMI: `$TUXDIR/samples/atmi/simpapp`
- CORBA: `$TUXDIR/samples/corba/simpapp`

## Step 1: Set Up Your Environment

To set up your sample application environment, follow these steps:

1. Create a new directory called `simpapp`:

| Platform | Command |
|----------|---------|
| UNIX | `mkdir simpapp` |
| Windows 2000 | `md simpapp` |

2. Change to the `simpapp` directory:

   `cd simpapp`

3. Depending on the operating system, set and export your environment variables:

| Platform | Command |
|----------|---------|
| UNIX | • Set `TUXDIR` to your Oracle Tuxedo installation directory<br>• Set `TUXCONFIG` to the `'pwd'/tuxconfig` directory (`simpapp`)<br>• Add `$TUXDIR/bin` to the beginning of `PATH`<br>• Add `$TUXDIR/lib` to the beginning of `LD_LIBRARY_PATH`<br><br>**Note:** If you are running the HP-UX operating system (OS), set `SHLIB_PATH`; if you are running the AIX OS, set `LIBPATH`, instead of `LD_LIBRARY_PATH`. |
| Windows 2000 | • Set `TUXDIR` to the Oracle Tuxedo installation directory<br>• Set `APPDIR` to the current application directory (`simpapp`)<br>• Set `TUXCONFIG` to the `%APPDIR%\TUXCONFIG` directory<br><br>**Note:** You can also use the `setenv.cmd` script in the `%TUXDIR%\samples\atmi\simpapp` directory to automatically set the `simpapp` environment variables. |

4. For Windows 2000, make sure you have installed MS Visual C++ 6.0 so that the `PATH` variable includes `msdev\bin` and the `LIB` variable includes `msdev\lib`.

5. Copy the `simpapp` domain to your current application directory (`simpapp`):

| Platform | Command |
|----------|---------|
| UNIX | **ATMI**<br>`cp $TUXDIR/samples/atmi/simpapp/* .`<br>**CORBA**<br>`cp $TUXDIR/samples/corba/simpapp/* .` |
| Windows 2000 | **ATMI**<br>`xcopy %TUXDIR%\samples\atmi\simpapp\*`<br>**CORBA**<br>`xcopy %TUXDIR%\samples\corba\simpapp\*` |

6. Change the permissions on all sample application files to allow full access:

| Platform | Command |
|----------|---------|
| UNIX | `chmod 777 *` |
| Windows 2000 | `attrib -R /S *.*` |

7. Depending on the Oracle Tuxedo environment, build the client by entering one of the following commands from the current application directory:

| Environment | Command |
|-------------|---------|
| ATMI | `buildclient -o simpcl -f simpcl.c` |
| CORBA | `buildobjclient -o simple_client -f "simple_c.cpp simplec.cpp"` |

8. Depending on the Oracle Tuxedo environment, build the server by typing one of the following commands from the current application directory:

| Environment | Command |
|-------------|---------|
| ATMI | `buildserver -o simpserv -f simpserv.c -s TOUPPER` |
| CORBA | `buildobjserver -o simple_server -f "simple_c.cpp simple_s.cpp simple_i.cpp simples.cpp"` |

9. Start the Web server by entering the following command:

   **Note:** In this tutorial, the name of the machine is `machine1` and the port assigned to the Web server is `8080`. You can use `8080` for the port if you have not already used it for another Web server.

| Platform | Command |
|----------|---------|
| UNIX | `tuxwsvr -l //machine1:8080 -i $TUXDIR/udataobj/tuxwsvr.ini` |
| Windows 2000 | `tuxwsvr -l //machine1:8080 -i %TUXDIR%\udataobj\tuxwsvr.ini` |

10. If you get the following error message:

    ```
    Could not establish listening endpoint
    ```

    you must use a port number other than 8080. Select a valid number that is not already being used.

11. Start the wlisten process by typing the following command:

    ```
    wlisten
    ```

12. If the `wlisten` process starts, skip to the next step.

    **Note:** If the `wlisten` process does not start, edit the line, `NADDR=//machine1:4003` in the `$TUXDIR/udataobj/webgui/webgui.ini` file by replacing the port number (4003) with a valid port number. Then run `wlisten` again.

13. Start the browser. In this example, we are using Netscape Navigator 4.73 on a machine running Windows 2000.

14. Enter the following URL:

    ```
    http://machine1:8080/webguitop.html
    ```

    **Note:** If you are using a Web server on the default port (8080), you can use the following URL: `http://machine1/webguitop.html`

    The initial Oracle window is displayed, including the Oracle logo and several legal notices.

15. Scroll to the bottom of the window and click the `Click Here to Run the Oracle Administration Console` button.

    After the page reloads, the LOGIN window is displayed.

16. Enter your login name and the `tlisten` password after the appropriate prompts, then click on the LOGIN button. As shown in Figure 1-2, "Oracle Administration Console Main Window," on page 1-7, the Oracle Administration Console main window is displayed in your browser.

For a complete description of the main window, refer to the section "A Tour of the Main Window" on page 2-1

# Step 2: Create a New Domain

To create a new domain, follow these steps:

1. From the main window menu bar, select Domain —>New.
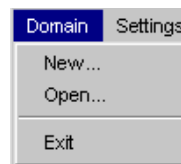


2. On the Create New Object window, enter the values indicated in the appropriate fields, as shown in Figure 1-3.

**Figure 1-3  Create New Object Window**
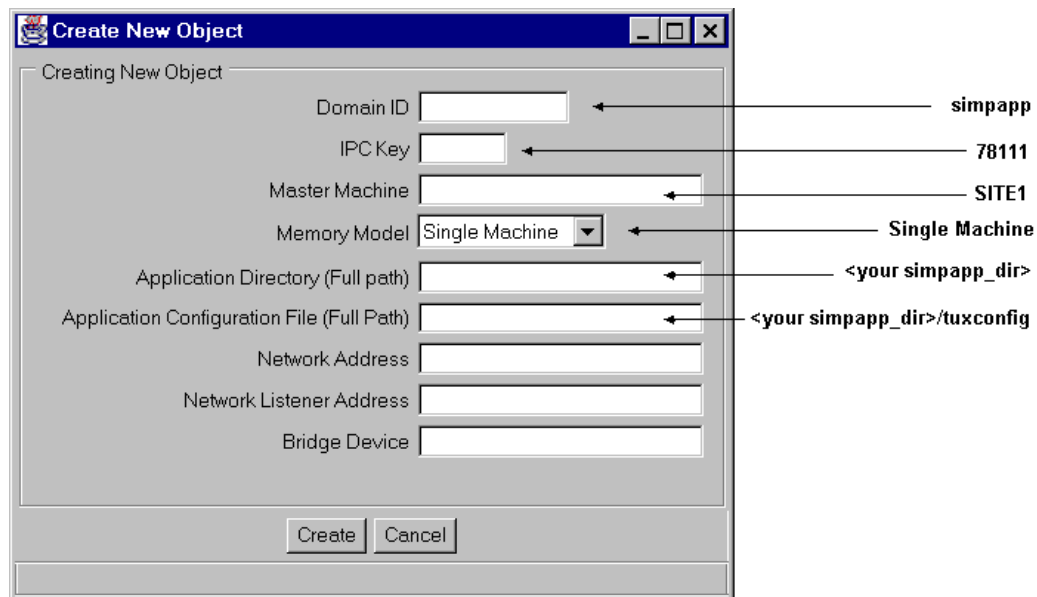


3. Leave the remaining fields blank and click Create.

   If successful, a window displays the following message:

   ```
   The SET operation was successful.
   ```

4. Click OK to close the window.

**Note:**  If you get an error message, enter a different value in the IPC Key field.

As shown in the following figure, the tree view area of the main window displays the newly created `simpapp` domain.

**New Domain**
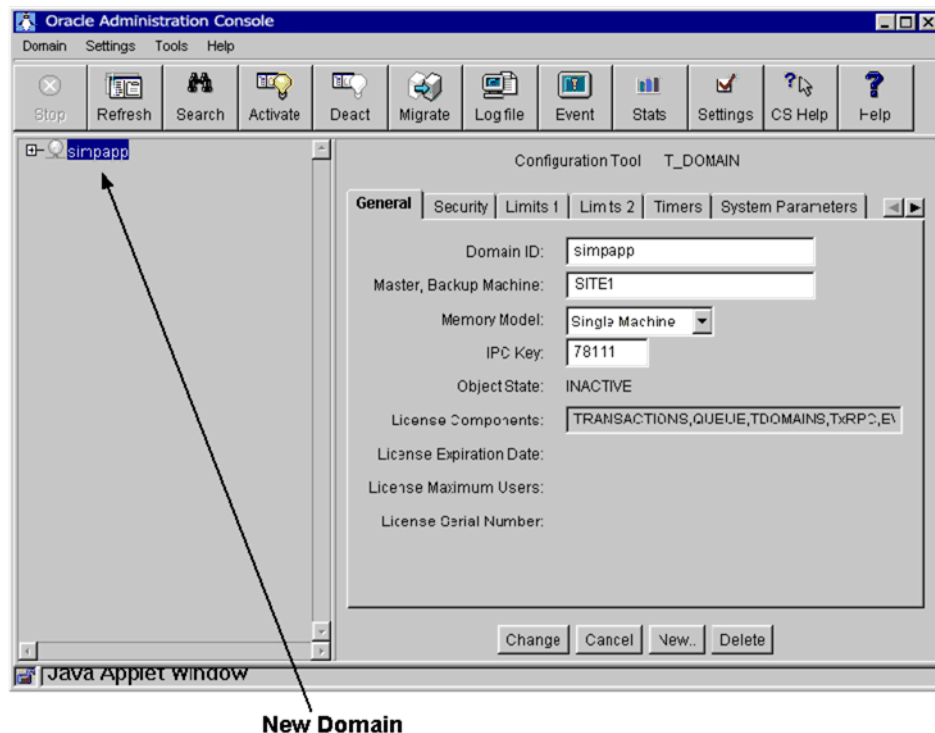
Step 3: Add GROUP1 to the simpapp Domain

To add GROUP1 to the `simpapp` domain, follow these steps:

1. Expand the tree by clicking on the plus sign (+) in front of the `simpapp` domain name. The tree view displays the listings for the Machines and Service Defaults folders.

2. Expand the Machines folder to display the inactive `SITE1` machine.

3. Expand the `SITE1` machine to display the administrative objects under its control, as shown in the following figure.
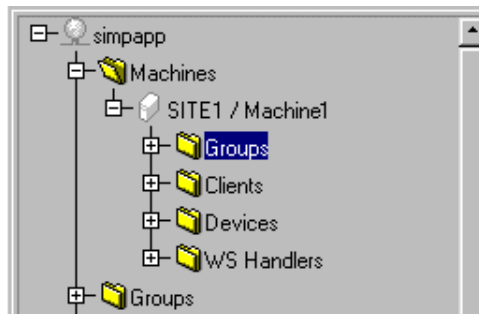
4. Right-click the Groups folder to display the Quick Menu, and then select New.

5. When the Create New Object window is displayed, enter the appropriate values shown in the appropriate fields and click Create.
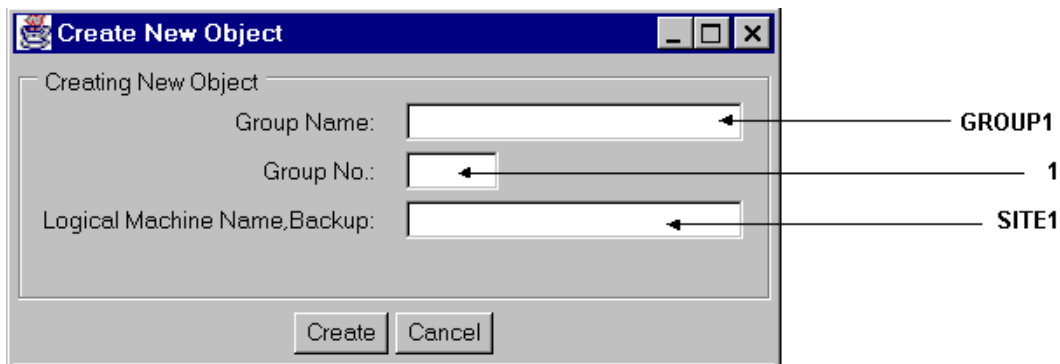


6. When the operation has succeeded, click OK to close the confirmation window.

# Step 4: Add Server simpserv to GROUP1

To add the `simpserv` server to GROUP1, follow these steps:

1. In the tree, expand the Groups folder to view GROUP1.

2. Expand GROUP1 to display the folders for Servers and Transactions.

3. Right-click the Servers folder to display the Quick Menu, and then select New to display the Create New Object window.
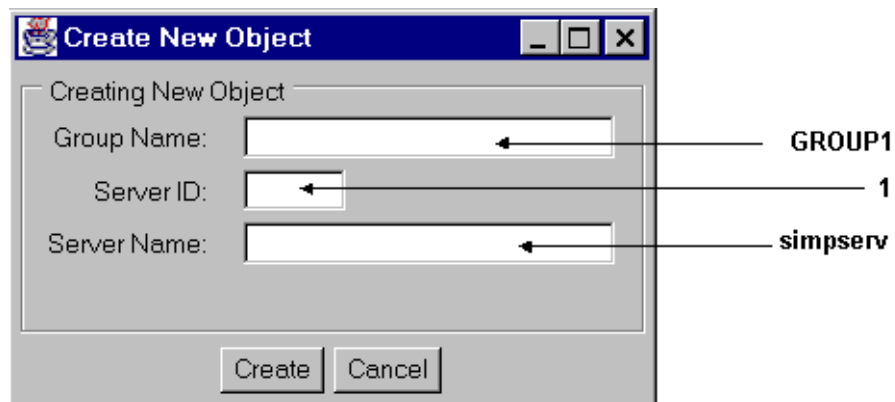


4. Enter the appropriate values as shown in the figure above, and then click Create.

5. When the operation has succeeded, a confirmation message window is displayed. Click OK to close the window.

## Step 5: Activate the simpapp Domain

To activate the `simpapp` domain, follow these steps:

1. Click the Activate toolbar button to display the Activate Objects window.
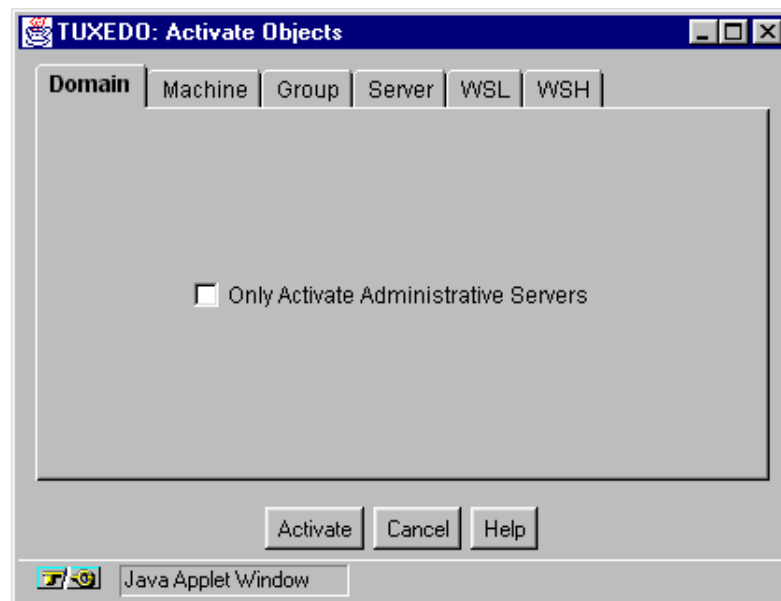


2. Click the Activate button. The Progress of Activate (Entire Domain) window is displayed, as shown in the following figure.
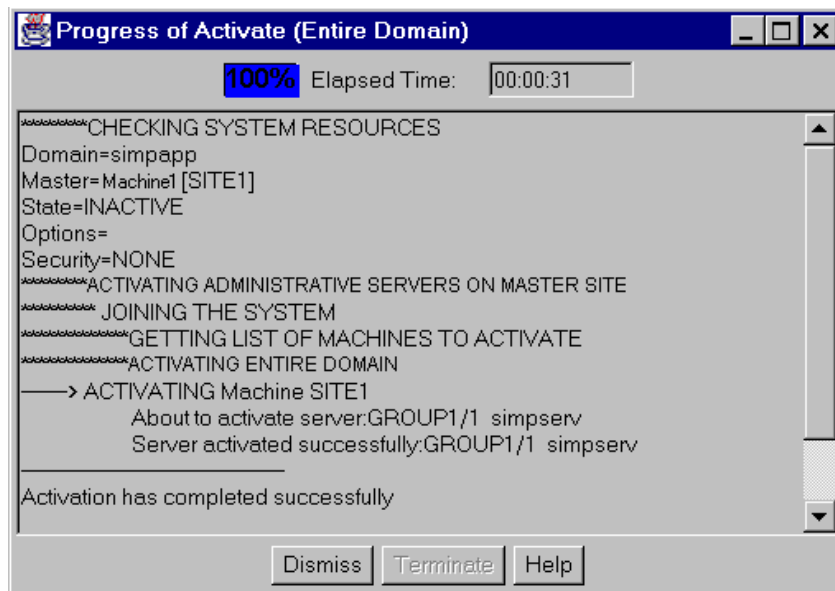
3. When the "Activation has completed successfully" message displays, click Dismiss to close the window.

# Step 6: Run simpapp

To run `simpapp`, follow these steps:

1. Expand the Servers folder. The Tree will display the `simpserv` server in an active state.

2. From a command prompt, execute the client by entering the following command:

   ```
   simpcl "hello"
   ```

3. If `simpcl` runs successfully, it will display the following output:

   ```
   Returned string is: HELLO
   ```

4. Select the `simpserv` server from the tree. Notice that the Configuration Tool is updated with the current parameters for the `simpserv` server.

5. Click the Deact toolbar button (Deactivate) to display the Deactivate Objects window.
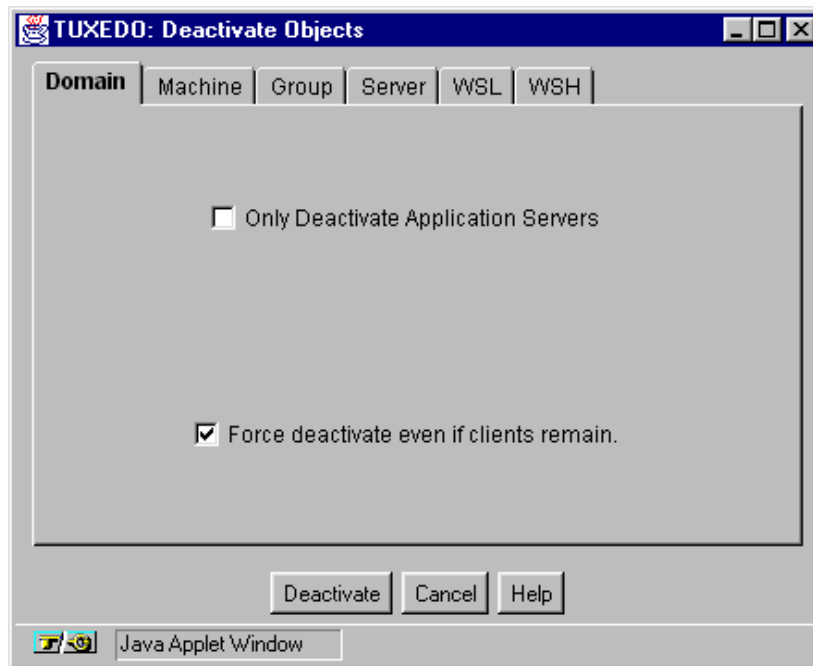
6. Select the Force deactivation even if clients remain check box.

7. Click Deactivate to start the deactivate process and to display the Progress of Deactivate (Entire Domain) window.

8. When you see the confirmation message, "Deactivation has completed successfully," click Dismiss to close the window.

9. To exit the Oracle Administration Console applet, select Domain—>Exit from the menu bar.

You have successfully configured and run `simpapp` through the Oracle Administration Console. Now you can start setting up your environment for your own application domain.

**CHAPTER 2**

# A Tour of the Main Window

The Oracle Tuxedo Administration Console is a Java-based applet that you can download from your company Web server into your Java enabled Web browser. The Oracle Tuxedo Administration Console enables you to manage your BEA Tuxedo system administrative resources with the convenience of Web access.

This topic includes the following sections:

- Main Window—describes the structure of the main window

- Parts of the Main Window—defines main window components

- The Tree View—describes the hierarchical tree view of administrative objects

- The Tree View Quick Menu—explains how to use the Quick Menu to create and manipulate the administrative objects in the tree

- The Configuration Tool Pane—describes the attributes display for administrative objects

# Main Window

Figure 2-1 illustrates the Oracle Tuxedo Administration Console main window. The main window consists of the following views:

- A hierarchical tree of a Oracle Tuxedo Management Information Base (MIB) objects

- Tabbed pages containing configuration information

**Figure 2-1 Oracle Tuxedo Administration Console Main Window**
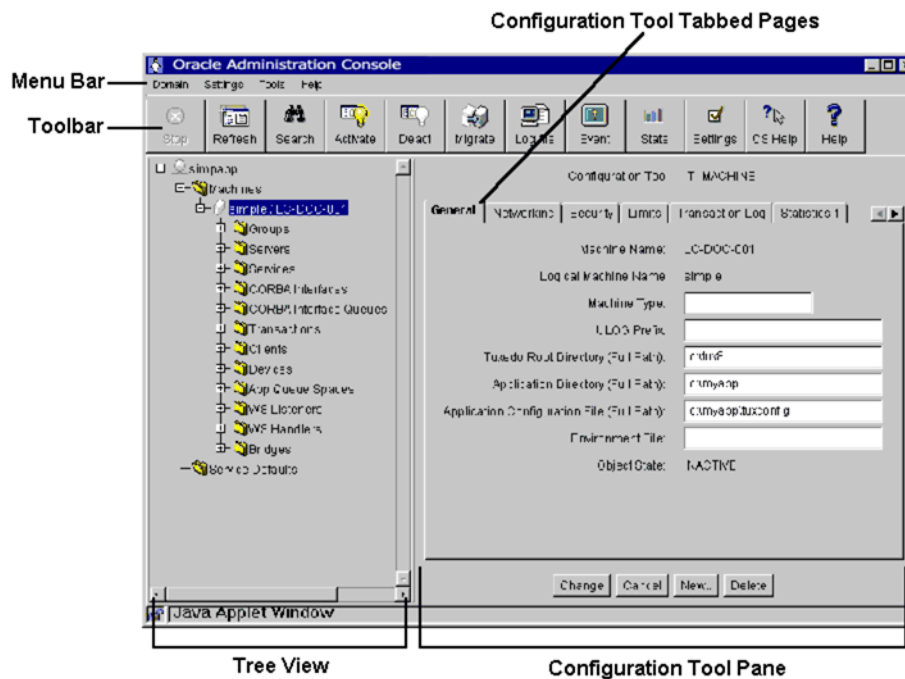


## Why Are the Main Window Items Blank or Inactive?

The initial Oracle Tuxedo Administration Console window may only display empty panes and some toolbar buttons and menu bar items may be disabled. To access the BEA Tuxedo system resources for system administration, you need to establish a connection to a domain or have domain connections specified in your initialization file. For more information refer to the section, "Oracle Administration Console Product Overview" on page 1-1

The following items are unavailable until you establish a domain connection:

- The Tree View pane and Configuration Tool pane

- All the buttons in the toolbar (except Options and Help)

- All the commands on the Tools menu

- The CS Help and Help commands on the Help menu

As shown in Figure 2-1, after you have set up and activated a domain, the Oracle Tuxedo Administration Console populates the Tree View pane with labeled icons, representing the administrative class objects in a domain. When you start using the Configuration Tool, the Oracle Tuxedo Administration Console displays tabbed pages in the Configuration Tool pane that enable you to specify configuration information.

## Resizing the Tree View and Configuration Tool Panes

If you want more display space for the Tree View pane or the Configuration Tool pane, you can adjust the width of either area by doing the following:

1. Using the mouse, point to the border that separates the two window panes (immediately to the right of the scrollbar in the Tree View area).

2. Pressing the mouse button changes the pointer to a double-headed arrow.

3. Drag the border to the right (to widen the Tree View pane) or to the left (to widen the Configuration Tool area).

## Parts of the Main Window

This section describes the parts of the main window and the actions available from the main window menus, toolbar buttons, Tree View objects, and Configuration Tool.

Figure 2-1 shows an Oracle Tuxedo Administration Console that has an established domain connection. Table 2-1 describes the major parts of the Main Window.

**Table 2-1 Main Window**

| Item | Description |
|------|-------------|
| Menu Bar | Provides access to all actions. |
| Toolbar | Provides a row of buttons that allow you to activate frequently used actions or administrative tools. |
| Tree View pane | Displays a hierarchical representation of the administrative class objects (such as servers and clients) in a BEA Tuxedo domain. |
| Configuration Tool pane | Provides a set of tab pages on which you can display, define, and modify the attributes of objects, such as the name of a machine. |
| Tab pages | A set of pages and labeled tabs containing fields that only display configuration information and fields that let you specify configuration information. |
| Configuration Tool buttons | Allow you to control the modifications you make to the configuration fields on the tab pages. |

The following topics describe each of these major parts in more detail.

## Menu Bar

The Menu bar appears at the top left side of the Oracle Tuxedo Administration Console main window. Table 2-2 describes the available menu actions.

**Note:** Unless specified otherwise, the phrase *click* means to quickly press and release the left mouse button.

**Table 2-2  Menu Bar Options**

| Menu | Description |
| --- | --- |
| Domain | The Domain menu contains options for specifying the domain (BEA Tuxedo application) you will work with. This menu has the following options:<br><br>• New—creates a new domain.<br>• Open—opens an existing domain.<br>• Exit—closes the current domain and exits the Oracle Tuxedo Administration Console applet.<br><br>This menu also lists the names of all created domain names. To switch to a different domain, simply click the desired domain. |
| Settings | The Settings menu lets you specify the following general default settings for the session:<br><br>• The location (online) of your BEA Tuxedo documentation.<br>• The method by which your data is sorted (by state or by name).<br>• The default mode in which you want to work (Edit mode or View mode).<br><br>The Settings menu provides the same function as the Settings toolbar button. |
| Tools | The Tools menu contains the same options provided by the toolbar buttons. For more information, refer to the section "Toolbar Buttons" on page 2-7. |
| Help | The Help menu provides access to the main help topics and Context-Sensitive (CS) Help. These same help options are also available on the toolbar as Help and CS Help, respectively.<br><br>To use CS Help, click Help—>CS Help to get the cross-hairs pointer, then click a field or specific area in the window to get information about the selected item. |

Throughout this online help, a menu choice is indicated in the following form:

Menu—>Menu Item
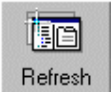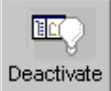
For example, an instruction to "Click Domain—>New" means click the Domain menu to display the menu items, and then click New to select the command.

```
<a name="P_STOP"></a>
<a name="P_REFRESH"></a>
<a name="P_SEARCH"></a>
<a name="P_ACTIVATE"></a>
<a name="P_DEACTIVATE"></a>
<a name="P_MIGRATE"></a>
<a name="P_ULOG"></a>
<a name="P_EVENT"></a>
<a name="P_STATS"></a>
<a name="P_SETTINGS"></a>
<a name="P_CSHELP">
</a> <a name="P_HELP"></a>
```
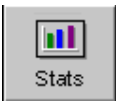
# Toolbar Buttons

The toolbar appears across the top of the main window, immediately below the menu bar.

Table 2-3 describes the toolbar buttons available for frequently used administrative operations.

**Table 2-3  Administrative Operations**

| Button | Description |
|--------|-------------|
|  Stop | Interrupts the current process and returns control to the administrator (who can then request a new operation). The Stop button allows you to interrupt the process currently underway, thus regaining control of the Oracle Tuxedo Administration Console. You can then start a new process. <br><br> You cannot assume that a process you interrupt with the Stop button has been completed; it may or may not be completed before control of the Oracle Tuxedo Administration Console is returned to you. |
|  Refresh | Updates the tree view and configuration tool pane with the most up-to-date data. |
|  Search | Searches for a particular administrative object class or object in the expanded Oracle Tuxedo Administration Console tree. Click the Search button to open the search pop-up window. Enter the name of the object for which you are searching. If necessary, the Tree View is scrolled to display your target object. |
|  Activate | Activates all or part of a BEA Tuxedo domain. The Activate button displays the Activate Objects window. This window contains a set of class-specific tab pages that enable you to activate all or part of a BEA Tuxedo domain. |
|  Deactivate | Deactivates all or a part of a BEA Tuxedo domain. |
|  Migrate | Migrates a server group or machine to another location, or swaps the master and backup machines. |

| Button | Description (Continued) |
|---|---|
| **Log file** | Displays the ULOG file from a particular machine in the active domain. |
| **Event** | Displays the window for monitoring system-generated events. |
| **Stats** | Displays the tab pages that allow you to view a graphical representation of BEA Tuxedo system activity. |
| **Options** | Provides the option to set the following default settings for the session:<br>• The location (online) of your BEA Tuxedo documentation.<br>• The method by which your data is sorted (by state or by name).<br>• The default mode in which you want to work (edit mode or view mode). |
| **CS Help** | Invokes Context-Sensitive (CS) Help. Click the CS Help button to get the cross-hairs cursor, then click a field or specific area in a GUI window to get information about the selected item. |
| **Help** | Opens the BEA Administration Console Online Help in a separate Web browser. |

# The Tree View

The Tree View pane appears in the left column of the main GUI window. The tree is a hierarchical representation of the administrative objects in a single BEA Tuxedo system domain. The GUI graphically depicts the relationship between each object and the others by showing its nesting level and parent objects. You can choose to view a complete tree (comprising all configurable objects of all types in the domain) or a subset of objects.

Table 2-2 shows the Tree View displayed in the main window.

**Figure 2-2  The Tree in the Main GUI Window**
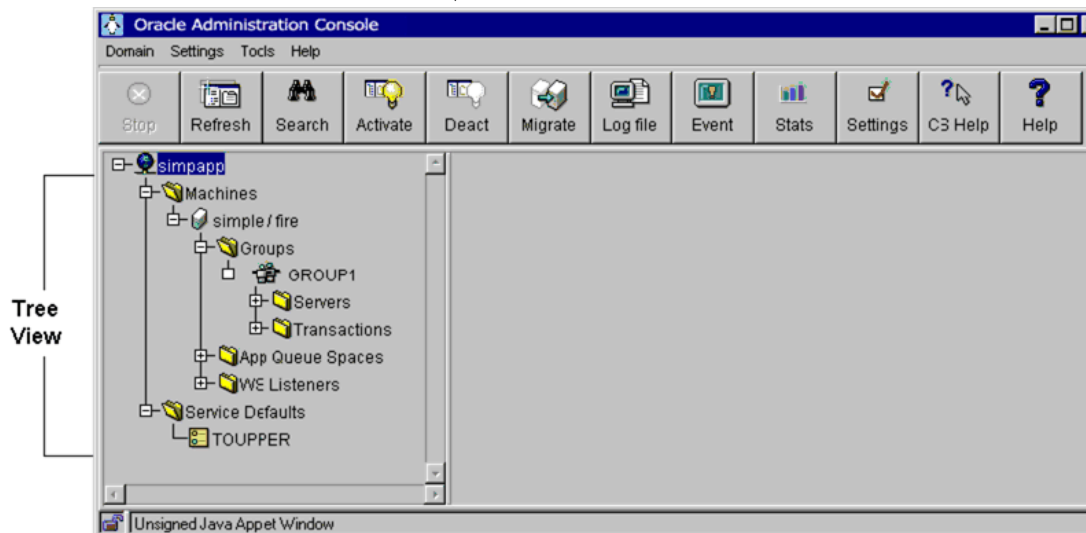


You can add or omit items from the Tree View. In addition, you can add or omit items from the display. By default, the Tree View displays all administrative classes available for a selected domain.

## Parts of the Tree View

The Oracle Tuxedo Administration Console Tree View contains multiple roots, one root for each administrative object. The first root consists of the application domain. The next root displays the object classes defined in the BEA Tuxedo TMIB. Each set of object classes is a part of an application domain. The third level represents an instance of an object belonging to an object class.

The Oracle Tuxedo Administration Console can display only one domain at a time. The current domain is represented in the Tree View as multiple sets of object classes containing administrative objects. The Oracle Tuxedo Administration Console gives you control over which object classes and objects are displayed for a domain through a combination of the collapsible tree structure and a Folders checklist dialog.

Figure 2-3 shows the top part of the sample tree shown in Figure 2-2. (Machines, Groups, Servers, Routing Tables, and so on, are examples of *object classes*. The FML and VIEW tables shown under Routing Tables are examples of *objects* belonging to the Routing Tables class.)

**Figure 2-3  Domain Tree Hierarchy**
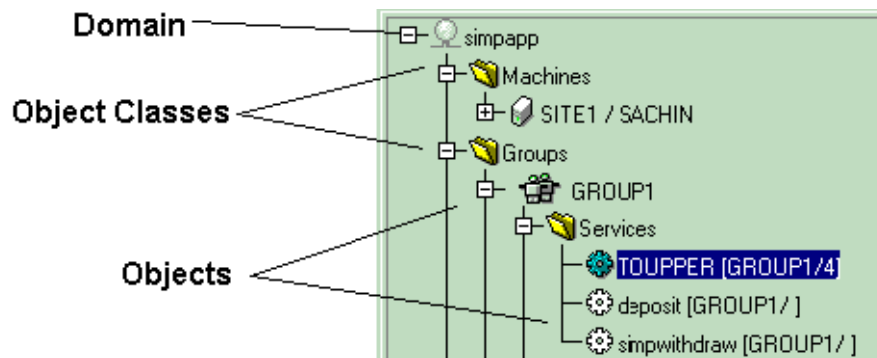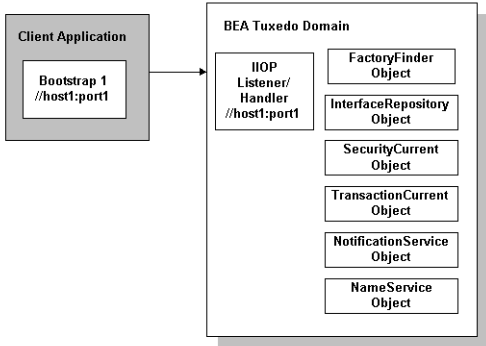


## Tree View Hierarchy

The following sections describe all the administrative objects that can appear in the Tree View.

### BEA Tuxedo System Domains and Classes

Table 2-4 lists the domain and class parts of the Tree View, their associated icons, and briefly describes each part.

**Table 2-4  Tree View**

| Tree Item | Icon | Description |
|---|---|---|
| Domain | | The Oracle Tuxedo Administration Console displays the domain icon (Tuxedo system application) as the tree unconnected, and inactive. |
| Administrative Class | closed<br>open | The folder icon and administrative name represent each object class (such as Machines, Servers, Clients, Factories, and so forth).<br>• Closed folders with a plus sign (+) indicate the folders contain objects.<br>• Open folders with a minus indicate all folder objects are displayed.<br>The Tree View displays the individual objects that are part of a class below the class folder. |

### Oracle Tuxedo Administration Console Class Objects

Table 2-5 describes the individual objects contained within a BEA Tuxedo system TMIB class and their associated icons.

**Table 2-5  TMIB Class**

| Tree Items | Icon | Description |
| --- | --- | --- |
| Machines | active<br>suspended<br>unknown | Machines represent the machines on which the applications reside. The machines are listed in the Tree View by their configuration name and have three states: active, suspended, and unknown. |
| Groups | active<br>inactive<br>migrating<br>unknown | Groups represent the administrative resources that are configured as groups and reside on a machine. These resources include servers. As an administrative unit, you can activate, deactivate, and migrate groups. |
| Servers | active<br>inactive<br>migrating<br>cleaning<br>partition<br>unknown<br>dead | Servers represent the server-side of a client/server application pair. The servers respond to requests from clients. The server icons represent BEA Tuxedo system-based application servers. Servers correspond to an operating system (OS) process. |
| Routing Tables | active | The routing tables enable a system administrator to set up a name and target service where messages and other data can be sent. |
| Service Defaults | active<br>inactive<br>suspended | Service defaults enable a system administrator to define standard settings for attributes for services such as priority, timeouts, loads, and routing. |
| Services | active<br>unadvertised<br>suspended | Services represent modules of application code that carry out a service request. These are the BEA Tuxedo system services. |
| Devices | active | Devices represent BEA Tuxedo system files (TUXCONFIG, TLOG, and so on) that reside on physical storage devices. |

| Tree Items | Icon | Description |
|---|---|---|
| Clients | active<br>inactive<br>suspended<br>dead | Clients represent the processes that attach to the domain of the BEA Tuxedo servers. |
| WS Listeners | active<br>inactive<br>migrating<br>cleaning<br>partition<br>unknown<br>dead | The Oracle Tuxedo Workstation Listeners (WSL) are processes that act as the single point of contact for Workstation clients. They also handle the distribution of workstation connections to Oracle Tuxedo Workstation Handlers (WSH). WSLs feature the same icons as servers. |
| WS Handlers | active<br>inactive<br>suspended<br>dead | The Oracle Tuxedo Workstation Handlers (WSH) are surrogate clients responsible for managing a set of Workstation client connections. These handlers are started dynamically by the workstation listener. |

| Tree Items | Icon | Description |
|---|---|---|
| Bridges | active<br>inactive<br>suspended<br>dead | Bridges are used to maintain virtual circuits to other nodes participating in an application for the purpose of transferring application messages between nodes. |
| ACLs<br>ACL Principals<br>ACL Groups | ACLs<br>principals<br>groups | Access Control List (ACLs) are the administrative resources used to manage standard *access control list* security. The following types of ACL objects are included:<br>• ACLs represent a list of groups allowed access to services, applications, queues, and events. This list is used to check authorization.<br>• ACL Principals refer to application administrators, application operators, clients, and remote domain names that can access a BEA Tuxedo system application.<br>• ACL Groups are a group or collection of principals who have access to a BEA Tuxedo system application. |

## Oracle Tuxedo Administration Console Application Queue Class Objects

Table 2-6 describes the individual application queue and related class objects contained within a BEA Tuxedo system TMIB class and their associated icons.

**Table 2-6  TMIB Class**

| Tree Items | Icon | Description |
|---|---|---|
| Application Queues | active | An application queue represents the storage area where messages are queued. |

| Tree Items | Icon | Description |
|---|---|---|
| Application Queue Messages, Spaces, and Transactions | messages <br> queue space <br> transactions | • Application Queue Messages are messages stored in application queues. <br> • Application Queue Spaces are storage spaces that contain one or more application queues. <br> • Application Queue Transactions are transactions associated with a queue space. |
| Transactions | | A transaction is a unit of activity that a BEA Tuxedo system application handles in a distributed client-server context. |

### Oracle Tuxedo CORBA System Class Objects

Table 2-7 lists the individual Oracle Tuxedo CORBA administrative class objects and their associated icons.

**Table 2-7  CORBA Administrative Class**

| Label | Icon | Description |
|---|---|---|
| Factory | | A CORBA Factory is an interface used by a client to obtain an object reference to a CORBA object. |
| Interface | active <br> partitioned <br> suspended | The interface icons represent CORBA IDL interfaces. In a CORBA development environment, an interface represents a contract with a client. Servers represent interfaces at startup. You can define the default attributes for interfaces at the domain and server group levels. |
| Interface Queue | active <br> partitioned <br> suspended | An Interface Queue controls the activation of interfaces in a server. The Interface Queue represents run-time attributes as it pertains to a particular server queue. |

# Expanding and Collapsing the Tree View

Figure 2-4 illustrates an expanded tree for the sample `simpapp` domain. The Groups folder only contains one object GROUP1 and is fully expanded as shown by the minus (-) sign next to the Groups icon. However, GROUP1 contains more items as indicated by the plus (+) sign.

**Figure 2-4 Expanded Tree**

Click on minus sign
to collapse lower
tree levels ————→

Click on plus sign
to expand lower
tree levels ————→
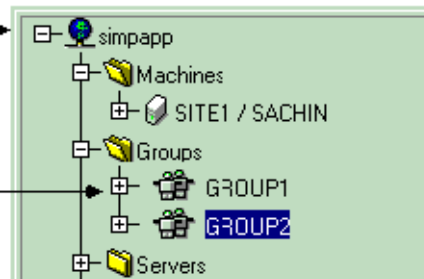


To display a domain tree, on the menu bar click Domain —>DomainName and select a domain. (You can also click Domain—>Open to bring up the Connect window and enter a Domain Name and `TUXCONFIG` path.) To expand the domain tree or any section of the domain tree, click the plus (+) sign next to the tree item.

**Note:** The plus (+) sign next to a tree icon changes to a minus (-) sign when the tree cannot be expanded further.

To collapse any section of the tree, click the minus sign beside any expanded tree item. Click the minus sign beside the domain icon and the Tree View displays only the domain icon and name.

## Searching for Objects in the Tree

To find a specific object in the Tree View, click the Search toolbar button. The Oracle Tuxedo Administration Console displays a Find dialog. Enter the name (or partial name) of the object you want to find and click the Next or Previous buttons to search forward or backward through the objects in the tree.

# The Tree View Quick Menu

The Oracle Tuxedo Administration Console Tree View contains a convenient shortcut menu that lets you quickly select frequently performed tasks. This Quick Menu includes options for creating new objects, activating and deactivating applications and application resources, as well as other administrative tasks.

To display the Quick Menu, right-click any object in the Tree View item (domain, class, or object) and the Quick Menu is displayed in the Tree View area.

If you select an option from the Quick Menu, the Quick Menu closes and the window associated with the selected option is displayed.

To close the Quick Menu without selecting a menu item, click an area of the main window outside the Quick Menu.

Figure 2-5 illustrates a sample pop-up Quick Menu.

**Figure 2-5  Quick Menu in the Main GUI Window**
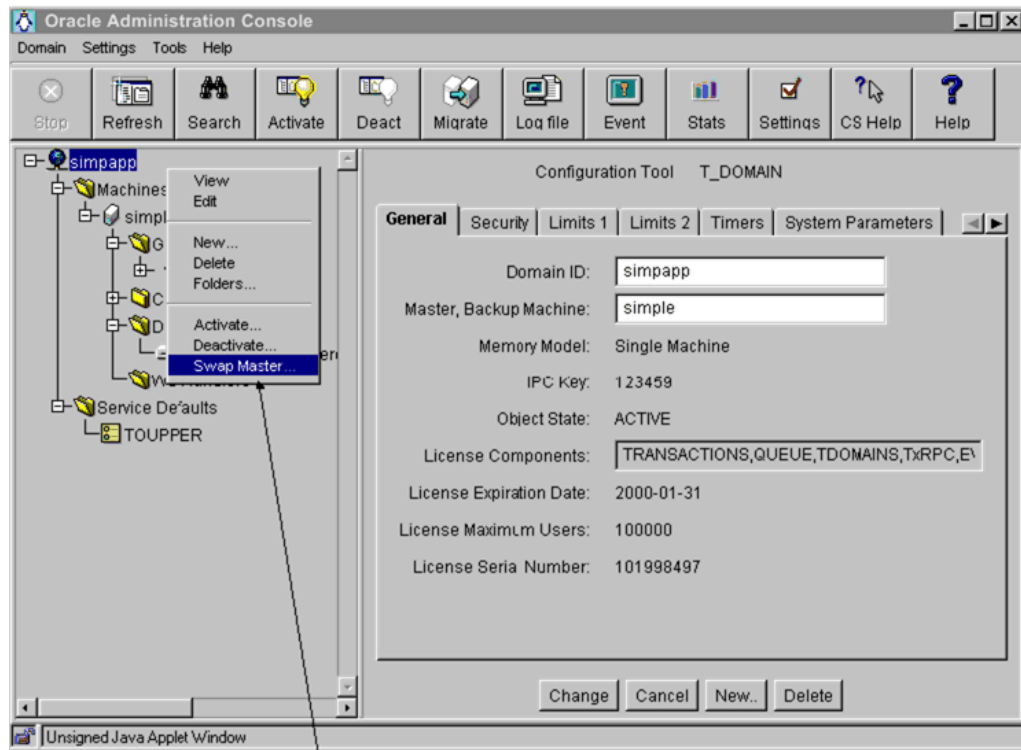


Quick Menu

The Oracle Tuxedo Administration Console always displays the Quick Menu in the Tree View area. The Quick Menu options invoke windows and dialog boxes, or toggle between modes. What you see on the Quick Menu depends upon which item the pointer is positioned on the tree.

