Oracle® Tuxedo

Oracle Tuxedo/Oracle Exalogic Environment Deployment Guide

11*g* Release 1 (11.1.1.2.0)

February 2011

ORACLE®

Oracle Tuxedo/Oracle Exalogic Environment Deployment Guide, 11*g* Release 1 (11.1.1.2.0)

Oracle Tuxedo/Oracle Exalogic Environment Deployment Guide

Oracle Tuxedo/Oracle Exalogic Environment Deployment Guide

This chapter contains the following sections:

Oracle Tuxedo Enterprise Deployment Overview

This section introduces the Oracle Tuxedo enterprise deployment reference topologies and configuration scenario for Oracle Exalogic. It contains the following:

What is Enterprise Deployment?

Enterprise deployment is an Oracle best-practices blueprint based on proven Oracle high-availability, security technologies and recommendations for Oracle Exalogic. The best practices described in these blueprints span across Oracle products various technology stacks: Oracle Database, Oracle Fusion Middleware and Oracle Exalogic machine.

An Oracle Tuxedo System enterprise application deployment:

- considers various business scenarios to make high-availability best practices as widely applicable as possible
- uses results from extensive performance impact studies for different configurations to ensure that the high-availability architecture is optimally configured to perform and scale to business needs
- can still provide persistent service when one of the nodes fails due to various reasons
- uses Oracle best practices and recommended architecture.

Prerequisites

Setup and commissioning of an Oracle Exalogic machine, including initial storage and networking configuration, as described in the *Oracle Exalogic Machine Owner's Guide*.

Terminology

This section provides information about Oracle Tuxedo concepts and terminologies that are related to administering an Oracle Tuxedo application.

- Oracle Home: An Oracle Home directory serves as the central support directory for all Oracle products installed on the target system.
- Compute Node: A physical machine in an Exalogic rack that is meant for running an Oracle Fusion Middleware instance at any given time.

- Shared Storage: Shared storage refers to the Sun ZFS Storage 7320 appliance that is accessible by all compute nodes in the Oracle Exalogic Machine. All compute nodes in the Exalogic machine can access this storage appliance simultaneously for both read and write operations.
- UBBCONFIG: Each Tuxedo domain is controlled by a configuration file in which installation-dependent parameters are defined. The text version of the configuration file is referred to as UBBCONFIG. The UBBCONFIG file for a Tuxedo domain contains all the information necessary to boot the application.
- TUXCONFIG: The TUXCONFIG file is a binary version of the UBBCONFIG file. It is created by running the tmloadcf(1) command, which parses UBBCONFIG and loads the binary TUXCONFIG file to the location referenced by the TUXCONFIG environment variable.
- XA: The eXtended Architecture (XA) is the standard-defined interface between the Transaction Manager and the Resource Manager. XA allows program control of Resource Managers that are involved in distributed transactions.
- TMS: The Oracle Tuxedo transaction management server, can be named TMS or other names, is responsible for coordinating global transactions.
- TLOG: Tuxedo transaction log.

Benefits of Oracle Recommendations

The Oracle Tuxedo configurations discussed in this guide are designed to maximize hardware resources, and provide a reliable, standards-compliant system for enterprise computing with a variety of applications.

- High Availability
- Performance
- Application Isolation

High Availability

The enterprise deployment architectures are highly available, because each Tuxedo application server or Tuxedo system server can be replicated on a different computer. They can provide persistent services even if one of the paired computers shuts down due to any problem.

For more information, see Achieving High Availability with Oracle Tuxedo at http://www.oracle.com/technetwork/middleware/tuxedo/overview/index.html.

Performance

Oracle Exalogic uses InfiniBand as the I/O fabric technology. InifiniBand provides a high throughput, low latency, and scalable fabric that is suitable for fabric consolidation of inter-processor communication, network and Storage. It is optimized for cluster and storage traffic.

Regardless of the design of the application, Oracle Exalogic offers a multitude of capabilities that dramatically improves overall performance and reliability of the application. To benefit from the features and capabilities of Oracle Exalogic, Oracle Tuxedo users only need to deploy applications to the Exalogic machine; no code changes or application re-architecture is necessary.

Application Isolation

Oracle Exalogic provides a high degree of isolation among concurrently deployed applications that have diverse security, reliability, and performance requirements. It creates a default IP over InfiniBand (IPoIB) link and an Ethernet over InfiniBand (EoIB) interface during initial configuration. All compute nodes in an Exalogic Machine are members of the default InfiniBand partition.

The most common model for application isolation involves multiple IP subnetting, in which the most mission-critical applications are assigned their own IP subnets layered above the default IPoIB link. In this model, some subnets may also contain applications that have less stringent or otherwise different resource requirements.

Overview of Oracle Exalogic Configured Environment

Before you start implementing the Oracle enterprise deployment topology, you should understand the current state of the Exalogic environment.

It is assumed that you have completed all tasks described in the *Oracle Exalogic Machine Owner's Guide*, which discusses your data center site preparation, Oracle Exalogic machine commissioning, initial networking configuration including IP address assignments, and initial setup of the Sun ZFS Storage 7320 appliance.

This section contains the following topics:

- Network
- Sun ZFS Storage 7320 Appliance
- Oracle Software

Network

Before you start configuring the enterprise deployment topology, you must run the Oracle `OneCommand` tool to complete the following tasks (as described in "Initial Configuration of an Exalogic Machine Using Oracle `OneCommand`" in the *Oracle Exalogic Machine Owner's Guide)*:

- Configuration of IP addresses for all Exalogic compute nodes and the Sun ZFS Storage 7320 appliance.
- Configuration of InfiniBand gateway switches.
- Configuration of the Cisco Ethernet management switch.
- Setup and verification of the default IP over InfiniBand (IPoIB) link spanning all compute nodes.
- Setup and verification of the default Ethernet over InfiniBand (EoIB) link for connectivity with components of the topology running on Ethernet.
- Configuration of the default InfiniBand partition that covers all of the compute nodes in Exalogic Machine.

Sun ZFS Storage 7320 Appliance

The initial configuration of the Sun ZFS Storage 7320 appliance on your Oracle Exalogic machine is completed at the time of manufacturing. For more information about default shares (Exported File Systems), see the “Default Storage Configuration" section in the *Oracle Exalogic Machine Owner's Guide*.

After completing this initial configuration, you can proceed to create custom shares as needed.

Oracle Software

Oracle Linux 5.5 is pre-installed on each of the compute nodes in your Oracle Exalogic machine.

Enterprise Deployment Reference Topology

The instructions and diagrams in this guide describe a reference topology, to which variations may be applied.

This guide provides configuration instructions for a reference enterprise topology in the following scenarios:

- Scenario 1: Oracle Exalogic machine connected to Oracle database or RAC over 10 GB Ethernet; the requests from workstation which acts as a remote