Table 2-8 provides a quick list of what the Quick Menu displays for each type of tree item.

**Table 2-8  Quick Menus for Types of Objects**

| Your Object Selection | Quick Menu Options |
|---|---|
| Domain | • Edit and View<br>• New<br>• Delete<br>• Folders<br>• Activate and Deactivate<br>• Swap Master |
| Folder (represents an object class) | New only. |
| Administrative object (such as a specific server or device) | The options vary for each type of selected object. Each administrative object has a subset of all options.<br><br>Table 2-9 contains a complete list of all possible Quick Menu options. |

## Quick Menu Options

To perform most BEA Tuxedo system operations, you can select options from the Quick Menu that displays when you right-click a tree object. The options available in the Quick Menu vary depending upon the object class.

Table 2-9 briefly describes all the menu options listed on the collective set of Quick Menus. No single Quick Menu offers all these operations; a single Quick Menu includes a subset of these options. Click the menu option to perform the operation for that domain, object class, or individual object.

**Table 2-9  Quick Menu Options**

| Select... | To... |
|---|---|
| View | Display the tab pages containing detailed configuration information about a selected object in read-only (or View) mode. For more information, refer to the section "Switching Edit and View Modes" on page 3-4. |
| Edit | Display the tab pages containing detailed configuration information about a selected object in Edit mode. (You can use this option to override a default View mode setting for the session and change the attributes for the selected object. For more information, refer to "Switching Edit and View Modes" on page 3-4.) |

**Table 2-9  Quick Menu Options**

| Select... | To... (Continued) |
| --- | --- |
| New | Add a new BEA Tuxedo system object. |
| Delete | Delete a BEA Tuxedo system object. You cannot delete all BEA Tuxedo system objects in this fashion. |
| Folders | Display a dialog box listing all available object classes for a domain and indicate which object classes to display in the tree. |
| Activate | Activate a selected object. |
| Deactivate | Deactivate a selected object. |
| Swap | Migrate the master and backup servers in a distributed configuration. |
| Clean | Detect and remove dead clients and servers; remove partitioned machines. |
| Migrate | Move groups of servers to an alternate processor. |
| Connect | Establish a connection (bridge) between two machines. |
| Advertise | Advertise a service. |
| Unadvertise | Hide a currently advertised object from other applications. |
| Suspend | Prevent the application from accessing a service. |
| Resume | Make a currently suspended service available. |
| Kill | Terminate a BEA Tuxedo system client or service. |
| Abort | Stop a transaction that is currently in progress and roll back all affected data. |
| Open | Allocate and initialize shared memory and semaphores for a /Q queue space. |
| Close | Remove shared memory and other IPC resources for a /Q queue space. |

## About the New Option

To create a new BEA Tuxedo system object, right-click the desired object class in the Tree View to display the Quick Menu, and then click New. The Create New Object window is displayed prompting you to enter required information about the new object. For detailed information about filling in these fields, see Chapter 3, "Creating New Domains and Administrative Class Objects."

## Using the Folders Option to Display or Hide Tree Items

You can use the Quick Menu Folders option to customize your view of the tree so that only the type(s) of objects you want to see are displayed.

You can use the Quick Menu Folders option to do the following tasks:

- Display all classes of objects

- Filter out types of objects that you do not want to view

- Limit the Tree View to only one class of object

To use the Folders option, complete the following steps:

1. Right-click a domain or administrative class object in the tree to display the Quick Menu, and then select Folders to display the Select Folders window.

   **Note:** The items shown in the Select Folders window will vary, depending on which type of object you have selected in the tree.

2. The Select Folders window contains a list of all possible administrative classes and a check box beside each class name. If the check box is selected, click it to exclude that administrative class from the Tree View. If the check box is not selected, click it to display that administrative class in the Tree View.

3. Click OK at the bottom of the list to apply your changes.

shows the Select Folders window for domains.

**Figure 2-6  Select Folders Window**
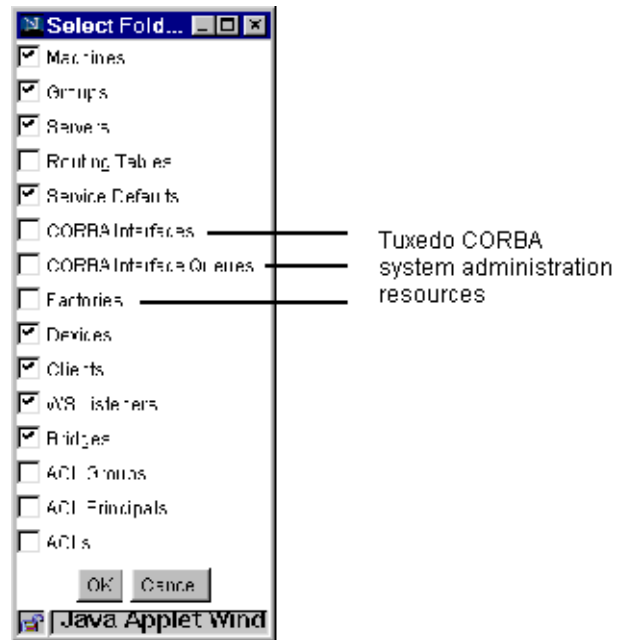
# The Configuration Tool Pane

The Configuration Tool is a utility that lets you set or change the attributes for a selected class of BEA Tuxedo system objects. When you select an object in the tree, the Configuration Tool pane for that object is displayed on the right side of the main window. As illustrated in Figure 2-7, the Configuration Tool pane keeps the attributes for each object in a set of tabbed pages.

**Figure 2-7  Configuration Tool**
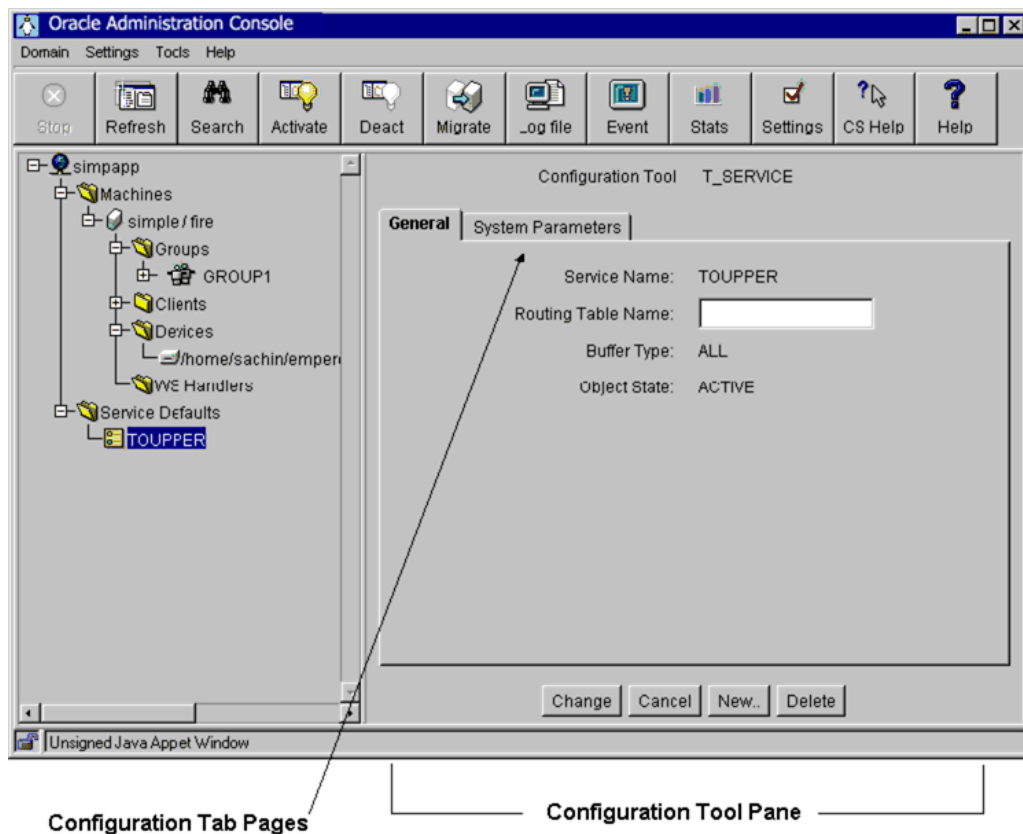


The Configuration Tool pane displays a set of tab pages for your application resources and for each class of objects and services shown in the Oracle Tuxedo Administration Console Tree View.

Each collection of tabbed pages is a set of logically grouped attributes. For example, the Domains class Security tab pages contain the settings for user IDs, group IDs, and the type of security to be used in that domain.

# Accessing the Configuration Tool Information for a Tree Object

To access the Configuration Tool from the tree:

1. Click the plus sign in front of the name of a class. The Tree View expands to display a list of the objects available for that class.

2. Click the icon or name of an object. The Configuration Tool pane displays the configuration tabs for the relevant class and the current settings for each class attribute. (Figure 2-7 shows the Configuration Tool for the TOUPPER Service Default.)

The Configuration Tool consists of a set of tab pages and a row of buttons for controlling the changes you make in the tab pages.

## The Tab Pages

The tab pages in the Configuration Tool pane are electronic forms that display and solicit (from you) information about the attributes of an administrative object.

A set of tab pages is provided for each administrative class of objects (such as machines and servers). The number of attributes associated with a class varies, depending on the class. Therefore, anywhere from one to eight tab pages may be displayed when you invoke the Configuration Tool by selecting an object in the tree.

To display a set of tab pages select an object in the tree by clicking on it.

The Configuration Tool pane displays the tab pages available for the object you have selected, as shown in Figure 2-8.

**Figure 2-8  Tab Page in the Configuration Tool Pane**
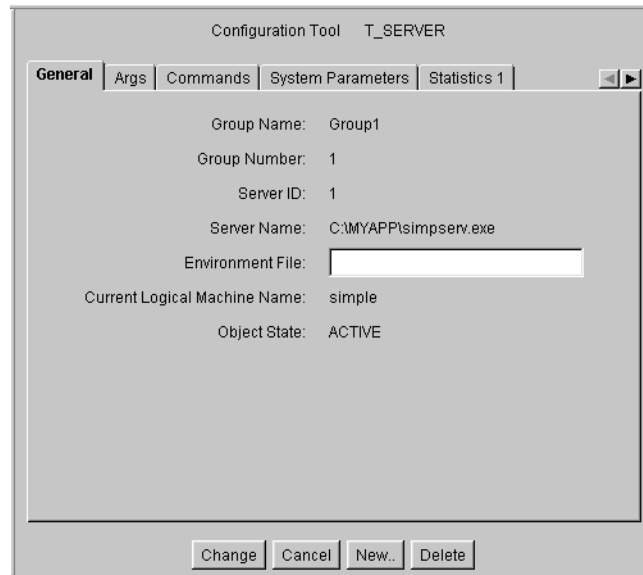


## The Configuration Tool Buttons

Use the buttons at the bottom of the Configuration Tool pane to implement the changes you make in the tab pages.

&lt;a name=T_GENERAL&gt;&lt;/a&gt;

| Use This Button . . . | To . . . |
| --- | --- |
| Change | Implement the changes you have made in the tab pages. |
| Cancel | Nullify changes you have made in the tab pages (as long as you have not clicked Change). |
| New | Add an administrative object to the current domain. |
| Delete | Remove an object from the current domain. |

**C H A P T E R** **3**

# Administration Tasks

The Oracle Tuxedo Administration Console simplifies the task of managing your Oracle Tuxedo application resources. The Oracle Tuxedo Administration Console enables you to quickly and easily perform the administrative tasks associated with managing mission-critical applications. With the Oracle Tuxedo Administration Console, you can create domains and administrative class objects (such as machines and services), as well as manage Tuxedo CORBA interfaces, factories, and interface queues. The Oracle Tuxedo Administration Console provides a browser-based solution for remote system administration.

This topic describes the administration tasks you can perform using the Oracle Tuxedo Administration Console.

This topic includes the following sections:

- Setting Up Your Oracle Tuxedo Administration Console

- Creating Administrative Class Objects

- Configuring Administrative Objects

- Connecting to and Disconnecting from a Domain

- Activating Your Domain and Domain Resources

- Deactivating Your Domain

- Migrating Tree Objects

- Swapping the Master and Backup Machines

- Cleaning a Machine

- Advertising, Unadvertising, Suspending, and Resuming Services

- Suspending, Resuming, and Killing Clients

- Suspending, Resuming, and Killing Workstation Handlers

- Opening and Closing Application Queue Spaces

- Deleting Application Queue Messages

- Aborting Transactions

- Creating Bridges Between Machines

- Security Administration

- Managing Tuxedo CORBA Resources

- Monitoring System Events

# Setting Up Your Oracle Tuxedo Administration Console

The Oracle Tuxedo Administration Console allows you to customize your administrative environment and set default parameters. Setting up and customizing the Oracle Tuxedo Administration Console includes the following tasks:

- Setting General Default Parameters

- Switching Edit and View Modes

- Controlling What the Tree Displays

## Setting General Default Parameters

To set general default parameters for your administrative session:

1. Select Settings—>Defaults from the menu bar (or click the Settings toolbar button). The Settings window is displayed as shown in Figure 3-1.

**Figure 3-1  Settings Window**
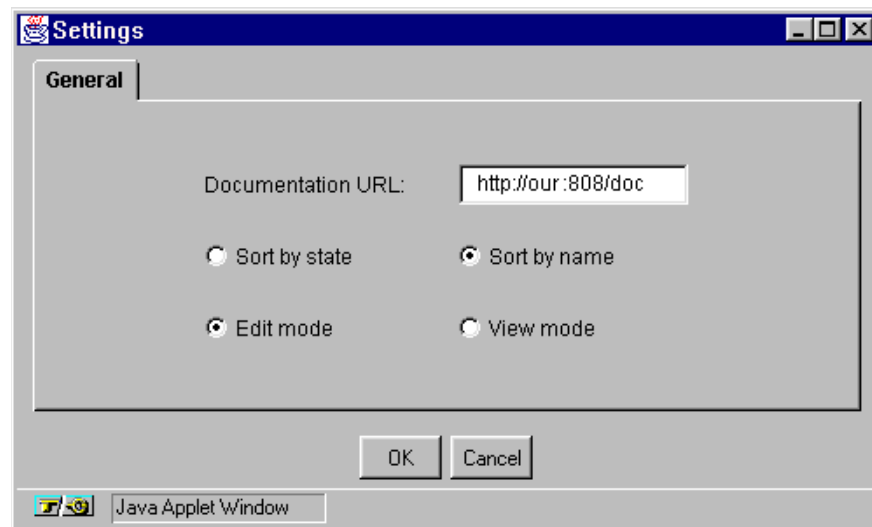


Table 3-1 explains the options available on the Settings window.

**Table 3-1  Settings Window**

| Field | Description |
|---|---|
| Documentation URL | Specifies the location of your Oracle online documentation. |
| Sort Criteria | Indicates the criterion (name or state) by which administrative objects will be sorted. Here are two examples: |
| | • If you choose to sort the objects in your domain, all the machines with names that begin with the same string (such as main1, main2, and main3) are grouped together in the Oracle Tuxedo Administration Console display. |
| | • If you choose to sort the objects in your domain by state, all the active machines are grouped together and all the inactive machines grouped together in the Oracle Tuxedo Administration Console display. |
| Default Work Mode (View or Edit) | Indicates the default work mode (view-only or edit) for the current session as follows: |
| | • In View mode, you can look at information about your domain but you cannot change it. None of the attribute fields are editable. |
| | • In Edit mode, you can create and configure domains. Where appropriate, the attribute fields are editable. |
| | **Note:** You can override this default setting for individual tree objects. For more information, see the section Switching Edit and View Modes. |

2. When you have specified the settings you want, click OK to apply your changes and close the window.

## Switching Edit and View Modes

You can set the Oracle Tuxedo Administration Console to the following default work modes:

- View mode—you can only view atttributes set in the Configuration tab pages; you cannot change the attributes.

- Edit mode—you can change the attributes in the Configuration tab fields.

By default, your Oracle Tuxedo Administration Console is set to Edit mode so that you can create or modify parameters by editing the appropriate fields in the configuration tab pages. (To change

the default mode for the session, select Settings—>Defaults from the menu bar and follow the steps in Setting General Default Parameters.)

You can momentarily *override* the default setting for individual tree items. Most likely you will use this feature to override a default View mode to edit an attribute on an object. If you return to the same object later in the session, it reverts to the default mode (in this case, View only) unless you override the default again.

To change the mode for a specific object in the Tree View:

1. Select an object in the tree, and click the right mouse button to display the Quick Menu.

2. Select either the View or Edit from the Quick Menu. This momentarily overrides the default mode setting for the selected tree object only. All of the other tree objects retain the default setting.

## Controlling What the Tree Displays

You can customize the configuration Tree View so that only the administrative classes (tree objects) you want to see are displayed. To control what class folders are displayed in the tree, follow these steps:

1. Right-click a domain or administrative class object in the tree, and then select Quick Menu —>Folders.

   As shown in Figure 3-2, the Select Folders window is displayed.

**Figure 3-2  Select Folders Dialog Box**
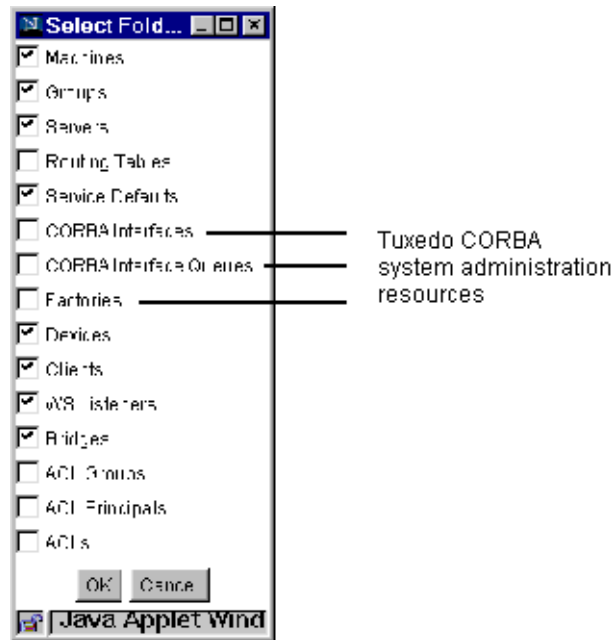


Note:    The items listed in the Folders window will vary, depending on which type of tree object you select. Figure 3-2 shows the Select Folders window displayed for the domain class object.

2.  To *select* a class object for display, click it to enter a check mark. (Clicking again  removes a check mark.) The Tree View will display only the selected classes (those with check marks).

3.  After selecting the classes you want to display, click OK to close the window.

4.  Click the plus sign in front of the domain name to expand the tree and display the selected class folders.

# Creating Administrative Class Objects

To create new domains and individual administrative class objects, you use the Create New Object window to define identifying attributes for the object such as name, location, and so on. The Oracle Tuxedo Administration Console provides a class-specific Create New Object window

for each class to which you can add an object. The fields on the Create New Object window vary, depending on the administrative class.

You cannot create new objects for all of the classes displayed in the Tree View. (For example, you cannot create new Client class objects.) For a complete list of the administrative resources you can create, see Table 3-2.

This section provides general instructions on how to create a new object via the Oracle Tuxedo Administration Console, using domains as an example. For detailed information on how to fill in attribute fields for a specific type of object, refer to the help topics listed in Table 3-2.

## Adding New Administrative Objects

To create any new administrative object, use the following general procedure:

1. Right-click a folder in the tree that contains the type of object you want to create. (Alternatively, you can also select another object of the same type as that you want to create.)

2. On the Quick Menu, click New.

**Figure 3-3  Quick Menu on a Device**
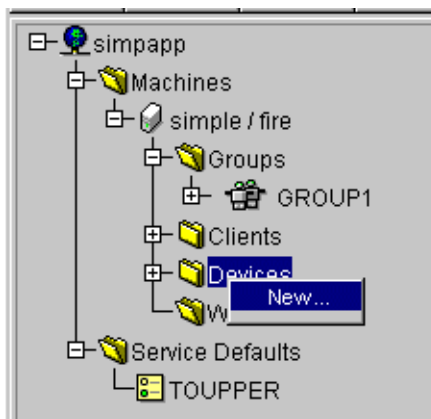


**Figure 3-4  Quick Menu on a CORBA Interfaces Class Object**
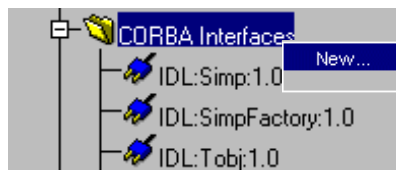
The Create New Object window is displayed as shown in Figure 3-5, prompting you for the identifying attributes needed to establish a new device object, or as shown in Figure 3-6, prompting you to establish a CORBA interface class object.

**Figure 3-5  Create New Object Window for a Device**
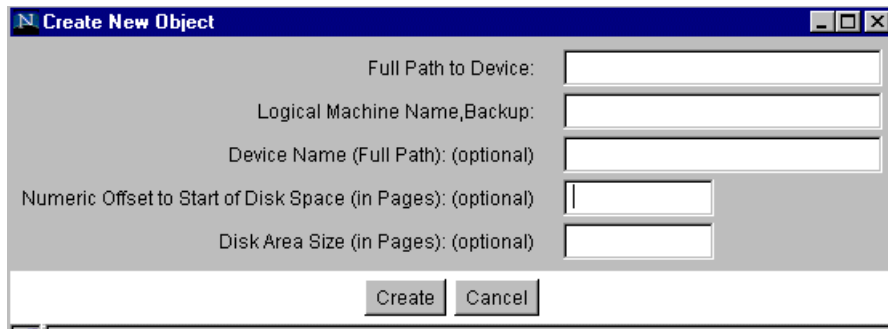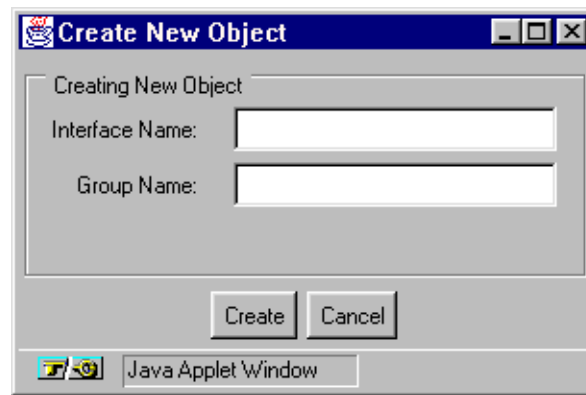


**Figure 3-6  Create New Object Window for the CORBA Interfaces Class Object**



3. Fill in the fields as needed, and click Create. (For detailed information on how to fill in the fields for a specific type of object, refer to the help topics listed in Table 3-2.)

   The new object appears in the tree contained under the folder you selected originally. (Or, at the same level as whatever non-folder object you selected originally.) The new object automatically includes empty folders of the appropriate subclasses in which you can create new member objects if needed.

Table 3-2 lists topics that give detailed information on how to fill in the fields on the Create New Object window for a specific type of object.

**Table 3-2  Creating New Class Objects**

| Class Resource | Help Topic |
| --- | --- |
| Domains | Creating a New Domain |
| Machines | Creating Machines |
| Application Queues | Creating Application Queues |
| Application Queue Spaces | Creating Application Queue Spaces |
| Groups | Creating Groups |
| Servers | Creating Servers |
| Services | Creating Services |
| Routing Tables | Creating Routing Tables |
| Service Defaults | Creating Service Defaults |
| CORBA Interfaces | Creating a New Interface |
| Devices | Creating Devices |
| WS Listeners | Creating Workstation Listeners |
| ACL Groups | Creating ACL Groups |
| ACL Principals | Creating ACL Principals |
| ACLs | Creating ACLs |

# Configuring Administrative Objects

To configure a domain and its administrative objects shown in the Tree View, you use the Configuration Tool tab pages displayed on the right side of the main window. The tab pages (also called tab folders) show the current attribute settings for the administrative object selected in the tree.

Depending on the type of administrative object you select, anywhere from one to eight tab pages is displayed—each page containing a set of attributes related to a particular administrative area for that object such as Networking, Security, Transactions, and so on. Many of the attribute fields on the tab pages are editable, so you can specify or change attributes as needed.

To configure an administrative object:

1. Select an administrative object in the Tree View.

   The Configuration Tool tab pages for that administrative object are displayed on the right side of the main window, as shown in Figure 3-7.

**Figure 3-7  Selecting an Administrative Object**
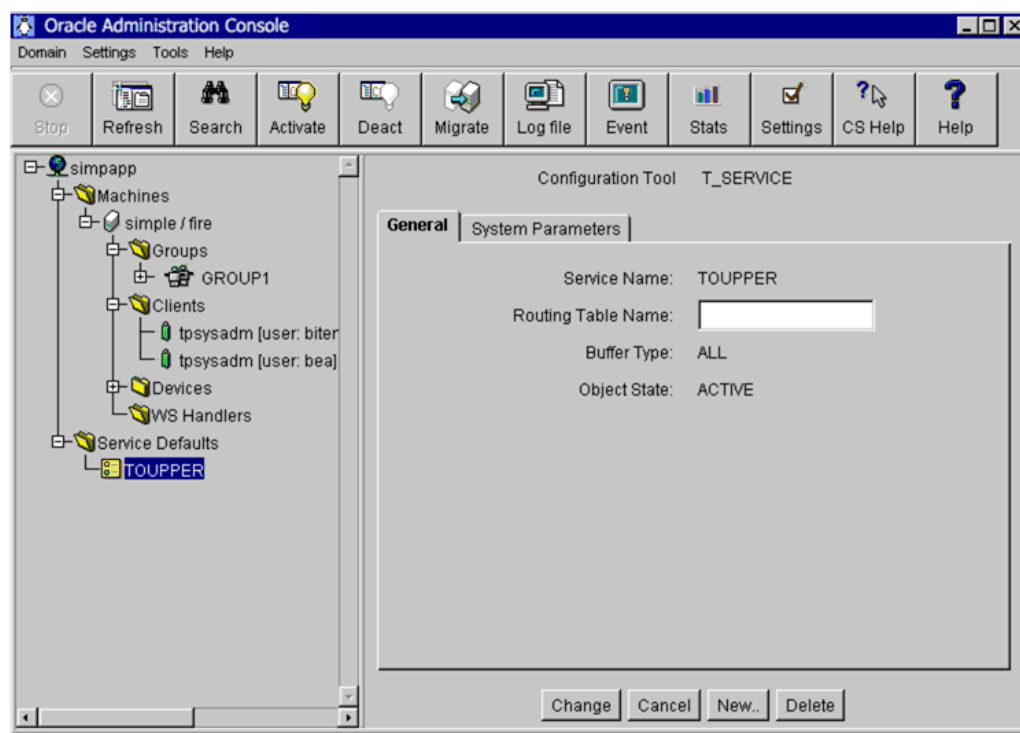


2. If you are in View mode, switch to Edit mode. (By default, the Oracle Tuxedo Administration Console is set to Edit mode so that you can create or modify parameters by editing the appropriate fields in the configuration tab pages.)

   **Note:**  For more information about setting your default configuration mode or switching from one mode to another, see the section Switching Edit and View Modes.

3. Select the various tab pages you want to work in by clicking on the topic-labeled tabs. Enter the appropriate information in the attribute fields. (To change an attribute, simply highlight its current value and enter a new one. If an attribute field is empty, enter a value for that attribute or press the Enter key on your keyboard to accept the default.)

**Note:** Defaults are only available for some fields. The defaults are listed in the help topic for the specific field. If you use the default, you need to click the Refresh button to display the default value.

Not all the fields on the tabbed folders can be edited; some information that cannot be changed, such as statistics for the current operation of the domain.

Table 3-3 lists topics that give detailed information on how to fill in attribute fields on the Configuration Tool tab pages for a specific type of object.

**Table 3-3  Configuring Administrative Resources**

| Class Resource | Help Topic |
| --- | --- |
| Domains | Configuring Domain Resources (T_DOMAIN Class) |
| Machines | Configuring Machines (T_MACHINE Class) |
| Application Queues | Configuring Application Queues (T_APPQ Class) |
| Application Queue Spaces | Configuring Application Queue Spaces (T_APPQSPACE Class) |
| Application Queue Messages | Configuring Application Queue Messages (T_APPQMSG Class) |
| Application Queue Transactions | Configuring Application Queue Transactions (T_APPQTRANS Class) |
| Groups | Configuring Groups (T_GROUP Class) |
| Servers | Configuring Servers (T_SERVER Class) |
| Services | Configuring Services (T_SVCGRP Class) |
| Routing Tables | Configuring Routing Tables (T_ROUTING Class) |
| Service Defaults | Configuring Service Defaults (T_SERVICE Class) |
| CORBA Interfaces | CORBA Interfaces (T_INTERFACE) |
| CORBA Interface Queues | CORBA Interface Queues (T_IFQUEUE) |
| Factories | CORBA Factories (T_FACTORY) |

**Table 3-3  Configuring Administrative Resources**

| Class Resource | Help Topic (Continued) |
| --- | --- |
| Devices | Configuring Devices (T_DEVICE Class) |
| Clients | Configuring Clients (T_CLIENT Class) |
| WS Listeners | Configuring Workstation Listeners (T_WSL Class) |
| WS Handlers | Configuring Workstation Handlers (T_WSH Class) |
| Bridges | Configuring Bridges (T_BRIDGE Class) |
| ACL Groups | Configuring ACL Groups (T_ACLGROUP Class) |
| ACL Principals | Configuring ACL Principals (T_ACLPRINCIPAL Class) |
| ACLs | Configuring ACLs (T_ACLPERM Class) |

# Connecting to and Disconnecting from a Domain

There are a number of operations that you can perform only when you are connected to a domain. These include:

- Activating machines and other objects

- Modifying the Oracle Tuxedo configuration

- Creating and using the Event Monitor

## How to Connect to a Domain

You have two options for connecting to an existing domain.

To connect to a domain listed on the menu bar Domain list, select *Domain—>Domain Name*.

To open a domain not on the list:

1.  Select Domain —>Open from the menu bar.

    The Connect window is displayed as shown in Figure 3-8.

**Figure 3-8  Connect Window**
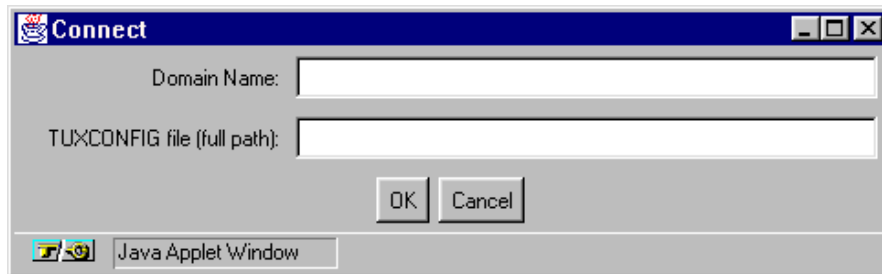


2. Fill in the fields as explained in .

**Table 3-4  Connecting to a Domain**

| Field | Description |
| --- | --- |
| Domain Name | The name of the domain you want to open. |
| TUXCONFIG File | The full path and filename of the `TUXCONFIG` file. |

3. Click OK to connect to the requested domain.

   If the connection is successful, the Tree View displays the domain icon and name, and the Configuration Tool tab pages display information about the domain.

## How to Disconnect from a Domain

The Oracle Tuxedo Administration Console automatically disconnects you from a domain whenever necessary (for example, when you select another domain name from the pull-down menu of Domains).

To disconnect from the domain to which you are currently connected, exit the GUI by selecting Domain—>Exit Application from the menu bar.

\<A name="A_0"\>\</a\>

\<a name="A_4"\>\</a\>

\<a name="A_2"\>\</a\>

\<a name="A_3"\>\</a\>

\<a name="A_5"\>\</a\>

<a name="A_6"></a>

<a name="A_7"></a>

<a name="A_8"></a>

<a name="A_9"></a>

# Activating Your Domain and Domain Resources

As the system administrator, you are the only user who is authorized to activate the Oracle Tuxedo domain and domain resources. You must perform the initial activation process on the Master machine on which the application configuration file is located.

You can activate all or part of a Oracle Tuxedo domain. Specifically, you can activate:

- An entire domain

- Only administrative servers in a domain or on a machine

- Only specific machines, groups, servers, or workstation listeners

## How to Activate a Domain

To activate a domain:

1. Select Tools—>Activate from the menu bar. (Alternatively, you can click the Activate toolbar button or right-click the domain in the tree and select Quick Menu—>Activate.)

   The Oracle Tuxedo Administration Console displays the Activate Objects window. Click the Domain tab to display the Domain information as shown in Figure 3-9.

**Figure 3-9  Activate Domain Tab Page**
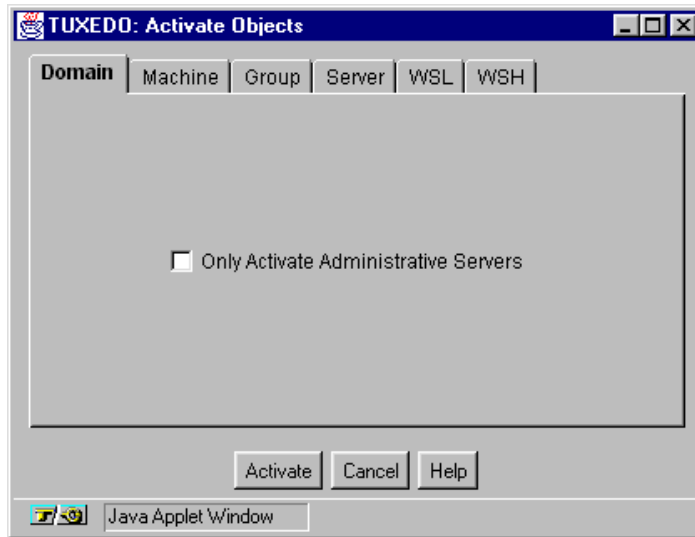


The Domain tab page contains a single option that indicates whether you want to activate only administrative servers for the domain, or the entire domain.

– To select only administrative servers, click the check box (a check mark indicates the option is selected).  (For more information on this option, refer to Activating Only Administrative Servers.)

– To select the entire domain, leave the check box unselected (no check mark).

2. To start the activate process for the domain, click Activate.

The Oracle Tuxedo Administration Console displays the Progress of Activate window. For more information about this window, see How to Monitor the Progress of Activate.

## How to Activate Class Objects

To activate domain and class objects:

1. Select Tools —>Activate from the menu bar. (Alternatively, you can click the Activate toolbar button or right-click an object in the tree and select Quick Menu—>Activate.)

The Oracle Tuxedo Administration Console displays the Activate Objects window containing a set of class-specific tab pages (see Figure 3-10). Click a tab to display the page for the type of object you want to activate.

The Activate Objects window contains six tab pages as shown in the following table.

| Select the Tab | To Choose an Activation Policy for... |
| --- | --- |
| Domain | The entire domain. (Refer to How to Activate a Domain.) |
| Machine | A selected list of machines. |
| Group | A selected list of groups. |
| Server | A selected list of servers. |
| WSL | A selected list of workstation listeners. |
| WSH | For future use. |

2. Click a tab to select the type of object that you want to activate, and then edit the tab page as explained in the following:

As shown in Figure 3-10, all Activate class objects tab pages include the following list boxes:

– ACTIVATE THESE. The Oracle Tuxedo Administration Console objects in this list will be activated.

– DO NOT ACTIVATE THESE. The Oracle Tuxedo Administration Console objects in this list will not be activated.

All objects shown indicate their *current* state next to the object name.

**Figure 3-10  Activate Objects Window**
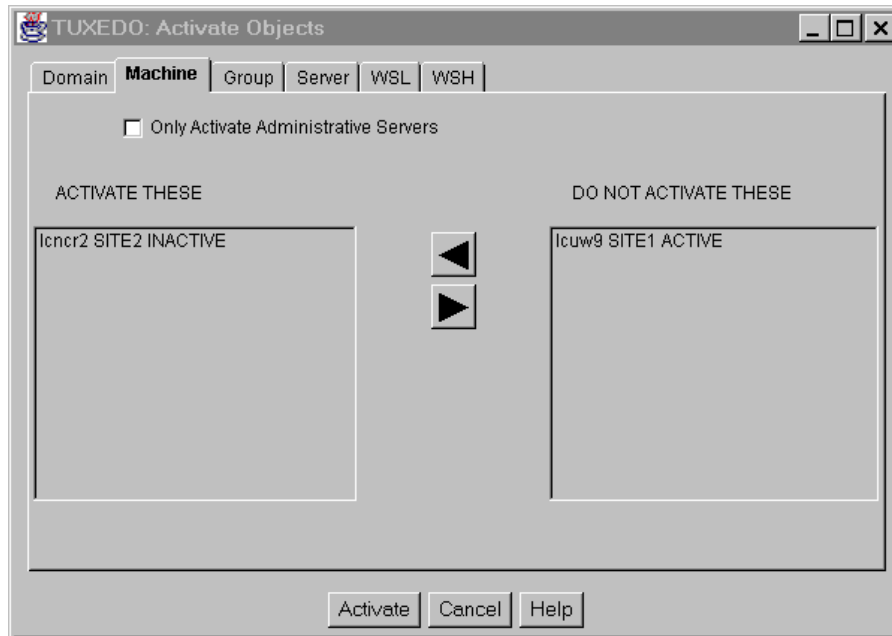


Move the object(s) you want to activate into the ACTIVATE THESE list and all other objects into the DO NOT ACTIVATE THESE list. (To move an object from one list (or column) to another, highlight the object name and click one of the arrow buttons between the two columns as shown in Table 3-5.)

**Table 3-5  Using Arrow Buttons to Modify Lists**

| Selection | | Action |
|---|---|---|
|  | Right arrow button | Moves the highlighted item to the list: DO NOT ACTIVATE THESE |
|  | Left arrow button | Moves the highlighted item to the list: ACTIVATE THESE |

The Machines tab page also contains an option to Only Activate Administrative Servers. For more information about this option, see Activating Only Administrative Servers.

3.  To start the activate process *for the objects on a particular tab*, click Activate.

**Note:** When you click Activate, you are activating only the selected objects on the current tab, not on all tabs. To activate several types of objects (such as machines, groups, servers, and so on) you need to step through this procedure for each type of object—opening the Activate window, clicking on a particular tab, making your selections, and clicking Activate.

The Oracle Tuxedo Administration Console displays the Progress of Activate window. For more information about this window, see How to Monitor the Progress of Activate.

## Activating Only Administrative Servers

The Bulletin Board Liaison Processes (BBLs) and the Distinguished Bulletin Board Liaison Process (DBBL) are Oracle Tuxedo administrative servers. Generally, these servers are activated first. If a BBL fails to activate, the other servers on that machine will not be activated. If the DBBL fails to activate, the entire application's activation process fails.

You can choose whether to limit your activation to BBLs and the DBBL for the entire domain or for selected machines as follows:

- On the Domain tab, choosing (with a check mark) the Only Activate Administrative Servers option means that only BBLs and the DBBL will be activated for the entire domain.

- On the Machines tab, choosing (with a check mark) the Only Activate Administrative Servers option means that only BBLs and the DBBL will be activated on the selected machines (those in the ACTIVATE THESE list).

## How to Monitor the Progress of Activate

As objects are activated, the Progress of Activate window is displayed (as shown in Figure 3-11). The Progress of Activate window tracks the status of the activation process as it occurs. You can:

- View the progress of the activation process

- Stop the activation process

If you do not want to view the Progress of Activate window during the activation process, click Dismiss to close the window. The activation process continues in the background.

**Figure 3-11  Progress of Activate Window**
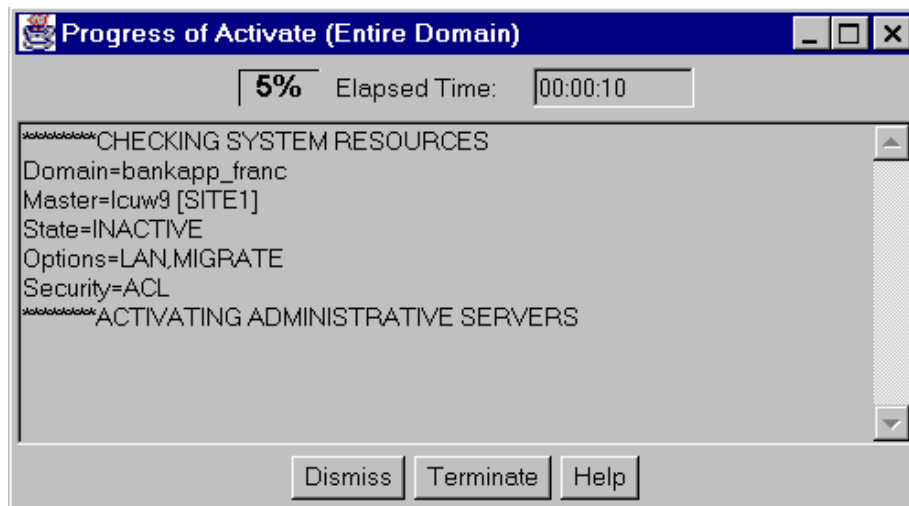


**Note:** Alternatively, if you decide you do not want to activate the chosen objects, you can click Terminate to end the activation process at any time. However, because terminating the operation in mid-process could yield unexpected results, it is recommended that you allow the process to conclude and then *deactivate* the undesired objects.

**Note:**  <a name="D_0"></a>

**Note:**  <a name="D_2"></a>

**Note:**  <a name="D_3"></a>

**Note:**  <a name="D_4"></a>

**Note:**  <a name="D_5"></a>

**Note:**  <a name="D_6"></a>

**Note:**  <a name="D_7"></a>

**Note:**  <a name="D_8"></a>

**Note:**  <a name="D_9"></a

# Deactivating Your Domain

The Oracle Tuxedo Administration Console enables you to deactivate all or parts of a  Oracle Tuxedo domain. Specifically, you can deactivate:

- An entire domain

- Only administrative servers in a domain or on a machine

- Only specific machines, groups, servers, or workstation listeners

**Note:** You can also deactivate bridges. For more information about deactivating bridges, see Creating Bridges Between Machines.

# How to Deactivate a Domain

To deactivate a domain:

1. Select Tools —>Deactivate from the menu bar. (Alternatively, you can click the Deactivate toolbar button or right-click an object in the tree and select Quick Menu—>Deactivate.)

   The Oracle Tuxedo Administration Console displays the Deactivate Objects window. Click the Domain tab to display the Domain information as shown in Figure 3-12.

**Figure 3-12  Deactivate Domain Tab Page**
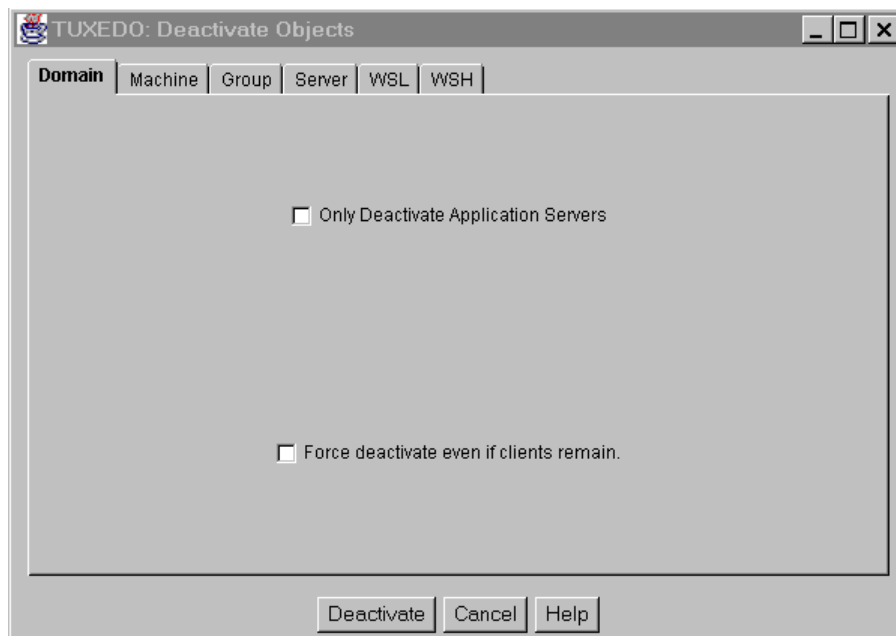
The Domain tab page contains the following options:

– Only Deactivate Administration Servers. (For more information, refer to Deactivating Only Application Servers.)

– Force deactivate even if clients remain. (For more information, refer to Forcing Deactivation.)

To select either of these options, click the check boxes. (A check mark indicates the associated action will be performed as a part of the deactivation).

2.  To start the deactivate process for the domain, click Deactivate.

The Oracle Tuxedo Administration Console displays the Progress of Deactivate window. For more information about this window, see How to Monitor the Progress of Deactivate.

## How to Deactivate Class Objects

To deactivate class objects:

1.  Select Tools —>Deactivate from the menu bar. (Alternatively, you can click the Deactivate toolbar button or right-click an object in the tree and select Quick Menu—>Deactivate.)

The Oracle Tuxedo Administration Console displays the Deactivate Objects window containing a set of class-specific tab pages (see Figure 3-13). Click a tab to display the page for the type of object you want to deactivate.

The Deactivate Objects window contains six tab pages as shown in the following table.

| Select the Tab | To Choose a Deactivation Policy for... |
| --- | --- |
| Domain | The entire domain. (Refer to How to Deactivate a Domain.) |
| Machine | A selected list of machines. |
| Group | A selected list of groups. |
| Server | A selected list of servers. |
| WSL | A selected list of workstation listeners. |
| WSH | For future use. |

2.  Click a tab to select the type of object you want to deactivate, and then edit the tab page as explained in the following paragraphs:

As shown in Figure 3-13, all Deactivate class objects tab pages include the following list boxes:

– DEACTIVATE THESE. The Oracle Tuxedo Administration Console objects in this list will be deactivated.

– DO NOT DEACTIVATE THESE. The Oracle Tuxedo Administration Console objects in this list will not be deactivated.

All objects shown indicate their *current* state next to the object name.

**Figure 3-13  Deactivate Objects Window**
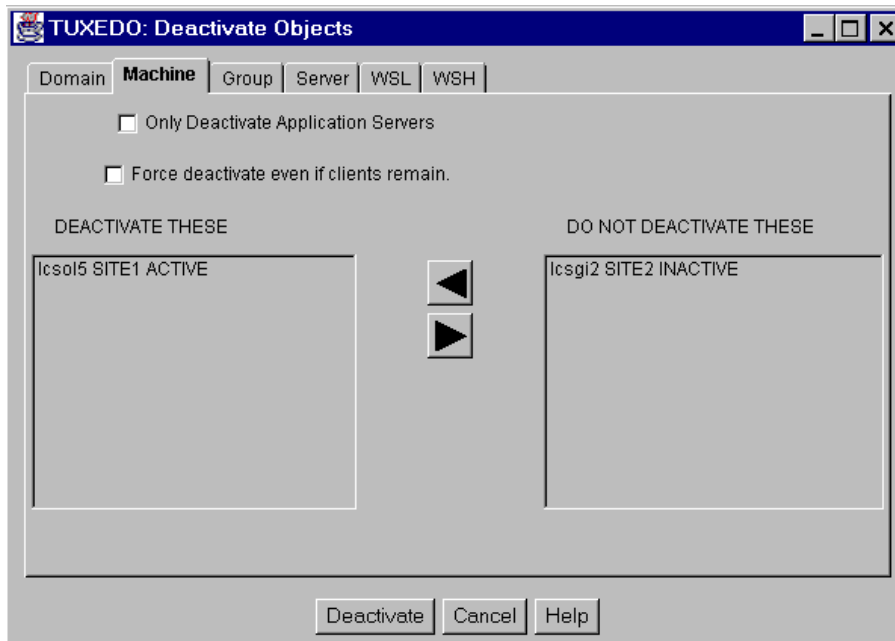


Move the object(s) you want to deactivate into the DEACTIVATE THESE list and all other objects into the DO NOT DEACTIVATE THESE list. (To move an object from one list (or column) to another, highlight the object name and click one of the arrow buttons between the two columns as shown in Table 3-5.)

The Machine tab page also contains options to:

– Only Deactivate Administrative Servers (For more information, refer to Deactivating Only Application Servers.)

– Force deactivate even if clients remain. (For more information, refer to Forcing Deactivation.)

3. To start the deactivate process *for the objects on a particular tab*, click Deactivate.

   **Note:** When you click Deactivate, you are deactivating only the selected objects on the current tab, not on all tabs. To deactivate several types of objects (such as machines, groups, servers, and so on) you need to step through this procedure for each type of object—opening the Deactivate window, clicking on a particular tab, making your selections, and clicking Deactivate.

The Oracle Tuxedo Administration Console displays the Progress of Deactivate window. For more information about this window, see How to Monitor the Progress of Deactivate.

## Deactivating Only Application Servers

Application servers are user-defined or Oracle Tuxedo defined servers configured by the user in the UBBCONFIG file. You can deactivate only application servers for the entire domain or for selected machines as follows:

- On the Domain tab, choosing (with a check mark) the Only Deactivate Application Servers option means that all application servers in the domain will be deactivated and only BBLs and the DBBL will remain active.

- On the Machine tab, choosing (with a check mark) the Only Deactivate Application Servers option means that all application servers on the selected machines will be deactivated and only BBLs and the DBBL will remain active. (In this context, *selected machines* refers to those in the DEACTIVATE THESE list.)

See also Forcing Deactivation.

## Forcing Deactivation

If you attempt to deactivate application servers for a domain or machine while there are clients still connected to the machine, the deactivation fails.

However, you can force deactivation even if clients remain connected as follows:

- On the Domain tab, choosing "Force deactivate even if clients remain" causes deactivation of all application servers in the domain even if there are clients still connected to some of the machines in the domain.

- On the Machine tab, choosing "Force deactivate even if clients remain" causes deactivation of all application servers on the selected machines even if there are clients still connected

to some of these machines. (In this context, *selected machines* refers to those in the DEACTIVATE THESE list.)

# How to Monitor the Progress of Deactivate

As objects are deactivated, the Progress of Deactivate window is displayed (as shown in Figure 3-14). The Progress of Deactivate window tracks the status of the deactivation process as it occurs. You can:

- View the progress of the deactivation process

- Stop the deactivation process

If you do not want to view the Progress of Deactivate window during the deactivation process, click Dismiss to close the window. The deactivation process continues in the background.

**Figure 3-14  Progress of Deactivate Window**
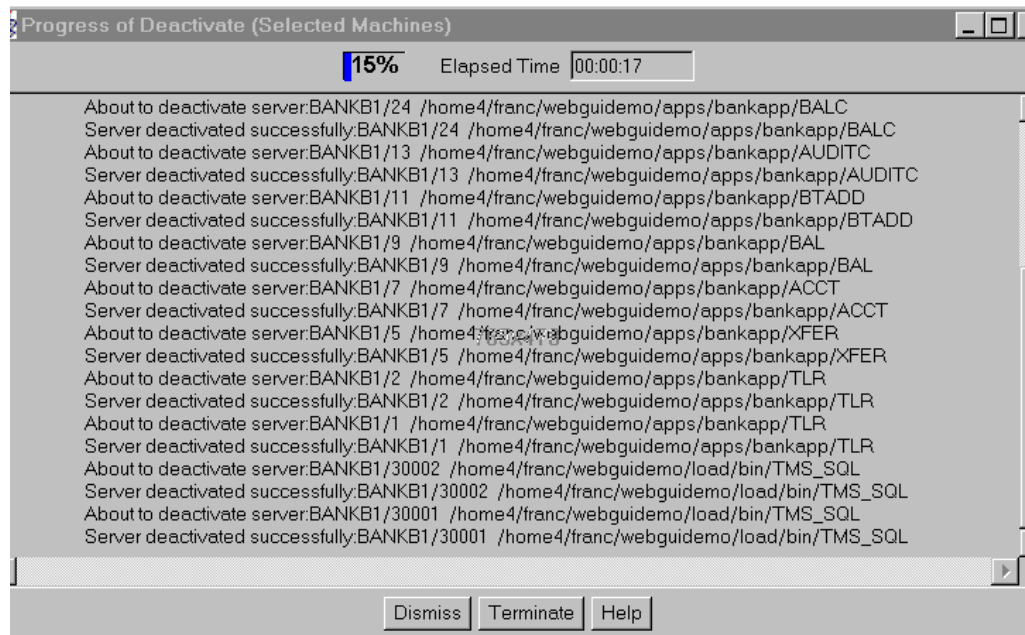


**Note:** Alternatively, if you decide you do not want to deactivate the chosen objects, you can click Terminate to end the deactivation process at any time. However, because terminating the operation in mid-process could yield unexpected results, it is recommended that you allow the process to conclude and then *activate* the objects again.

# Migrating Tree Objects

Using the Oracle Tuxedo Administration Console, you can migrate server groups and machines to alternate processors. This allows you to schedule maintenance for the original machine. You can also move groups of services from one machine to an alternate machine. By doing so you can schedule maintenance for the original machine.

**Note:** A server group is the smallest unit that can be migrated to an alternate location.

## Before Starting a Migrate Task

Before you can migrate server groups and machines, MIGRATE must be an option for the domain (see the Options field on the System Parameters folder for the domain) and a backup machine must be specified (see the Master, Backup Machine field on the General folder for the domain).

When you migrate a server group, the environment on the destination machine must duplicate the environment on the source machine, including all packages such as database software and any other application software necessary to run the application.

## Displaying the Migrate Objects Window

You use the Migrate Objects window to select the object(s) you want to migrate. To display the Migrate Objects window, select Tools —>Migrate from the menu bar. (Alternatively, you can click the Migrate toolbar button or right-click a Groups object in the tree and select Quick Menu—>Migrate.)

The Oracle Tuxedo Administration Console displays the Migrate Objects window containing a set of class-specific tab pages. Click a tab to display the page for the type of object you want to migrate.

## Migrating a Server Group from One Machine to Another

To migrate a server group from one machine to another:

1. Display the Migrate Objects window and select the name of the server group you want to move, using the drop-down menu in the Migrate Group field.

2. Enter the name of the machine on which that server group is currently located (source) in the From Machine field.

3. Enter the name of the machine to which you want to migrate the server group (destination) in the To Machine field.

4. Click Migrate. The Progress of Migrate window is displayed. For more information about this window, refer to Monitoring the Progress of Migrate.

Your next steps depend entirely on the applications you are running. To ensure an orderly migration, you might need to drain queues, shut down groups, migrate databases using tools provided by the database vendor, and so on.

## Migrating All Servers from One Machine to Another Machine

To migrate all servers from one machine to another:

1. Display the Migrate Objects window and enter the name of the machine from which you are migrating servers in the Migrate Servers From field.

2. Enter the name of the machine to which you want to migrate the servers in the To Machine field.

3. Click Migrate. The Progress of Migrate window is displayed.

Your next steps depend entirely on the applications you have running. You may need to drain queues, shut down groups, migrate databases using tools provided by the database vendor, and so on, to ensure an orderly migration.

## Monitoring the Progress of Migrate

As objects are migrated, the Progress of Migrate window is displayed. The Progress of Migrate window tracks the status of the migration process as it occurs. You can:

- View the progress of the migration process

- Stop the migration process

If you do not want to view the Progress of Migrate window during the migration process, click Dismiss to close the window. The migration process continues in the background.

**Note:** Alternatively, if you decide you do not want to migrate the chosen objects, you can click Terminate to end the migration process at any time. However, because terminating the operation in mid-process could yield unexpected results, consider allowing the process to conclude and then migrating the undesired objects back to the original machine.

# Swapping the Master and Backup Machines

If you have identified one machine as a Master and a second machine as a Backup, you can choose to run the Oracle Tuxedo application from either machine. For example, you can swap the application from the Master to the Backup so that you can perform down-time maintenance on the Master machine. Or, if the Master processor loses the virtual circuit to other processors while the application is running, you can log in to the Backup machine. You can then bring up the DBBL on the backup node, and allow the application to continue running (although possibly in a degraded state).

## Setting up Master and Backup Machines

To have the option to swap machines, you must have set the names of the machines that serve as the Master and Backup nodes of your system. You can set the Master and Backup machines, by either:

- Specifying a Master and Backup machine name, when creating a new domain.

- Changing the Master, Backup Machine field in the Configuration Tab for a Domain.

For more information about the Master, Backup Machine field, see the section Master Machine (TA_MASTER) in the section Configuring Domain Resources (T_DOMAIN Class) and Machine Role (TA_ROLE) in the section Configuring Machines (T_MACHINE Class).

To swap a Oracle Tuxedo domain from one machine to another:

1. In the tree, select the domain.

2. Right-click to select Quick Menu—>Swap Master. (To be able to do this, you must have already identified one machine as a Master and a second machine as a Backup.)

3. The Administration Console displays a confirmation window, indicating whether the swap was successful.

# Cleaning a Machine

You can clean machines by removing dead, partitioned machines from the booted machine list or by checking for dead clients and servers if the machine is not partitioned.

Select Clean to clean the bulletin board on the specified machine to restart or clean up any servers that may require it. If the machine in question is partitioned, entries for processes and services identified as running on that machine are removed from all non-partitioned bulletin boards.

Occasionally, you may find it necessary to clean your machine (e.g., when the network has gone down or when clients or servers have been killed). The clean operation has the following effects:

- Checks the integrity of all accessers of the bulletin board residing on your machine, and the DBBL as well.

- Removes dead servers and restarts them if they are marked as restartable.

- Removes those resources no longer associated with any processes.

- Causes the DBBL to check the status of each BBL. If any BBL does not respond within the specified time, it is marked as partitioned.

If the machine is partitioned, entries for processes and services identified as running on the machine are removed from all non-partitioned bulletin boards. If the machine is not partitioned, any processes or services that cannot be restarted or cleaned up are removed.

If you clean a partitioned machine but the machine is still running, you have to kill the Oracle Tuxedo system application and processes and remove all of the Oracle Tuxedo system application shared memory, semaphores, and message queues by hand. A reboot is often faster.

## How to Clean a Machine

To clean a machine:

1. Select a machine in the tree.

2. Select Quick Menu—>Clean (right-click).

   A message is displayed indicating whether the Clean operation is successful or not.

# Advertising, Unadvertising, Suspending, and Resuming Services

You can perform the following service-related operations.

| To... | You can... |
| --- | --- |
| Make a service available | Advertise the service |
| Make a currently advertised service unavailable | Unadvertise the service |

| To... | You can... |
| --- | --- |
| Prevent the application from accessing a service | Suspend the service |
| Make a suspended service available to the application | Resume the service |

## Advertise a Service

To advertise a service (either from an existing service or from a service default object), complete the following steps:

1. In the tree, select the server that contains the service you want to advertise.

2. Select Quick Menu—>Advertise (right-click).

   The Advertise Service window is displayed as shown in Figure 3-15.

**Figure 3-15  Advertise Service Window**



3. Enter the name of the service to be advertised in the Service Name field.

4. Optionally, enter the name of the program function that will handle the service request in the Function Name field. The function name must be one of the functions that is allowed to receive service requests (see `buildserver(1)` ). If you leave this field blank, the entry in the Service Name field is used as the Function Name.

5. Click OK to close the dialog box and advertise the service.

   A confirmation message is displayed indicating whether the Advertise operation was successful or not.

## Unadvertise a Service

To make a currently advertised service unavailable:

1. In the tree, select the server that contains the service you want to unadvertise.

2. Select Quick Menu—>Unadvertise (right-click).

## Suspend a Service

To suspend a service:

1. In the tree, select the service you want to suspend.

2. Select Quick Menu—>Suspend (right-click).

    If the Suspend is successful, a Result of Requested Change menu is displayed.

    If the service cannot be suspended, an error message is displayed specifying the reason.

## Resume a Suspended Service

To reactivate a suspended service:

1. In the tree, select the suspended service you want to resume.

2.  Select Quick Menu—>Resume (right-click).

    A Result of Requested Change window is displayed, indicating whether the reactivation is successful.

# Suspending, Resuming, and Killing Clients

You can perform the following client-related tasks using the client Quick Menu.

| To... | You can... |
| --- | --- |
| Prevent a client from accessing the application | Suspend the client |
| Allow a suspended client to access the application again | Resume the client |
| Remove a client from the Oracle Tuxedo application | Kill the client |

## Suspend a Client

If you want to isolate your client temporarily, you can suspend it.

To suspend a client:

1. Select a client in the tree.

2. Select Quick Menu—>Suspend (right-click).

    If suspension is successful, the Oracle Tuxedo Administration Console displays a Result of Requested Change dialog box and the tree displays the suspended client icon.

    If the client cannot be suspended, the Oracle Tuxedo Administration Console displays an error message.

When you suspend a client, the client is prevented from performing the following tasks:

- Making service requests using Oracle Tuxedo tpcall(3) or tpacall(3)

- Initiating conversations using Oracle Tuxedo tpconnect(3)

- Beginning transactions using Oracle Tuxedo tpbegin(3)

- Enqueuing requests using Oracle Tuxedo tpenqueue(3)

Clients within a transaction can make these calls until they abort or commit the current transaction, at which time they are suspended.

## Resume a Suspended Client

To reactivate a suspended client:

1. In the tree, select the suspended client you want to resume.

2. Select Quick Menu—>Resume (right-click).

   The Oracle Tuxedo Administration Console displays the Result of Requested Change window, indicating whether the reactivation is successful.

## Kill a Client

A client can be killed, or abortively deactivated, to remove it from the Oracle Tuxedo application. The recommended method to kill a client is:

1. Suspend the desired client(s). (For more information about suspending clients, see Suspend a Client.)

2. In the tree, select the suspended client and select Quick Menu—>Kill (right-click).

   If the abortive deactivation is successful, a Result of Requested Change window is displayed, and the client icon is displayed as dead (icon with an X). To remove any dead items from the tree view, you need to use the Quick Menu Clean option.

   If the client cannot be killed, an error message is displayed specifying the reason. Although the Oracle Tuxedo Administration Console process will refuse to kill itself; one Oracle Tuxedo Administration Console process can be used to kill another.

   The application may not be able to kill the client due to platform or signaling restrictions. In this case, a native client is abortively terminated at its next access to Application-to-Transaction Manager Interface (ATMI), and a Workstation client connection to a workstation handler (WSH) will be preemptively torn down. The connections to all clients attached to that WSH are terminated.

## Suspending, Resuming, and Killing Workstation Handlers

You can perform the following workstation handler-related tasks using the Client Quick Menu.

| To... | You can... |
| --- | --- |
| Prevent clients from accessing a workstation handler | Suspend the workstation handler |

| To... | You can... |
| --- | --- |
| Make a suspended workstation handler available to clients | Resume the workstation handler |
| Remove a workstation handler from the Oracle Tuxedo application | Kill the workstation handler |

## Suspending Workstation Handlers

If you want to isolate the clients connected to a particular workstation handler temporarily, you can suspend it. When you suspend a workstation handler, the clients connected to it are prevented from performing any of the following tasks:

- Making service requests using tpcall(3) or tpacall(3)

- Initiating conversations using tpconnect(3)

- Beginning transactions using tpbegin(3)

- Enqueuing requests using tpenqueue(3)

In addition, the suspended workstation handler will not accept any new client connections.

**Note:** A workstation handler involved in a transaction when you suspend it is not actually suspended until the transaction is aborted or committed.

## Suspend a Workstation Handler

To suspend a workstation handler:

1. In the tree, select the workstation handler you want to suspend.

2. Select Quick Menu—>Suspend (right-click).

   If suspend is successful, a Result of Requested Change dialog box is displayed. The WorkStation Handler icon changes to indicate the suspended state.

   If the workstation handler cannot be suspended, an error message is displayed specifying the reason.

## Resume a Suspended Workstation Handler

To reactivate a suspended workstation handler:

1. In the tree, select the workstation handler you want to resume.

2. Select Quick Menu—>Resume (right-click).

   A Result of Requested Change dialog box is displayed, indicating if the reactivation is successful**.**

## Kill a Workstation Handler

A workstation handler can be killed, or abortively deactivated, to remove it from the Oracle Tuxedo application. The recommended method to kill a workstation handler is as follows:

1. Suspend the desired workstation handlers.

1. In the tree, select a suspended workstation handler.

2. Select Quick Menu—>Kill (right-click).

   If the abortive deactivation is successful, a Result of Requested Change dialog box is displayed and the object is displayed as dead (icon with an X) until you clean it out with the Quick Menu Clean option. The connections to all clients attached to that workstation handler are terminated.

   If you cannot kill the workstation handler, an error message is displayed, specifying the reason. The application may not be able to kill the workstation handler due to platform or signaling restrictions. In this case, a Workstation client connection to a workstation handler will be preemptively torn down.

# Opening and Closing Application Queue Spaces

You can use the Oracle Tuxedo Administration Console to allocate and initialize shared memory and to remove shared memory and other IPC resources (for example, message queues and semaphores). The options that perform these functions are Open and Close.

## Open an Application Queue Space

To open an application queue space:

1. In the tree, select the application queue space.

2. Select Quick Menu—>Open (right-click).

   A message is displayed indicating whether the application queue space opened successfully.

## Close an Application Queue Space

To close an application queue space:

1. In the tree, select the application queue space.

2. Select Quick Menu—>Close (right-click).

   A message is displayed to tell you whether the application queue space closed successfully.

# Deleting Application Queue Messages

You can delete an application queue message from its queue space if you do not need to keep it any longer.

## Delete Application Queue Messages

To delete the application queue message:

1. In the tree, select the queue message you want to delete. (The message must be in the ACTIVE state to be deleted.)

2. Select Quick Menu—>Delete (right-click).

   A Result of Requested Change dialog box is displayed, indicating whether the application queue transaction was deleted successfully.

# Aborting Transactions

If you must stop a transaction before it is complete, you can abort the transaction. Any data that changed as a result of that transaction is rolled back to its value at the start of the transaction. The transaction must be in either the ACTIVE, ABORTONLY, or COMCALLED state.

**Caution:** There is a very slight risk that the data roll back may not be successful. You can minimize the risk by aborting a transaction early in the process.

To abort a transaction:

1. In the tree, select the transaction.

2. Select Quick Menu—>Abort (right-click).

   The Result of Requested Change dialog box is displayed, indicating whether the transaction aborted successfully.

If you cannot abort the transaction, an error message is displayed specifying the reason.

# Creating Bridges Between Machines

A bridge is a connection or *link* between two machines in a multi-processor domain. From the Oracle Tuxedo Administration Console, you can perform the following tasks:

- Create a Bridge
- Suspend a Bridge
- Resume a Suspended Bridge
- Deactivate a Bridge

## Create a Bridge

To create a bridge between two machines, complete the following steps:

1. Right-click a machine in the tree and select Quick Menu—>Connect to display the Connect window.

2. In the Connect window, enter the logical machine identifier (LMID) of the machine to which you want to be connected.

   The LMID is the first of two strings (separated by a /) that identify a machine in the tree. The second string is the physical machine ID, or PMID. For example, suppose the Machines class in your tree includes the following entry: `SITE1/rcuw9`. In this case, the LMID is `SITE1` and the PMID is `rcuw9`.

3. Click OK to close the window and create the bridge.

   The Oracle Tuxedo Administration Console displays a message indicating whether the bridge was created successfully. If so, a bridge now connects the two sites.

## Suspend a Bridge

To suspend a bridge:

1. Select the bridge in the tree.

2. Select Quick Menu—>Suspend (right-click).

   The Oracle Tuxedo Administration Console displays a message indicating whether the bridge was suspended successfully.

## Resume a Suspended Bridge

To resume a bridge:

1. Select the suspended bridge in the tree.

2. Select Quick Menu—>Resume (right-click).

   The Oracle Tuxedo Administration Console displays a message indicating whether the bridge resumed successfully.

## Deactivate a Bridge

To deactivate a bridge:

1. Select an active bridge in the tree.

2. Select Quick Menu—>Deactivate (right-click).

   The Oracle Tuxedo Administration Console displays a message indicating whether the bridge was deactivated successfully.

# Security Administration

You can manage application security with the Oracle Tuxedo Administration Console ACL objects. The Oracle Tuxedo Administration Console allows you to create new ACL groups, set configuration parameters for ACLs, and ACL Principals. Domain ACLs also allow you to control access to an entire domain.

<a name="N_ACLGROUP"></a>

## Creating ACL Groups

To create new ACL Groups, display the Create New Object form and enter values for the fields:

1. Right-click the ACL Groups folder in the tree, and then select Quick Menu —>New.

   The Create New Object window is displayed.

2. Enter values for the fields:

   – Group Name

   – Group ID

3. Click Create to close the window and create the new ACL group.

   (Or click Cancel to close the window without creating a new group.)

# Managing Tuxedo CORBA Resources

Oracle Tuxedo system resources include CORBA interfaces, CORBA interface queues, and CORBA factories that are used in Oracle Tuxedo client and server applications. You can use the Oracle Tuxedo Administration Console to manage Oracle Tuxedo resources during run time as follows:

- Suspending and Resuming Interfaces

- Suspending and Resuming Interface Queues

- Viewing Attributes for Factories

## Suspending and Resuming Interfaces

During run time, you can perform the following actions on a CORBA interface:

| To... | You can... |
|---|---|
| Prevent clients from accessing an interface on the application | Suspend the interface |
| Allow clients to access an interface on the application | Resume the interface |

### Suspending a CORBA Interface

If you want to isolate your interface temporarily, you can suspend it.

To suspend an interface:

1. Select the interface object in the tree.

2. Select Quick Menu—>Suspend (right-click).

If the suspend is successful, the Oracle Tuxedo Administration Console displays a Result of Requested Change dialog box and the tree represents the interface as suspended.

**Note:**   If the interface cannot be suspended, the Oracle Tuxedo Administration Console displays an error message.

### Resuming a Suspended CORBA Interface

To reactivate a suspended interface:

1.  Select the suspended interface object in the tree.

2.  Select Quick Menu—>Resume (right-click).

    The Oracle Tuxedo Administration Console displays the Result of Requested Change dialog box, indicating whether the reactivation was successful.

## Suspending and Resuming Interface Queues

During run time, you can perform the following actions on a CORBA interface queue:

| To...                                                                   | You can...                  |
| ----------------------------------------------------------------------- | --------------------------- |
| Prevent clients from accessing an interface queue on the application    | Suspend the interface queue |
| Allow clients to access an interface queue on the application           | Resume the interface queue  |

### Suspending a CORBA Interface Queue

If you want to isolate your interface queue temporarily, you can suspend it.

To suspend a CORBA interface queue:

1.  Select the interface queue in the tree.

2.  Select Quick Menu—>Suspend (right-click).

    If the suspend is successful, the Oracle Tuxedo Administration Console displays a Result of Requested Change dialog box and the tree represents the interface queue as suspended.

    If the interface cannot be suspended, the Oracle Tuxedo Administration Console displays an error message.

### Resuming a Suspended CORBA Interface Queue

To reactivate a suspended CORBA interface queue:

1.  Select the suspended interface queue in the tree.

2.  Select Quick Menu—>Resume (right-click).

The Oracle Tuxedo Administration Console displays the Result of Requested Change dialog box, indicating whether the reactivation is successful.

## Viewing Attributes for Factories

The Oracle Tuxedo Administration Console only allows you to view Oracle Tuxedo CORBA factory information. You cannot make configuration changes to a factory and you cannot create new factories. You can use the Configuration Tool pane to view the interface name for a Factory, the Factory ID, and the object state.

# Monitoring System Events

You can use the Oracle Tuxedo Administration Console to monitor and track system events. The following sections describe how to use the Oracle Tuxedo Administration Console for:

- Viewing System Statistics
- Viewing System Events
- Viewing a ULOG File

## Viewing System Statistics

You can use the Oracle Tuxedo Administration Console to view current statistics for administrative class objects using the Statistics Tool window.

### Displaying the Statistics Tool Window

The Statistics Tool window provides statistics for domains, machines, servers, services, and clients. To display the Statistics Tool window, select Tools —>Stats from the menu bar or click the Stats toolbar button. The Oracle Tuxedo Administration Console displays the Statistics Tool window as shown in Figure 3-16.

The Statistics Tool displays the domain tab pages by default. You can click a tab to view the statistics for any of the other listed administrative class objects. The domain tab displays statistics for the following administrative classes in a domain:

- Machines
- Groups
- Servers

- Services

- Clients

**Figure 3-16  Statistics Tool window**
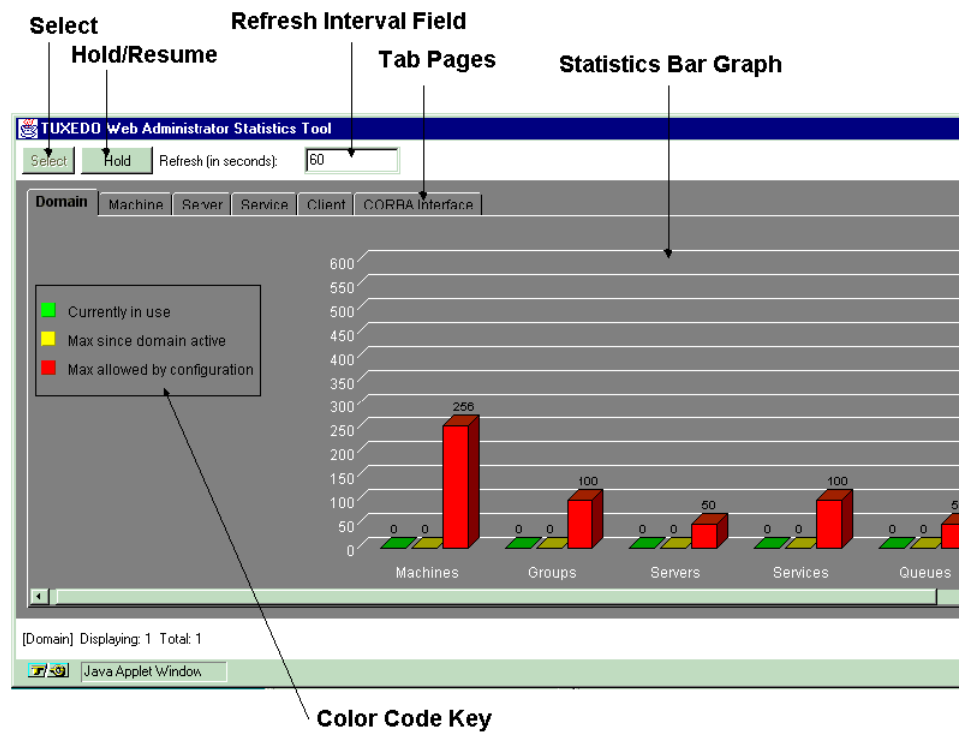
The following table lists the parts of the Statistics Tool window.

| Part | Description |
|------|-------------|
| Select button | Displays the Select Objects dialog box containing a list of all the objects for which statistics are available. |
| Hold button | Temporarily stops your statistics from being updated. The bar graph remains unchanged until you resume the update. The Hold button label is replaced by the word Resume. |
| Resume button | Resumes updating statistics. The bar graph starts displaying the changing statistics, as your statistics are updated. The button is re-labeled Hold. |
| Refresh field | Controls update rate for statistics. Intervals are specified in seconds. |
| Tab Pages | Displays the statistics for a specific administrative class: Domain, Machine, Server, Service, and Client. |
| Bar graph | Displays either a three-dimensional or two-dimensional bar graph of the statistics for an administrative object. |
| Color key | Lists the color and label for specific statistics displayed in the bar graph. |

## Changing the Refresh Rate

The Oracle Tuxedo Administration Console updates statistics at a regular interval. You can specify this interval in the Refresh field at the top of the Statistics Tool window. The interval value represents the number of seconds between Refresh operations. The default interval for refreshing the statistics is 60 seconds.

To suspend updating statistics, enter a zero in the Refresh field. To resume updating statistics, enter a number greater than zero. When you enter a zero in the Refresh field, the Hold button is re-labeled Resume; when you enter a value greater than zero, the Resume button toggles back to Hold.

## Changing the Bar Graph View

You can display the statistics on any of the tab pages as either a two-dimensional or three-dimensional bar graph. To switch between these two types of graphs:

1. Place the cursor in the bar graph image.

2. Hold down the left mouse button and move the cursor around within the bar graph.

3.  Release the button to select the type of bar graph you prefer.

## Displaying More Statistics

The Oracle Tuxedo Administration Console can display statistics for all of the available objects in a class. At the bottom of the Statistics Tool window, the status line displays the number of class objects for which statistics are currently being displayed and the total number of objects of this type for which statistics are available. Figure 3-16 lists `[Domain] Displaying: 5 Total: 5`. This line indicates statistics are displayed for all five of five possible objects.

You can select other available administrative objects and display their statistics using the Select Objects dialog box. To display this dialog box, click Select in the Statistics Tool window (upper left). As shown the following example, the Select Objects dialog box lists the available administrative objects for which you can display statistics.



To select one or more items from the list, highlight the item. Click OK to show statistics for your selection and close the Select Objects dialog box. Click Apply to show statistics for your selection and keep the Select Objects dialog box open for further selections.

<a name="EVENT"></a>

# Viewing System Events

If the system event broker server (TMSYSEVT) is configured, you can use the Oracle Tuxedo Administration Console to monitor system-generated events.

To view system events via the Event Tool:

1.  Select Tools—>Event from the menu bar or on the Event toolbar button.

**Figure 3-17  Sample Initial Event Tool Window**
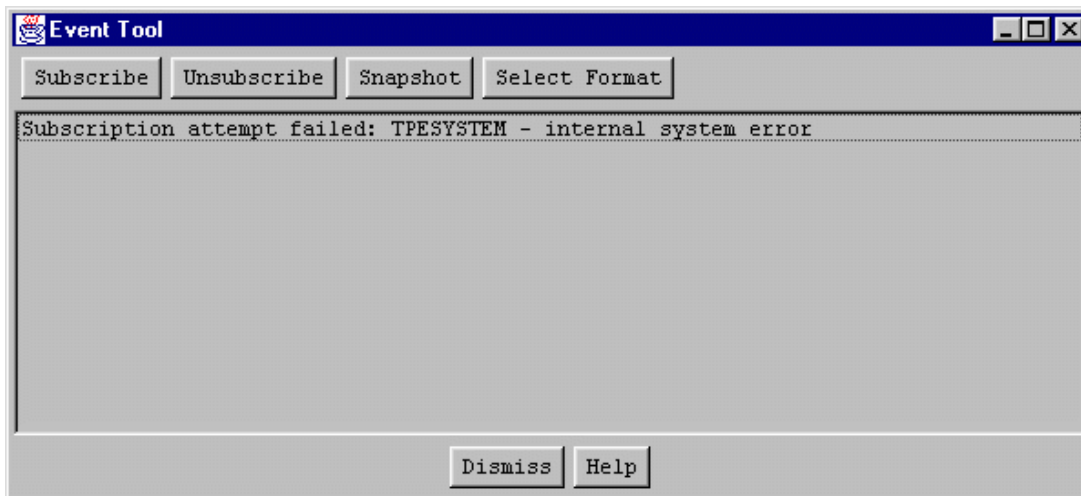


The menu bar across the top of this window offers four options as shown in the following table.

| Select This Button . . . | To . . . |
| --- | --- |
| Subscribe | Request notification of specified system events. |
| Unsubscribe | Reject further notification of specified system events. |
| Snapshot | Create a record of the data currently held by the Event Tool. |
| Select Format | Display the Format Options dialog box to select parameters for the information being collected by the Event Tool. See Figure 3-18. |
| Dismiss button | Closes the Event Tool window. |
| Help button | Provides help on this window. |

2.  To set the format for the Event Tool display, click Select Format to display the Event Tool Format Options window. Select (with a check mark) the options you want by clicking on the choice boxes, then click OK to close the window and apply your changes.

**Figure 3-18  Event Tool Format Options Window**
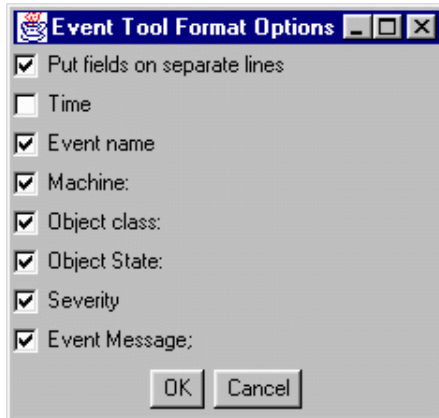


3.  Once the format options are set as needed, use the Subscribe, Unsubscribe, and Snapshot options as desired.

4.  Click Dismiss to close the Event Tool window.

5.  <a name="ULOG"></a>

# Viewing a ULOG File

Using the Oracle Tuxedo Administration Console, you can view the ULOG file from a selected machine in the active domain. You use the Logfile Tool window to display a ULOG file.

## Displaying the Logfile Tool Window

To display the Logfile Tool window, select Tools —>Logfile from the menu bar or click the Logfile toolbar button. The Logfile button displays the ULOG file from a particular machine in the active domain.

The Logfile Tool window is shown in Figure 3-19. The Logfile Tool initially displays a blank viewing area, a drop-down list box and several buttons. The following table describes the Logfile Tool window components.

| Part | Description |
| --- | --- |
| Drop-down menu | Contains a scrollable list of nodes for which you may view the log files. |
| Select Format button | Invokes the Select Format pop-up dialog box that allows you to specify the type of information you want from the log file and how you want it to be displayed. |
| Snapshot button | Captures a record from a particular log as it is being displayed. |
| Plus sign button | Invokes other Logfile function-related buttons. |
| Dismiss button | Closes the Logfile Tool. |
| Help button | Provides help on this window. |

## Selecting a Machine to View its ULOG File

To view a ULOG file for a specific object, select a machine from the drop-down menu in the top left corner of the window. (To display the menu, press and hold the left mouse button on the down arrow. To select a menu item, drag the cursor to highlight a menu item and release the the mouse button.) As shown in Figure 3-19, the Logfile Tool displays the ULOG file for the selected machine.

**Figure 3-19  Sample ULOG File in Logfile Tool Window**
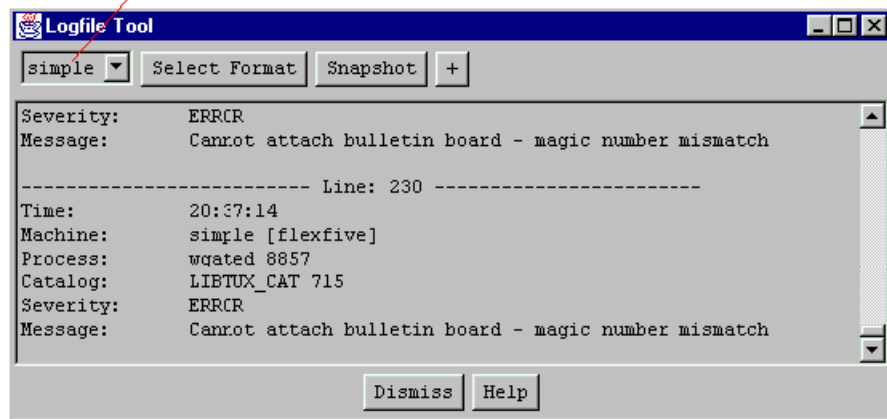


## Taking a Snapshot of the ULOG File

To save a record of the displayed ULOG file, click Snapshot on the Logfile Tool window.

## Setting the Format for the ULOG file

To modify the format of the ULOG display:

1.  Click Select Format on the Logfile Tool window.

    The Logfile Format window is displayed, as shown in Figure 3-20.

**Figure 3-20  Logfile Format OptionsWindow**
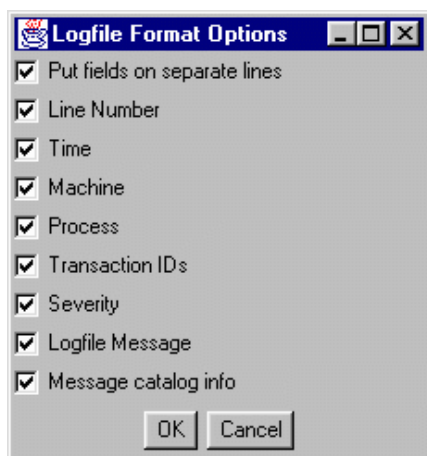


2.  Click the check boxes to enable or disable the format options. (A check mark indicates the option is selected.)

3.  Click OK to close the window and save your changes.

## Customizing the Logfile Tool Options

As shown in Figure 3-21, the Logfile Tool provides additional options. When you first invoke the Logfile Tool only a single row of buttons appears in the upper left corner of the Logfile Tool window. To display the additional Logfile Tool options, click the + sign button. The six fields that are then displayed allow you to select a ULOG file and specify which parts of it you want to see.

| Use the . . . | To . . . |
|---|---|
| Message Type scrollable menu | Choose the types of messages (such as error messages and warnings) you want to view. |
| Date field | Specify the date of the ULOG file you want to view. By default, the ULOG file for the current day (on the server) is displayed. |
| Filter field | Request a subset of entries (within the set of lines you request through the Initial Download Records field) that match a string that you specify in this field. |

| Use the . . . | To . . . (Continued) |
|---|---|
| Search field | Look for a particular record (in the portion of the ULOG file that you are viewing) by matching a string that appears in that record. Enter the string to be matched in this field. |
| Initial download records field | Specify the number of records you want to view when you download the ULOG file. (The default is 100.) |
| Maximum lines in view field | Specify the highest number of lines you want to have available for display in the window at a given time. (The default is 1000.) |

**Figure 3-21  Logfile Tool with All Options Displayed**
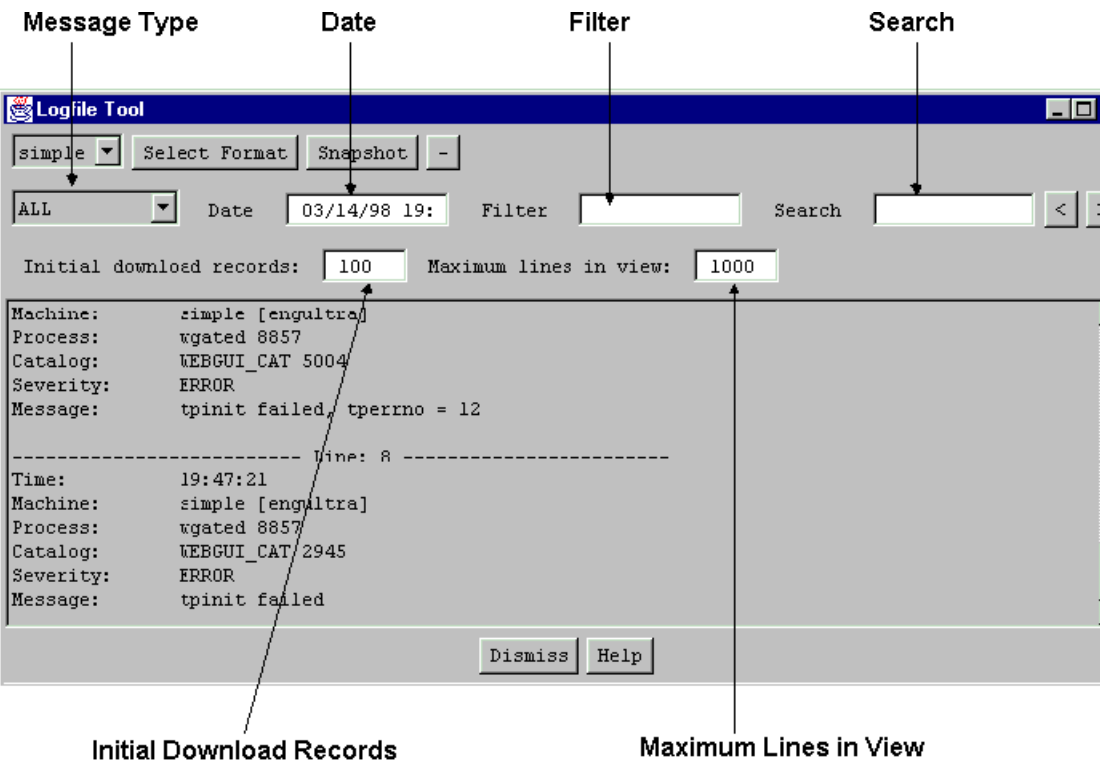
# Creating and Configuring Core Administrative Resources

The Oracle Tuxedo system core administrative resources consist of both ATMI and CORBA domains and their associated resources. The Oracle Administration Console enables the system administrator to manage the core Oracle Tuxedo ATMI resources for configuration and run-time administration.

**Note:** For detailed instructions on creating and configuring Oracle Tuxedo CORBA administrative resources, refer to Chapter 8, "Creating and Configuring Tuxedo CORBA Resources."

The following help topics define the fields for the Create New Object window and Configuration Tool tab pages when creating or configuring the Oracle Tuxedo ATMI administrative resources.

This topic includes the following sections:

- Domains

  - Opening an Existing Domain

  - Creating a New Domain

  - Configuring Domain Resources (T_DOMAIN Class)

- Machines

  - Creating Machines

  - Configuring Machines (T_MACHINE Class)

- Transactions

  - Configuring Transactions (T_TRANSACTION Class)

- Groups
  - Creating Groups
  - Configuring Groups (T_GROUP Class)
- Servers
  - Creating Servers
- Routing Tables
  - Creating Routing Tables
  - Configuring Routing Tables (T_ROUTING Class)
- Service Defaults
  - Creating Service Defaults
  - Configuring Service Defaults (T_SERVICE Class)
- Services
  - Creating Services
  - Configuring Services (T_SVCGRP Class)
- Devices
  - Creating Devices
  - Configuring Devices (T_DEVICE Class)
- Clients
  - Configuring Clients (T_CLIENT Class)
- Bridges
  - Configuring Bridges (T_BRIDGE Class)

**Note:** Descriptions fields that do not allow you to enter text or make a choice of entries are described briefly.

# Opening an Existing Domain

## &lt;a name="C_DOMAIN"&gt;&lt;/a&gt;

To open an existing domain, select Domain—>DomainName from the menu bar.

Figure 4-1 shows a sample domain menu listing the `simple` domain.

**Figure 4-1  Sample Domain Menu**



The following table describes the other Domain menu options.
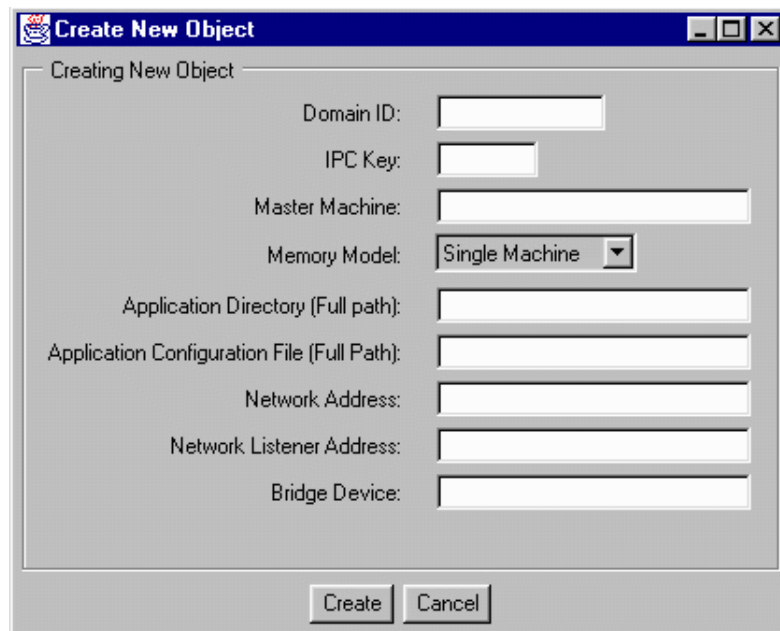
| Menu Option | Description |
| --- | --- |
| New | Creates a new domain. Use this option if the domain you want is not listed on the menu. After clicking on New, enter the pathname of the `tuxconfig` file for your domain. The GUI will then invoke the specified domain and add its name to the menu for future reference. |
| Open | Displays an existing domain that is not listed on the menu. |
| simple | Opens and displays the domain called `simple` in the tree view. |

# Creating a New Domain

Using the BEA Tuxedo Administration Console applet, you can perform all of the administrative tasks required to manage a Oracle Tuxedo domain.

To create a new domain, select Domain—>New from the menu bar to display the Create New Object window as shown in Figure 4-2. (You can also select an existing domain in the tree and select Quick Menu—>New.)

**Figure 4-2  Create New Object Window for a Domain**



Enter values for the following fields:

- Domain ID (TA_DOMAINID)

- IPC Key (TA_IPCKEY)

- Master Machine (TA_MASTER)

- Memory Model (TA_MODEL)

- Application Directory (Full Path) (TA_APPDIR)

- Application Configuration File (Full Path) (TA_TUXCONFIG)

- Network Address (TA_NADDR)

- Network Listener Address (TA_NLSADDR)

- Bridge Device (TA_BRIDGE)

Click Create to create the new domain. (Or click Cancel to close the window without creating a new domain.)

The following sections explain in detail the attribute fields on the Create New Object window for domains.

# Hexadecimal Addresses

Anywhere within the interface, fields that take an address in decimal format can also be specified in hexadecimal format when preceded by the characters 0x. Each character after the initial 0x is a number between 0 and 9 or a letter between A and F (not case sensitive). The hexadecimal format is useful for arbitrary binary network addresses such as IPX/SPX or TCP/IP. For example:

`0x0002091E9B02C112`

`0002` is the first part of a TCP/IP address and `091E` is the port number `2334` translated into a hexadecimal number. The rest of the line contains hexadecimal numbers that are translations of the elements of the IP address 155.2.193.12: `155` becomes `9B`, `2` becomes `02`, and so on.

## Domain ID (TA_DOMAINID)

## <a name="13000"></a>

## <a name="12037"></a>

### Description

The name of your Oracle Tuxedo application.

### Valid Values

A string of 0 to 30 characters. The default is `NoName`; this name is used only for display purposes in the GUI; it is not recorded as a value in the MIB.

## Example

```
bankapp
```

## IPC Key (TA_IPCKEY)

## <a name="12017"></a>

## <a name="13001"></a>

### Description

A numeric key that identifies the shared memory segment in which the structures used by your application are located. In a single-processor environment, the interprocess communication (IPC) key identifies the bulletin board. In a multiple-processor or LAN environment, this key names the message queue of the Distinguished Bulletin Board Liaison (DBBL).

### Valid Values

An integer in the range from 32769 to 262144. If you expect to define a network of machines using TCP/IP addressing, you may find it helpful to express the value of the IPC key in hexadecimal notation.

### Example

```
34956
```

## Master Machine (TA_MASTER)

## <a name="13002"></a>

## <a name="12039"></a>

### Description

The name of the master machine, which is the node on which the master copy of the configuration file (TUXCONFIG) for your application is located. In a multiprocessor environment, this machine is also the one that should run the DBBL.

### Valid Values

The format for this value is:

LMID1[,LMID2]

where:

LMID1

Is the master machine.

LMID2

Is the backup machine.

### Notes

Each logical machine identifier (LMID) contains a maximum of 30 characters. The name of the master machine (LMID1) must correspond to the local machine for inactive applications. For a single-machine application, you cannot specify a backup machine.

The administrator can assign the name or it can be specified in your application.

A master machine name is required; specifying a backup machine is optional. For more information about using a Master and Backup machine to swap machines, refer to the section, Setting up Master and Backup Machines. If you specify a backup machine, it will be used as an alternate location (for the TUXCONFIG file and the DBBL) during process relocation and booting.

### Example

```
branch1a,branch1b
```

## Memory Model (TA_MODEL)

<a name="13003"></a>

<a name="12040"></a>

### Description

The classification of your Oracle Tuxedo application as a single-machine (SHM) or multiprocessor (MP) application, specified by selecting one of two memory models (SHM or MP) from a drop-down menu.

### Valid Values

Either SHM or MP. The default is SHM.

## Application Directory (Full Path) (TA_APPDIR)

<a name="13004"></a>

<a name="13009"></a>

<a name="12081"></a>

### Description

All directories are searched when application servers are booted. The first directory serves as the current directory for all application and administrative servers booted on this machine.

### Valid Values

The format for this value is:

`pathname1[:pathname2[:pathname3]...]`

where:

`pathname`
Is a colon-separated list (for UNIX) of absolute pathnames for directories that contain Oracle Tuxedo applications.

Is a semicolon-separated list for Windows 2000. Each pathname is a string of 2 to 78 characters. A full pathnamePathnamepathnames must be specified for every application for which SECURITY has been set.

## Example

```
/home/apps/bankapp:/home/apps/personnelapp
```

## Application Configuration File (Full Path) (TA_TUXCONFIG)

## &lt;a name="13005"&gt;&lt;/a&gt;

## &lt;a name="13011"&gt;&lt;/a&gt;

## &lt;a name="12092"&gt;&lt;/a&gt;

## Description

The Application Configuration File field is equivalent to the MIB attribute `TA_TUXCONFIG`. `TUXCONFIG` is also the name of an environment variable that is used to compile the configuration file and activate the application. This internal entry is used by the Oracle Tuxedo system software after the application is activated.

## Valid Values

The format for this value is:

*pathname*

where:

*pathname*

Is the absolute pathname for the location (either a file or device) of the configuration file for your Oracle Tuxedo application. Each pathname is a string of 2 to 78 characters.

## Example

```
/home/bankapp/bankconfig
```

## Network Address (TA_NADDR)

<a name="13079"></a>

<a name="13006"></a>

<a name="12085"></a>

### Description

This field specifies the complete network address to be used by the bridge process placed on the LMID as its listening address. The listening address for a bridge is the means by which it is contacted by other bridge processes participating in the application.

### Valid Values

A TCP/IP address is specified in one of two formats:

```
//hostname:port_number
```

```
//#.#.#.#:port_number
```

where:

```
hostname
```
> The local name for the address of the machine on which the bridge resides.

```
port_number
```
> The TCP port number or name at which the bridge will listen for incoming requests. The value of *port_number* can be either a number (between 0 and 65535) or a name. If *port_number* is a name, then it must be found in the network services database on your local machine.

### Notes

In the first format, the bridge finds an address for `hostname` using the local name resolution facilities (usually DNS). `hostname` must unambiguously resolve to the address of the machine on which the bridge resides.

In the second format, the `#.#.#.#` is in **dotted decimal format**. Each `#` is a number from 0 to 255. This dotted decimal number represents the IP address of the local machine.

The address can also be specified in hexadecimal format.

The address can also be specified as an arbitrary string (with a maximum of 78 characters). For example, string addresses are used in STARLAN networks.

### Example

A local machine on which a TDomain is being run is using TCP/IP addressing and is named `backup.company.com`, with address 155.2.193.18. The port number at which the TDomain should accept requests is 2334; this port number has been added to the network services database under the name `bankapp-bridge`. The address can be specified in any of the following five formats:

```
//155.2.193.18:bankapp-bridge
```

```
//155.2.193.18:2334
```

```
//backup.company.com:bankapp-bridge
```

```
//backup.company.com:2334
```

```
//0x0002091E9B02C112
```

The last example shows the Hexadecimal Addresses format.

## Network Listener Address (TA_NLSADDR)

### &lt;a name="13080"&gt;&lt;/a&gt;

### &lt;a name="13007"&gt;&lt;/a&gt;

### &lt;a name="12086"&gt;&lt;/a&gt;

### Description

This field specifies the network address used by the UNIX `tlisten(1)` process servicing the network on the node identified by the LMID.

### Valid Values

The format for this field is the same as that of Network Address (TA_NADDR).

### Notes

`tmloadcf(1)` prints an error if you do not enter a value in this field for most machines. If you do not enter a value in this field for the MASTER LMID, `tmloadcf` prints a warning, and `tmadmin(1)` will not be able to run in administrator mode on remote machines; it will be limited to read-only operations. This also means that the backup site will be unable to reboot the master site after failure.

### Example

See the example under Network Address (TA_NADDR).

## Bridge Device (TA_BRIDGE)

&lt;a name="13081"&gt;&lt;/a&gt;

&lt;a name="13008"&gt;&lt;/a&gt;

&lt;a name="12082"&gt;&lt;/a&gt;

### Description

The device name to be used by the bridge process placed on that LMID to access the network.

### Valid Values

A string of 0 to 78 characters.

### Notes

If your networking functionality is TLI-based, the device name must be an absolute pathname. The network transport endpoint file path has the following form:

`/dev/provider_name`

For STARLAN this is `/dev/starlan`. In a multiprocessor configuration, the value takes the form `/dev/starlanN`, where `N` is the PE number.

If your networking functionality is sockets-based, you need not enter a value in this field.

### Example

`/dev/starlan`

# Configuring Domain Resources (T_DOMAIN Class)

Application resources refers to global application attributes; that is, the attributes that apply to an entire application, or domain. These attribute values serve to identify, customize, size, secure, and tune a Oracle Tuxedo application. Many of the attribute values that you set on the tabs for your application resources serve as application default values for other classes of administrative objects.

Information about your application resources is listed in eight tabbed pages:

- General Tab
- Security Tab
- Limits 1 Tab
- Limits 2 Tab
- Timers Tab
- System Parameters Tab
- Statistics 1 Tab
- Statistics 2 Tab

## General Tab

To configure general domain attributes, select a domain from the BEA Tuxedo Administration Console tree view and select the General tab in the Configuration Tool pane. The tab contains the following configuration fields:

- Domain ID (TA_DOMAINID)
- Master Machine (TA_MASTER)
- Memory Model (TA_MODEL)
- IPC Key (TA_IPCKEY)
- Object State (TA_STATE)
- License Components (TA_COMPONENTS)
- License Expiration Date (TA_LICEXPIRE)

- License Maximum Users (TA_LICMAXUSERS)

- License Serial Number (TA_LICSERIAL)

## Object State (TA_STATE)

### <a name="12034"></a>

Description

This field displays the state of your domain. A domain can be in one of the following states:

- ACTIVE

- INACTIVE

## License Components (TA_COMPONENTS)

### <a name="12390"></a>

Description

This field displays the Oracle Tuxedo products that you have licensed.

## License Expiration Date (TA_LICEXPIRE)

### <a name="12391"></a>

Description

This field displays the date on which your Oracle Tuxedo license will expire.

### License Maximum Users (TA_LICMAXUSERS)

### <a name="12392"></a>

Description

This field displays the maximum number of people allowed to use your Oracle Tuxedo license.

### License Serial Number (TA_LICSERIAL)

### <a name="12393"></a>

Description

This field displays your Oracle Tuxedo license number.

## Security Tab

The Security tab page contains the following fields:

- User ID (TA_UID)
- Group ID (TA_GID)
- Security (TA_SECURITY)
- Permission (TA_PERM)
- Authorization Service Name (TA_AUTHSVC)
- Password (TA_PASSWORD)
- Default Commit Control (TA_CMTRET)

### User ID (TA_UID)

### <a name="12033"></a>

Description

The default user ID of the Oracle Tuxedo administrator's login for this domain.

### Valid Values

An integer greater than or equal to 0 (up to the limits imposed by your operating system). The default is the effective user ID of the person running the `tmloadcf` command. Specify 0 for Windows 2000 systems.

You can override the value specified here, on individual machines, by specifying other values in the User ID field of the Security tab for the Machines class.

### Example

33

## Group ID (TA_GID)

## <a name="12010"></a>

### Description

The default group ID of the Oracle Tuxedo administrator's login for this domain.

### Valid Values

An integer greater than or equal to 0 (up to the limits imposed by your operating system). The default is the effective group ID of the person running the `tmloadcf` command. Specify 0 for Windows 2000 systems.

You can override the value specified here, on individual machines, by specifying other group IDs in the Security tab for the Machines class.

### Example

601

## Security (TA_SECURITY)

## <a name="12043"></a>

### Description

This field specifies a method of application security.

## Valid Values

Any of the methods presented in the drop-down menu. The available possibilities are shown in the following table.

| Security | Description |
|---|---|
| NONE (default) | Security is turned off. |
| APP_PW | Application Password security is enforced. Clients will be required to provide the application password during initialization. |
| USER_AUTH | User Authentication security is enforced during client initialization:<br>• Clients will be required to provide the application password.<br>• Authentication will be performed for each user. |
| ACL | ACL security is enforced during client initialization:<br>• Clients will be required to provide the application password.<br>• Authentication will be performed for each user.<br>• Access control checks will be run on service names, queue names, and event names. (If an associated ACL is not found for a name, it will be assumed that permission is granted.) |
| MANDATORY_ACL | Mandatory ACL security is enforced. The method is the same as the ACL method with one additional feature: if an associated ACL is not found for the specified service, queue, or event, then permission is denied. |

## Notes

A prerequisite for selecting a security method is the assignment of a password.

## Permission (TA_PERM)

### &lt;a name="12000"&gt;&lt;/a&gt;

Description

This field holds a number that controls read and write access to the IPC structure files to:

- The designated user

- The group with which that user is associated

- Any user who is not a member of the group

Valid Values

A four-digit octal number (of which the first character is always 0) in the range 0001 to 0777 that represents the read and write permissions (as defined by the UNIX system) that you set for the IPC structures that implement the bulletin board, and default UNIX system permissions for system and application message queues. The default is 0666, signifying that permission to read and write (6) has been given to the designated user, the designated group, and any other user.

You can override the value specified here, on individual machines, by specifying other values for Permission on the Security tab for the Machines class.

**Note:** Modifications to this attribute for an active object will not affect running servers or clients.

This attribute is specific to UNIX systems; the BEA Tuxedo Administration Console might not return this attribute for applications running on other than UNIX platforms.

## Authorization Service Name (TA_AUTHSVC)

### &lt;a name="12035"&gt;&lt;/a&gt;

Description

The name of an application authentication service that is invoked by the system for each client joining the system.

### Valid Values

A string of 0 to 15 characters, one of the following values as specified in Security (TA_SECURITY): `USER_AUTH`, `ACL`, or `MANDATORY_ACL`.

For upward compatibility, setting both `APP_PW` and `AUTHSVC` implies `USER_AUTH`. If you make those two specifications in Security (TA_SECURITY), *service_name* will be set, without notification, to `USER_AUTH` no matter what specification you make here.

If you selected the `ACL` or `MANDATORY_ACL` security method in Security (TA_SECURITY), then *service_name* must be `AUTHSVC`; this name will be used no matter what specification you make here.

If you selected the USER_AUTH in Security (TA_SECURITY), then the default for *service_name* is AUTHSVC.

A value in this field is not required if you selected no security (NONE) or APP_PW in Security (TA_SECURITY).

## Password (TA_PASSWORD)

## <a name="12046"></a>

### Description

Clear-text application password. The system automatically encrypts this information.

### Valid Values

A string of 0 to 8 characters. This field is not required if you selected no security (NONE) in Security (TA_SECURITY).

### Example

```
two4one
```

### Default Commit Control (TA_CMTRET)

### <a name="12036"></a>

Description

Initial setting of the TP_COMMIT_CONTROL characteristic for all client and server processes in a Oracle Tuxedo application.

Valid Values

Select one of the following security methods presented in the drop-down menu.

COMPLETE

> Initializes the TP_COMMIT_CONTROL characteristic to TP_CMT_COMPLETE.

**LOGGED**

> Initializes the TP_COMMIT_CONTROL characteristic to TP_CMT_LOGGED.

Notes

For details about setting this characteristic, refer to the description of the tpscmt(3) function in the *BEA Tuxedo ATMI C Function Reference* manual.

**Note:** Run-time modifications to this attribute do not affect active clients and servers.

## Limits 1 Tab

The Limits 1 tab page contains the following fields:

- Max Accessors (TA_MAXACCESSORS)
- Max Buffer Types (TA_MAXBUFTYPE)
- Max Buffer Subtypes (TA_MAXBUFSTYPE)
- Max Conversations (TA_MAXCONV)
- Max Routing Entries (TA_MAXDRT)
- Max Routing Ranges (TA_MAXRFT)
- Max Routing Data (TA_MAXRTDATA)
- Limits 2 Tab

## Max Accessors (TA_MAXACCESSORS)

### &lt;a name="12018"&gt;&lt;/a&gt;

Description

The default maximum number of processes that can have access to a bulletin board on a particular processor at any one time.

Valid Values

An integer in the range 0 to 32767. You do not need to include system administration processes, such as the BBL and the tmadmin command, when calculating this number. The default is 50.

Limitation

Changes in this field do not affect active or already configured objects in the Machines class.

## Max Buffer Types (TA_MAXBUFTYPE)

### &lt;a name="12020"&gt;&lt;/a&gt;

Description

Maximum number of buffer types that can be accommodated in the bulletin board buffer type table.

Valid Values

Displays an integer in the range 1 to 32767. The default is 6.

## Max Buffer Subtypes (TA_MAXBUFSTYPE)

### &lt;a name="12019"&gt;&lt;/a&gt;

Description

Maximum number of buffer subtypes that can be accommodated in the bulletin board buffer subtype table.

Valid Values

Displays an integer in the range 1 to 32767. The default is 50.

## Max Conversations (TA_MAXCONV)

### &lt;a name="12021"&gt;&lt;/a&gt;

Description

The maximum number of simultaneous conversations in which processes on a particular machine can be involved.

Valid Values

An integer in the range 1 to 32767. The default is 0.

This is the default attribute setting for newly configured objects in the Machines class. You can override the value specified here, on individual machines, by specifying other values in the Max Conversations field of the Limits tab for the Machines class.

## Max Routing Entries (TA_MAXDRT)

### &lt;a name="12022"&gt;&lt;/a&gt;

Description

Maximum number of routing table entries that can be accommodated in the bulletin board routing table.

Displays an integer in the range 1 to 32767. The default is 0.

One entry is required for each object in the Routing Tables class. Additional entries should be allocated to allow for run-time growth.

### Example

100

## Max Routing Ranges (TA_MAXRFT)

### <a name="12027"></a>

### Description

Maximum number of routing criteria range table entries to be accommodated in the bulletin board range criteria table.

### Valid Values

Displays an integer in the range 1 to 32767. The default is 0.

One entry is required for each range within a specification in the Routing Table Ranges field of the General tab for the Routing class. In addition, one entry is required for each object in the Routing Tables class. Additional entries should be allocated to allow for run-time growth.

### Example

1000

## Max Routing Data (TA_MAXRTDATA)

### <a name="12028"></a>

### Description

Maximum amount of string pool space to be accommodated in the bulletin board string pool table. Strings and carrays specified in the Max Routing Ranges (TA_MAXRFT) are stored in the string pool. (A carray, *character array*, is a typed buffer format in which the data is an array

of uninterpreted bytes; it always requires an associated length value.) Additional space should be allocated to allow for run-time growth.

### Valid Values

Displays an integer in the range 0 to 32760. The default is 0.

### Example

```
3500
```

# Limits 2 Tab

The Limits 2 tab page contains the following fields:

- Max Machines (TA_MAXMACHINES)
- Max Groups (TA_MAXGROUPS)
- Max Servers (TA_MAXSERVERS)
- Max Services (TA_MAXSERVICES)
- Max Queues (TA_MAXQUEUES)
- Max Global Transactions (MAXGTT)
- Max ACL Groups (TA_MAXACLGROUPS)

## Max Machines (TA_MAXMACHINES)

## &lt;a name="12025"&gt;&lt;/a&gt;

### Description

Maximum number of configured machines that can be accommodated in the bulletin board machine table.

### Valid Values

Displays an integer in the range 256 to 8191. The default is 256.

## Max Groups (TA_MAXGROUPS)

### <a name="12023"></a>

Description

Maximum number of configured server groups that can be accommodated in the bulletin board server group table.

Valid Values

Displays an integer in the range 100 to 32767. The default is 100.

## Max Servers (TA_MAXSERVERS)

### <a name="12029"></a>

Description

Maximum number of servers to be accommodated in the bulletin board server table.

Valid Values

Displays an integer in the range 1 to 8191. The default is 50.

Allowances should be made in setting this attribute for system-supplied administrative servers. Administration of each Oracle Tuxedo site adds approximately one server. Additionally, if TMSs are specified for any server groups (in the TMS Name field of the TMS tab for the Groups class), then they will be booted along with their server group and should be accounted for when determining the value to be entered in this field.

## Max Services (TA_MAXSERVICES)

### <a name="12030"></a>

Description

Maximum number of services to be accommodated in the bulletin board service table.

### Valid Values

Displays an integer in the range 1 to 32767. The default is 100.

Allowances should be made in setting this attribute for system supplied servers offering services for administrative purposes. Administration of each Oracle Tuxedo site adds approximately five services. Other administrative components such as Oracle Tuxedo Workstation, /Q, and Domain may also add administrative services that should be accounted for.

## Max Queues (TA_MAXQUEUES)

## &lt;a name="12026"&gt;&lt;/a&gt;

### Description

Maximum number of queues to be accommodated in the bulletin board queue table.

### Valid Values

Displays an integer in the range 1 to 8191. The default is 50.

## Max Global Transactions (MAXGTT)

## &lt;a name="12403"&gt;&lt;/a&gt;

### Description

Maximum number of simultaneous global transactions in which a particular machine can be involved.

### Valid Values

An integer in the range 0 to 32767. The default is 100.

You can override the value specified here, on individual machines, by specifying other values in the Max Global Transactions field of the Limits tab for the Machines class.

### Max ACL Groups (TA_MAXACLGROUPS)

### <a name="12024"></a>

Description

Maximum number of group identifiers that can be used for ACL permissions checking.

Valid Values

Displays an integer in the range 1to 16384. The default is 16384. The maximum group identifier that can be defined is 1 less than `max_aclgroups`.

## Timers Tab

The Timers tab contains the following fields:

- Basic Time Unit for Sanity Checks (TA_SCANUNIT)

- Bulletin Board Query Time (TA_BBLQUERY)

- Default Blocking Time (TA_BLOCKTIME)

- Bulletin Board Waiting Time (TA_DBBLWAIT)

- Sanity Checking Timer (TA_SANITYSCAN)

### Basic Time Unit for Sanity Checks (TA_SCANUNIT)

### <a name="12032"></a>

Description

Interval of time (in seconds) between periodic scans by the system. Periodic scans are used to detect old transactions and timed-out blocking calls within service requests.

Valid Values

An integer in the range 0 to 60. The default is 0. The basic time unit (or "scan unit") must be a multiple of 5.

Passing a value of 0 for this attribute on a SET operation causes the attribute to be reset to its default value.

## Bulletin Board Query Time (TA_BBLQUERY)

### <a name="12001"></a>

### Description

The amount of time (expressed in scan units) between DBBL status checks on registered BBLs. The DBBL checks to ensure that all BBLs have reported in within the period specified in this field. If a BBL has not been heard from, the DBBL sends a message to that BBL asking for status. If no reply is received, the BBL is partitioned.

### Valid Values

An integer in the range 0 to 32767. The default is 0. This attribute value should be set to at least twice $scan\_unit$ (the value specified in the Basic Time Unit for Sanity Checks (TA_SCANUNIT)). The default is an integer number that is approximately equal to 300 divided by $scan\_unit$.

## Default Blocking Time (TA_BLOCKTIME)

### <a name="12002"></a>

### Description

The minimum amount of time (expressed in scan units) that a blocking ATMI call will block before timing out. The DBBL checks to ensure that all BBLs have reported in within the period specified in this field. If a BBL has not been heard from, the DBBL sends a message to that BBL asking for status. If no reply is received, the BBL is partitioned.

### Valid Values

An integer in the range 0 to 32767. The default is an integer number that is approximately equal to 50 divided by $scan\_unit$.

Passing a value of 0 for this attribute on a SET operation will cause the attribute to be reset to its default value.

## Bulletin Board Waiting Time (TA_DBBLWAIT)

### <a name="12009"></a>

### Description

The maximum amount of time (expressed in scan units) that a DBBL should wait for replies from its BBLs before timing out. The DBBL checks to ensure that all BBLs have reported in within the period specified in this field. If a BBL has not been heard from, the DBBL sends a message to that BBL asking for status. If no reply is received, the BBL is partitioned.

### Valid Values

An integer in the range 0 to 32767. The default is an integer number that is approximately equal to 20 divided by *scan_unit*.

Passing a value of 0 for this attribute on a SET operation will cause the attribute to be reset to its default value.

## Sanity Checking Timer (TA_SANITYSCAN)

### <a name="12031"></a>

### Description

The amount of time (expressed in scan units) between basic sanity checks of the system. The DBBL checks to ensure that all BBLs have reported in within the period specified in this field. If a BBL has not been heard from, the DBBL sends a message to that BBL asking for status. If no reply is received, the BBL is partitioned.

Sanity checking includes client/server viability checks done by each BBL for clients/servers running on the local machine as well as BBL status check-ins (MP mode only).

### Valid Values

An integer in the range 0 to 32767. The default is an integer number that is approximately equal to 20 divided by *scan_unit*.

Passing a value of 0 for this attribute on a SET operation will cause the attribute to be reset to its default value.

# System Parameters Tab

The System Parameters tab page contains the following fields:

- Load Balancing Enabled (TA_LDBAL)

- Notification (TA_NOTIFY)

- User Signal (TA_USIGNAL)

- Options (TA_OPTIONS)

- System Access (TA_SYSTEM_ACCESS)

## Load Balancing Enabled (TA_LDBAL)

## <a name="12038"></a>

### Description

Whether to enable load balancing, specified by selecting either `Yes` or `No` from a drop-down menu.

### Valid Values

Enter either `Yes` or `No`.

### Limitation

Modifications to this field do not affect clients and servers.

## Notification (TA_NOTIFY)

## <a name="12041"></a>

### Description

Default notification detection method to be used by the system for unsolicited messages sent to client processes, chosen from a drop-down menu.

### Valid Values

Any of the methods presented in the drop-down menu. The available choices are shown in the following table.

| Method | Description |
| --- | --- |
| Dip in (poll) | Specifies that dip-in-based notification detection should be used. The system will only detect notification messages on behalf of a client process while within ATMI calls. The point of detection within any particular ATMI call is not defined by the system, and dip-in detection will not interrupt blocking system calls. |
| Signal | Specifies that signal-based notification detection should be used. The system sends a signal to the target client process after the notification message has been made available. The system installs a signal catching routine on behalf of clients selecting this method of notification. |
| Ignore | Specifies that by default, notification messages are to be ignored by application clients. This is appropriate in applications that specify that only clients requesting notification at `tpinit(3)` time should receive unsolicited messages |

### Notes

Once unsolicited messages are detected, they are made available to the application through the unsolicited message handling routine that has been defined by the application and identified by the UNIX `tpsetunsol(3)` function.) This default value can be overridden on a per-client basis using the appropriate `tpinit(3)` flag value.

### Limitation

Run-time modifications to this attribute do not affect active clients. All signaling of client processes is done by administrative system processes and not by application processes. Therefore, only clients running with the same UNIX system user identifier can be notified using the SIGNAL method.

## User Signal (TA_USIGNAL)

## &lt;a name="12045"&gt;&lt;/a&gt;

### Description

Displays which of two signals to be used for signal-based notification.

## Valid Values

Either `SIGUSR1` or `SIGUSR2`.

## Options (TA_OPTIONS)

## <a name="12042"></a>

### Description

Comma-separated list of application options in effect.

### Valid Values

The available choices are shown in the following table.

| Option | Description |
|--------|-------------|
| LAN | Networked application |
| MIGRATE | Allow server group migration |
| ACCSTATS | Exact statistics (SHM mode only) |

### Notes

The default is no options.

### Limitation

Only the `ACCSTATS` option can be set or reset in an active application.

### Example

`LAN,MIGRATE,ACCSTATS`

### System Access (TA_SYSTEM_ACCESS)

### <a name="12044"></a>

Description

Default mode used by Oracle Tuxedo libraries within application processes to gain access to Oracle Tuxedo internal tables.

Valid Values

The available choices are shown in the following table.

| Option | Description |
| --- | --- |
| FASTPATH | Specifies that Oracle Tuxedo's internal tables are accessible by Oracle Tuxedo libraries via unprotected shared memory for fast access. |
| PROTECTED | Specifies that Oracle Tuxedo's internal tables are accessible by Oracle Tuxedo libraries via protected shared memory for safety against corruption by application code. |
| NO_OVERRIDE | Specifies that the mode selected cannot be overridden by an application process using flags available for use with tpinit(3). |

Notes

The default is FASTPATH.

Limitation

Updates to this attribute value in a running application affect only newly started clients and newly configured objects in the Servers class.

Example

FASTPATH, NO_OVERRIDE

## Statistics 1 Tab

The Statistics 1 tab contains the following fields:

- Highest Number of Machines (TA_HWMACHINES)

- Highest Number of Queues (TA_HWQUEUES)

- Highest Number of Routing Table Entries (TA_HWDRT)

- Highest Number of Routing Criteria Entries (TA_HWRFT)

- Largest Size of Routing Table String Pool (TA_HWRTDATA)

- Highest Number of BBL Servers in Use (TA_HWSERVERS)

- Highest Number of Groups (TA_HWGROUPS)

- Highest Number of Services (TA_HWSERVICES)

## Highest Number of Machines (TA_HWMACHINES)

## &lt;a name="12012"&gt;&lt;/a&gt;

### Description

This field displays the high-water number of configured machines (since Oracle Tuxedo was booted or you reset the counter for this field).

## Highest Number of Queues (TA_HWQUEUES)

## &lt;a name="12013"&gt;&lt;/a&gt;

### Description

This field displays the high-water number of bulletin board queue table entries that have been used since Oracle Tuxedo was booted.

## Highest Number of Routing Table Entries (TA_HWDRT)

### <a name="12011"></a>

Description

> This field displays the maximum number of routing table entries that can be accommodated in the bulletin board routing table. One entry is required for each object in the Routing Tables class. Additional entries should be allocated to allow for run-time growth.

## Highest Number of Routing Criteria Entries (TA_HWRFT)

### <a name="12014"></a>

Description

> This field displays the maximum number of routing criteria range table entries accommodated in the bulletin board range criteria table. One entry is required for each individual range specified in the Routing Table Ranges field (on the General tab for the Routing Tables class). In addition, one entry is required for each object in the Routing Tables class. Additional entries should be allocated to allow for run-time growth.

## Largest Size of Routing Table String Pool (TA_HWRTDATA)

### <a name="12015"></a>

Description

> This field displays the maximum string pool space accommodated in the bulletin board string pool table. Strings and `carray`s specified in the Routing Table Ranges field (on the General tab for the Routing Tables class) are stored in the string pool. Additional space should be allocated to allow for run-time growth.

### Highest Number of BBL Servers in Use (TA_HWSERVERS)

<a name="12016"></a>

Description

This field displays the high-water number of in-use bulletin board server table entries.

### Highest Number of Groups (TA_HWGROUPS)

<a name="12048"></a>

Description

This field displays the high-water number of in-use bulletin board server group table.

### Highest Number of Services (TA_HWSERVICES)

<a name="12049"></a>

Description

This field displays the high-water number of in-use bulletin board service table entries.

## Statistics 2 Tab

The Statistics 2 tab contains the following fields:

- Current Number of Machines (TA_CURMACHINES)
- Current Number of Queues (TA_CURQUEUES)
- Current Number of Routing Table Entries (TA_CURDRT)
- Current Number of Routing Criteria Entries (TA_CURRFT)
- Current Size of Routing Table String Pool (TA_CURRTDATA)
- Current Number of BBL Servers In Use (TA_CURSERVERS)
- Current Number of Groups (TA_CURGROUPS)

- Current Number of Services (TA_CURSERVICES)
- Current Number of Type Entries (TA_CURTYPE)
- Current Number of Subtype Entries (TA_CURSTYPE)

## Current Number of Machines (TA_CURMACHINES)

### <a name="12004"></a>

Description

This field displays the number of machines currently configured.

## Current Number of Queues (TA_CURQUEUES)

### <a name="12005"></a>

Description

This field displays the number of bulletin board queue table entries currently being used.

## Current Number of Routing Table Entries (TA_CURDRT)

### <a name="12003"></a>

Description

This field displays the number of bulletin board routing table entries currently being used.

## Current Number of Routing Criteria Entries (TA_CURRFT)

### <a name="12006"></a>

Description

This field displays the number of bulletin board routing criteria range table entries currently being used.

### Current Size of Routing Table String Pool (TA_CURRTDATA)

<a name="12007"></a>

Description

This field displays the current size of the routing table string pool.

### Current Number of BBL Servers In Use (TA_CURSERVERS)

<a name="12008"></a>

Description

This field displays the number of bulletin board server table entries currently being used.

### Current Number of Groups (TA_CURGROUPS)

<a name="12404"></a>

Description

This field displays the number of bulletin board server group table entries that are currently being used.

### Current Number of Services (TA_CURSERVICES)

<a name="12405"></a>

Description

This field displays the number of bulletin board service table entries that are currently being used.

### Current Number of Type Entries (TA_CURTYPE)

### &lt;a name="12406"&gt;&lt;/a&gt;

Description

This field displays the number of bulletin board type table entries that are currently being used.

### Current Number of Subtype Entries (TA_CURSTYPE)

### &lt;a name="12407"&gt;&lt;/a&gt;

Description

This field displays the number of bulletin board subtype table entries that are currently being used.

# Creating Machines

# &lt;a name="N_MACHINE"&gt;&lt;/a&gt;

The following help topics describe what information to enter in the Create New Object window for creating machines.

To create a new Machine:

1. Select the Machine folder in the tree view.

2. Select Quick Menu —>New (right-click) to display the New Object window.

3. Enter values for the following fields:

   – Machine Name (TA_PMID)

   – Logical Machine Name (TA_LMID)

   – Tuxedo Root Directory (Full Path) (TA_TUXDIR)

   – Application Configuration File (Full Path) (TA_TUXCONFIG)

   – Application Directory (Full Path) (TA_APPDIR)

   – Network Address (TA_NADDR)

- Network Listener Address (TA_NLSADDR)

- Bridge Device (TA_BRIDGE)

4. Click the Create button to close the window and create the new Machine.

The rest of this section describes the fields on the Create New Object window.

## Machine Name (TA_PMID)

## &lt;a name="13010"&gt;&lt;/a&gt;

## &lt;a name="12087"&gt;&lt;/a&gt;

### Description

Physical machine identifier used by Oracle Tuxedo (specifically, in the TM_MIB) for mapping application resources to machines configured for this application.

### Valid Values

A string of 1 to 30 characters.

For UNIX users: The value of Machine Name is the node name returned when you run the `uname -a` command on this machine.

### Example

```
jupiter
```

## Logical Machine Name (TA_LMID)

**<a name="13013"></a>**

**<a name="12096"></a>**

**<a name="12087"></a>**

### Description

Logical machine identifier used, in the TM_MIB definition, as the sole means of mapping application resources to configured machines.

### Valid Values

A string of 1 to 30 characters.

You have the option of assigning a logical name to a backup machine in case the first machine fails.

### Example

```
SITE1
```

### Tuxedo Root Directory (Full Path) (TA_TUXDIR)

<a name="13012"></a>

<a name="12093"></a>

Description

TUXDIR is the absolute pathname of the directory (on this machine) in which the Oracle Tuxedo software resides.

Valid Values

A string of 2 to 78 characters. This directory should not be on a remote filesystem.

Example

```
/home/tuxedo
```

## Configuring Machines (T_MACHINE Class)

<a name="T_MACHINE"></a>

<a name="C_MACHINE"></a>

Machine attributes are listed in nine tab pages with the following labels:

- General Tab
- Networking Tab
- Security Tab
- Limits Tab
- Transaction Log Tab
- Statistics 1 Tab
- Statistics 2 Tab
- Statistics 3 Tab

- ACL Statistics Tab

# General Tab

The General tab for machine attributes contains the following fields:

- Machine Name (TA_PMID)

- Logical Machine Name (TA_LMID)

- Machine Type (TA_TYPE)

- ULOG Prefix (TA_ULOGPFX)

- Tuxedo Root Directory (Full Path) (TA_TUXDIR)

- Application Directory (Full Path) (TA_APPDIR)

- Application Configuration File (Full Path) (TA_TUXCONFIG)

- Environment File (TA_ENVFILE)

- Object State (TA_STATE)

## Machine Type (TA_TYPE)

## <a name="12094"></a>

### Description

Machine type. Machines are grouped on the basis of like data representations.

### Valid Values

A string of 0 to 15 characters. This value is used only for comparison. The default is no entry, that is, the null string, which matches any other machine for which a value has not been specified in this field.

This parameter is used for comparing machines. If the same type has been assigned to two machines, data encoding/decoding is bypassed when data is sent between them. We recommend specifying a type when your application involves a heterogeneous network of machines or when different compilers are used on the machines in the network.

Example

486

## ULOG Prefix (TA_ULOGPFX)

### <a name="12095"></a>

Description

The directory in which the user log error message file (for all servers, clients, and administrative processes executed on this machine) will be stored. The directory is specified as an absolute pathname (not including the filename) for the user log error message file.

Valid Values

A string of 0 to 78 characters. This directory should not be on a remote filesystem. The default is `$APPDIR/ULOG`.

Modifications to this attribute for an active object will not affect running servers or clients.

Example

/usr2/apps/bank/ULOG

where `/usr2/apps/bank` is the value of `$APPDIR`.

## Environment File (TA_ENVFILE)

### <a name="12084"></a>

Description

The pathname of the file that contains variables to be added to the environment for all servers on this machine.

Valid Values

A string of 0 to 78 characters. The default is no entry, that is, the null string.

### Example

```
/home/bankapp/bankenv
```

## Object State (TA_STATE)

### Description

<a name="12080"></a>

This field displays the state of your machine. A machine can be in one of the following states:

- ACTIVE

- INACTIVE

- PARTITIONED

# Networking Tab

The Networking tab for machine attributes contains the following fields:

- Network Address (TA_NADDR)

- Network Listener Address (TA_NLSADDR)

- Bridge Device (TA_BRIDGE)

- Network Load (TA_TMNETLOAD)

## Network Load (TA_TMNETLOAD)

<a name="12075"></a>

### Description

The additional load to be added during load balancing, specifically, when computing the cost of sending a service request from this machine to another.

### Valid Values

An integer in the range 0 to 32767. The default is 0.

# Security Tab

The Security tab for machine attributes contains the following fields:

- User ID (TA_UID)
- Group ID (TA_GID)
- Permission (TA_PERM)
- Minimum Encryption Bits (TA_MINENCRYPTBITS)
- Maximum Encryption Bits (TA_MAXENCRYPTBITS)

## User ID (TA_UID)

### &lt;a name="12077"&gt;&lt;/a&gt;

### Description

UNIX system user identifier for the Oracle Tuxedo application administrator on this machine.

### Valid Values

An integer in the range 0 to 2147483647. The default is the value entered in the User ID field on the Security tab for the Application Resources (T_DOMAIN) class. Administrative commands such as `tmboot(1)`, `tmshutdown(1)`, and `tmadmin(1)` must be run, on this machine, by the owner of this ID. Application and administrative servers on this machine will be started as this user. IPC structures created for the bulletin board are associated with this ID.

### Limitation

This is a UNIX system-specific attribute that cannot be returned if the platform on which the application is being run is not UNIX-based.

### Example

466

## Group ID (TA_GID)

### <a name="12053"></a>

Description

UNIX system group identifier for the Oracle Tuxedo application administrator on this machine. (The administrator for this machine is the owner of the user ID, as defined by the value you entered in the User ID (TA_UID).)

Valid Values

An integer in the range 0 to 2147483647. The default is the value entered in the User ID field on the Security tab for the Application Resources (T_DOMAIN) class.

Limitation

This is a UNIX system-specific attribute that cannot be returned if the platform on which the application is being run is not UNIX-based.

Example

512

## Permission (TA_PERM)

### <a name="12049"></a>

Description

The numeric permissions associated with the IPC structures that implement the bulletin board.

Valid Values

A four-digit octal number (of which the first character is always 0) in the range 0001 to 0777 that represents the read and write permissions (as defined by the UNIX system) that you set for the IPC structures that implement the bulletin board, and default UNIX system permissions for system and application message queues. For example, the value 0666 signifies that permission to read and write (6) has been given to the designated user, the designated group, and any other user. The default is the value entered in the User ID field on the Security tab for Application Resources (T_DOMAIN)).

This attribute is specific to UNIX systems; the BEA Tuxedo Administration Console might not return this attribute for applications running on other than UNIX platforms.

### Example

0600

## Minimum Encryption Bits (TA_MINENCRYPTBITS)

### <a name="12047"></a>

### Description

The minimum encryption level that will be accepted by a process when data flowing over Oracle Tuxedo network links has been encrypted.

### Valid Values

One of the methods presented in the drop-down menu. The available choices are shown in the following table.

| Security | Description |
|----------|-------------|
| 0 | No encryption |
| 40 | Encryption key length (in bits) |
| 128 | Encryption key length (in bits) |

**Note:** Modifications to this attribute for an active object will not affect running servers or clients.

### Notes

The encryption level is the size of the encryption key used for decrypting (expressed in bits). The default is 0, that is, no encryption. When establishing a network link to this machine, at least this minimum level of encryption is required. If this minimum level of encryption cannot be met, link establishment will fail.

Minimum and maximum encryption levels must be specified for every machine participating in a Oracle Tuxedo domain. During a communication session, the minimum encryption level will

be used if either participating process does not support the maximum level that has been specified for the current machine.

## Maximum Encryption Bits (TA_MAXENCRYPTBITS)

### <a name="12048"></a>

Description

The maximum encryption level that will be accepted by a process when data flowing over Oracle Tuxedo network links has been encrypted.

Valid Values

One of the methods presented in the drop-down menu. The available choices are shown in the following table.

| Security | Description |
|----------|-------------|
| 0 | No encryption |
| 40 | Encryption key length (in bits) |
| 128 | Encryption key length (in bits) |

Notes

The encryption level is the size of the encryption key used for decrypting (expressed in bits). The default is 28. When establishing a network link to this machine, up to this level of encryption will be accepted.

Minimum and maximum encryption levels must be specified for every machine participating in a Oracle Tuxedo domain. During a communication session, the minimum encryption level will be used if either participating process does not support the maximum level that has been specified for the current machine.

# Limits Tab

The Limits tab for machine attributes contains the following fields:

- Max Accessors (TA_MAXACCESSORS)

- Max Conversations (TA_MAXCONV)

- Max Workstation Clients (TA_MAXWSCLIENTS)

- Max Global Transactions (TA_MAXGTT)

- Spin Count (TA_SPINCOUNT)

- Compression Limit (TA_CMPLIMIT)

- Max ACL Entries in Cache (TA_MAXACLCACHE)

## Max Accessors (TA_MAXACCESSORS)

### <a name="12057"></a>

Description

The maximum number of clients and servers that can have access to a bulletin board on a particular processor at any one time.

Valid Values

An integer in the range 0 to 32767. You do not need to include system administration processes, such as the BBL, when calculating this number, but all application servers, clients, and TMS servers should be counted. If the application is activating workstation listeners on this site, then both the listeners and the potential number of workstation handlers that may be activated should be counted. The default is the value entered in the Max Accessors (TA_MAXACCESSORS) on the Limits 1 Tab tab for Application Resources (T_DOMAIN).

Example

150

## Max Conversations (TA_MAXCONV)

### &lt;a name="12058"&gt;&lt;/a&gt;

Description

The maximum number of simultaneous conversations in which processes on a particular machine can be involved.

Valid Values

An integer in the range 1 to 32767. The default is the value you specified in the Max Conversations (TA_MAXCONV) on the Limits 1 Tab tab for Application Resources (T_DOMAIN).

This is the default attribute setting for newly configured objects in the Machines class. The maximum number of simultaneous conversations per server is 64. The maximum number of simultaneous conversations per server is 64.

Example

400

## Max Workstation Clients (TA_MAXWSCLIENTS)

### &lt;a name="12060"&gt;&lt;/a&gt;

Description

The number of accesser entries on this machine to be reserved for Workstation clients only.

Valid Values

An integer in the range 1 to 32767. The default is 0.

This field is active only when the WorkStation feature is being used. The number specified here takes a portion of the total accesser slots specified with the value entered in the Max Accessors (TA_MAXACCESSORS). The appropriate setting of this parameter helps to conserve IPC resources since Workstation client access to the system is multiplexed through a Oracle Tuxedo supplied surrogate, the workstation handler.

The value of this field must be less than the value of the Max Accessors (TA_MAXACCESSORS); setting this field to a greater value is an error.

Example

150

## Max Global Transactions (TA_MAXGTT)

### <a name="12059"></a>

Description

Maximum number of simultaneous global transactions in which this machine can be involved

Valid Values

An integer in the range 1 to 32767. The default is the value you specified in the Max Global Transactions (MAXGTT) on the Limits 2 Tab tab for Application Resources (T_DOMAIN).

Example

10000

## Spin Count (TA_SPINCOUNT)

### <a name="12072"></a>

Description

The number of attempts that should be made at the user level to lock the bulletin board before blocking processes on a UNIX semaphore.

Valid Values

An integer greater than or equal to 0. The default is 0. A value of 0 indicates that the spin count built into the delivered binary should be used. If set, this parameter causes the TMSPINCOUNT environment variable to be ignored. This behavior varies from platform to platform.

Default values are built into the Oracle Tuxedo binaries on each machine. These defaults can be overridden at run time for tuning purposes using this attribute. The spin count can be reset to the

default built-in value for the site by resetting this attribute value to 0. The system uses the TMSPINCOUNT environment variable if the value is not set here or in the ubbconfig file.

### Example

10

## Compression Limit (TA_CMPLIMIT)

## <a name="12083"></a>

### Description

Threshold message size at which compression will occur for remote traffic and, optionally, for local traffic.

### Valid Values

*remote*[,*local*]

### Notes

The values of *remote* and *local* can be either nonnegative numeric values or the string MAXLONG, which is dynamically translated to the maximum long setting for the machine. The default is MAXLONG,MAXLONG.

If you enter a value only for *remote*, the value of *local* will be set to MAXLONG.

## Max ACL Entries in Cache (TA_MAXACLCACHE)

## <a name="12410"></a>

### Description

Number of entries in the cache used for ACL entries when TA_SECURITY is set to ACL or MANDATORY_ACL. The appropriate setting of this parameter helps to conserve shared memory resources and yet reduce the number of disk access needed to perform ACL checking.

### Valid Values

An integer in the range 1 to 32767. The default is 100.

# Transaction Log Tab

The Transaction Log tab for machine attributes contains the following fields:

- TLOG Name (TA_TLOGNAME)
- TLOG Size (TA_TLOGSIZE)
- TLOG Device (TA_TLOGDEVICE)

## TLOG Name (TA_TLOGNAME)

### <a name="12091"></a>

### Description

The name of the DTP transaction log for this machine.

### Valid Values

A string of 0 to 30 characters. The default is TLOG.

If more than one DTP transaction log (or TLOG) exists on the same TLOG device (TLOG Device (TA_TLOGDEVICE)), they must have unique names. The name you specify in this field must not match any other name in any other table in the configuration in which the TLOG table is created.

### Example

JUPITERLOG

## TLOG Size (TA_TLOGSIZE)

### <a name="12074"></a>

### Description

The number of pages of the DTP transaction log (or "TLOG") for this machine.

## Valid Values

An integer in the range 1 to 2048, represents the number of pages. The default is 100. The value you enter is entirely application-dependent. After a global transaction completes, the TLOG records are no longer needed and are thrown away.

### TLOG Device (TA_TLOGDEVICE)

### &lt;a name="12090">&lt;/a>

#### Description

The full pathname for the device (raw slice) or UNIX system file that contains the DTP transaction log (TLOG) for this machine. (The TLOG is stored as a Oracle Tuxedo VTOC table on the device.)

#### Valid Values

A string of 0 to 64 characters. If you do not enter a value in this field, it is assumed that there is no TLOG on this machine.

You can specify the same value that you entered (for this machine) in the Tuxedo Root Directory (Full Path) (TA_TUXDIR) on the General tab for the Machines class. If no TLOG device exists for a given machine, the associated LMID cannot be used by server groups that participate in distributed transactions.

#### Example

```
/usr2/apps/bank/TLOG
```

## Statistics 1 Tab

The Statistics 1 tab for machine attributes contains the following fields:

- Number of Conversations (TA_NUMCONV)
- Number of Dequeues (TA_NUMDEQUEUE)
- Number of Enqueues (TA_NUMENQUEUE)
- Number of Posts (TA_NUMPOST)
- Number of Requests (TA_NUMREQ)

- Number of In Use Transaction Table Entries (TA_CURGTT)

- Number of Transactions (TA_NUMTRAN)

- Number of Transactions Aborted (TA_NUMTRANABT)

- Number of Transactions Committed (TA_NUMTRANCMT)

## Number of Conversations (TA_NUMCONV)

### \<a name="12061">\</a>

### Description

Number of `tpconnect`(3) operations performed from this machine since Oracle Tuxedo was booted (or since you reset this field).

### Valid Values

*conversations*

### Notes

An integer greater than or equal to 0. If the machine is in an active state, you can reset the counter by replacing the number shown with a number that will be used as the starting point for a new count. We recommend entering 0 when you reset the counter.

Interpret the number shown in this field in terms of the starting point for the counter. For example, if you reset the counter to 1000 and the number displayed in this field later is 10000, the number of operations performed is 9000.

### Example

100

## Number of Dequeues (TA_NUMDEQUEUE)

## <a name="12062"></a>

### Description

Number of `tpdequeue`(3) operations performed from this machine since Oracle Tuxedo was booted (or since you reset this field).

### Valid Values

An integer greater than or equal to 0. If the machine is in an active state, you can reset the counter by replacing the number shown with a number that will be used as the starting point for a new count. We recommend entering 0 when you reset the counter.

Interpret the number shown in this field in terms of the starting point for the counter. For example, if you reset the counter to 1000 and the number displayed in this field later is 10000, the number of operations performed is 9000.

### Example

100

## Number of Enqueues (TA_NUMENQUEUE)

## <a name="12063"></a>

### Description

Number of `tpenqueue`(3) operations performed from this machine since Oracle Tuxedo was booted (or since you reset this field).

### Valid Values

An integer greater than or equal to 0. If the machine is in an active state, you can reset the counter by replacing the number shown with a number that will be used as the starting point for a new count. We recommend entering 0 when you reset the counter.

Interpret the number shown in this field in terms of the starting point for the counter. For example, if you reset the counter to 1000 and the number displayed in this field later is 10000, the number of operations performed is 9000.

Example

100

## Number of Posts (TA_NUMPOST)

### <a name="12064"></a>

### Description

Number of tppost(3) operations performed from this machine since Oracle Tuxedo was booted (or since you reset this field).

### Valid Values

An integer greater than or equal to 0. If the machine is in an active state, you can reset the counter by replacing the number shown with a number that will be used as the starting point for a new count. We recommend entering 0 when you reset the counter.

Interpret the number shown in this field in terms of the starting point for the counter. For example, if you reset the counter to 1000 and the number displayed in this field later is 10000, the number of operations performed is 9000.

### Example

100

## Number of Requests (TA_NUMREQ)

### <a name="12065"></a>

### Description

Number of tpacall(3) or tpcall(3) operations performed from this machine since Oracle Tuxedo was booted (or since you reset this field).

### Valid Values

An integer greater than or equal to 0. If the machine is in an active state, you can reset the counter by replacing the number shown with a number that will be used as the starting point for a new count. We recommend entering 0 when you reset the counter.

Interpret the number shown in this field in terms of the starting point for the counter. For example, if you reset the counter to 1000 and the number displayed in this field later is 10000, the number of operations performed is 9000.

### Example

100

## Number of In Use Transaction Table Entries (TA_CURGTT)

### <a name="12052"></a>

This field displays the number of transaction table entries on this machine that are currently in use.

## Number of Transactions (TA_NUMTRAN)

### <a name="12067"></a>

### Description

The number of transactions initiated (with `tpbegin`(3)) from this machine since Oracle Tuxedo was booted (or since you reset this field).

### Valid Values

An integer greater than or equal to 0. If the machine is in an active state, you can reset the counter by replacing the number shown with a number that will be used as the starting point for a new count. We recommend entering 0 when you reset the counter.

Interpret the number shown in this field in terms of the starting point for the counter. For example, if you reset the counter to 1000 and the number displayed in this field later is 10000, the number of operations performed is 9000.

### Example

2

## Number of Transactions Aborted (TA_NUMTRANABT)

<a name="12068"></a>

Description

The number of transactions aborted (tpabort(3)) from this machine since Oracle Tuxedo was booted (or since you reset this field).

Valid Values

An integer greater than or equal to 0. If the machine is in an active state, you can reset the counter by replacing the number shown with a number that will be used as the starting point for a new count. We recommend entering 0 when you reset the counter.

Interpret the number shown in this field in terms of the starting point for the counter. For example, if you reset the counter to 1000 and the number displayed in this field later is 10000, the number of operations performed is 9000.

Example

0

## Number of Transactions Committed (TA_NUMTRANCMT)

<a name="12069"></a>

Description

The number of transactions committed (with tpcommit(3)) from this machine since Oracle Tuxedo was booted (or since you reset this field).

Valid Values

An integer greater than or equal to 0. If the machine is in an active state, you can reset the counter by replacing the number shown with a number that will be used as the starting point for a new count. We recommend entering 0 when you reset the counter.

Interpret the number shown in this field in terms of the starting point for the counter. For example, if you reset the counter to 1000 and the number displayed in this field later is 10000, the number of operations performed is 9000.

Example

2

# Statistics 2 Tab

The Statistics 2 tab for machine attributes contains the following fields:

- Current Number of Clients (TA_CURCLIENTS)

- Current Number of WS Clients (TA_CURWSCLIENTS)

- Current Number of Conversations (TA_CURCONV)

- Current Number of Accessors (TA_CURACCESSORS)

- Current Service Load (TA_CURRLOAD)

- Highest Number of Accessors (TA_HWACCESSORS)

- Highest Number of Conversations (TA_HWCONV)

- Highest Number of In Use Transaction Table Entries (TA_HWGTT)

- Highest Number of Clients (TA_HWCLIENTS)

- Highest Number of WS Clients (TA_HWWSCLIENTS)

## Current Number of Clients (TA_CURCLIENTS)

### &lt;a name="12050"&gt;&lt;/a&gt;

Description

This field displays the number of clients, both native and workstation, currently logged in to this machine.

### Current Number of WS Clients (TA_CURWSCLIENTS)

#### &lt;a name="12051"&gt;&lt;/a&gt;

Description

This field displays the number of Workstation clients currently logged in to this machine

&lt;a name="12051"&gt;&lt;/a&gt;

### Current Number of Conversations (TA_CURCONV)

#### &lt;a name="12412"&gt;&lt;/a&gt;

Description

This field displays the number of active conversations with participants on this machine.

### Current Number of Accessors (TA_CURACCESSORS)

#### &lt;a name="12054"&gt;&lt;/a&gt;

Description

This field displays the number of clients and servers currently accessing the application either directly on this machine or through a workstation handler on this machine.

### Current Service Load (TA_CURRLOAD)

#### &lt;a name="12055"&gt;&lt;/a&gt;

Description

This field displays the current service load on this machine.

**Note:** If the T_DOMAIN:TA_LDBAL attribute is N or the T_DOMAIN:TA_MODEL attribute is MP, then an FML32 NULL value is returned (0).

### Highest Number of Accessors (TA_HWACCESSORS)

**&lt;a name="12056"&gt;&lt;/a&gt;**

Description

This field displays the high-water number of clients and servers accessing the application (either directly, on this machine, or through a workstation handler on this machine).

### Highest Number of Conversations (TA_HWCONV)

**&lt;a name="12055"&gt;&lt;/a&gt;**

Description

This field displays the highest number of simultaneous conversations that have occurred on this machine since Oracle Tuxedo was booted.

### Highest Number of In Use Transaction Table Entries (TA_HWGTT)

**&lt;a name="12056"&gt;&lt;/a&gt;**

Description

This field displays the high-water number of transaction table entries currently being used on this machine.

### Highest Number of Clients (TA_HWCLIENTS)

**&lt;a name="TA_HWCLIENTS"&gt;&lt;/a&gt;**

Description

This field displays the high-water number of clients (both native and workstation) logged in to this machine.

### Highest Number of WS Clients (TA_HWWSCLIENTS)

### &lt;a name="TA_HWWSCLIENTS"&gt;&lt;/a&gt;

Description

This field displays the high-water number of Workstation clients currently logged in to this machine.

## Statistics 3 Tab

The Statistics 3 tab for machine attributes contains the following fields:

- Number of Subscriptions (TA_NUMSUBSCRIBE)

- Disk Page Size (TA_PAGESIZE)

- Offset to the TLOG File (TA_TLOGOFFSET)

- Offset to the Oracle Tuxedo File (TA_TUXOFFSET)

- Total Service Load Completed (TA_WKCOMPLETED)

- Total Service Load Initiated (TA_WKINITIATED)

- Machine Role (TA_ROLE)

- Software Release (TA_SWRELEASE)

- Current Number of Accessors (TA_CURACCESSORS)

### Number of Subscriptions (TA_NUMSUBSCRIBE)

### &lt;a name="12066"&gt;&lt;/a&gt;

Description

The number of `tpsubscribe(3)` operations performed from this machine since Oracle Tuxedo was booted (or since you reset this field).

Valid Values

An integer greater than or equal to 0. If the machine is in an active state, you can reset the counter by replacing the number shown with a number that will be used as the starting point for a new count. We recommend entering 0 when you reset the counter.

Interpret the number shown in this field in terms of the starting point for the counter. For example, if you reset the counter to 1000 and the number displayed in this field later is 10000, the number of operations performed is 9000.

Example

1

## Disk Page Size (TA_PAGESIZE)

## <a name="12070"></a>

Description

This field displays the disk page size used on this machine.

## Offset to the TLOG File (TA_TLOGOFFSET)

## <a name="12073"></a>

Description

This field displays the size (in pages) of the offset between the beginning of the device and the beginning of the Oracle Tuxedo filesystem that contains the DTP transaction log for this machine.

### Offset to the Oracle Tuxedo File (TA_TUXOFFSET)

<a name="12076"></a>

Description

This field displays the size (in pages) of the offset between the beginning of the device and the beginning of the Oracle Tuxedo filesystem that contains the TUXCONFIG file for this machine. (If the offset is greater than 0, then it is placed in the environment of all servers booted on a machine.)

### Total Service Load Completed (TA_WKCOMPLETED)

<a name="12078"></a>

Description

This field displays the total service load dequeued and processed successfully by servers running on this machine.

### Total Service Load Initiated (TA_WKINITIATED)

<a name="12079"></a>

Description

This field displays the total service load enqueued by clients/servers running on this machine.

### Machine Role (TA_ROLE)

<a name="12088"></a>

Description

This field displays the role of this machine, as shown in the following table.

| Role | Description |
| --- | --- |
| MASTER | This machine is the master machine. |
| BACKUP | This machine is the backup master machine. |
| OTHER | This machine is neither the master nor the backup master machine. |

For more information about using a Master and Backup machine to swap machines, refer to the section, Setting up Master and Backup Machines.

## Software Release (TA_SWRELEASE)

## &lt;a name="12089"&gt;&lt;/a&gt;

Description

This field displays the software release for the binary on this machine (or a 0-length string if the binary is not a Oracle Tuxedo master binary).

# ACL Statistics Tab

The ACL Statistics tab for machine attributes contains the following fields:

- Highest Number of Entries in ACL Cache (TA_HWACLCACHE)
- Number of ACL Cache Hits (TA_ACLCACHEHITS)
- Number of Accesses to ACL Cache (TA_ACLCACHEACCESS)
- Number of Failed Accesses to ACL Cache (TA_ACLFAIL)

## Highest Number of Entries in ACL Cache (TA_HWACLCACHE)

## &lt;a name="12416"&gt;&lt;/a&gt;

Description

This field displays the high-water number of entries used in the ACL cache.

### Number of ACL Cache Hits (TA_ACLCACHEHITS)

<a name="12417"></a>

Description

This field displays the number of accesses to the ACL cache that resulted in a "hit" (that is, the entry was already in the cache).

### Number of Accesses to ACL Cache (TA_ACLCACHEACCESS)

<a name="12418"></a>

Description

This field displays the number of accesses to the ACL cache.

### Number of Failed Accesses to ACL Cache (TA_ACLFAIL)

<a name="12419"></a>

Description

This field displays the number of accesses to the ACL cache that resulted in an access control violation.

# Configuring Transactions (T_TRANSACTION Class)

<a name="T_TRANSACTION"></a>

<a name="C_TRANSACTION"></a>

Transaction attributes are listed in two tabbed tabs with the following labels:

- General Tab
- Miscellaneous Tab

# General Tab

The General tab of the transactions class contains the following fields:

- Logical Machine Name, Backup (TA_LMID)

- Transaction Identifier (TA_XID)

- Transaction Identifier (from tpsuspend) (TA_TPTRANID)

- Coordination Logical Machine (TA_COORDLMID)

- Object State (TA_STATE)

## Logical Machine Name, Backup (TA_LMID)

## &lt;a name="12323"&gt;&lt;/a&gt;

### Description

This field displays the retrieval machine logical machine identifier. Note that transaction attributes are primarily kept local to a site and coordinated with common transaction identifiers by transaction management servers (TMSs).

## Transaction Identifier (TA_XID)

## &lt;a name="12322"&gt;&lt;/a&gt;

### Description

This field displays the transaction identifier as returned from `tx_info(3)` mapped to a string representation. The data in this field should not be interpreted directly by the user except for equality comparison.

### Transaction Identifier (from tpsuspend) (TA_TPTRANID)

<a name="12321"></a>

Description

This field displays the transaction identifier as returned from `tpsuspend(3)` mapped to a string representation. The data in this field should not be interpreted directly by the user except for equality comparison.

### Coordination Logical Machine (TA_COORDLMID)

<a name="12319"></a>

Description

This field displays logical machine identifier of the server group responsible for coordinating the transaction.

### Object State (TA_STATE)

<a name="12318"></a>

This field displays the state of the transaction. A transaction can be in one of the following states:

- ACTIVE
- ABORTONLY
- ABORTED
- COMCALLED
- READY
- DECIDED
- SUSPENDED

# Miscellaneous Tab

The Miscellaneous tab of the transactions class contains the following fields:

- Number of Groups (TA_GRPCOUNT)
- First Group Index (TA_GRPINDEX)
- Group Number (TA_GRPNO)
- Timeout (TA_TIMEOUT)
- Runtime State (TA_GSTATE)

## Number of Groups (TA_GRPCOUNT)

## <a name="12314"></a>

### Description

This field displays the number of groups identified as participants in the transaction by the information returned from the retrieval site.

## First Group Index (TA_GRPINDEX)

## <a name="12315"></a>

### Description

This field displays the index of the first group specific attribute values (`TA_GRPNO` and `TA_GSTATE`) corresponding to this object.

## Group Number (TA_GRPNO)

## <a name="12316"></a>

### Description

This field displays the group number of the participating group.

### Timeout (TA_TIMEOUT)

**<a name="12317"></a>**

Description

This field displays time left, in seconds, before the transaction will time out on the retrieval site. This attribute value is returned only when the transaction state (TA_STATE) is ACTIVE.

### Runtime State (TA_GSTATE)

**<a name="12320"></a>**

Description

This field displays the state, at run time, of the selected transactions for the specified group. Such a transaction can be in one of the following states:

- ACTIVE
- ABORTED
- READONLY
- READY
- HCOMMIT
- HABORT
- DONE

## Creating Groups

**<a name="N_GROUP"></a>**

To create new Groups:

1. Select the Groups folder in the tree view.

2. Select Quick Menu —>New (right-click) to display the New Object window.

3. Enter values for the following fields:

   – Group Name (TA_SRVGRP)

   – Group Number (TA_GRPNO)

   – Logical Machine Name, Backup (TA_LMID)

4. Click the Create button to close the window and create the new Group.

## Group Name (TA_SRVGRP)

&lt;a name="13016"&gt;&lt;/a&gt;

&lt;a name="13022"&gt;&lt;/a&gt;

&lt;a name="13019"&gt;&lt;/a&gt;

&lt;a name="12135"&gt;&lt;/a&gt;

&lt;a name="12105"&gt;&lt;/a&gt;

&lt;a name="12151"&gt;&lt;/a&gt;

### Description

Logical name of the server group.

### Valid Values

A string of 0 to 30 characters.

The group name cannot contain an asterisk, a comma, or a colon. The group name must be unique among all group names in the Groups class and all values entered in the Logical Machine Name (TA_LMID) for objects in the Machines class.

This name is used as the value of the SRVGRP parameter in two places: in the Servers tabs (to identify a server as a part of this group), and the Services tab (to identify an instance of a service with its occurrences in the group).

Example

```
BANKB1
```

## Group Number (TA_GRPNO)

<a name="13014"></a>

<a name="12097"></a>

Description

Number associated with this server group.

Valid Values

An integer in the range 1 to 30000. The number must be unique among all entries in the Groups tabs.

Example

100

## Logical Machine Name, Backup (TA_LMID)

<a name="13015"></a>

<a name="12104"></a>

Description

Primary logical machine identifier for this server group (*LMID1*) and optional secondary logical machine identifier (*LMID2*).

Valid Values

*LMID1*[,*LMID2*]

### Notes

Each LMID (logical machine identifier) contains a maximum of 30 characters. Logical machine identifiers specified in this field must be configured already. Each name must be unique within this group's configuration.

The name of a new group cannot contain an asterisk, comma, or colon.

You are required to specify one machine; specifying a backup machine is optional. If you specify a backup machine and server group migration is enabled, then you will be able to migrate this server group to the backup machine, if necessary.

### Example

```
SITE1, SITE2
```

## Configuring Groups (T_GROUP Class)

### <a name="T_GROUP"></a>

### <a name="C_GROUP"></a>

Group attributes are listed in three tabbed tabs:

- General Tab
- TMS Tab
- Args Tab

## General Tab

The General tab contains the following fields:

- Group Name (TA_SRVGRP)
- Group Number (TA_GRPNO)
- Logical Machine Name, Backup (TA_LMID)
- Current Logical Machine (TA_CURLMID)
- Object State (TA_STATE)

### Current Logical Machine (TA_CURLMID)

### <a name="12101"></a>

Description

Name (expressed as a logical machine identifier) of the machine on which this server group is currently running.

Valid Values

A string of 1 to 30 characters. This attribute is not returned for server groups that are not active.

Example

MACH1

### Object State (TA_STATE)

### <a name="12099"></a>

Description

This field displays the state of your group. A group can be in one of the following states:

- ACTIVE

- INACTIVE

- MIGRATING

## TMS Tab

The TMS tab provides two fields.

- TMS Name (TA_TMSNAME)

- TMS Count (TA_TMSCOUNT)

## TMS Name (TA_TMSNAME)

### <a name="12103"></a>

Description

Name of the transaction manager server `a.out` file associated with this group. This file is executed by `tmboot` when you boot this server group.

Valid Values

A string of 0 to 78 characters. The value `TMS` is reserved to indicate use of the null XA interface. The default is no entry, that is, the null string.

You must enter a value in this field for any group that includes servers that will participate in distributed transactions. (A distributed transaction is a transaction executed across multiple resource managers, and possibly machines, that are started with `tpbegin`(3), and ended with a `tpcommit`(3) or `tpabort`(3) call.)

If you specify a nonempty value other than `TMS`, then you must also specify a TLOG device for the one or more machines associated with the primary and secondary logical machines for this group.

A unique server identifier is selected automatically for each TM server. You will be able to restart the servers an unlimited number of times.

Example

`srvrboot`

## TMS Count (TA_TMSCOUNT)

### <a name="12098"></a>

Description

Number of transaction manager servers to be started for the associated group, if you have specified a value for the TMS Name (TA_TMSNAME).

Valid Values

*tmscount*

## Notes

*tmscount* is an integer in the range 2 to 10. The default is 3. If you have not specified a value for the TMS Name (TA_TMSNAME), then this field is ignored.

The servers are set up in an MSSQ set automatically.

# Args Tab

The Args tab contains the following fields:

- Open Information (TA_OPENINFO)
- Close Information (TA_CLOSEINFO)

## Open Information (TA_OPENINFO)

### <a name="12102"></a>

### Description

Information needed to open a particular instance of a particular resource manager, or an indication that such information is not required for this group.

### Valid Values

A string of 0 to 256 characters, consisting, if not empty, of four substrings, separated by colons, of the format:

*rm_name*:*fsconfig*:*dbname*:*open_mode*

These subfields are described in the following table.

| Substring | Description |
|-----------|-------------|
| *rm_name* | TUXEDO/D for Oracle Tuxedo System/D databases.<br>TUXEDO/SQL for Oracle Tuxedo System/SQL databases.<br>TUXEDO/QM for Oracle Tuxedo /Q databases. |
| *fsconfig* | An environment variable that contains the pathname of the database. |

| Substring | Description |
|-----------|-------------|
| *dbname* | The name of the database. |
| *open_mode* | One of `readonly` or `readwrite`. |

**Note:** The `/D` and `/SQL` databases are only used by the sample applications provided with Oracle Tuxedo (for example, `bankapp`) and are *not* supported for use with the application code.

The default is no entry, that is, the null string.

On a Windows 2000 machine, semicolons are used instead of the colons after *fsconfig* and *dbname*.

The Open Information format depends on the requirements of the resource manager vendor. The string required by the vendor must be prefixed by *rm_name*, which is the published name of the vendor's transaction (XA) interface, followed immediately by a colon.

This field is ignored if you have not specified a name in the TMS Name (TA_TMSNAME) or if you have specified it as `TMS`. If you have specified a value for the TMS name but you have not done so for the Open Information field, it means that a resource manager exists for this group but does not require any information for executing an open operation.

Oracle Tuxedo /Q databases use the following format:

`TUXEDO/QM:`*qmconfig*`:`*qspace*

### Limitation

Run-time modifications to this attribute does not affect active servers in the group.

### Example

`TUXEDO/SQL:/usrq/apps/bank/bankdq:billing_db:readwrite`

## Close Information (TA_CLOSEINFO)

## &lt;a name="12100"&gt;&lt;/a&gt;

### Description

Information needed when the resource manager is being closed.

## Valid Values

A string of 0 to 256 characters, consisting, if not empty, of three substrings, separated by colons, of the format:

`rm_name:entity_name:open_mode`

These subfields are described in the following table.

| Substring | Description |
|-----------|-------------|
| `rm_name` | `TUXEDO/D` for Oracle Tuxedo System/D databases. `TUXEDO/QM` for Oracle Tuxedo /Q databases. |
| `entity_name` | The name of the database or queue space. |
| `open_mode` | One of `readonly` or `readwrite`. |

**The default is no entry, that is, the null string.**

On Windows 2000 and NetWare machine, semicolons are used instead of the colons after `fsconfig` and `dbname`.

The Close Information format depends on the requirements of the resource manager vendor. The string required by the vendor must be prefixed by `rm_name`, which is the published name of the vendor's transaction (XA) interface, followed immediately by a colon.

This field is ignored if you have not specified a name in the TMS Name (TA_TMSNAME) or if you have specified it as `TMS`. If you have specified a value for the TMS name but you have not done so for the Close Information field, it means that a resource manager exists for this group but does not require any information for executing a close operation.

## Limitations

Run-time modifications to this attribute will not affect active servers in the group.

## Example

`TUXEDO/D:billing_db:readwrite`

# Creating Servers

## \<a name="N_SERVER">\</a>

To create new Servers:

1. Select the Servers folder in the tree view.

2. Select Quick Menu —>New (right-click) to display the New Object window.

3. Enter values for the following fields:

   – Group Name (TA_SRVGRP)

   – Server ID (TA_SRVID)

   – Server Name (TA_SERVERNAME)

4. Click the Create button to close the window and create the new Server.

## Server ID (TA_SRVID)

## \<a name="13017">\</a>

## \<a name="12124">\</a>

### Description

A numeric identifier, selected by the administrator, that uniquely identifies a server within a group.

### Valid Values

An integer in the range 1 to 30000.

If multiple instances of servers are desired, do not use consecutive numbers. Instead, leave enough room for the system to assign additional server IDs up to the number specified in the Max Servers Activated (TA_MAX) field.

### Example

4

## Server Name (TA_SERVERNAME)

## &lt;a name="13018"&gt;&lt;/a&gt;

## &lt;a name="12133"&gt;&lt;/a&gt;

### Description

Pathname of the execpathnameutable file for the server.

### Valid Values

A string of 1 to 78 characters. The server identified in this field will run on the machines specified in the Logical Machine Name, Backup (TA_LMID) for the Groups class to which this server belongs. If you specify a relative pathname, a search will be done for the executable file in the following order:

1. The directory specified in the "Application Directory" field.

2. The bin subdirectory of the directory specified in the Tuxedo Root Directory (Full Path) (TA_TUXDIR).

3. `/bin`

4. `/usr/bin`

5. The pathname specified as the first `PATH=` entry in the machine environment file (if one exists).

### Example

```
/home/bankapp/credit_dept/ACCT
```

# Configuring Servers (T_SERVER Class)

`<a name="T_SERVER"></a>`

`<a name="C_SERVER"></a>`

Group attributes are listed in six tabbed folders:

- General Tab

- Args Tab

- Commands Tab

- System Parameters Tab

- Statistics 1 Tab

- Statistics 2 Tab

- Statistics 3 Tab

## General Tab

The General tab contains the following fields:

- Group Name (TA_SRVGRP)

- Group Number (TA_GRPNO)

- Server ID (TA_SRVID)

- Server Name (TA_SERVERNAME)

- Environment File (TA_ENVFILE)

- Current Logical Machine Name (TA_LMID)

- Object State (TA_STATE)

### Group Number (TA_GRPNO)

<a name="12420"></a>

Description

This field displays the group number associated with this server's group.

### Environment File (TA_ENVFILE)

<a name="12128"></a>

Description

This field displays the name of the file that defines the environment in which the `a.out` file is executed by `tmboot`.

Valid Values

A string of 0 to 78 characters. The default is no entry, that is, the null string.

Example

`environ`

### Current Logical Machine Name (TA_LMID)

<a name="12422"></a>

Description

This field displays the current logical machine on which the server is running.

## Object State (TA_STATE)

## <a name="12125"></a>

### Description

This field displays the state of your server. A server can be in one of the following states:

- ACTIVE

- INACTIVE

- MIGRATING

- CLEANING

- RESTARTING

- SUSPENDED

- PARTITIONED

- DEAD

# Args Tab

The Args tab contains the following fields:

- Grace Period (TA_GRACE)

- Max Servers Activated (TA_MAX)

- Min Servers Activated (TA_MIN)

- Max Generations (TA_MAXGEN)

- Activation Sequence Number (TA_SEQUENCE)

- System Access (TA_SYSTEM_ACCESS)

## Grace Period (TA_GRACE)

### <a name="12106"></a>

Description

The amount of time (in seconds) during which this server can be started.

Valid Values

An integer in the range 0 to 2147483647. The default is 86400 (the number of seconds in 24 hours. The Automatic Restart (TA_RESTART) must be set to `Yes`. The equation used to calculate the grace period is:

```
(MAXGEN - 1)
```

where `MAXGEN` is set in the Max Generations (TA_MAXGEN). As soon as one grace period ends, the next one begins. Thus setting the value of this field to `0` removes all limitations. The server can then be restarted an unlimited number of times.

## Max Servers Activated (TA_MAX)

### <a name="12107"></a>

Description

The maximum number of times this server can be reactivated.

Valid Values

An integer in the range 1 to 1001. The default is 1. Use this parameter to prevent having to repeat an entry for multiple occurrences of the same server.

Example

10

## Min Servers Activated (TA_MIN)

## <a name="12109"></a>

### Description

The minimum number of times this server can be reactivated.

### Valid Values

An integer in the range 1 to 1001. The default is 1. Use this parameter to prevent having to repeat an entry for multiple occurrences of the same server.

Limitation: Run-time modifications to this attribute will not affect a running server.

If you have specified a value in the Request Queue (TA_RQADDR) and the Min Servers Activated value is greater than 1, then an MSSQ set is formed.

## Max Generations (TA_MAXGEN)

## <a name="12108"></a>

### Description

The maximum number of times this server can be started within the period specified in the Grace Period (TA_GRACE).

### Valid Values

An integer in the range 1 to 255. The default is 1. To restart the server, the value in this field must be 2 or less.

### Limitation

Run-time modifications to this attribute will affect a running server and all other active servers with which it is sharing a request queue. However, only the selected server's configuration parameter is modified. Thus, the behavior of the application depends on the order of booting in subsequent activations unless the administrator ensures that all servers sharing a queue have the same value for this attribute.

## Activation Sequence Number (TA_SEQUENCE)

### <a name="12121"></a>

### Description

The order in which servers are activated and deactivated.

If a value is specified in this field for some servers but not for others, those with assigned numbers will be activated first. The servers for which numbers have not been assigned are then activated in the order in which they appear in the configuration file. If the same value (for "Activation Sequence Number") is assigned to more than one server, those servers may be activated in parallel.

### Valid Values

An integer in the range 1 to 9999. The default is for servers to be activated in the order in which they are listed in the configuration file, and deactivated in reverse order.

## System Access (TA_SYSTEM_ACCESS)

### <a name="12134"></a>

### Description

This field controls the allocation of shared memory space.

### Valid Values

One of the methods presented in the drop-down menu. The available choices are shown in the following table.

| Method | Description |
| --- | --- |
| FASTPATH | This method allows for faster processing, but is less secure. Processes that share memory space are vulnerable to potential software defects on one of the processes. This is the default method. |
| PROTECTED | The shared memory for Oracle Tuxedo Internal tables is not accessible from code outside of Oracle Tuxedo libraries. |

### Limitation

Run-time modifications to this attribute will not affect a running server.

# Commands Tab

The Commands tab contains the following fields:

- Executable Options (TA_CLOPT)

- Restart Command (TA_RCMD)

- Automatic Restart (TA_RESTART)

## Executable Options (TA_CLOPT)

### &lt;a name="12126"&gt;&lt;/a&gt;

### Description

Command-line options to be passed to the server when it is activated.

### Valid Values

A string of 0 to 256 characters. The default is `-A`.

### Limitation

Run-time modifications to this attribute will not affect a running server.

## Restart Command (TA_RCMD)

### &lt;a name="12129"&gt;&lt;/a&gt;

### Description

A command, specified by the application, that is executed in parallel with the system restart of an application server.

### Valid Values

A string of 0 to 78 characters. The default is no entry, that is, the null string. This command must be an executable UNIX system file.

Servers sharing a request queue (that is, equal values in the Request Queue (TA_RQADDR) on the System Parameters tab) should have equal values for this attribute. If they do not, then the first server activated will establish the run-time value associated with all servers on the queue.

### Limitation

Run-time modifications to this attribute will affect a running server and all other active servers with which it is sharing a request queue. However, only the selected server's configuration parameter is modified. Thus, the behavior of the application depends on the order of booting in subsequent activations, unless the administrator ensures that all servers sharing a queue have the same value for this attribute.

## Automatic Restart (TA_RESTART)

## <a name="12131"></a>

### Description

Determines whether or not the server is restartable.

### Valid Values

Enter either `Yes` or `No`.

The default is `No`. You must specify `Yes` in this field if server migration has been specified for this server group in the following manner: You selected MIGRATE in the Options (TA_OPTIONS) of the General tab for Configuring Domain Resources (T_DOMAIN Class), and you designated an alternate site in the Logical Machine Name, Backup (TA_LMID) of the General tab in the Groups class.

Servers sharing a request queue (that is, equal values in the Request Queue (TA_RQADDR) on the System Parameters tab) should have equal values for this attribute. If they do not, then the first server activated will establish the run-time value associated with all servers on the queue.

### Limitation

Run-time modifications to this attribute will affect a running server and all other active servers with which it is sharing a request queue. However, only the selected server's configuration parameter is modified. Thus, the behavior of the application depends on the order of booting in subsequent activations, unless the administrator ensures that all servers sharing a queue have the same value for this attribute.

# System Parameters Tab

The System Parameters tab contains the following fields:

- Request Queue (TA_RQADDR)

- Reply Queue (TA_REPLYQ)

- Request Queue Perm (TA_RQPERM)

- Reply Queue Perm (TA_RPPERM)

- Reply Message Queue ID (TA_RPID)

- Request Message Queue ID (TA_RQID)

- Process ID (TA_PID)

- Conversations Allowed (TA_CONV)

## Request Queue (TA_RQADDR)

### &lt;a name="12132"&gt;&lt;/a&gt;

Description

A symbolic name that points to the address queue of the server.

Valid Values

A string of 0 to 30 characters. The default is $GRPNO.SRVID$, where:

- $GRPNO$ is the value specified in the Group Number (TA_GRPNO) of the General tab for the Groups class.

- $SRVID$ is the value specified in the Server ID (TA_SRVID) of the General tab for the Servers class).

Multiple Server, Single Queue (MSSQ) sets are established by using the same symbolic name for more than one server or by specifying the Min Servers Activated (TA_MIN) to be greater than 1. All members of an MSSQ set must offer an identical set of services and must be in the same server group.

### Reply Queue (TA_REPLYQ)

### &lt;a name="12130"&gt;&lt;/a&gt;

#### Description

Determines whether or not a reply queue, separate from the request queue, is to be set up for this server.

#### Valid Values

Enter either `Yes` or `No`.

#### Notes

The default is `Yes`.

If you specify `Yes`, the server is a member of a Multiple Server, Single Queue (MSSQ) set and contains services programmed to receive reply messages. This ensures that the individual reply queue is created for this server. If this is not done, the reply is sent to the request queue shared by all servers of the MSSQ set, and there is no way of assuring that it will be picked up by the server waiting for it. It should be standard practice to specify `Yes` for all member servers of an MSSQ set if replies are anticipated. Servers in an MSSQ set must have identical service offerings, so it is reasonable to expect that if one server in the set expects replies, any server in the set may do likewise.

If you specify `No`, only one server uses the request queue. Replies are then picked up from the request queue without difficulty.

### Request Queue Perm (TA_RQPERM)

### &lt;a name="12120"&gt;&lt;/a&gt;

#### Description

The numeric permissions assigned to the request queue for this server.

#### Valid Values

A four-digit octal number (of which the first character is always 0) in the range 0001 to 0777 that represents the UNIX system read and write permissions (as defined by the UNIX system) that you set for the request queue for this server. For example, the value 0666 signifies that permission

to read and write (6) has been given to the designated user, the designated group, and any other user. The default is 0666. If you do not enter a value in this field, the permissions of the bulletin board, as specified by the Permission (TA_PERM) on the Security tab for the Application Resources class (T_DOMAIN), are implemented.

If the default is active, your application could be open to the owner of any login on the system, which is not recommended.

### Limitation

This attribute is specific to UNIX systems; it might not be returned if the platform on which the application is being run is not UNIX-based.

## Reply Queue Perm (TA_RPPERM)

## <a name="12119"></a>

### Description

UNIX system-style permissions assigned to the server's reply queue.

### Valid Values

A four-digit octal number, in the range 0001 to 0777, as described under Request Queue Perm (TA_RQPERM). The default is the value you entered for this field in the Permission (TA_PERM) on the Security tab for the Application Resources class (T_DOMAIN).

### Limitation

This attribute is specific to UNIX systems; it might not be returned if the platform on which the application is being run is not UNIX-based.

### Example

0666

### Reply Message Queue ID (TA_RPID)

#### &lt;a name="12425"&gt;&lt;/a&gt;

Description

This field displays the UNIX system message queue identifier for the server's reply queue. If a separate reply queue is not allocated (the Reply Queue (TA_REPLYQ) specified No), then this attribute will be the same as the value of the Request Message Queue ID (TA_RQID).

Limitation

This field is specific to UNIX systems; there might be no value displayed here if the platform on which the application is being run is not UNIX-based.

### Request Message Queue ID (TA_RQID)

#### &lt;a name="12426"&gt;&lt;/a&gt;

Description

This field displays the UNIX system message queue identifier for the server's request queue. If a separate queue is not allocated (the Reply Queue (TA_REPLYQ) specified No), then this attribute will be the same as the value of the Reply Message Queue ID (TA_RPID).

Limitation

This field is specific to UNIX systems; there might be no value displayed here if the platform on which the application is being run is not UNIX-based.

### Process ID (TA_PID)

#### &lt;a name="12424"&gt;&lt;/a&gt;

Description

This field displays the UNIX system process identifier for the server. Note that this may not be a unique attribute since servers may be located on different machines allowing for duplication of process identifiers.

### Limitation

This field is specific to UNIX systems; there might be no value displayed here if the platform on which the application is being run is not UNIX-based.

## Conversations Allowed (TA_CONV)

## <a name="12127"></a>

### Description

This field causes the number of conversations initiated through this server by `tpconnect(3)` since Oracle Tuxedo was booted (or you reset the counter) to be displayed in the Number of Conversations (TA_NUMCONV).

### Valid Values

Enter either `Yes` or `No`.

# Statistics 1 Tab

The Statistics 1 tab contains the following fields:

- Number of Conversations (TA_NUMCONV)
- Number of Dequeues (TA_NUMDEQUEUE)
- Number of Enqueues (TA_NUMENQUEUE)
- Number of Posts (TA_NUMPOST)
- Number of Requests (TA_NUMREQ)
- Total Number of Workload Completed (TA_TOTWORKL)
- Restarting Time (TA_TIMERESTART)
- Starting Time (TA_TIMESTART)
- Time Left in Current Service (TA_SVCTIMEOUT)
- Time Left for Reply (TA_TIMELEFT)

## Number of Conversations (TA_NUMCONV)

### <a name="12110"></a>

Description

This field displays the number of `tpconnect(3)` conversations initiated by this server since Oracle Tuxedo was booted (or you reset the counter).

Valid Values

An integer greater than or equal to 0.

Example

1500

## Number of Dequeues (TA_NUMDEQUEUE)

### <a name="12111"></a>

Description

This field displays the number of `tpdequeue(3)` operations initiated by this server since Oracle Tuxedo was booted (or since you reset the counter).

Valid Values

An integer greater than or equal to 0.

Example

0

## Number of Enqueues (TA_NUMENQUEUE)

### <a name="12112"></a>

Description

This field displays the number of `tpenqueue(3)` operations performed from this machine since Oracle Tuxedo was booted (or since you reset the counter).

Valid Values

An integer greater than or equal to 0.

Example

0

## Number of Posts (TA_NUMPOST)

### <a name="12113"></a>

Description

This field displays the number of `tppost(3)` operations performed from this machine since Oracle Tuxedo was booted (or you reset the counter).

Valid Values

An integer greater than or equal to 0.

Example

0

## Number of Requests (TA_NUMREQ)

<a name="12114"></a>

Description

This field displays the number of tpacall(3) or tpcall(3) operations performed from this machine since Oracle Tuxedo was booted (or since you reset the counter).

Valid Values

An integer greater than or equal to 0.

Example

0

## Total Number of Workload Completed (TA_TOTWORKL)

<a name="12123"></a>

Description

This field displays the total work load completed by this server.

Valid Values

An integer greater than or equal to 0.

For conversational servers, the value of this field indicates the workload of completed incoming conversations. This is a run-time attribute that is kept across server restart but is lost at server shutdown.

Example

0

## Restarting Time (TA_TIMERESTART)

### &lt;a name="12398"&gt;&lt;/a&gt;

### Description

This field displays the amount of time since restart.

### Valid Values

An integer greater than or equal to 0, representing the seconds since 00:00:00 UTC, January 1, 1970, as returned by the `time(2)` system call on the machine identified in the Current Logical Machine Name (TA_LMID). This value shows when the server was last started or restarted.

## Starting Time (TA_TIMESTART)

### &lt;a name="12399"&gt;&lt;/a&gt;

### Description

This field displays the amount of time since the server was first started.

### Valid Values

An integer greater than or equal to 0, representing the seconds since 00:00:00 UTC, January 1, 1970, as returned by the `time(2)` system call on the machine identified in the Current Logical Machine Name (TA_LMID). This value shows when the server was first started.

Restarts of the server do not reset this value; however, if the value entered in the Max Generations (TA_MAXGEN) on the Args tab and the amount of time specified in the Grace Period (TA_GRACE) on the Args tab has expired, the value in this field (Starting Time) will be reset to the time of the restart.

### Time Left in Current Service (TA_SVCTIMEOUT)

#### &lt;a name="12436"&gt;&lt;/a&gt;

Description

This field displays the time remaining for this server to process the current service request (if any).

Valid Values

An integer greater than or equal to 0, representing the number of seconds. A value of 0 for an active service means that no timeout processing is being performed.

### Time Left for Reply (TA_TIMELEFT)

#### &lt;a name="12437"&gt;&lt;/a&gt;

Description

This field displays the time remaining for this server to receive the reply for which it is currently waiting before it will time out.

Valid Values

An integer greater than or equal to 0, representing the number of seconds. This timeout can be a transactional timeout or a blocking timeout.

## Statistics 2 Tab

The Statistics 2 tab contains the following fields:

- Number of Subscriptions (TA_NUMSUBSCRIBE)
- Number of Transactions (TA_NUMTRAN)
- Number of Transactions Aborted (TA_NUMTRANABT)
- Number of Transactions Committed (TA_NUMTRANCMT)
- Total Number of Requests Completed (TA_TOTREQC)

- Statistics 3 Tab

## Number of Subscriptions (TA_NUMSUBSCRIBE)

## &lt;a name="12115"&gt;&lt;/a&gt;

### Description

This field displays the number of subscriptions made by this server.

### Valid Values

An integer greater than or equal to 0, the number of subscriptions made by this server with tpsubscribe(3) since Oracle Tuxedo was booted (or you reset the counter).

### Example

0

## Number of Transactions (TA_NUMTRAN)

## &lt;a name="12116"&gt;&lt;/a&gt;

### Description

This field displays the number of transactions begun by this server since its last start or restart.

### Valid Values

An integer greater than or equal to 0.

### Example

0

## Number of Transactions Aborted (TA_NUMTRANABT)

### <a name="12117"></a>

Description

This field displays the number of transactions aborted by this server since its last start or restart.

Valid Values

An integer greater than or equal to 0.

Example

0

## Number of Transactions Committed (TA_NUMTRANCMT)

### <a name="12118"></a>

Description

This field displays the number of transactions committed by this server since its last start or restart.

Valid Values

An integer greater than or equal to 0.

Example

0

## Total Number of Requests Completed (TA_TOTREQC)

### <a name="12122"></a>

Description

This field displays the number of requests completed by this server.

Valid Values

An integer greater than or equal to 0. For conversational servers, the value in this field indicates the number of completed incoming conversations. This is a run-time attribute that is kept across server restart but is lost at server shutdown.

Example

0

# Statistics 3 Tab

The Statistics 3 tab contains the following fields:

- Client Logical Machine Name (TA_CLTLMID)

- Client Process ID (TA_CLTPID)

- Client Expecting Reply (TA_CLTREPLY)

- Commit Control (TA_CMTRET)

- Current Number of Conversations (TA_CURCONV)

- Current Number of Requests (TA_CURREQ)

- Current Service (TA_CURRSERVICE)

- Current Time (TA_CURTIME)

- Last Group Contacted (TA_LASTGRP)

- Current Transaction Level (TA_TRANLEV)

### Client Logical Machine Name (TA_CLTLMID)

<a name="12427"></a>

Description

This field displays the logical machine for the initiating client or server. The initiating client or server is the process that made the service request on which the server is currently working.

### Client Process ID (TA_CLTPID)

<a name="12428"></a>

Description

This field displays the UNIX system process identifier for the initiating client or server.

Limitation

This field is specific to UNIX systems; there might be no value displayed here if the platform on which the application is being run is not UNIX-basedUNIX-based.

### Client Expecting Reply (TA_CLTREPLY)

<a name="12429"></a>

Description

This field displays whether or not the initiating client or server is expecting a reply.

Valid Values

Enter either `Yes` or `No`.

## Commit Control (TA_CMTRET)

### <a name="12430"></a>

Description

This field displays the setting of the TP_COMMIT_CONTROL characteristic for this server. For details about setting this characteristic, refer to the description of the tpscmt(3) function in the *BEA Tuxedo ATMI C Function Reference* manual, and the Default Commit Control (TA_CMTRET) in the *File Formats, Data Descriptions, MIBs, and System Processes* manual.

## Current Number of Conversations (TA_CURCONV)

### <a name="12431"></a>

Description

Number of tpconnect(3) conversations initiated by this server that are still active.

Valid Values

An integer greater than or equal to 0.

Example

1500

## Current Number of Requests (TA_CURREQ)

### <a name="12432"></a>

Description

This field displays the number of requests that have been initiated by this server with tpcall(3) that are still active.

## Current Service (TA_CURRSERVICE)

### &lt;a name="12433"&gt;&lt;/a&gt;

Description

This field displays the service name on which the server is currently working (if any).

## Current Time (TA_CURTIME)

### &lt;a name="12434"&gt;&lt;/a&gt;

Description

This field displays the current time (in seconds) since 00:00:00 UTC, January 1, 1970, as returned by the time(2) system call on the machine identified in the Current Logical Machine Name (TA_LMID). This attribute can be used to compute elapsed time from the Starting Time (TA_TIMESTART) and Restarting Time (TA_TIMERESTART) attribute values.

## Last Group Contacted (TA_LASTGRP)

### &lt;a name="12435"&gt;&lt;/a&gt;

Description

This field displays the server group number (as shown in the Group Number (TA_GRPNO)) of the last service request made or conversation initiated from this server outward.

## Current Transaction Level (TA_TRANLEV)

### &lt;a name="12438"&gt;&lt;/a&gt;

Description

This field displays the current transaction level for this server. A value of 0 means that the server is not currently involved in a transaction.

# Creating Routing Tables

## \<a name="N_ROUTING">\</a>

To create new Routing Tables:

1. Select the Routing Tables folder in the tree view.

2. Select Quick Menu —>New (right-click) to display the New Object window.

3. Enter values for the following fields:

   – Routing Table Name (TA_ROUTINGNAME)

   – Buffer Type (TA_BUFTYPE)

   – Field Name (TA_FIELD)

   – Routing Table Ranges (TA_RANGES)

4. Click the Create button to close the window and create the new Routing Tables.

## Routing Table Name (TA_ROUTINGNAME)

## \<a name="13052">\</a>

### Description

The name of the routing criteria table.

### Valid Values

A string of 1 to 15 characters.

### Example

BRNCH

## Buffer Type (TA_BUFTYPE)

<a name="13050"></a>

<a name="12303"></a>

### Description

A list of types and subtypes (specified with the relevant types) of data buffers for which this routing entry is valid.

### Valid Values

A string of 1 to 256 characters, of the following format:

```
type1[:subtype1[,subtype2 ...]][;type2[:subtype3[, ...]]] ...
```

Entries for types are separated by semicolons. Within a type/subtype pair, the type and subtype components are separated by a colon. Subtype names should not contain semicolon, colon, comma, or asterisk characters.

Duplicate type/subtype pairs can not be specified for the same routing criterion name; more than one routing entry can have the same criterion name as long as the type/subtype pairs are unique. If multiple buffer types are specified for a single routing entry, the data types of the routing field for each buffer type must be the same.

A maximum of 32 type/subtype combinations are allowed.

The valid types are:

- FML
- VIEW
- X_C_TYPE
- X_COMMON
- XML

No subtype can be specified for the FML type. Subtypes are required for VIEW, X_C_TYPE, and X_COMMON. (The wildcard character, *, is not allowed.)

### Example

```
FML
```

## Field Name (TA_FIELD)

## <a name="13051"></a>

## <a name="12304"></a>

### Description

Routing field name.

### Valid Values

A string of 1 to 30 characters.

This field is assumed to be an FML buffer or view field name that is identified in an FML field table (using the FLDTBLDIR and FIELDTBLS environment variables) or an FML view table (using the VIEWDIR and VIEWFILES environment variables), respectively. This information is used to get the associated field value for data-dependent routing during the sending of a message.

### Example

```
ACCOUNT_ID
```

## Routing Table Ranges (TA_RANGES)

## <a name="13053"></a>

## <a name="12306"></a>

### Description

The ranges and associated server groups for the routing field.

### Valid Values

*lower*[-*upper*]:*group*[, *lower*[-*upper*]:*group*[, ...]]

These constructs have the following values:

| Construct | Description |
|-----------|-------------|
| *lower* | Signed numeric value or character string in single quotes. Its value must be less than or equal to that of upper. |
| upper | Signed numeric value or character string in single quotes. |
| group | The associated group to which the request is routed if the field matches the range. A group name of "*" indicates that the request can go to any group that has a server offering the desired service. |

## Notes

The values `MIN` and `MAX` can be used to indicate the minimum and maximum values for the data type of the associated field on the machine. The asterisk, a wildcard character, is a metacharacter that, when used in the position of a range, indicates any values not covered by any previous ranges. Only one wildcard range is allowed per entry and it should be listed last, since ranges following it will be ignored.

The routing field can be of any data type supported in FML. A numeric routing field must have numeric range values, and a string routing field must have string range values.

String range values for string, `carray`, and character field types must be delimited by single quotes and cannot be preceded by a sign. (To embed a single quotation mark in a character string value, precede it with two backslashes.) Short and long integer values are a string of digits, optionally preceded by a plus or minus sign. Floating point numbers are of the window accepted by the C compiler or `atof(3)`:

- An optional sign

- A string of digits optionally containing a decimal point

- An optional `e` or `E` followed by an optional sign or space, followed by an integer

## Examples

```
0:BANKB0
```

```
10000-59999:BANKB1
```

In the following example, entries in the range 60,000 to 69,999 are routed to `BANKB2`, those in the range 70,000 to 79,999 to `BANKB3`, and the rest to `BANKB4`.

```
60000-69999:BANKB2, 70000-79999:BANKB3, *:BANKB4
```

In the following example, all entries less than or equal to -5 are routed to BANKB6.

```
'MIN--5':BANKB6
```

In the following example, all entries greater than or equal to 6 are routed to BANKB7.

```
'6-MAX':BANKB7
```

# Configuring Routing Tables (T_ROUTING Class)

## <a name="T_ROUTING"></a>

## <a name="C_ROUTING"></a>

Routing table attributes are listed in one tabbed folder labeled General Tab.

# General Tab

The General tab contains the following fields:

- Routing Table Name (TA_ROUTINGNAME)
- Object State (TA_STATE)
- Buffer Type (TA_BUFTYPE)
- Field Name (TA_FIELD)
- Routing Table Ranges (TA_RANGES)

## Routing Table Name (TA_ROUTINGNAME)

## <a name="12305"></a>

### Description

This field displays the name of the routing criteria table specified in the Routing Table Name (TA_ROUTINGNAME) of the Create New Object window.

### Valid Values

*rctable*

### Notes

*rctable* is a string of 1 to 15 characters.

### Example

BRNCH_1

## Object State (TA_STATE)

## <a name="12302"></a>

### Description

This field displays the state of your routing table. A routing table has only one state: VALID.

# Creating Service Defaults

## <a name="N_SERVICE"></a>

To create new Service Defaults:

1. Select the Service Defaults folder in the tree view.

2. Select Quick Menu —>New (right-click) to display the New Object window.

3. Enter values for the following field:

   Service Name (TA_SERVICENAME)

4. Click the Create button to close the window and create the new Service Defaults.

## Service Name (TA_SERVICENAME)

## <a name="13020"></a>

## <a name="13021"></a>

## <a name="12144"></a>

## <a name="12150"></a>

Description

The name of the service.

Valid Values

A string of 1 to 15 characters.

Example

OPEN_ACCT

# Configuring Service Defaults (T_SERVICE Class)

## &lt;a name="T_SERVICE"&gt;&lt;/a&gt;

## &lt;a name="C_SERVICE"&gt;&lt;/a&gt;

## &lt;a name="12305"&gt;&lt;/a&gt;

Service defaults class attributes are listed in two tabbed folders:

- General Tab
- System Parameters Tab

## General Tab

The General tab contains the following fields:

- Service Name (TA_SERVICENAME)
- Routing Table Name (TA_ROUTINGNAME)
- Buffer Type (TA_BUFTYPE)
- Object State (TA_STATE)

### Routing Table Name (TA_ROUTINGNAME)

## &lt;a name="12143"&gt;&lt;/a&gt;

### Description

The name of the routing criteria table used for this service when doing data-dependent routing.

### Valid Values

A string of 1 to 15 characters. The default is no entry, that is, the null string; it means data-dependent routing is not done for this service.

If multiple entries exist for the same service name but with different SRVGRP (see Group Name (TA_SRVGRP)) parameters, the Routing Table Name parameter must be the same for all of these entries.

## Buffer Type (TA_BUFTYPE)

### <a name="12142"></a>

### Description

A list of types and subtypes of data buffers accepted by this service.

### Valid Values

A string of 1 to 256 characters, of the following format:

*type1*[:*subtype1*[,*subtype2* ...]][;*type2*[:*subtype3*[, ...]]] ...

Types of data buffers provided with Oracle Tuxedo:

- FML (for FML buffers)
- VIEW
- X_C_TYPE
- X_COMMON (for FML views)
- STRING (for null-terminated character arrays)
- CARRAY or X_OCTET (for a character array that is neither encoded nor decoded during transmission)
- XML

Of these types, only VIEW, X_C_TYPE, and X_COMMON have subtypes. A view subtype gives the name of the particular view expected by the service. Application types and subtypes can also be added (see the UNIX manpage on tuxtypes(5)). For a type that has subtypes, an asterisk can be specified for the subtype to indicate that the service accepts all subtypes for the associated type.

A single service can interpret only a fixed number of buffer types, namely those found in its buffer type switch (see the UNIX manpage on tuxtypes(5)). If this parameter is set to ALL, that service will accept all buffer types found in its buffer type switch. Leaving this field blank is

equivalent to setting it to `ALL`. If multiple entries exist for the same service name but with different SRVGRP parameters, this parameter must be the same for all of these entries.

A maximum of 32 type/subtype combinations are allowed. A type name can be 8 characters or less in length and a subtype name can be 16 characters or less in length. Type and subtype names should not contain semicolon, colon, comma, or asterisk characters; otherwise it is difficult to see where type and subtype values end.

The default is `ALL`.

### Example

The following specifies that the service takes FML buffers:

`FML`

The following specifies that the service takes all FML views:

`VIEW:*`

## Object State (TA_STATE)

### <a name="12140"></a>

This field displays the state of your default service. A default service can be in one of the following states:

- ACTIVE
- INACTIVE

# System Parameters Tab

The System Parameters tab contains the following fields:

- Auto Transaction Mode (TA_AUTOTRAN)
- Service Load (TA_LOAD)
- Priority (TA_PRIO)
- Service Time Out (TA_SVCTIMEOUT)
- Transaction Time (TA_TRANTIME)

## Auto Transaction Mode (TA_AUTOTRAN)

**&lt;a name="12141"&gt;&lt;/a&gt;**

**&lt;a name="12146"&gt;&lt;/a&gt;**

### Description

This field specifies whether or not a transaction should automatically be started if a request message is received for this service and the request is not already in transaction mode.

### Valid Values

Enter either `Yes` or `No`. The default is `No`.

### Limitation

Run-time updates to this attribute are not reflected in active service group objects.

**Note:**   This field is available only for DTP (distributed transaction processing) applications.

## Service Load (TA_LOAD)

**&lt;a name="12136"&gt;&lt;/a&gt;**

**&lt;a name="12145"&gt;&lt;/a&gt;**

### Description

The relative load that the service imposes on the system.

### Valid Values

An integer in the range 1 to 32767. The default is 50.

Service loads are used for load balancing purposes; that is, queues with higher enqueued workloads are less likely to be chosen for a new request. Service loads have meaning only if the value of the Load Balancing Enabled (TA_LDBAL) for the domain class is `Yes`.

## Limitation

Run-time updates to this attribute are not reflected in active groups (in the Groups class).

## Priority (TA_PRIO)

&lt;a name="12137"&gt;&lt;/a&gt;

&lt;a name="12147"&gt;&lt;/a&gt;

### Description

A number that assigns a dequeueing priority to the service request.

### Valid Values

An integer in the range 1 to 100. The default is 50. Higher numbers mean higher priority.

This number is considered relative to the priority assigned to other services offered in the same server. Oracle Tuxedo software prevents services with low priorities from remaining enqueued forever behind a continuing stream of higher priority requests. After every tenth request dequeued by priority, Oracle Tuxedo takes one off the bottom of the queue. When a value is specified in this field, it affects only priority when the system is activated.

### Limitation

Run-time updates to this attribute are not reflected in active service group objects.

## Service Time Out (TA_SVCTIMEOUT)

&lt;a name="12138"&gt;&lt;/a&gt;

&lt;a name="12148"&gt;&lt;/a&gt;

### Description

The time limit (in seconds) for processing requests for this service name.

### Valid Values

An integer greater than or equal to 0, representing the number of seconds. The default is 0.

Servers processing service requests for this service are terminated if they exceed the specified time limit in processing the request. A value of 0 for this attribute indicates that the service should not be terminated.

### Limitations

Run-time updates to this attribute are not reflected in active objects in the Groups class.

## Transaction Time (TA_TRANTIME)

**&lt;a name="12139"&gt;&lt;/a&gt;**

**&lt;a name="12149"&gt;&lt;/a&gt;**

### Description

The default timeout period for a transaction automatically started for the associated service.

### Valid Values

An integer in the range 0 to 2147483647, representing seconds. The default is 30 seconds. A value of 0 implies the maximum timeout value for the machine.

(Transactions are started automatically when a request not in transaction mode is received and the value for the service (in the Auto Transaction Mode (TA_AUTOTRAN) is `Yes`.

### Limitation

Run-time updates to this attribute are not reflected in active GROUPS.

### Example

90

# Creating Services

**&lt;a name="N_SVCGRP"&gt;&lt;/a&gt;**

To create new Services:

1. Select the Services folder in the tree view.

2. Select Quick Menu —>New (right-click) to display the New Object window.

3. Enter values for the following field:

   – Service Name (TA_SERVICENAME)

   – Group Name (TA_SRVGRP)

4. Click the Create button to close the window and create the new Service.

# Configuring Services (T_SVCGRP Class)

## <a name="T_SVCGRP"></a>

## <a name="C_SVCGRP"></a>

Services class attributes are listed in two tabbed folders:

- General Tab
- System Parameters Tab

## General Tab

The General tab contains the following fields:

- Service Name (TA_SERVICENAME)
- Group Name (TA_SRVGRP)
- Service Type (TA_SVCTYPE)
- Service Request Name (TA_SVCRNAM)
- Object State (TA_STATE)

## Service Type (TA_SVCTYPE)

### <a name="12152"></a>

### Description

This field displays the type of service.

### Valid Values

Enter one of the following values.

| Construct | Description |
| --- | --- |
| APP | An application-defined service name. |
| CALLABLE | A system-provided, callable service. |
| SYSTEM | A system provided and system callable service. |

## Service Request Name (TA_SVCRNAM)

### <a name="12153"></a>

### Description

Function name within the associated server assigned to process requests for this service.

### Valid Values

A string of 1 to 15 characters. The default is the value you entered in the Service Name (TA_SERVICENAME).

On a SET request, this server must be able to map the function name to a function using its symbol table to successfully advertise the service. In some situations (for example, direct calls to tpadvertise(3) by the server), the function name for an active service object will not be known and a question mark will be returned as the attribute value.

### Limitation

You can enter a value in this field only when changing the state of the service from inactive to active.

Example

```
SERVC2
```

### Object State (TA_STATE)

## <a name="12154"></a>

Description

This field displays the current state of the service. A service can be in one of the following states:

- ACTIVE

- INACTIVE

- SUSPENDED

- PARTITIONED

# System Parameters Tab

The System Parameters tab contains the following fields:

- Service Load (TA_LOAD)

- Auto Transaction Mode (TA_AUTOTRAN)

- Priority (TA_PRIO)

- Service Time Out (TA_SVCTIMEOUT)

- Transaction Time (TA_TRANTIME)

# Creating Devices

## <a name="N_DEVICE"></a>

To create new Devices:

1. Select the Devices folder in the tree view.

2. Select Quick Menu —>New (right-click) to display the New Object window.

3. Enter values for the following fields:

 – Full Path to Device (TA_CFGDEVICE)

 – Logical Machine Name, Backup (TA_LMID)

 – Device Name (Full Path) (TA_DEVICE)

 – Numeric Offset to Start of Disk Space (in Pages) (TA_DEVOFFSET)

 – Disk Area Size (in Pages) (TA_DEVSIZE)

4. Click the Create button to close the window and create the new Device.

## Full Path to Device (TA_CFGDEVICE)

## &lt;a name="13047"&gt;&lt;/a&gt;

## &lt;a name="12311"&gt;&lt;/a&gt;

### Description

Absolute pathname of the file or device in which the Oracle Tuxedo filesystem is stored (or is to be stored).

### Valid Values

A string of 2 to 64 characters.

### Example

```
/home/apps/bankapp/tuxconfig
```

## Logical Machine Name, Backup (TA_LMID)

## &lt;a name="13049"&gt;&lt;/a&gt;

## &lt;a name="12313"&gt;&lt;/a&gt;

### Description

Name (expressed as a logical machine identifier) of the machine on which this device is located

## Valid Values

`LMID1`[`,LMID2`]

## Notes

Each LMID (logical machine identifier) contains a maximum of 30 characters. The default is the local machine, as a single-machine application.

This attribute can be used as a key field in both unbooted and booted applications as long as they are already configured (that is, at least one machine is defined). This name is required as a key field on SET operations when you are accessing a booted application.

You can assign a value to this field only in a configured application. If you enter a value in this field when accessing the Devices class in an unconfigured application, it will be ignored.

## Device Name (Full Path) (TA_DEVICE)

<a name="13048"></a>

## Description

Absolute pathname of the device list entry.

## Valid Values

A string of 2 to 64 characters. The default is the pathname entered in the Full Path to Device (TA_CFGDEVICE).

This field is optional.

## Numeric Offset to Start of Disk Space (in Pages) (TA_DEVOFFSET)

### <a name="13045"></a>

### <a name="12308"></a>

Description

The offset between the beginning of this device and the point at which the space on this device is reserved for use by the Oracle Tuxedo VTOC specified by the value you entered in the Full Path to Device (TA_CFGDEVICE).

Valid Values

An integer greater than or equal to 0. The default is 0.

Limitation

This attribute must be set to at creation time 0 for the first device list entry (TA_DEVICE) on the Oracle Tuxedo filesystem (TA_CFGDEVICE).

This field is optional.

## Disk Area Size (in Pages) (TA_DEVSIZE)

### <a name="13046"></a>

### <a name="12309"></a>

Description

The offset (specified in pages) from the start of the disk space to this device.

Valid Values

An integer greater than or equal to 0. The default is 1000. If you use a file-based device, this is normally 0. The only time you can specify a value for this field is when you are creating this device.

This field is optional.

# Configuring Devices (T_DEVICE Class)

## &lt;a name="T_DEVICE"&gt;&lt;/a&gt;

## &lt;a name="C_DEVICE"&gt;&lt;/a&gt;

Device attributes are listed in one tabbed folder labeled General Tab.

## General Tab

The General tab contains the following fields:

- Logical Machine Name, Backup (TA_LMID)
- Object State (TA_STATE)
- Full Path to Device (TA_CFGDEVICE)
- Numeric Offset in Pages (TA_CFGOFFSET)
- Device Name (Full Path) (TA_DEVICE)
- Numeric Offset to Start of Disk Space (in Pages) (TA_DEVOFFSET)
- Disk Area Size (in Pages) (TA_DEVSIZE)

### Object State (TA_STATE)

### &lt;a name="12310"&gt;&lt;/a&gt;

Description

This field displays the state of your routing table. A device has only one state: VALID.

### Numeric Offset in Pages (TA_CFGOFFSET)

### <a name="12307"></a>

Description

The offset at which space on this device begins for use within the Oracle Tuxedo VTOC specified in the Device Name (Full Path) (TA_DEVICE) on this tab.

Valid Values

An integer, representing blocks, in the range 0 to the page size of the disk. The default is 0.

We strongly recommend leaving this field set to 0.

### Device Name (Full Path) (TA_DEVICE)

### <a name="12312"></a>

Description

Absolute pathname of the device list entry.

Valid Values

A string of 2 to 64 characters. The default is the pathname entered in the Full Path to Device (TA_CFGDEVICE).

## Configuring Clients (T_CLIENT Class)

## <a name="T_CLIENT"></a>

## <a name="C_CLIENT"></a>

Routing table attributes are listed in three tabbed folders:

- General Tab
- System Parameters Tab
- Statistics Tab

# General Tab

The General tab contains the following fields:

- Client Name (TA_CLTNAME)
- User Name (TA_USRNAME)
- Client ID (TA_CLIENTID)
- WorkStation Client (TA_WSC)
- WorkStation Handler (TA_WSH)
- WorkStation Client ID (TA_WSHCLIENTID)
- WorkStation Client Address (TA_NADDR)
- Object State (TA_STATE)

## Client Name (TA_CLTNAME)

## <a name="12172"></a>

### Description

This field displays the client name associated with the client at `tpinit(3)` by the `cltname` element of the TPINIT structure.

## User Name (TA_USRNAME)

## <a name="12173"></a>

### Description

This field displays the username associated with the client at `tpinit(3)` time by the `usrname` element of the TPINIT structure.

## Client ID (TA_CLIENTID)

### <a name="12174"></a>

Description

> This field displays the client identifier. The data in this field should not be interpreted directly by the end user except for equality comparison.

## WorkStation Client (TA_WSC)

### <a name="12175"></a>

Description

> This field displays the Workstation client. If this attribute is set to Yes, then the indicated client is logged in to the application from a remote workstation.

## WorkStation Handler (TA_WSH)

### <a name="12176"></a>

Description

> This field displays the workstation handler. If this attribute is set to Yes, then the indicated client is a workstation handler process.

## WorkStation Client ID (TA_WSHCLIENTID)

### <a name="12177"></a>

Description

> This field displays the client identifier for the associated workstation handler (WSH) if this client is a Workstation client; otherwise, this attribute will be returned as a zero-length string.

### WorkStation Client Address (TA_NADDR)

### <a name="12439"></a>

### Description

This field displays the network address of a Workstation client. Network addresses with unprintable characters are converted to the Hexadecimal Addresses format. If the address is a TCP/IP address, then it is returned in the **dotted decimal format** with appended port number, as described under Network Address (TA_NADDR).

No value is displayed (that is, the field is blank) for a non-Workstation client.

### Limitation

The ability of the system to provide this information is determined by the transport provider in use. In some cases, Workstation clients may not have addresses associated with them if the provider does not make this information available.

### Object State (TA_STATE)

### <a name="12178"></a>

### Description

This field displays the state of your client. A client can take one of the following states:

- ACTIVE

- SUSPENDED

- DEAD

## System Parameters Tab

The System Parameters tab contains the following fields:

- Logical Machine Name, Backup (TA_LMID)

- Process ID (TA_PID)

- Group Name (TA_SRVGRP)

- Up Time (TA_TIMESTART)

- Idle Time (TA_IDLETIME)

- Current Time (TA_CURTIME)

- Current Encryption Bits (TA_CURENCRYPTBITS)

- Seconds Remaining Before Timeout (TA_TIMELEFT)

## Logical Machine Name, Backup (TA_LMID)

## <a name="12155"></a>

### Description

This field displays the logical machine identifier for the machine on which the client is running (native clients) or to which the client is connected (Workstation clients).

## Process ID (TA_PID)

## <a name="12156"></a>

### Description

This field displays the process identifier of the client. For Workstation clients, this identifier indicates the workstation handler through which the Workstation client is connected. A negative number can be specified on a GET operation for the purpose of retrieving client information for the calling process. If the calling process is not a client, then an error will be returned.

## Group Name (TA_SRVGRP)

## <a name="12157"></a>

### Description

This field displays the server group with which the client is associated. This information is set by the `grpname` element of the TPINIT structure at `tpinit`(3) time.

## Up Time (TA_TIMESTART)

### &lt;a name="12158"&gt;&lt;/a&gt;

Description

This field displays the time (specified in seconds) since the client joined the application (as reported by the time(2) system call on the machine identified in the Logical Machine Name, Backup (TA_LMID).

## Idle Time (TA_IDLETIME)

### &lt;a name="12397"&gt;&lt;/a&gt;

Description

This field displays the approximate amount of time (in seconds) since this client last interacted with the system through an ATMI call. This value is accurate to within the value the scan units, as described in Basic Time Unit for Sanity Checks (TA_SCANUNIT).

When specified as a key field:

- A positive value indicates that all clients with idle times of at least the indicated value match.

- A negative value indicates that all clients with no more than the indicated value match.

- A value of 0 indicates that all clients match.

### Current Encryption Bits (TA_CURENCRYPTBITS)

### <a name="12160"></a>

Description

This field displays the current encryption level for this client. The level is negotiated when the link is established.

Valid Values

Enter one of the following: 0, 40, or 128, representing bits.

### Seconds Remaining Before Timeout (TA_TIMELEFT)

### <a name="12161"></a>

Description

This field displays the time left (in seconds) for this client to receive the reply for which it is currently waiting before it will time out.

Valid Values

An integer greater than or equal to 0, representing the number of seconds. This timeout can be a transactional timeout or a blocking timeout.

## Statistics Tab

The Statistics tab contains the following fields:

- Number of Conversations (TA_NUMCONV)
- Current Number of Conversations (TA_CURCONV)
- Number of Dequeues (TA_NUMDEQUEUE)
- Number of Enqueues (TA_NUMENQUEUE)
- Number of Posts (TA_NUMPOST)
- Number of Requests (TA_NUMREQ)

- Number of Subscriptions (TA_NUMSUBSCRIBE)

- Number of Transactions (TA_NUMTRAN)

- Number of Transactions Aborted (TA_NUMTRANABT)

- Number of Transactions Committed (TA_NUMTRANCMT)

## Number of Conversations (TA_NUMCONV)

## <a name="12162"></a>

### Description

Number of `tpconnect`(3) conversations initiated by this client since Oracle Tuxedo was booted.

### Valid Values

An integer greater than or equal to 0.

### Example

0

## Current Number of Conversations (TA_CURCONV)

## <a name="12163"></a>

### Description

Number of `tpconnect`(3) conversations initiated by this client that are still active.

### Valid Values

An integer greater than or equal to 0.

### Example

1500

## Number of Dequeues (TA_NUMDEQUEUE)

### <a name="12164"></a>

Description

Number of tpdequeue(3) operations initiated by this client since Oracle Tuxedo was booted.

Valid Values

An integer greater than or equal to 0.

Example

0

## Number of Enqueues (TA_NUMENQUEUE)

### <a name="12165"></a>

Description

Number of tpenqueue(3) operations performed from this client since Oracle Tuxedo was booted (or since you reset the counter).

Valid Values

An integer greater than or equal to 0.

Example

0

## Number of Posts (TA_NUMPOST)

### <a name="12166"></a>

Description

Number of tppost(3) operations performed from this client since Oracle Tuxedo was booted.

## Valid Values

An integer greater than or equal to 0.

## Example

0

# Number of Requests (TA_NUMREQ)

## <a name="12167"></a>

### Description

This field displays the number of `tpacall(3)` or `tpcall(3)` operations performed from this client since Oracle Tuxedo was booted.

### Valid Values

An integer greater than or equal to 0.

### Example

0

# Number of Subscriptions (TA_NUMSUBSCRIBE)

## <a name="12168"></a>

### Description

This field displays the number of subscriptions made by this client.

### Valid Values

An integer greater than or equal to 0, the number of subscriptions made by this server with `tpsubscribe(3)` since Oracle Tuxedo was booted (or you reset the counter).

### Example

0

## Number of Transactions (TA_NUMTRAN)

### <a name="12169"></a>

Description

This field displays the number of transactions begun by this client since its last start or restart.

Valid Values

An integer greater than or equal to 0.

Example

0

## Number of Transactions Aborted (TA_NUMTRANABT)

### <a name="12170"></a>

Description

This field displays the number of transactions aborted by this client since its last start or restart.

Valid Values

An integer greater than or equal to 0.

Example

0

### Number of Transactions Committed (TA_NUMTRANCMT)

<a name="12171"></a>

Description

This field displays the number of transactions committed by this client since its last start or restart.

Valid Values

An integer greater than or equal to 0.

Example

0

# Configuring Bridges (T_BRIDGE Class)

<a name="T_BRIDGE"></a>

<a name="C_BRIDGE"></a>

Bridge attributes are listed in three tabbed folders:

- General Tab

- Timers Tab

- Statistics Tab

## General Tab

The General tab contains the following fields:

- Logical Machine Name, Backup (TA_LMID)

- Object State (TA_STATE)

- Current Encryption Bits (TA_CURENCRYPTBITS)

## Logical Machine Name, Backup (TA_LMID)

### <a name="12291"></a>

Description

This field displays the source logical machine identifier (*LMID1*) and optionally, the destination logical machine identifier (*LMID2*) for network connection.

Valid Values

*LMID1*[,*LMID2*]

## Object State (TA_STATE)

### <a name="12292"></a>

Description

This field displays the state for your bridge. A bridge can take one of the following states:

- ACTIVE
- INACTIVE
- SUSPENDED

## Current Encryption Bits (TA_CURENCRYPTBITS)

### <a name="12293"></a>

Description

This field displays the current encryption level for this link. The level is negotiated between machines when the link is established.

Valid Values

Enter one of the following: 0, 40, or 128, representing bits.

# Timers Tab

The Timers tab contains the following fields:

- Connection Time (TA_CONTIME)
- Time Remaining in Suspension (TA_SUSPTIME)

## Connection Time (TA_CONTIME)

## <a name="12295"></a>

### Description

This field displays the amount of time (in seconds) that has elapsed since the connection was established.

## Time Remaining in Suspension (TA_SUSPTIME)

## <a name="12296"></a>

### Description

This field displays the time remaining in the suspension of this connection.

### Valid Values

An integer greater than or equal to 0, representing seconds. The default is 300. After this amount of time, the state of the connection will automatically change to inactive and can be activated by normal application traffic.

# Statistics Tab

The Statistics tab contains the following fields:

- Bytes Sent (TA_SENTBYT)
- Bytes Received (TA_RCVDBYT)
- System Messages Sent (TA_SENTNUM)

- System Messages Received (TA_RCVDNUM)
- Flow Control Events (TA_FLOWCNT)

## Bytes Sent (TA_SENTBYT)

## <a name="12297"></a>

### Description

Number of bytes sent from the source logical machine to the destination logical machine.

### Valid Values

An integer greater than or equal to 0.

### Example

0

## Bytes Received (TA_RCVDBYT)

## <a name="12298"></a>

### Description

Number of bytes received by the source logical machine the destination logical machine.

### Valid Values

An integer greater than or equal to 0.

### Example

0

### System Messages Sent (TA_SENTNUM)

#### <a name="12299"></a>

Description

Number of messages sent from the source logical machine to the destination logical machine.

Valid Values

An integer greater than or equal to 0.

Example

0

### System Messages Received (TA_RCVDNUM)

#### <a name="12300"></a>

Description

Number of messages received by the source logical machine the destination logical machine.

Valid Values

An integer greater than or equal to 0.

Example

0

### Flow Control Events (TA_FLOWCNT)

#### <a name="12301"></a>

Description

Number of times flow control has been encountered over this connection.

## Valid Values

An integer greater than or equal to 0.

## Example

0

# Creating and Configuring Application Queues

The Oracle Tuxedo system uses application queues for time-independent communication. In Oracle Tuxedo system applications, messages are stored on a queue, and queues are defined within a particular queue space. Queueing and dequeueing is done within a transaction. The BEA Tuxedo Administration Console provides the administrative environment required for managing and controlling access to application queues. The following help topics contain information about the application queue resources:

- Application Queues
    - Creating Application Queues
    - Configuring Application Queues (T_APPQ Class)

- Application Queue Spaces
    - Creating Application Queue Spaces
    - Configuring Application Queue Spaces (T_APPQSPACE Class)

- Application Queue Messages
    - Configuring Application Queue Messages (T_APPQMSG Class)

- Application Queue Transactions
    - Configuring Application Queue Transactions (T_APPQTRANS Class)

## Application Queues

The Application Queues tree resources allow you to manage the attributes of an application queue such an pathnames, machine IDs, and processing order.

<a name="N_APPQ"></a>

## Creating Application Queues

To create new Application Queues:

1. Select the Application Queue folder in the tree and select Quick Menu —>New (right mouse click) to display the Create New Object window.

2. Enter values for the following fields:

   – App Queue Name (TA_APPQNAME)

   – Queue Messages Processing Order (TA_APPQORDER)

   – App Queue Space Name (TA_APPQSPACENAME)

   – App Queue Space Device (TA_QMCONFIG)

   – Logical Machine Name (TA_LMID)

3. Click Create to close the window and create the new application queue.

For information about using the Create New Object window, see Creating Administrative Class Objects.

<a name="T_APPQ"></a>

## Configuring Application Queues (T_APPQ Class)

The attributes of application queues are listed in the following tab pages:

- General Tab

- Queue Message Tab

- Retries Tab

- Command Tab

The Configuration Tool contains some fields that are read-only. The information provided by these display-only fields is described briefly. For general information about using the

Configuration Tool to configure administrative resources, see Configuring Administrative Objects.

<a name="C_APPQ"></a>

# General Tab

The General tab page contains the following fields:

- App Queue Name (TA_APPQNAME)

- App Queue Space Name (TA_APPQSPACENAME)

- App Queue Space Device (TA_QMCONFIG)

- Logical Machine Name (TA_LMID)

- Object State (TA_STATE)

- <a name="13029"></a><a name="12238"></a>

## App Queue Name (TA_APPQNAME)

### Description

The name of the application queue to be operated upon.

### Valid Values

A string of 1 to 15 characters in length.

### Example

```
Q_OPENACCT
```

<a name="13030"></a><a name="12240">

## App Queue Space Name (TA_APPQSPACENAME)

### Description

The name of the application queue space associated with the application queue. Filling in this field is half of what you must do to identify such a queue space; you must also enter a value in the App Queue Space Device (TA_QMCONFIG).

## Valid Values

A string of 1 to 15 characters in length.

## Example

```
QSP_BANKAPP
```

```
<a name="13031"></a><a name="12245"></a>
```

# App Queue Space Device (TA_QMCONFIG)

## Description

The pathname for an application queue configuration device, which must be on the current machine when you are using a deactivated configuration. Filling in this field is half of what you must do to identify a queue space associated with the application queue; you must also enter a value in the App Queue Space Name (TA_APPQSPACENAME).

## Valid Values

A string of 1 to 78 characters in length.

## Example

```
Q_DEVICE
```

```
<a name="13028"></a><a name="12246"></a>
```

# Logical Machine Name (TA_LMID)

## Description

Logical machine identifier for the machine on which the application queue space is located.

## Valid Values

A string of 1 to 30 characters in length

## Example

```
SITE1
```

Notes

The value in this field must be specified as a key field except when the application is unconfigured (i.e., the TUXCONFIG environment variable is not set).

<a name="12237"></a>

## Object State (TA_STATE)

Description

This field displays the state of your application queue. An application queue may be in only one of the following states:

- VALID
- INVALID
- NEW

All operations (whether GET or SET) on application queues silently open the associated queue space (i.e., they implicitly set the state of the queue space to OPEN if it is not already OPEN or ACTIVE). This may be a time-consuming operation if the queue space is large.

# Queue Message Tab

The Queue Message tab page contains the following fields:

- Queue Messages Processing Order (TA_APPQORDER)
- Out of Order Messages Processing (TA_OUTOFORDER)
- Current Number of Messages (TA_CURMSG)
- Current Number of Disk Pages Used (TA_CURBLOCKS)
- <a name="12239"></a><a name="13028"></a>

## Queue Messages Processing Order (TA_APPQORDER)

Description

The order in which you want messages in the queue to be processed. Consider your needs carefully before selecting an order for your application: the value in this field cannot be modified after the application queue is created.

### Valid Values

Select one of the combinations listed below. The commas indicate a secondary (or lower) processing order within the primary processing order. For example, `PRIO, LIFO` means that the queue is processed in priority order, but if two items in the queue share the same priority, they are then processed in last in first out order.

`FIFO` — first in first out

`PRIO, LIFO` — Priority (then last in first out)

`PRIO, FIFO` — Priority (then first in first out)

`TIME, LIFO` — Execution Time (then last in first out)

`PRIO, TIME, LIFO` — Priority (then Execution Time (then last in first out)

`PRIO, TIME, LIFO` — Priority (then Execution Time (then first in last out)

`TIME, PRIO, LIFO` — Execution Time (then Priority (then last in first out)

### Default

`FIFO`

`<a name="12244"></a>`

## Out of Order Messages Processing (TA_OUTOFORDER)

### Description

Method by which out-of-order message processing will be handled.

### Valid Values

Select one of the following:

- `NONE`
- `TOP`
- `MSGID`

### Default

`NONE`

`<a name="12234"></a>`

## Current Number of Messages (TA_CURMSG)

### Description

This field displays the number of messages currently in the queue.

<a name="12233"></a>

## Current Number of Disk Pages Used (TA_CURBLOCKS)

### Description

This field displays the number of disk pages currently in the queue.

<a name="retries"></a>

# Retries Tab

The Retries tab page contains the following fields:

- Max. Number of Retries (TA_MAXRETRIES)
- Delay Between Retries (in Sec) (TA_RETRYDELAY)
-  <a name="12235"></a>

## Max. Number of Retries (TA_MAXRETRIES)

### Description

The maximum number of retries for a failed queue message. When this number of retries is exhausted, the message is placed on the error queue of the queue's application queue space, as defined by Error Queue Name (TA_ERRORQNAME). If there is not error queue, the message is dropped.

### Valid Values

A number greater than or equal to zero

### Default

Zero

<a name="12236"></a>

### Delay Between Retries (in Sec) (TA_RETRYDELAY)

#### Description

The delay, in seconds, between retries for failed queue messages.

#### Valid Values

A number greater than or equal to zero

#### Default

Zero

<a name="c_retries"></a>

# Command Tab

The Command tab page contains the following fields:

- High-Water Mark for Command Execution (TA_CMDHW)

- Low-Water Mark for Command Execution (TA_CMDLW)

- Command to Execute (TA_CMD)

-     <a name="12242"></a>

### High-Water Mark for Command Execution (TA_CMDHW)

#### Description

The high-water mark for command execution is expressed as an integer greater or equal to zero, optionally followed by one of the following:

B or b — the high-water marks are expressed in disk blocks.

M or m — the high-water marks are expressed in messages.

% — the high-water marks are expressed as a percentage of queue capacity.

#### Syntax

num [b | m | %] (where num is a number greater than or equal to zero)

#### Default

The default is 100%.

<a name="12243"></a>

## Low-Water Mark for Command Execution (TA_CMDLW)

### Description

The low-water marks for command execution are expressed in an integer greater or equal to zero, optionally followed by one of the following:

B or b — the low-water marks are expressed in disk blocks.

M or m — the low-water marks are expressed in messages.

% — the low-watermarks are expressed as a percentage of queue capacity.

### Valid Values

*num* [b | m | %] (where *num* is a number greater than or equal to zero)

### Default

The default is 0%.

<a name="12241"></a>

## Command to Execute (TA_CMD)

### Description

The command to be executed when the high-water mark is reached. If the high-water mark is reached after the low-water mark, the command is re-executed.

### Valid Values

A shell command. This must be a string up to 78 characters in length.

### Default

No entry

# Application Queue Spaces

The Application Queue Spaces objects allow you to manage the attributes of an application queue spaces.

<a name="N_APPQSPACE"></a>

# Creating Application Queue Spaces

To create new Application Queue Spaces:

1. Select the application queue spaces folder in the tree.

2. Select Quick Menu —>New (right-click) to display the Create New Object window.

3. Enter values for the following fields:

   – IPC Key (TA_IPCKEY)

   – Max Queues (TA_MAXQUEUES)

   – Max Messages (TA_MAXMSG)

   – Max Disk Pages (TA_MAXPAGES)

   – Max Processes (TA_MAXPROC)

   – Max Open Transactions (TA_MAXTRANS)

   – App Queue Space Name (TA_APPQSPACENAME)

   – App Queue Space Device (TA_QMCONFIG)

   – Logical Machine Name (TA_LMID)

For information about using the Create New Object window, see Creating Administrative Class Objects.

<a name="T_APPQSPACE"></a>

# Configuring Application Queue Spaces (T_APPQSPACE Class)

The attributes of application spaces are listed in six tab pages with the following labels:

- General Tab

- Disk/Pages Tab

- Queue Tab

- Transaction Tab

- Process Tab

- Message Tab

For general information about using the Configuration Tool to configure administrative resources, see Configuring Administrative Objects.

<a name="C_APPQSPACE"></a>

# General Tab

The General tab page contains the following fields:

- App Queue Space Name (TA_APPQSPACENAME)
- App Queue Space Device (TA_QMCONFIG)
- Logical Machine Name (TA_LMID)
- IPC Key (TA_IPCKEY)
- Error Queue Name (TA_ERRORQNAME)
- Object State (TA_STATE)

This section describes these fields.

<a name="13038"></a> <a name="12265"></a>

### App Queue Space Name (TA_APPQSPACENAME)

#### Description

The name used to identify the application queue space.

#### Valid Values

A string of 1 to 15 characters in length.

#### Example

```
QSP_BANKAPP
```

<a name="12268"></a>

### App Queue Space Device (TA_QMCONFIG)

#### Description

The absolute pathname of the file or device where the application queue space is located.

## Valid Values

A string of 1 to 78 characters in length.

## Example

```
/HOME4/APPS/BANKAPP/QSPDVC
<a name="12269"></a>
```

# Logical Machine Name (TA_LMID)

## Description

Identifier of the logical machine on which the application queue space is located.

## Valid Values

A string of 1 to 30 characters in length.

## Example

```
SITE1
```

## Notes

You must specify that this field is a key field except when the application is unconfigured (that is, when the TUXCONFIG environment variable is not set).

```
<a name="13032"></a> <a name="12249"></a>
```

# IPC Key (TA_IPCKEY)

## Description

A numeric key used to access queue space shared memory.

## Valid Values

A number in the range of 32769 to 262143.

## Example

```
88945
<a name="12266"></a>
```

### Error Queue Name (TA_ERRORQNAME)

#### Description

The name of the error queue associated with the queue space.

#### Valid Values

A string of 0 to 15 characters in length.

#### Default

No entry

#### Example

```
Q_ERROR
<a name="12264"></a>
```

### Object State (TA_STATE)

#### Description

This field displays the state of the application queue space. An application queue space may be in one of the following states:

- INACTIVE
- INITIALIZING
- OPEN
- ACTIVE

All operations on application queue classes silently open the associated queue space (i.e., implicitly set the state of the queue space to OPEN if it is not already OPEN or ACTIVE). This may be a time-consuming operation if the queue space is large.

`<a name="diskpg"></a>`

## Disk/Pages Tab

The Disk/Pages tab page contains the following fields:

- Max Disk Pages (TA_MAXPAGES)
- Blocking Factor (TA_BLOCKING)

- Current Number of Extents (TA_CUREXTENT)

- Percent Initialized on Disk (TA_PERCENTINIT)

- Force Initialization of New Extents (TA_FORCEINIT)

This section describes these fields.

&lt;a name="13035"&gt;&lt;/a&gt; &lt;a name="12260"&gt;&lt;/a&gt;

## Max Disk Pages (TA_MAXPAGES)

### Description

The maximum number of disk pages used by all queues in the queue space. Each time this attribute is increased, a new extent is used. You cannot decrease the number of pages by entering a lower number in this field.

### Valid Values

A number greater than or equal to zero.

### Example

```
200
```

&lt;a name="12251"&gt;&lt;/a&gt;

## Blocking Factor (TA_BLOCKING)

### Description

The blocking factor used in the queue space for disk space management. The default when a new queue space is created is 16.

### Valid Values

A number greater than or equal to zero.

### Default

```
16
```

&lt;a name="12252"&gt;&lt;/a&gt;

### Current Number of Extents (TA_CUREXTENT)

Description

This field displays the number of extents currently in use by the queue space. The maximum number of extents that can be used is 100.

<a name="12263"></a>

### Percent Initialized on Disk (TA_PERCENTINIT)

Description

This field displays the percentage of disk space that has been initialized for the queue space.

<a name="12267"></a>

### Force Initialization of New Extents (TA_FORCEINIT)

Description

Select `Yes` to force the application queue space to initialize immediately, or `No` to delay the initialization.

Valid Values

`Yes` or `No`

Default

`No`

<a name="queuefolder"></a>

## Queue Tab

The Queue tab page contains the following fields:

- Max Queues (TA_MAXQUEUES)
- Current Number of Queues (TA_CURQUEUES)
- Highest Number of Queues (TA_HWQUEUES)

This section describes these fields.

<a name="13033"></a><a name="12250"></a>

### Max Queues (TA_MAXQUEUES)

Description

The maximum number of queues that the queue space can accommodate.

Valid Values

A number greater than or equal to zero.

Example

```
20
```

<a name="12247"></a>

### Current Number of Queues (TA_CURQUEUES)

Description

This field displays the number of server queues that currently exist in the queue space. This number can be determined only if the queue space is new, open, or active. Otherwise a value of -1 is displayed.

<a name="12248"></a>

### Highest Number of Queues (TA_HWQUEUES)

Description

This field displays the highest number of queues that existed at any one time in the queue space since the queue space was last opened.

<a name="transaction"></a>

# Transaction Tab

The Transaction tab page contains the following fields:

- Max Open Transactions (TA_MAXTRANS)
- Current Number of Open Transactions (TA_CURTRANS)

- Highest Number of Open Transactions (TA_HWTRANS)

This section describes these fields.

<a name="13037"></a> <a name="12262"></a>

## Max Open Transactions (TA_MAXTRANS)

### Description

The maximum number of open transactions allowed in the queue space.

### Valid Values

A number greater than or equal to zero.

### Example

5

<a name="12255"></a>

## Current Number of Open Transactions (TA_CURTRANS)

### Description

This field displays the current number of outstanding transactions involving the queue space.

<a name="12258"></a>

## Highest Number of Open Transactions (TA_HWTRANS)

### Description

This field displays the highest number of outstanding transactions involving the queue space, since the queue space was last opened. The highest number of queues existing in the queue space since the queue space was last opened. The number is reset to 0 when the queue space state is set to CLEANING.

<a name="process"></a>

# Process Tab

The Process tab page contains the following fields:

- Max Processes (TA_MAXPROC)

- Current Number of Processes (TA_CURPROC)

- Highest Number of Processes (TA_HWPROC)

This section describes the following fields.

`<a name="13036"></a> <a name="12261"></a>`

## Max Processes (TA_MAXPROC)

### Description

The maximum number of processes that can access the queue space simultaneously.

### Valid Values

A number greater than or equal to zero.

### Example

```
5
<a name="12254"></a>
```

## Current Number of Processes (TA_CURPROC)

### Description

This field displays the number of processes currently accessing the queue space.

`<a name="12257"></a>`

## Highest Number of Processes (TA_HWPROC)

### Description

This field displays the highest number of processes simultaneously attached to the queue space at any one time since the queue space was last opened.

`<a name="trans_message"></a>`

# Message Tab

The Message tab page contains the following fields:

- Max Messages (TA_MAXMSG)
- Current Number of Messages (TA_CURMSG)
- Highest Number of Messages (TA_HWMSG)

This section describes these fields.

<a name="13034"></a> <a name="12259"></a>

## Max Messages (TA_MAXMSG)

### Description

The maximum number of messages allowed in the queue space.

### Valid Values

A number greater than or equal to zero.

### Example

```
200
```

<a name="12253"></a>

## Current Number of Messages (TA_CURMSG)

### Description

This field displays the current number of messages in the queue space. This number can be determined only if the queue space is OPEn or ACTive, or if the queue space is newly created. If none of these conditions apply, a value of -1 is returned.

<a name="12256"></a>

## Highest Number of Messages (TA_HWMSG)

### Description

This field displays the highest number of messages in the queue space at any one time since the queue space was last opened.

# Application Queue Messages

The Application Queue Messages objects allow you to manage the attributes of application queue messages.

<a name="T_APPQMSG"></a>

# Configuring Application Queue Messages (T_APPQMSG Class)

The attributes of application queue messages are listed in three tab pages with the following labels:

- General Tab

- Queue Message Tab

- Statistics Tab

For general information about using the Configuration Tool to configure administrative resources, see Configuring Administrative Objects.

<a name="C_APPQMSG"></a>

## General Tab

The General tab page contains the following fields:

- Object State (TA_STATE)

- App Queue Message ID (TA_APPQMSGID)

- App Queue Name (TA_APPQNAME)

- App Queue Space Name (TA_APPQSPACENAME)

- App Queue Space Device (TA_QMCONFIG)

- Logical Machine Name (TA_LMID)

This section describes these fields.

## Object State (TA_STATE)

### Description

This field displays the state of the application queue message. An application queue message may be in only one state: VALID.

**Note:** All operations on application queue messages (both GET and SET) silently open the associated queue space (i.e., implicitly set the state of the queue space to OPEN if it is not already OPEN or ACTIVE). This may be a time-consuming operation if the queue space is large.

**Note:** <a name="12276"></a>

## App Queue Message ID (TA_APPQMSGID)

### Description

This field displays a unique identifier for the queue message, which can be used to select the message for GET or SET operations. No significance should be placed on this value; it is provided for your convenience in doing equality comparisons.

<a name="12277"></a>

## App Queue Name (TA_APPQNAME)

### Description

This field displays the name of the application queue in which the message is stored.

<a name="12278"></a>

## App Queue Space Name (TA_APPQSPACENAME)

### Description

This field displays the name of the application queue space containing the message.

<a name="12283"></a>

### App Queue Space Device (TA_QMCONFIG)

Description

This field displays the absolute pathname of the file or device where the application queue space is located.

<a name="12285"></a>

### Logical Machine Name (TA_LMID)

Description

This field displays the logical machine identifier of the logical machine on which the application queue space is located.

<a name="q_msgs">

# Queue Message Tab

The Queue Message tab page contains the following fields:

- Number of Retries (TA_CURRETRIES)

- Message Size (TA_MSGSIZE)

- Message Priority (TA_PRIORITY)

- Correlation Identifier (TA_CORRID)

- New App Queue Name (TA_NEWAPPQNAME)

- Time to Process the Message (TA_TIME)

This section describes these fields.

<a name="12270"></a>

### Number of Retries (TA_CURRETRIES)

Description

This field displays the number of retries that have been attempted so far on this message.

<a name="12273"></a>

## Message Size (TA_MSGSIZE)

### Description

This field displays the size of the message (in bytes).

<a name="12274"></a>

## Message Priority (TA_PRIORITY)

### Description

The priority of the message. The value in this field is valid only for PRIO-based queues. A value of -1 is returned by a GET operation if the queue is not PRIO-based.

### Valid Values

A number in the range of 1 to 100.

### Example

3

<a name="12279"></a>

## Correlation Identifier (TA_CORRID)

### Description

This field displays the correlation identifier for this message provided by the application in the tpenqueue(3) request. The empty string indicates that a correlation identifier is not present.

<a name="12282"></a>

## New App Queue Name (TA_NEWAPPQNAME)

### Description

Name of the queue into which to move the selected message. This queue must be an existing queue in the same queue space. This operation will not succeed unless the message is in a VALID state.

### Valid Values

A string from 1 to 15 characters in length.

### Example

```
APPQ3
```

```
<a name="12284"></a>
```

## Time to Process the Message (TA_TIME)

### Description

The time when the message will be processed. This field is valid only for TIME-based queues. (If the queue is not TIME-based, an empty string is returned by a GET operation.)

### Valid Values

Use either of the following formats.

Format 1:

```
+seconds
```

Specifies that the message will be processed *seconds* in the future. The value 0 (zero) specifies that the message should be processed immediately.

Format 2:

```
YY[MM[DD[hh[mm[ss]]]]]
```

Specifies the year, month, day, hour, minute, and second when the message should be processed.

Omitted units default to their minimum possible values.

### Example

Example of Format 2:

```
9506
```

This is equivalent to 950601000000. The years 00 through 37 are treated as 2000 through 20037, 70 through 99 are treated as 1970 through 1999, and 38 through 69 are invalid.

```
<a name="q_stats">
```

# Statistics Tab

The Statistics tab page contains the following fields:

- High Priority (TA_HIGHPRIORITY)

- Low Priority (TA_LOWPRIORITY)

- End Time (TA_MSGENDTIME)

- Start Time (TA_MSGSTARTTIME)

This section describes these fields.

<a name="12271">

## High Priority (TA_HIGHPRIORITY)

### Description

The highest priority in the range within which to search for occurrences of application messages.

<a name="12272"></a>

## Low Priority (TA_LOWPRIORITY)

### Description

The lowest priority in the range within which to search for occurrences of application messages.

<a name="12280"></a>

## End Time (TA_MSGENDTIME)

### Description

The time marking the end of the period during which the search for occurrences of application queue messages is performed.

<a name="12281"></a>

## Start Time (TA_MSGSTARTTIME)

### Description

The time marking the start of the period during which the search for occurrences of application queue messages is performed.

# Application Queue Transactions

The Application Queue Transactions objects allow you to manage the attributes of an application queue transactions.

<a name="T_APPQTRANS"></a>

# Configuring Application Queue Transactions (T_APPQTRANS Class)

The attributes of application queue transactions are listed on the tab page labeled General. The fields for application queue transactions are information fields and cannot not be edited. The General tab page fields for Application Queue Transactions are the following.

<a name="C_APPQTRANS"></a>

## General Tab

The General tab page contains the following fields.

- Object State (TA_STATE)

- Transaction Identifier (TA_XID)

- App Queue Space Name (TA_APPQSPACENAME)

- App Queue Space Device (TA_QMCONFIG)

- Logical Machine Name (TA_LMID)

- <a name="12286"></a>

### Object State (TA_STATE)

This field displays the state of the application queue transaction. An application queue transaction may be in any of the following states: ACTIVE, ABORTONLY, ABORTED, COMCALLED, READY, DECIDED, or SUSPENDED.

All operations on application queue transactions (both GET and SET) silently open the associated queue space (i.e., implicitly set the state of the queue space to OPEN if it is not already OPEN or ACTIVE). This may be a time-consuming operation if the queue space is large.

<a name="12287"></a>

### Transaction Identifier (TA_XID)

This field displays transaction identifier as returned by `tx_info(3)` and mapped to a string representation. The data in this field should not be interpreted directly by the user except for equality comparison.

<a name="12288"></a>

### App Queue Space Name (TA_APPQSPACENAME)

This field displays the name of the application queue space associated with the transaction.

<a name="12289"></a>

### App Queue Space Device (TA_QMCONFIG)

This field displays the absolute pathname of the file or device where the application queue space is located.

<a name="12290"></a>

### Logical Machine Name (TA_LMID)

This field displays the identifier of the logical machine where the application queue space is located.

For general information about using the Configuration Tool to configure administrative resources, see Configuring Administrative Objects.

# Creating and Configuring Workstation Listeners and Handlers

The Oracle Tuxedo system includes the capability to require that clients run on a workstation for purposes of security, performance, and convenience. A network administrator can define the environment required to control the Workstation clients. The Oracle Tuxedo Administration Console workstation objects allow you to specify the information required to control access to an application from multiple workstations.

These workstation objects are grouped on the Oracle Tuxedo Administration Console tree view in the following categories:

- Workstation Listeners (WSLs)
    - Creating Workstation Listeners
    - Configuring Workstation Listeners (T_WSL Class)
- Workstation Handlers (WSHs)
    - Configuring Workstation Handlers (T_WSH Class)

You can define new workstation listeners and you can obtain information about active workstation handlers.

# Workstation Listeners (WSLs)

The Workstation Listener (WSL) allows you to manage workstation groups, configuration, and run-time attributes of WSL server processes.

<a name="N_WSL"></a>

# Creating Workstation Listeners

<a name="WSL_NewObjectForm"></a>

To create new Workstation Listeners (WSLs):

1.  Select Workstation Listeners in the tree view.

2.  Select Quick Menu—>New to display the Create New Object window.

3.  Enter values for the following fields:

    – Group Name (TA_SRVGRP)

    – Server ID (TA_SRVID)

    – Network Address (TA_NADDR)

    – Bridge Device (TA_DEVICE)

4.  Click Create to close the window and create the new WS Listener group.

5.  <a name="T_WSL"></a>

# Configuring Workstation Listeners (T_WSL Class)

To configure WSLs, select WS Listeners in the Tree view. Enter values for the fields in the tab pages. The Configuration Tool pane contains the following tab pages for configuring Workstation Listener attributes: General, Options, System Parameters, and Statistics.

<a name="C_WSL"></a>

<a name="WSL_GeneralTabPage"></a>

# General Tab

The General tab page contains the following fields:

- Group Name (TA_SRVGRP)

- Object State (TA_STATE)

- Server ID (TA_SRVID)

- WorkStation Handler Name (TA_WSHNAME)

- Network Address (TA_NADDR)

- Bridge Device (TA_DEVICE)

- Group Number (TA_GRPNO)

- Process ID (TA_PID)

- Logical Machine Name, Backup (TA_LMID)

This section describes these fields.

<a name="13026"></a> <a name="12207"></a>

## Group Name (TA_SRVGRP)

### Description

Logical name of the server group.

### Valid Values

A string of 1 to 30 characters. The string may not include asterisks (*), commas, or colons.

### Example

```
WSGROUP1
```

<a name="12195"></a>

## Object State (TA_STATE)

### Description

This field displays the state of your workstation listener within the application. A workstation listener may be in any state defined for the Default Servers class:

- `ACTIVE`
- `INACTIVE`
- `MIGRATING`
- `CLEANING`
- `RESTARTING`
- `SUSPENDED`
- `PARTITIONED`
- `DEAD`
- `<a name="13023"></a> <a name="12194"></a>`

## Server ID (TA_SRVID)

### Description

A server identification number that is unique within the server group.

### Valid Values

A number in the range of 1 to 30,001 inclusive.

### Example

```
500
```

`<a name="12205"></a>`

## WorkStation Handler Name (TA_WSHNAME)

### Description

The name of the executable providing workstation handler services for this workstation listener. The default value for this is `WSH`, which corresponds to the system provided workstation handler. Workstation handlers may be customized using the command `buildwsh(1)`.

## Valid Values

A string of 1 to 78 characters.

## Default

```
WSH
```

```
<a name="13025"></a> <a name="12200"></a>
```

# Network Address (TA_NADDR)

## Description

The complete network address to be used by the WSL process as its listening address. The listening address for a WSL is the means by which it is contacted by Workstation client processes participating in the application.

If the string has the form `0xhex-digits` or `\\xhex-digits`, it must contain an even number of valid hex digits. These forms are translated internally into a character array containing the hexadecimal representations of the string specified.

## Valid Values

TCP/IP addresses may be specified in either of the following formats.

Format 1: `//hostname:port_number`

In Format 1, the bridge finds an address for `hostname` using the local name resolution facilities (usually DNS). `hostname` must unambiguously resolve to the address of the machine on which the bridge resides.

Format 2: `//#.#.#.#:port_number`

In Format 2, the `#.#.#.#` is in dotted decimal format. In dotted decimal format, each `#` should be a number from 0 to 255. This dotted decimal number represents the IP address of the local machine.

In both of the above formats, `port_number` is the TCP port number at which the bridge will listen for incoming requests. The value of `port_number` can be either a number (between 0 and 65535) or a name. If `port_number` is a name, then it must be found in the network services database on your local machine.

The address can also be specified in hexadecimal format when preceded by the characters `0x`. Each character after the initial `0x` is a number between 0 and 9 or a letter between A and F (case

insensitive). The hexadecimal format is useful for arbitrary binary network addresses such as IPX/SPX or TCP/IP.

The address can also be specified as an arbitrary string (with a maximum of 78 characters). For example, string addresses are used in STARLAN networks.

## Example

Suppose the local machine on which a TDomain is being run is using TCP/IP addressing and is named `backup.company.com`, with address `155.2.193.18`. Further suppose that the port number at which the TDomain should accept requests is `2334`. Assume that port number `2334` has been added to the network services database under the name `bankapp-wsl`. You can specify the address in any of the following five formats:

```
//155.2.193.18:bankapp-wsl
```

```
//155.2.193.18.2334
```

```
//backup.company.com:bankapp-wsl
```

```
//backup.company.com:2334
```

```
0x0002091E9B02C112
```

The last line shows an address in hexadecimal format where `0002` is the first part of a TCP/IP address and `091E` is the port number 2334 translated into a hexadecimal number. The rest of the line contains hexadecimal numbers that are translations of the elements of the IP address 155.2.193.12: "155" becomes "9B", "2" becomes "02", and so on.

<a name="13024"></a> <a name="12198"></a>

<a name="WSL_GeneralTabPage_TA_DEVICE"></a>

<a name="WSL_NewObjectForm_TA_DEVICE"></a>

# Bridge Device (TA_DEVICE)

## Description

The full pathname for the device to be used by the WSL process to access the network. This is a required value for access to a network via a TLI-based Oracle Tuxedo system binary. This attribute is not needed for sockets-based Oracle Tuxedo system binaries.

## Valid Values

A string from 0 to 78 characters in length.

### Example

```
/dev/tcp
```

<a name="12182"></a>
<a name="WSL_GeneralTabPage_TA_GRPNO"></a>

## Group Number (TA_GRPNO)

### Description

This field displays the group number associated with this group of servers.

<a name="12184"></a>
<a name="WSL_GeneralTabPage_TA_PID"></a>

## Process ID (TA_PID)

### Description

This field displays the native operating system process identifier for the WSL server. Note that this may not be a unique attribute since servers may be located on different machines allowing for duplication of process identifiers.

<a name="12206"></a>
<a name="WSL_GeneralTabPage_TA_LMID"></a>

## Logical Machine Name, Backup (TA_LMID)

### Description

This field displays the logical machine identifier for the machine on which the server is currently running.

<a name="wsl_opt"></a>
<a name="WSL_OptionsTabPage"></a>

# Options Tab

The Options tab page contains the following fields:

- Executable Options (TA_CLOPT)
- Environment File (TA_ENVFILE)
- Restart Command (TA_RCMD)
- Automatic Restart (TA_RESTART)
- Grace Period (TA_GRACE)
- Activation Sequence Number (TA_SEQUENCE)

This section describes these fields.

<a name="12196"></a>

<a name="WSL_OptionsTabPage_TA_CLOPT"></a>

## Executable Options (TA_CLOPT)

### Description

Command-line options to be passed to the WSL server when it is activated.

### Valid Values

A string of 0 to 128 characters.

### Default

```
-A
```

### Notes

Run-time modifications to this attribute will not affect a running WSL server.

<a name="12199"></a>

<a name="WSL_OptionsTabPage_TA_ENVFILE"></a>

## Environment File (TA_ENVFILE)

### Description

The environment file specific to the WSL server.

## Valid Values

A string of 0 to 78 characters.

## Default

No entry

## Notes

Run-time modifications to this attribute will not affect a running WSL server.

<a name="12201"></a>

<a name="WSL_OptionsTabPage_TA_RCMD"></a>

# Restart Command (TA_RCMD)

## Description

An application-specified command to be executed in parallel with the system restart of an application server. This command must be an executable file in the native operating system.

## Valid Values

A string of string 0 to 78 characters.

## Default

No entry

<a name="12202"></a>

<a name="WSL_OptionsTabPage_TA_RESTART"></a>

# Automatic Restart (TA_RESTART)

## Description

Restartable (Y) or non-restartable (N) WSL server. If server migration is specified for this server group, then this attribute must be set to Y.

## Valid Values

Y or N

### Default

Y

<a name="12181"></a>

<a name="WSL_OptionsTabPage_TA_GRACE"></a>

## Grace Period (TA_GRACE)

### Description

The period of time, in seconds, over which the limit specified in the Max Generations (TA_MAXGEN) for the Workstation Listener class applies.

The value in this field is meaningful only for restartable WSL servers, i.e., if you have entered Y in the Automatic Restart (TA_RESTART) for the Workstation Listeners class. When a restarting server would exceed the limit specified in the Max Generations field but the period specified in the Grace Period field has expired, the system resets the current generation to 1 and resets the initial boot time to the current time.

A value of 0 for this attribute indicates that the WSL server should always be restarted.

### Valid Values

A number greater than or equal to zero.

### Default

0

<a name="12185"></a>

<a name="WSL_OptionsTabPage_TA_SEQUENCE"></a>

## Activation Sequence Number (TA_SEQUENCE)

### Description

Specifies when this server should be booted (tmboot(1)) or shutdown (tmshutdown(1)) relative to other servers. If two servers are given the same sequence number, it is possible for tmboot(1) to boot them in parallel and for tmshutdown(1) to shut them down in parallel. If you add a workstation listener without specifying a value in the Activation Sequence Number field (or if you enter an invalid value), the workstation listener that is generated will be 10,000 or more — higher than any other automatically-selected default value.

Servers are booted by tmboot(1) in increasing order of sequence number and shut down by tmshutdown(1) in decreasing order. Run-time modifications to this attribute affect only tmboot(1) and tmshutdown(1) and will affect the order in which running servers may be shut down by a subsequent invocation of tmshutdown(1).

### Valid Values

A number in the range of 1 to 10,000 inclusive.

<a name="wsl_sysp"></a>

# System Parameters Tab

The System Parameters tab page contains the following fields:

- Compression Limit (TA_CMPLIMIT)
- Suspend Status (TA_SUSPENDED)
- Current Number of Handlers (TA_CURHANDLERS)
- Highest Number of Handlers (TA_HWHANDLERS)
- Version Number (TA_WSPROTO)
- Minimum Encryption Bits (TA_MINENCRYPTBITS)
- Maximum Encryption Bits (TA_MAXENCRYPTBITS)
- Minimum Port Number (TA_MINWSHPORT)
- Maximum Port Number (TA_MAXWSHPORT)

This section describes these fields.

<a name="12197"></a>

<a name="WSL_SysParamsTabPage_TA_CMPLIMIT"></a>

## Compression Limit (TA_CMPLIMIT)

### Description

Message size threshold at which compression will occur for traffic to and from Workstation clients. *threshold* may be either non-negative numeric values or the string MAXLONG, which is dynamically translated to the maximum long setting for the machine.

## Valid Values

*threshold*

## Default

MAXLONG

## Notes

This attribute value is not used for Workstation clients running Oracle Tuxedo Workstation release 6.1 or an earlier release.

&lt;a name="12203"&gt;&lt;/a&gt;

&lt;a name="WSL_SysParamsTabPage_TA_SUSPENDED"&gt;&lt;/a&gt;

# Suspend Status (TA_SUSPENDED)

## Description

This field displays the status of your workstation listener in regard to suspension issues.

## Valid Values

One of the following values is displayed in this field:

NEW

Indicates that new incoming clients may not connect through this WSL object.

ALL

Indicates that Workstation clients already connected to the application through this workstation listener have been SUSPENDED and new incoming connections have been disallowed.

NONE

Indicates that no suspension characteristics are in effect.

## Example

ALL

&lt;a name="12186"&gt;&lt;/a&gt;

<a name="WSL_SysParamsTabPage_TA_CURHANDLERS"></a>

## Current Number of Handlers (TA_CURHANDLERS)

### Description

This field displays the number of currently active handlers associated with this workstation listener (WSL).

<a name="12186"></a>

<a name="WSL_SysParamsTabPage_TA_HWHANDLERS"></a>

## Highest Number of Handlers (TA_HWHANDLERS)

### Description

The maximum number of handlers that should be available in conjunction with this WSL at any given time. Handlers are started as necessary to meet the demand of Workstation clients attempting to access the system. This attribute must be greater than or equal to the setting for the minimum number of handlers.

### Valid Values

A number in the range of zero to 32K inclusive.

### Default

0

<a name="12193"></a>

<a name="WSL_SysParamsTabPage_TA_WSPROTO"></a>

## Version Number (TA_WSPROTO)

### Description

This field displays the Oracle Tuxedo Workstation protocol version number for this Workstation group. Note that Workstation clients connecting to this group may themselves have a different protocol version number associated with them.

<a name="12179"></a>

<a name="WSL_SysParamsTabPage_TA_MINENCRYPTBITS"></a>

## Minimum Encryption Bits (TA_MINENCRYPTBITS)

### Description

When connecting to Oracle Tuxedo, require at least this minimum level of encryption. If this level of encryption cannot be met, link establishment will fail.

### Valid Values

`0` or `40` or `128` where `0` means no encryption and `40` and `128` specify encryption key lengths (in bits).

### Default

`0`

<a name="12180"></a>
<a name="WSL_SysParamsTabPage_TA_MAXENCRYPTBITS"></a>

## Maximum Encryption Bits (TA_MAXENCRYPTBITS)

### Description

When connecting to Oracle Tuxedo, negotiate encryption up to this level.

### Valid Values

`0` or `40` or `128` where `0` means no encryption and `40` and `128` specify encryption key lengths (in bits).

### Default

`128`

<a name="12208"></a>

<a name="WSL_SysParamsTabPage_TA_MINWSHPORT"></a>

## Minimum Port Number (TA_MINWSHPORT)

### Description

The lower end of the range of available port numbers that may be allocated to WSH processes by this listener.

### Valid Values

A number in the range of 0 to 65,535 inclusive.

### Default

2048

### Example

4096

<a name="12209"></a>

<a name="WSL_SysParamsTabPage_TA_MAXWSHPORT"></a>

## Maximum Port Number (TA_MAXWSHPORT)

### Description

The upper end of the range of available port numbers that may be allocated to WSH processes by this listener.

### Valid Values

A number in the range of 0 to 65,535 inclusive.

### Default

65,535

### Example

60,000

<a name="wsl_stats"></a>

# Statistics Tab

The Statistics tab page contains the following fields:

- Max Generations (TA_MAXGEN)

- Min Number of Handlers (TA_MINHANDLERS)

- Max Number of Handlers (TA_MAXHANDLERS)

- Max Number of Clients (TA_MULTIPLEX)

- Max Idle Time (TA_MAXIDLE)

- Max Initialization Time (TA_MAXINIT)

This section describes these fields.

<a name="12183"></a>

<a name="WSL_StatsTabPage_TA_MAXGEN"></a>

## Max Generations (TA_MAXGEN)

### Description

The number of generations allowed for a restartable WSL server over the specified grace period. The initial activation of the WSL server counts as one generation; each restart also counts as one. Processing after the maximum generations is exceeded is discussed in the description of the Grace Period (TA_GRACE) on the Options tab page for Workstation Listeners.

### Valid Values

A number in the range of one to 1 to 255 inclusive.

### Default

```
1
```

<a name="12191"></a>

<a name="WSL_StatsTabPage_TA_MINHANDLERS"></a>

## Min Number of Handlers (TA_MINHANDLERS)

### Description

The minimum number of handlers that should be available in conjunction with this Workstation Listener (WSL) at any given time. The WSL will start this many Workstation Handlers (WSH) immediately upon being activated and will not deplete the supply of WSHs below this number until the administrator issues a shutdown to the WSL. Modifications to this attribute for a running WSL may cause additional handlers to be activated.

### Valid Values

A number in the range of 0 to 255 inclusive.

### Default

0

<a name="12188"></a>
<a name="WSL_StatsTabPage_TA_MAXHANDLERS"></a>

## Max Number of Handlers (TA_MAXHANDLERS)

### Description

The maximum number of handlers that should be available in conjunction with this Workstation Listener at any given time. Handlers are started as necessary to meet the demand of Workstation clients attempting to access the system. This attribute must be greater than or equal to the setting for the minimum number of handlers.

### Valid Values

A number in the range of 0 to 32,766 inclusive.

### Default

0

### Tips

A value of 0 in this field means that the effective value is determined at activation time from the current value in the Max Handlers field and the Max Workstation Clients field for the Machines

class (T_MACHINE). Note that a GET operation with the MIB_LOCAL flag set will get the effective value for objects with an activation time default setting.

<a name="12192"></a>

<a name="WSL_StatsTabPage_TA_MULTIPLEX"></a>

## Max Number of Clients (TA_MULTIPLEX)

### Description

Maximum number of clients to be supported by any one handler process concurrently.

### Valid Values

A number in the range of 1 to 32,767 inclusive.

### Default

10

<a name="12189"></a>

<a name="WSL_StatsTabPage_TA_MAXIDLE"></a>

## Max Idle Time (TA_MAXIDLE)

### Description

The maximum amount of time (in minutes) that a Workstation client is permitted to be idle before it is abortively disconnected from the application by the handler. A value of 35,204,649 allows clients to be idle as long as they wish without being timed out. A value of 0 means that clients may be terminated after any period of inactivity greater than 1 second.

### Valid Values

A number in the range of 0 to 35,204,649 inclusive.

### Default

35,204,649

<a name="12190"></a>

<a name="WSL_StatsTabPage_TA_MAXINIT"></a>

### Max Initialization Time (TA_MAXINIT)

Description

The minimum amount of time (in seconds) that should be allowed for a Workstation client to complete initialization processing through the Workstation Handler before being timed out by the Workstation Listener.

Valid Values

A number in the range of 1 to 32,766 inclusive.

Default

60

<a name="T_WSH"></a>

## Workstation Handlers (WSHs)

The Workstation Handler object allows you to specify the workstation statistics specific to a particular WSH client process.

<a name="T_WSH"></a>

## Configuring Workstation Handlers (T_WSH Class)

To configure Workstation Handlers, select WS Handlers in the Tree view. Enter values for the fields in the tab pages.

The Configuration Tool pane contains the following tab pages for specifying the attributes of WS Handlers: General, Handler Info, and Statistics.

<a name="C_WSH"></a>

## General Tab

<a name="WSH_GeneralTabPage"></a>

The General tab page contains the following fields:

- Client ID (TA_CLIENTID)

- WorkStation Client ID (TA_WSHCLIENTID)

- Logical Machine Name, Backup (TA_LMID)

- Process ID (TA_PID)

- Group Name (TA_SRVGRP)

- Server ID (TA_SRVID)

- Group Number (TA_GRPNO)

- Object State (TA_STATE)

This section describes these fields.

\<a name="12227"\>\</a\>

\<a name="WSH_GeneralTabPage_TA_CLIENTID"\>\</a\>

## Client ID (TA_CLIENTID)

### Description

This field displays the client identifier for this workstation handler. The data in this field should not be interpreted directly by the user except for equality comparison.

\<a name="12230"\>\</a\>

\<a name="WSH_GeneralTabPage_TA_WSHCLIENTID"\>\</a\>

## WorkStation Client ID (TA_WSHCLIENTID)

### Description

This field displays the client identifier for this workstation handler. The data in this field should not be interpreted directly by the end user except for an equality comparison. This field can be used to link the workstation handler to its associated Workstation client objects in the Clients class. The value in this field is always equal to the value in the Client ID (TA_CLIENTID) for this class.

\<a name="12231"\>\</a\>

<a name="WSH_GeneralTabPage_TA_LMID"></a>

## Logical Machine Name, Backup (TA_LMID)

### Description

This field displays the logical machine identifier for the machine on which the workstation handler is currently running.

<a name="12214"></a>

<a name="WSH_GeneralTabPage_TA_PID"></a>

## Process ID (TA_PID)

### Description

This field displays the native operating system process identifier for the Workstation client. Note that this may not be a unique attribute since clients may be located on different machines, allowing for the duplication of process identifiers.

<a name="12232"></a>

<a name="WSH_GeneralTabPage_TA_SRVGRP"></a>

## Group Name (TA_SRVGRP)

### Description

This field displays the logical name of the server group for the associated workstation listener.

<a name="12225"></a>

<a name="WSH_GeneralTabPage_TA_SRVID"></a>

## Server ID (TA_SRVID)

### Description

This field displays the server identification number for the associated workstation listener. This number is unique within the server group.

<a name="12212"></a>

<a name="WSH_GeneralTabPage_TA_GRPNO"></a>

## Group Number (TA_GRPNO)

### Description

This field displays the group number associated with this group of servers.

<a name="12226"></a>

<a name="WSH_GeneralTabPage_TA_STATE"></a>

## Object State (TA_STATE)

### Description

This field displays the state of the workstation handler client within the application. Any state defined for the Clients class in TM_MIB(5) may be returned or set as indicated on that reference page.

State changes to the SUSPENDED state affect all clients associated with this workstation handler as does the resetting of a SUSPENDED workstation handler to ACTIVE. Additionally, SUSPENDED workstation handler clients will not be assigned any additional incoming clients by the workstation listener.

Note that the state of a workstation handler client may not be set to DEAD when the Clients class is being accessed. However, the state transition to DEAD is allowed via the Workstation Handlers class. Such a transition will result in all connections that are being handled by the targeted workstation handler being dropped abortively.

<a name="wsh_haninfo"></a>

# Handler Info Tab

<a name="WSH_HanderInfoTabPage"></a>

The Handler Info tab page contains the following fields:

- Max Number of Clients (TA_MULTIPLEX)
- Highwater Number of Clients (TA_HWCLIENTS)
- Current Number of Clients (TA_CURCLIENTS)

- Seconds Remaining Before Timeout (TA_TIMELEFT)

- Current Status of WSH (TA_ACTIVE)

- Total Active Time (TA_TOTACTTIME)

- Total IDLE Time (TA_TOTIDLTIME)

- Network Address (TA_NADDR)

This section describes these fields.

<a name="12221"></a>

<a name="WSH_HandlerInfoTabPage_TA_MULTIPLEX"></a>

## Max Number of Clients (TA_MULTIPLEX)

### Description

This field displays the maximum number of clients that may access the application via this workstation handler.

<a name="12213"></a>

<a name="WSH_HandlerInfoTabPage_TA_HWCLIENTS"></a>

## Highwater Number of Clients (TA_HWCLIENTS)

### Description

The number displayed in this field is the high-water mark of clients accessing the application via this workstation handler.

<a name="12210"></a>

<a name="WSH_HandlerInfoTabPage_TA_CURCLIENTS"></a>

## Current Number of Clients (TA_CURCLIENTS)

### Description

This field displays the number of clients currently accessing the application via this workstation handler.

<a name="12219"></a>

<a name="WSH_HandlerInfoTabPage_TA_TIMELEFT"></a>

## Seconds Remaining Before Timeout (TA_TIMELEFT)

### Description

If there is a value other than zero displayed in this field, it means that the new Workstation client that has just been assigned to this workstation handler has the indicated number of seconds in which to complete the initialization process with the workstation handler.

<a name="12229"></a>

<a name="WSH_HandlerInfoTabPage_TA_ACTIVE"></a>

## Current Status of WSH (TA_ACTIVE)

### Description

This field displays the state of the workstation handler. A value of Y indicates that the workstation handler is currently performing work on behalf of one of its associated Workstation clients. A value of N indicates that the workstation handler is currently waiting for work to perform on behalf of one of its associated Workstation clients.

<a name="12223"></a>

<a name="WSH_HandlerInfoTabPage_TA_TOTACTTIME"></a>

## Total Active Time (TA_TOTACTTIME)

### Description

This field displays the time, in seconds, that the workstation handler has been active since it started processing.

## Total IDLE Time (TA_TOTIDLTIME)

### Description

This field displays the time, in seconds, that the workstation handler has been idle since it started processing.

### Network Address (TA_NADDR)

Description

This field displays the network address of a Workstation client. Network addresses with unprintable characters are converted to the "0x . . ." network address format (as described for the Network Address field on the Networking tab page for the Machines class). If the address is a TCP/IP address, then it is returned in the *dotted_decimal*:*port_number* format:

```
//#.#.#.#.port_number
```

Each # (pound sign) represents a decimal number in the range of 0 to 255. The value of *port_number* is a decimal number in the range 0 to 65535. No value will be displayed (that is, the field will be blank) for a non-Workstation client.

&lt;a name="wsh_stats"&gt;&lt;/a&gt;

# Statistics Tab

&lt;a name="WSH_StatsTabPage"&gt;&lt;/a&gt;

The Statistics tab page contains the following fields:

- Current Workload (TA_CURWORK)
- Flow Control Events (TA_FLOWCNT)
- Queue Blocking Events (TA_NUMBLOCKQ)
- Bytes Received (TA_RCVDBYT)
- System Messages Received (TA_RCVDNUM)
- Bytes Sent (TA_SENTBYT)
- System Messages Sent (TA_SENTNUM)

This section describes these fields.

&lt;a name="12220"&gt;&lt;/a&gt;

<a name="WSH_StatsTabPage_TA_CURWORK"></a>

## Current Workload (TA_CURWORK)

### Description

This field displays the amount of work processed by this workstation handler since the last workstation handler assignment by the workstation listener. The workstation listener uses this value to load-balance new incoming connections among a set of workstation handler processes.

<a name="12211"></a>

<a name="WSH_StatsTabPage_TA_FLOWCNT"></a>

## Flow Control Events (TA_FLOWCNT)

### Description

This field displays the number of times flow control has been encountered by this workstation handler. This attribute should be considered only in relation to recent past values as it may wrap around during the lifetime of the workstation handler.

<a name="12222"></a>

<a name="WSH_StatsTabPage_TA_NUMBLOCKQ"></a>

## Queue Blocking Events (TA_NUMBLOCKQ)

### Description

This field displays the number of times this workstation handler has been unable to enqueue a message to a local UNIX system message queue due to queue blocking conditions. This attribute should be considered only in relation to recent past values as it may wrap around during the lifetime of the workstation handler.

<a name="12215"></a>

<a name="WSH_StatsTabPage_TA_RCVDBYT"></a>

## Bytes Received (TA_RCVDBYT)

### Description

This field displays the number of bytes received from the network by this workstation handler from all of its present and past Workstation clients. This attribute should be considered only in relation to recent past values as it may wrap around during the lifetime of the workstation handler.

<a name="12216"></a>

<a name="WSH_StatsTabPage_TA_RCVDNUM"></a>

## System Messages Received (TA_RCVDNUM)

### Description

This field displays the number of Oracle Tuxedo system messages received from the network by this workstation handler from all of its present and past Workstation clients. This attribute should be considered only in relation to recent past values as it may wrap around during the lifetime of the workstation handler.

## Bytes Sent (TA_SENTBYT)

### Description

This field displays the number of bytes sent to the network by this workstation handler to all of its present and past Workstation clients. This attribute should be considered only in relation to recent past values as it may wrap around during the lifetime of the workstation handler.

<a name="12218"></a>

<a name="WSH_StatsTabPage_TA_SENTNUM"></a>

## System Messages Sent (TA_SENTNUM)

### Description

This field displays the number of Oracle Tuxedo system messages sent to the network by this workstation handler to all of its present and past Workstation clients. This attribute should be

considered only in relation to recent past values as it may wrap around during the lifetime of the workstation handler.

# Creating and Configuring Access Control Lists

An *access control list* (ACL) is a list that specifies who and what is authorized to access BEA Tuxedo system objects. The ACL enables a system manager to administer security through authenticating users, setting permissions, and controlling access. The BEA Tuxedo Administration Console ACL objects allow the system manager to create and configure ACL objects. These objects are grouped into the following major categories:

- ACL Groups

  - Creating ACL Groups

  - Configuring ACL Groups (T_ACLGROUP Class)

- ACL Principals (users or domains)

  - Creating ACL Principals

  - Configuring ACL Principals (T_ACLPRINCIPAL Class)

- ACLs

  - Creating ACLs

  - Configuring ACLs (T_ACLPERM Class)

# ACL Groups

The ACL Groups represent the groups of BEA Tuxedo application users and domains.

<a name="N_ACLGROUP"></a>

# Creating ACL Groups

To create new ACL Groups:

1. Select the ACL Groups folder in the tree view.

2. Select Quick Menu —>New (right-click) to display the New Object window.

3. Enter values for the following fields:

   – Group Name (TA_GROUPNAME)

   – Group ID (TA_GROUPID) (optional)

4. Click the Create button to close the window and create the new ACL group.

5. <a name="T_ACLGROUP"></a>

# Configuring ACL Groups (T_ACLGROUP Class)

To configure ACL Groups, choose an ACL Groups object in the tree view. Enter values for the fields in the General tab displayed in the Configuration Tool pane.

<a name="C_ACLGROUP"></a>

## General Tab

The General tab page lists the ACL Groups Class attributes and includes the following fields:

- Group Name (TA_GROUPNAME)
- Group ID (TA_GROUPID)
- Object State (TA_STATE)
- <a name="13041"></a> <a name="12379"></a>

## Group Name (TA_GROUPNAME)

### Description

Logical name of an ACL group. An ACL group is a set of users characterized by the type of permission to access particular administrative objects, such as services.

### Valid Values

A string of one to 30 characters. The group name must be printable and it may not include a pound sign (#), a comma, a colon, or a newline character.

### Example

```
banktellers
```

<a name="13084"></a> <a name="12377"></a>

## Group ID (TA_GROUPID)

### Description

A decimal number representing the ACL group specified in the Group Name (TA_GROUPNAME). A value of 0 indicates the default group other. If not specified at creation time, it defaults to the next available (unique) identifier greater than 0.

### Valid Values

A number greater than 0 but less than 16,384.

### Default

A unique ID number is assigned.

### Example

```
201
```

### Notes

0 is reserved for a group named other which is provided by BEA Tuxedo. (You can add users to this group but you cannot modify or delete the group.)

<a name="12378"></a>

### Object State (TA_STATE)

Description

This field displays the state of your ACL group. An ACL group may be in only one state: VALID.

# ACL Principals

The users and remote domains in an application that need authentication and authorization are collectively known as *principals*. To join an application as a specific user, it is necessary to present a user-specific password.

<a name="N_ACLPRINCIPAL"></a>

# Creating ACL Principals

To create new ACL Principals:

1. Select the ACL Principals folder in the tree view.

2. Select Quick Menu —>New (right-click) to display the New Object window.

3. Enter values for the following fields:

   – Principal Name (TA_PRINNAME)

   – Password (TA_PRINPASSWD) (optional)

4. Click the Create button to close the window and create the new ACL principals.

The rest of this section describes the fields on the Create New Object window.

<a name="T_ACLPRINCIPAL"></a>

# Configuring ACL Principals (T_ACLPRINCIPAL Class)

To configure ACL Principals, choose an ACL Principals object in the tree view. Enter values for the fields in the Configuration Tool General tab page.

<a name="C_ACLPRINCIPAL"></a>

## General Tab

The General tab page lists the ACL Principals Class attributes and includes the following fields:

- Principal Name (TA_PRINNAME)

- Principal Client Name (TA_PRINCLTNAME)

- Principal ID (TA_PRINID)

- Principal Group (TA_PRINGRP)

- Password (TA_PRINPASSWD)

- Object State (TA_STATE)

- <a name="13044"></a> <a name="12388"></a>

## Principal Name (TA_PRINNAME)

### Description

Logical name of a principal. A principal is a user or domain that can access a BEA Tuxedo application.

### Valid Values

A string from 1 to 30 characters in length.

The principal name must be printable and it cannot contain a pound sign (#), a colon, or a newline character.

### Example

```
bill_jones
```

<a name="12387"></a>

## Principal Client Name (TA_PRINCLTNAME)

### Description

The client name associated with a principal (that is, a user or domain that can access a BEA Tuxedo application). The client name provides a further qualifier on the user entry that is checked for authentication.

### Valid Values

A string from 1 to 30 characters in length.

The client name must be printable and it cannot contain a colon or a newline character.

### Default

\* (wildcard character)

### Example

tpsysusr (logs you in as the BEA Tuxedo application administrator)

typsysop (logs you in as the BEA Tuxedo application operator)

<a name="12385"></a>

## Principal ID (TA_PRINID)

### Description

A unique identification number for an ACL principal.

### Valid Values

A number in the range of 1 to 131,071 inclusive.

### Default

A unique ID number is assigned.

### Example

2001

### Notes

BEA Tuxedo reserves principal client names for the administrator and operator of a Tuxedo application. The name tpsysadm logs in the associated user as the administrator; tpsysop logs in the associated user as the operator. If you plan to use either of these names, be sure to specify it in this field. If you do not, authentication will fail and the designated user will not be able to log in as the administrator (or operator).

<a name="12384"></a>

## Principal Group (TA_PRINGRP)

### Description

A group ID number that specifies the group to which a principal belongs. A value of 0 indicates the default group other. If not specified at creation time, the default value 0 is assigned.

### Valid Values

A number in the range of 0 to 16,383 inclusive.

### Default

0 (assigned to group other)

### Example

201

### Notes

0 is reserved for a group named other which is provided by BEA Tuxedo. If you do not assign principals to a group, they will be assigned, by default, to group other.

<a name="13076"></a> <a name="12389"></a>

## Password (TA_PRINPASSWD)

### Description

The authentication password for the associated ACL principal.

### Valid Values

A string.

### Example

obi1kenobi

### Notes

The system automatically encrypts this password.

<a name="12386"></a>

### Object State (TA_STATE)

Description

This field displays the state of your ACL principal. An ACL principal may be in only one state:
VALID.

## ACLs

The ACLs, as a whole, comprise the principals and access control lists for BEA Tuxedo
applications services, application queues, and events. The ACLs indicate what groups are
allowed to access BEA Tuxedo system entities.

<a name="N_ACLPERM"></a>

## Creating ACLs

To create new ACLs:

1. Select the ACLs folder in the tree view.

2. Select Quick Menu —>New (right-click) to display the New Object window.

3. Enter values for the following fields:

   – ACL Entity Name (TA_ACLNAME)

   – ACL Entity Type (TA_ACLTYPE) (optional)

   – Groups (TA_ACLGROUPIDS)

4. Click the Create button to close the window and create the new ACL group.

5. <a name="T_ACLPERM"></a>

## Configuring ACLs (T_ACLPERM Class)

The Configuration Tool pane displays one General tab page with the fields for configuring ACLs.

<a name="C_ACLPERM"></a>

## General Tab

The General tab page lists the ACLs Class attributes and include the following fields:

- ACL Entity Name (TA_ACLNAME)

- ACL Entity Type (TA_ACLTYPE)

- Groups (TA_ACLGROUPIDS)

- Object State (TA_STATE)

- <a name="13042"></a>

- <a name="12382"></a>

## ACL Entity Name (TA_ACLNAME)

### Description

The name of a service, event or queue (referred to, collectively, as an "entity") for which permissions are being granted.

### Valid Values

A string from 1 to 30 characters in length.

The ACL entity name must be printable and it cannot contain a colon, a pound sign (#), or a newline character.

### Example

```
TRANSFER
```

<a name="13043"></a> <a name="12383"></a>

## ACL Entity Type (TA_ACLTYPE)

### Description

The type of the entity for which permissions are being granted.

### Valid Values

ENQ or DEQ or SERVICE or POSTEVENT

### Example

```
SERVICE
```

```
<a name="13083"></a> <a name="12381"></a>
```

## Groups (TA_ACLGROUPIDS)

### Description

A comma-separated list of numeric IDs for groups that are permitted access to the associated entity.

### Valid Values

A string.

The length of this list is limited only by the amount of disk space on the machine.

### Example

```
201, 301
```

### Notes

The values in this list are the Group ID numbers you entered for the relevant ACL groups on the General tab page for the ACL Groups class. See Group ID (TA_GROUPID).

```
<a name="12380"><a>
```

## Object State (TA_STATE)

### Description

This field displays the state of your ACL. An ACL may be in only one state: VALID.

# Creating and Configuring Tuxedo CORBA Resources

The BEA Tuxedo system core administrative resources consists of both ATMI and CORBA domains and their associated resources. The CORBA interface queues and CORBA factories are used in BEA Tuxedo client and server applications. The Oracle Administration Console enables the system administrator to manage the BEA Tuxedo CORBA resources for configuration and run-time administration.

**Note:** For detailed instructions on creating and configuring core BEA Tuxedo ATMI administrative resources, refer to Chapter 4, "Creating and Configuring Core Administrative Resources."

The following sections define the fields for the Create New Object window and Configuration Tool tab pages when creating or configuring BEA Tuxedo CORBA resources:

- CORBA Interfaces (T_INTERFACE)
  - Creating a New CORBA Interface
  - Configuring a CORBA Interface (T_INTERFACE)
- CORBA Interface Queues (T_IFQUEUE)

  Viewing a CORBA Interface Queue Configuration (IF_QUEUE)

- CORBA Factories (T_FACTORY)

  Viewing a CORBA Factory Configuration

# CORBA Interfaces (T_INTERFACE)

A CORBA Interface represents a contract with a client. Interfaces are advertised by BEA Tuxedo server applications at startup. The BEA Tuxedo T_INTERFACE MIB represents configuration and run-time attributes of CORBA interfaces at both the domain and server group levels. Using the BEA Tuxedo Administration Console applet, you can perform the administrative tasks required to create and manage BEA Tuxedo CORBA interfaces including defining default attributes for interfaces at the domain and server group levels.

<a name="newcorbainterface"></a>

<a name="T_INTERFACE"></a>

<a name="C_INTERFACE"></a>

<a name="N_INTERFACE"></a>

# Creating a New CORBA Interface

Select the CORBA Interfaces folder and select Quick Menu—>New to open the Create New Object window. The Create New Object Window contains the following fields:

- Interface Name

- Group Name

To create a new interface, enter the required information for the new CORBA interface and click the Create button.

**Note:** For information about using the Quick Menu, see Quick Menu Options. For information about the Create New Object window, see Adding New Administrative Objects.

The following sections describe the information you need to enter in the Create New Object window for each interface field.The format of the fields you enter may be dependent upon the IDL that generates the interface implementation, For more information about the IDL format, see the CORBA 2.1 Specification Section 7.6.

<a name="ncif_interfacename"></a>

<a name="ac_ifcno_inf1"></a>

### Interface Name

#### Description

This name represents the interface repository ID for the interface. The format of the name is dependent on the options specified in the IDL that generates the interface implementation.

#### Valid Values

A fully qualified interface name, where name is a text string from 1 to a maximum of 128 characters. You cannot leave this field blank.

#### Example

```
IDL:Simp:1.0
```

```
<a name="ac_ifcno_grp1"></a><a name="ncif_groupname"></a>
```

### Group Name

#### Description

This name represents the name of the server group associated with this interface.

#### Valid Values

A group name, where the name is a text string from 0 to a maximum of 30 characters. Server group names cannot contain an asterisk, comma, or colon. You cannot leave this field blank.

#### Example

```
GRP9BANKERS
```

```
<a name="concorbainterface"></a>
```

# Configuring a CORBA Interface (T_INTERFACE)

The BEA Tuxedo Administration Console Interface (T_INTERFACE) configuration attributes are displayed on the following tab pages:

- General Tab

- System Parameters Tab

- Policies Tab

- Statistics Tab

To view or edit the configuration attributes for an Interface, highlight an interface icon in the main window tree. The Configuration Tool pane displays the tabs containing the configuration attributes for your selection. For more information about the Configuration Tool, see Configuring Administrative Objects.

The following sections describe each interface configuration attribute field.The format of the editable field may be dependent upon the IDL that generates the interface implementation. For more information about the IDL format, see the CORBA 2.1 Specification Section 7.6.

<a name="ccif_generaltab"></a>

## General Tab

The following describes the fields provided on the General tab and the values you need to enter for those fields.

<a name="16301"></a>

<a name="ac_ifgen_inf2"></a>

### Interface Name

#### Description

The Interface Name field displays the fully qualified interface name that was specified in the Create New Objects Interface Name field. For more information, see Creating a New CORBA Interface.

<a name="16302"></a>

<a name="ac_ifgen_grp2"></a>

### Group Name

#### Description

The Group Name field displays the name of the server group associated with this interface that was specified in the Create New Objects Group Name field. For more information, see Creating a New CORBA Interface.

<a name="16303"></a>

<a name="ac_ifgen_obs2"></a>

## Object State

### Description

The Object State field displays the state attributes for Interface Queue objects.

### Valid Values

This read-only field displays one of the following values:

ACT             Active state. The Interface Queue object is an available interface in the
                running system.

SUS             Suspend state. The Interface Queue object represents a currently suspended
                interface in the running system.

PAR             Partitioned state. The Interface Queue object represents a currently
                partitioned interface in the running system.

The Object States for an Interface Queue are set by selecting an Interface Queue icon and using
the Quick Menu to suspend or resume the object. If the Interface Queue object is suspended, you
can activate the object. To restore a suspended Interface Queue object to an active state, select
Resume from the Quick Menu. If the Interface Queue object is active, you can suspend the object.
To suspend an Interface Queue object, highlight the object and select Suspend from the Quick
Menu.

**Note:**  The BEA Tuxedo Administration Console does not support dynamic advertisement of
interfaces (i.e., state change from INActive or INValid to ACTive) or unadvertisement
of interfaces (i.e., state change from ACTive to INActive).

**Note:**  <a name="16310"></a>

**Note:**  <a name="ac_ifgen_lmn2"></a>

## Logical Machine Name

### Description

The Logical Machine Name field displays the name of the current logical machine on which the
queue offering this interface is loaded.

<a name="ccif_sysparmtab"></a>

# System Parameters Tab

The following sections describes the fields provided on the System Parameters tab and the values you need to enter for those fields.

<a name="16305"></a>

<a name="ac_ifsyp_ifl2"></a>

## Interface Load

### Description

The Interface Load field displays a value representing the load on the system imposed by the T_INTERFACE object. Interface loads are used for load balancing purposes. Queues with higher enqueued workloads are less likely to be chosen for a new request.

### Valid Values

Any integer from 1 to a maximum of 32K.

### Example

```
50
```

```
<a name="16306"></a>
```

```
<a name="ac_ifsyp_pri2"></a>
```

## Priority

### Description

The Priority field displays the dequeueing priority. If multiple interface requests are waiting on a queue for servicing, the higher priority requests are handled first.

### Valid Values

A number from 1 a maximum of 101.

### Example

```
101
```

```
<a name="16307"></a>

<a name="ac_ifsyp_ift2"></a>
```

## Interface Timeout

### Description

The Interface Timeout field displays the time limit (in seconds) for processing individual method invocations for the selected interface queue. Servers processing method invocations for this interface are abortively terminated if they exceed the specified time limit in processing the request.

### Valid Values

A number greater than or equal to zero. A value of zero for this attribute indicates that the server should not be abortively terminated.

### Example

```
30

<a name="16308"></a>

<a name="ac_ifsyp_trt2"></a>
```

## Transaction Timeout

### Description

The Transaction Timeout field displays the timeout value in seconds for transactions automatically started for this interface queue. Transactions are started automatically when a request not in transaction mode is received and the Interface AUTOTRAN attribute value is set to Y. For more information about the Interface AUTOTRAN attribute, see the Interface Auto Transaction Mode field.

### Valid Values

A number greater than or equal to zero, representing a timeout interval in seconds.

## Example

```
60

<a name="16309"></a>

<a name="ac_ifsyp_rtn2"></a>
```

## Routing Table Name

### Description

The Routing Table Name field displays the factory-based routing criteria associated with this interface.

### Valid Values

A string from 1 a maximum of 15 characters.

### Example

```
RTBLSOPAC

<a name="ccif_policies"></a>
```

# Policies Tab

The following topics describe the Policies tab fields provided for configuring CORBA Interfaces.

```
<a name="16304"></a>

<a name="ac_ifpol_atm2"></a>
```

## Auto Transaction Mode

### Description

The Auto Transaction Mode field indicates whether a transaction is automatically started for invocations made outside a transaction context. The transaction policy setting can override the AUTOTRAN attribute. See Transaction Policy for more information.

### Valid Values

Select Yes or No from the drop-down list box in this field. Select Yes to automatically start a transaction. Select No to prevent transactions from automatically starting.

<a name="16311"></a>

<a name="ac_ifpol_dep2"></a>

## Deactivation Policy

### Description

The Deactivation Policy field displays the TP Framework deactivation policy. This reflects the policy registered with the framework at server startup. The first server to register the interface sets the value in T_INTERFACE. You can not change this value.

The values for the deactivation policy are: method, transaction, process.

<a name="16312"></a>

<a name="ac_ifpol_trp2"></a>

## Transaction Policy

### Description

The Transaction Policy field displays the policy attribute for the interface. The Transaction Policy is a read-only field. The developer sets the value for this attribute when the server application is built and registered at server startup.

The transaction policy setting can override the AUTOTRAN attribute. For more information about how these settings affect the AUTOTRAN attribute, see the Interface Auto Transaction Mode field.

The values displayed for the Transaction Policy are listed in the following table and include a description of how these settings override the AUTOTRAN attribute.

| Valid Values | AUTOTRAN Setting | Description |
| --- | --- | --- |
| always | No or Yes | No change occurs at run time. The behavior is the same as if AUTOTRAN was set to Y. |
| never | Y | No change occurs. The interface will never be involved in a transaction. |
| ignore | Y | Overrides the AUTOTRAN setting. The interface will never be involved in a transaction. |
| optional | Y or N | Has no effect on AUTOTRAN setting. |

<a name="ccif_statstab"></a>

# Statistics Tab

The following topics describe the fields provided on the Configuration Statistics tab and the values you need to enter for those fields.

<a name="16313"></a>

<a name="ac_ifstat_rec2"></a>

## Requests Completed

### Description

The Requests Completed field contains the number of interface method invocations completed with respect to the corresponding T_IFQUEUE objects since they were initially offered. Local queries on domain level objects will return one object per machine with the statistics for the indicated interface on that machine.

### Valid Values

A number greater than or equal to zero.

<a name="16314"></a>

<a name="ac_ifstat_req2"></a>

### Requests Queued

Description

The Requests Queued field displays the number of requests currently enqueued for this interface. The displayed number is a long integer greater than or equal to zero. This is a read-only field.

## CORBA Interface Queues (T_IFQUEUE)

The T_IFQUEUE MIB represents run-time attributes of an interface as it pertains to a particular server queue (T_QUEUE). This is primarily a read-only class providing access to the inherited configuration attributes of an interface as well as statistics relating to the interface on the queue. Additionally, this class gives administrators finer granularity in suspending and activating interfaces. This class provides the link between an interface name and the server processes capable of processing method invocations on the interface (i.e., TA_RQADDR can be used as a key search field on the T_SERVER class).

<a name="newinterfaceq"></a>

<a name="T_IFQUEUE"></a>

<a name="C_IFQUEUE"></a>

<a name="N_IFQUEUE"></a>

## Viewing a CORBA Interface Queue Configuration (IF_QUEUE)

The BEA Tuxedo Administration Console Interface Queue (IF_QUEUE) configuration attributes are displayed on the following tab pages:

- General Tab

- Policies Tab

- System Parameters Tab

To view the configuration attributes for an Interface Queue, highlight an interface queue icon. The Configuration Tool pane displays the General tab containing the configuration attributes for your selection.

For more information about the Configuration Tool, see Configuring Administrative Objects. For more information about the IDL format for interface queue attributes, see the CORBA 2.1 Specification Section 7.6.

The following sections describe each interface queue configuration attribute. The Interface Queue configuration attributes are read-only. You cannot edit them.

<a name="nifq_generaltab"></a>

# General Tab

The following sections describe the fields provided on the General tab.These Interface Queue fields are read-only attributes.

<a name="15200"></a>

<a name="ac_ifqgen_ifn4"></a>

## Interface Name

### Description

The Interface Name field displays the fully qualified interface name. This is the repository ID for the interface. The interface name is a string from 0 to a maximum of 128 characters. The format of the name is dependent on the options specified in the IDL that generates the interface implementation.

<a name="15201"></a>

<a name="ac_ifqgen_grp4"></a>

## Group Name

### Description

The Group Name field displays the name of the server group associated with this interface. The name is a string from 0 to a maximum of 30 characters. Server group names cannot contain an asterisk, comma, or colon.

## Logical Machine Name

### Description

The Logical Machine Name field displays the name of the current logical machine on which the queue offering this interface is loaded.

&lt;a name="15202"&gt;&lt;/a&gt;

&lt;a name="ac_ifqgen_qad4"&gt;&lt;/a&gt;

## Queue Address

### Description

The Queue Address field displays the symbolic address of the request queue for an active server offering this interface.The address is a string from 1 to a maximum of 30 characters.

&lt;a name="15203"&gt;&lt;/a&gt;

&lt;a name="ac_ifqgen_obs4"&gt;&lt;/a&gt;

## Object State

### Description

The Object State field displays the state attributes for Interface Queue objects. This read-only value displays the following values:

| | |
|---|---|
| ACT | Active state. The Interface Queue object is an available interface in the running system. |
| SUS | Suspend state. The Interface Queue object represents a currently suspended interface in the running system. |
| PAR | Partitioned state. The Interface Queue object represents a currently partitioned interface in the running system. |

The Object States for an Interface Queue are set by selecting an Interface Queue icon and using the Quick Menu to suspend or resume the object. If the Interface Queue object is suspended, you can activate the object. To restore a suspended Interface Queue object to an active state, select Resume from the Quick Menu. If the Interfaces Queue object is active, you can suspend the object. To suspend an Interface Queue object, highlight the object and select Suspend from the Quick Menu.

**Note:** The BEA Tuxedo Administration Console does not support dynamic advertisement of interfaces (i.e., state change from INActive or INValid to ACTive) or unadvertisement of interfaces (i.e., state change from ACTive to INActive).

**Note:** &lt;a name="nifq_policytab"&gt;&lt;/a&gt;

# Policies Tab

The following sections describes the fields provided on the Policies tab.The Interface Queue policies are read-only attributes.

<a name="15204"></a>

<a name="ac_ifqpol_atm4"></a>

## Auto Transaction Mode

### Description

The Auto Transaction Mode field displays the value of Y or N indicating whether a transaction is automatically started for invocations made outside a transaction context. The transaction policy setting can override the AUTOTRAN attribute. For more information about how these settings affect the AUTOTRAN attribute, see the Interface Auto Transaction Mode field.

<a name="15211"></a>

<a name="ac_ifqpol_dep4"></a>

## Deactivation Policy

### Description

The Deactivation Policy field displays the TP Framework deactivation policy. This reflects the policy registered with the framework at server startup. The first server to register the interface sets the value in T_INTERFACE. You cannot change this value.

The values for the deactivation policy are: method, transaction, process.

## Transaction Policy

### Description

The Transaction Policy field displays the policy attribute for the interface. The Transaction Policy is a read-only field. The developer sets the value for this attribute when the server is built and registered at server startup.

The transaction policy setting can override the AUTOTRAN attribute. For more information about the values for this field and how these values affect the AUTOTRAN attribute, see the Interface Transaction Policy field.

<a name="nifq_sysparmtab"></a>

# System Parameters Tab

The following sections describes the fields provided on the System Parameters tab.The Interface Queue system parameters are read-only attributes.

<a name="15205"></a>

<a name="ac_ifqsyp_ifl4"></a>

## Interface Load

### Description

The Interface Load field displays a value representing the load on the system imposed by the T_INTERFACE object. Interface loads are used for load balancing purposes. Queues with higher enqueued workloads are less likely to be chosen for a new request.

The displayed value is a number from 1 to a maximum of 32K.

<a name="15206"></a>

<a name="ac_ifqsyp_pri4"></a>

## Priority

### Description

The Priority field displays the dequeueing priority. If multiple interface requests are waiting on a queue for servicing, the higher priority requests are handled first.

The displayed value is a number from 1 a maximum of 101.

<a name="15207"></a>

<a name="ac_ifqsyp_ift4"></a>

### Interface Timeout

Description

The Interface Timeout field displays the time limit (in seconds) for processing individual method invocations for the selected interface queue. Servers processing method invocations for this interface are abortively terminated if they exceed the specified time limit in processing the request.

The displayed value is a number greater than or equal to zero. A value of zero for this attribute indicates that the server should not be abortively terminated.

&lt;a name="15208"&gt;&lt;/a&gt;

&lt;a name="ac_ifqsyp_trt4"&gt;&lt;/a&gt;

### Transaction Timeout

Description

The Transaction Timeout field displays the timeout value in seconds for transactions automatically started for this interface queue. Transactions are started automatically when a request not in transaction mode is received and the Interface AUTOTRAN attribute value is set to Y. For more information about the Interface AUTOTRAN attribute, see the Interface Auto Transaction Mode field.

The displayed value is a number greater than or equal to 0 representing a timeout interval in seconds.

&lt;a name="15209"&gt;&lt;/a&gt;

&lt;a name="ac_ifqsyp_rtn4"&gt;&lt;/a&gt;

### Routing Table Name

Description

The Routing Table Name field displays the factory-based routing criteria associated with this interface.

The displayed value is a string from 1 to a maximum of 15 characters.

&lt;a name="15217"&gt;&lt;/a&gt;

<a name="ac_ifqsyp_nsi4"></a>

## Number of Servers with Interface

### Description

The Number of Servers with Interface field displays the number of corresponding servers offering this interface on this queue.

The displayed value is a number greater than or equal to zero.

<a name="nifq_statstab"></a>

# Statistics Tab

The following sections describe the fields provided on the Statistics tab.The Interface Queue statistics are read-only attributes.

<a name="15213"></a>

<a name="ac_ifqstat_rec4"></a>

## Requests Completed

### Description

The Requests Completed field displays the number of interface method invocations completed since the interface was initially offered.

The displayed value is a number greater than or equal to zero.

<a name="15214"></a>

<a name="ac_ifqstat_req4"></a>

## Requests Queued

### Description

The Requests Queued field displays the number of requests currently enqueued for this interface.

The displayed value is a number greater than or equal to zero.

<a name="15215"></a>

<a name="ac_ifqstat_cuo4"></a>

### Current Number of Objects

#### Description

The Current Number of Objects field displays the number of active objects for this interface for associated queues. This number represents the number of entries in the active object table for its queue on the associated machine. This included objects that are not in memory but that were invoked within an active transaction.

The displayed value is a number greater than or equal to zero.

<a name="15216"></a>

<a name="ac_ifqstat_cnt4"></a>

### Current Number of Transactions

#### Description

The Current Number of Transactions field displays the number of active global transactions associated with this interface for its associated queue.

The displayed value is a number greater than or equal to zero.

<a name="T_FACTORY"></a>

<a name="C_FACTORY"></a>

# CORBA Factories (T_FACTORY)

A CORBA Factory is an interface used by a client to obtain an object reference to a CORBA object. Object references to factories are obtained by the client using an object reference to a FactoryFinder interface. The FactoryFinder interface is advertised by the BEA Tuxedo server application and is made available to the client as part of client bootstrap processing.

The T_FACTORY MIB represents the occurrences of factories registered with the BEA Tuxedo FactoryFinder. The available factories for a domain are displayed in the BEA Tuxedo Administration Console. The BEA Tuxedo Administration Console only allows you to view a Factory configuration and run-time information using the configuration tool.

<a name="concorbafactory"></a>

# Viewing a CORBA Factory Configuration

The BEA Tuxedo Administration Console Factory configuration attributes are provided on the General tab page. To view the configuration attributes for a factory, highlight a factory icon. The Configuration Tool pane displays the General tab containing the configuration attributes for your selected factory.

For more information about the Configuration Tool, see Configuring Administrative Objects. For more information about the IDL format for factory interface names, see the CORBA 2.1 Specification Section 7.6 for details.

The following sections describe each factory configuration attribute.

<a name="cfac_generaltab"></a>

## General Tab

The following sections describes the read-only fields provided on the General tab.

<a name="15220"></a>

<a name="ac_facgen_inf6"></a>

### Interface Name

#### Description

The Interface Name field displays the interface repository ID for the factory. The format of this name is dependent on the options specified in the IDL that generates the interface implementation.

<a name="15221"></a>

<a name="ac_facgen_fid6"></a>

## Factory ID

### Description

The Factory ID field displays the registered ID for the factory. The Factory ID is a string value from one to 256 alphanumeric characters. The format of the ID depends upon the options specified in the IDL for the generated interface implementation.

<a name="15222"></a>

<a name="ac_facgen_obs6"></a>

## Object State

### Description

The Object State field displays the current state of the selected Factory. The format of the Object State is a string. The value of this field is ACT (ACTive) indicating that the Object State is active. The ACT object state indicates that the factory is registered with the FactoryFinder.

# Architecture and Implementation

This appendix is a guide to the architecture and implementation of the Oracle Tuxedo Administration Console. It is meant for advanced users and developers who are familiar with the concepts and protocols of the World Wide Web and Java programs.

## Architecture Overview

The Oracle Tuxedo Administration Console applet is an advanced client-server application, written in Java on the client side, and C on the server side. The client (user) machine runs a Java applet, and the BEA Tuxedo server machine runs a gateway program that interfaces with the BEA Tuxedo MIB (Management Information Base) via a custom protocol.

A Web (HTTP) server is not needed for the actual operation of the GUI, but is necessary for delivering the applet code to the Web browser, and for displaying the online help and snapshots. If you do not already have a Web server on your system, you can use `tuxwsvr`, which is delivered with the BEA Tuxedo software.

The Web server does two things:

- It retrieves remote "content" files (HTML, Images, and Java files).

- It runs special programs that deliver "content." Such programs are called CGI (Common Gateway Interface) programs. The Oracle Tuxedo Administration Console is delivered with one CGI program, called `tuxadm` (described in the following paragraphs).

The gateway program run on the server machine (as mentioned previously) is called `wgated`. It is not run directly, but is kicked off by the Oracle Tuxedo Administration Console listener,

`wlisten`. If you run the `ps` command while the GUI is running, you will see one or more `wgated` processes.

Both `tuxwsvr` and `wlisten` listen on distinct network ports waiting for requests to arrive. They know nothing about each other. Therefore, the GUI needs two network ports to operate. These are different from the network ports that are configured in the `UBBCONFIG` file for the Bridges and `tlisten` processes.

In order for a Web browser to run a Java applet, the browser must load an HTML file containing an <APPLET> element. The applet element contains the name of the program, and the parameters to it. It is similar to a command line in an ordinary program.

You could certainly hard-code the applet element for a particular invocation of the GUI, and then retrieve that HTML page with your Web browser. However, the flexibility of the Oracle Tuxedo Administration Console architecture makes it unnecessary for you to do so. An initialization file (or `.ini` file) controls the operation of the GUI. Some of the parameters in the `ini` file control the applet, some control the server gateway, and some are used by both.

The `tuxadm` CGI program mentioned above reads the `ini` file, and merges it with an HTML template file, generating the applet HTML on the fly. The template file, delivered with the BEA Tuxedo software, resides in:

`$TUXDIR/udataobj/webgui/webgui.html`

This is a bare bones template; customers are expected to add their corporate style to it. All that `tuxadm` does is replace the keyword %APPLET% with the actual applet element. You can specify an alternate location for the template by setting the `TEMPLATE` parameter in the server's initialization file (see the following instructions).

If you use the "View Source" feature of your Web browser while the GUI is running, you can see this expanded HTML page that is used to deliver the applet.

Before continuing, look at the following example of HTML code generated by `tuxadm`:

**Listing A-1   Sample HTML Code**

```
<HTML>
<HEAD>
<TITLE>BEA Administrator Console Applet Screen</TITLE>
</HEAD>
<BODY>
<CENTER>
<H1>The Applet is Loading</H1>
```

```
<APPLET ARCHIVE="bea/tuxadm/TuxApplet.jar"
CODE="bea/tuxadm/TuxApplet.class"
WIDTH="400" HEIGHT="150" CODEBASE="/java">
<PARAM NAME="TUXDIR" VALUE="/usr/tuxedo">
<PARAM NAME="PORT" VALUE="4009">
<PARAM NAME="WIDTH" VALUE="400">
<PARAM NAME="HEIGHT" VALUE="150">
<PARAM NAME="ENCRYPTBITS" VALUE="0">
<PARAM NAME="DOCBASE" VALUE="/docs">
<PARAM NAME="SNAPBASE" VALUE="/java/snapshot">
<PARAM NAME="TOTALDOMAINS" VALUE="2">
<PARAM NAME="DOMAINNAME0" VALUE="simpapp">
<PARAM NAME="TUXCONFIG0" VALUE="/home/sdo/simpapp/tuxconfig">
<PARAM NAME="DOMAINNAME1" VALUE="bankapp">
<PARAM NAME="TUXCONFIG1" VALUE="/home/sdo/bankapp/tuxconfig">
<B>Your browser does not appear to support JAVA.
Administration Console must be run on a JAVA capable browser.CCheck your
browser's options to see if JAVA processing has been disabled.</B>
</APPLET>
</CENTER>
</BODY>

</HTML>
```

**Note:** If you are using encryption for communication between the Administration Console applet and server, the HTML code generated by `tuxadm` contains some Javascript code. To see this generated code, use "View Source."

Now you can see why you do not want to create this by hand.

# Running the Web Server

To deliver the above HTML and the applet to the Web browser, a Web server must be running.

To run `tuxwsvr`, use the following command line:

```
tuxwsvr -l //machine:port -i ini-file
```

You may have to replace `/dev/tcp` with the pathname of the correct device name for the network device. This device will be the same one that you used for `tlisten`, and that you specify for the `BRIDGE` parameter in the `UBBCONFIG` file. The string `machine:port` is based on the standard DNS naming convention used by BEA Tuxedo. Web servers usually run at port 80. However, because you must be root in order to use that port, we recommend selecting a different one, for example:

```
//lcuw99:8080
```

**Note:** This is *not* the port number that you saw in the applet element shown previously. That port number was the one given to `WLISTEN`, *not* `TUXWSVR`. We haven't discussed `WLISTEN` yet.

The `ini-file` contains the directory mappings for the Web server.

No administrator wants to open up the entire file system to outsiders. Therefore, several aliases are provided for the directory names used by the Oracle Tuxedo Administration Console:

- One alias is for the main HTML page that launches the GUI. You will normally bring up the GUI by retrieving the page called `webguitop.html`. This page has a button that calls the `tuxadm` CGI program, which returns the above HTML needed to run the applet.

- One alias is for the directory in which CGI programs can be found.

- One alias is for the Java files and images.

- One alias is for the Help documents.

If you do not want the ULOG and Event snapshot files to be written in your Java directory, you may need another alias for a replacement snapshot directory.

When you installed BEA Tuxedo software from CD, you created a Web server `ini` file automatically. However, if you do not care for the default values in that file, you will need to create your own `ini` file.

Add the following to your `ini` file for the Web server:

```
CGI      /cgi-bin    TUXDIR/bin
```

```
HTML      /java TUXDIR/udataobj/webgui/java

HTML      /TUXDIR/udataobj/webgui
```

**Note:** *TUXDIR* should be replaced by the full path to the BEA Tuxedo root directory, e.g., `/usr/tuxedo`. Do not put the word `TUXDIR` in the `ini` file.

What does this mean? Well, when the Web server gets the initial request to run the CGI program, it will look like `/cgi-bin/tuxadm`.

The above file maps this to the BEA Tuxedo `bin` directory, where `tuxadm` resides. In a secure environment, you would not do this; you would copy `tuxadm` somewhere else (or you might need it to be a shell script that sets `LD_LIBRARY_PATH` and executes the real `tuxadm`). It also helps to have `TUXDIR` set in the environment before running `tuxwsvr`.

The APPLET element has an attribute called `CODEBASE` which, as you can see from above, is `/java`. When the Web browser attempts to retrieve the applet from the Web server using the path `/java/bea/tuxadm/TuxApplet.class`, the above `ini` file remaps that to the following:

`$TUXDIR/udataobj/webgui/java/bea/tuxadm/TuxApplet.class`

The mapping of `/` to `udataobj/webgui` is used to map the top-level Web page, `webguitop.html`. (Because the `java` directory is right under the `webgui` directory, the mapping of `/java` in the `tuxwsvr.ini` file is not strictly needed. We have put it there for illustration purposes, in case you want to move your Java files elsewhere.)

If you have installed the BEA Tuxedo online documentation, you will probably also have a mapping for the documentation.

The Web server's `ini` file can contain multiple CGI and HTML lines, and they are processed in sequential order until a match is reached.

This is a good place to describe the difference between the `webgui.ini` parameters that end in `DIR` and `BASE`. `DIR` means a directory on the server, where `wlisten` was run. The server programs use files there.

`BASE` means the URL prefix that a Web browser uses to retrieve a document. The `CODEBASE` for the Java code is `/java`, but the server directory for the Java code is `/usr/tuxedo/udataobj/webgui/java`.

## Configuring the Oracle Tuxedo Administration Console

The next step is to set up the `webgui.ini` file, which is used by `tuxadm` and `wlisten`. `tuxadm` will look in `$TUXDIR/udataobj/webgui/webgui.ini` if it is not given the `INIFILE` parameter (more on this later). `wlisten` requires the `ini` file path on the command line (more later). (If you

have installed BEA Tuxedo using the installation utility, you probably have a usable `webgui.ini` file already.)

**Note:** This is a different `ini` file from the one described for the Web Server. Listing A-2 shows what your `ini` file might look like:

**Listing A-2   Sample webgui.ini File**

```
TUXDIR=/usr/tuxedo
INIFILE=/home/sdo/gui/webgui.ini
NADDR=//lcuw99:4009
DEVICE=/dev/tcp
ENCRYPTBITS=128
DOCBASE=/tuxdocs
CODEBASE=/java
WIDTH=400 (optional)
HEIGHT=150 (optional)
FRAMEWIDTH=800
FRAMEHEIGHT=450
TEMPLATE=/usr/tuxedo/udataobj/webgui/webgui.html (optional)
DOMAIN=simpapp;/home/sdo/simpapp/tuxconfig
DOMAIN=bankapp;/home/sdo/bankapp/tuxconfig
DOMAIN=...
```

- `TUXDIR` is where BEA Tuxedo is installed.

- `NADDR` is the network address that you will use for `wlisten`, in the standard BEA Tuxedo syntax (DNS, 0x, . . . or another provider-specific form).

- `DEVICE` is the device filename for the TCP/IP device (not needed on some systems that use sockets, such as HP and IBM).

- `ENCRYPTBITS` is 0, 40, 56, or 128. This parameter specifies the strength of encryption used in communication between the GUI applet and the Administration Console server. The default is 128-bit.

- `CODEBASE` is the path of the directory in which the Java code can be found (i.e., the aliased path recognized by the Web server, not the physical path on the machine).

- `DOCBASE` is where the BEA Tuxedo help files will be found.

- `INIFILE` is the path of this file (the real path, not the aliased path).

- `TEMPLATE` is where the HTML template file that `tuxadm` uses to deliver the applet.

The `DOMAIN` list is optional, especially if you are using the GUI to create your domains from scratch.

For each domain in the domain list (up to 64), include a DOMAIN line that looks like:

```
DOMAIN=domain_name;tuxconfig-path
```

The value of `tuxconfig-path` is the real path, not the aliased path.

Now you can see how the APPLET HTML code is generated by `tuxadm`.

Once you start using the GUI, it will add some lines to the `ini` file to store such information as your folder selection lists, and the options you set from the options button.

## Running wlisten

Before running `wlisten`, you need to set the FML32 environment variables so that the programs can find the BEA Tuxedo MIB field tables. Set the following variables as shown:

```
FLDTBLDIR32=$TUXDIR/udataobj
FIELDTBLS32=tpadm,evt_mib
```

The command line for `wlisten` is:

```
wlisten -i ini-file
```

`wlisten` can operate in either of two security modes:

- Authentication mode

- Encryption mode

These modes can be used together or authentication can be used by itself.

Authentication is done based on the `tlisten` password file. If there is a file of passwords called `tlisten.pw`, in either `$TUXDIR/udataobj` or `$APPDIR/.adm`, then the user is required to enter a username and password when the applet starts. The password must match one of the passwords in the password file, or authentication fails. The username is passed to `tpinit` when the GUI gateway program, `wgated`, connects to BEA Tuxedo.

If encryption is also being used, then the communications that occur from then on, between the applet and the server, are encrypted using RSA encryption.

## Starting the Oracle Tuxedo Administration Console (for Nonexpert Users)

Now that you have a Web server and a `wlisten` running, it is time to run the Oracle Tuxedo Administration Console.

The installation script places a file called `webguitop.html` in the directory used by the Web Server for its general HTML files. The bottom of this file has a Submit FORM, which generates the URL to run `tuxadm` (the CGI program).

You would normally bring up this Web page by entering the following URL:

```
http://lcuw99:8080/webguitop.html
```

and clicking the button at the bottom.

If you want to edit the default `webguitop.html` file, you should go to the very bottom, where you will see:

```
<FORM ACTION="http://HOST/CGI-BIN/tuxadm">
```

```
<INPUT TYPE=HIDDEN NAME="TUXDIR" VALUE="TUXDIRVALUE">
```

You should edit it to something like:

```
<FORM ACTION="/cgi-bin/tuxadm">
```

```
<INPUT TYPE=HIDDEN NAME="INIFILE" VALUE="/home/sdo/gui/webgui.ini">
```

where `INIFILE` is set to the place where you have put your GUI `ini` file.

## Starting the Oracle Tuxedo Administration Console (for Developers)

If you have previously agreed to the software license on the `webguitop.html` page, you can proceed directly into the GUI by using the special URL:

```
http://machine:port/cgi-bin/tuxadm?TUXDIR=tuxdir-path&INIFILE=inifile-path
```

The port is the port at which you ran the Web server (`tuxwsvr`), not the `wlisten`. You will get strange results if you use the wlisten port by mistake.

The `tuxdir-path` is the absolute path of the BEA Tuxedo root directory on the server machine. The `inifile-path` is the absolute path of the `ini` file for the wlisten (not the Web server). `INIFILE` should match the `INIFILE` parameter within the `ini` file. You can leave out the entire

`&INIFILE= . . .` if the `INIFILE` is in the default location of
`$TUXDIR/udataobj/webgui/webgui.ini`.

The URL that would be used in our example is:

`http://lcuw99:8080/cgi-bin/tuxadm?TUXDIR=/home/units/`
`r63&INIFILE=/home/sdo/gui/webgui.ini`

# Using Appletviewer

This is an advanced topic not needed by most Administration Console users. However, if you are
a Java programmer, you might want to know how to run the Administration Console using the
Appletviewer (the standalone program for running a Java applet).

**Note:**  Oracle has not tested, nor does it support using Appletviewer, but in special
circumstances, (for example, if your Web Server is not working properly), you might
want to run the console using the Appletviewer.

The applet code and images can be installed on a machine where the Java Appletviewer program
will be run.

The HTML containing the APPLET element needs to be modified in two ways: the `CODEBASE`
needs to point to a local directory on the machine, and you need to add a `HOST` parameter to give
the DNS name of the machine on which `wlisten` is running.

In our example from above, your APPLET element would have a new PARAM tag that looks
like:

`<PARAM NAME="HOST" VALUE="lcuw99">`

This `HOST` parameter is undocumented, so do not look for it anywhere.

The Web server is not used in this configuration. However, since you are not using a browser,
you will not be able to use the Help system, nor see the ULOG and Event snapshots. The Oracle
Administration Console has not been tested on and therefore does not support Java version 1.1.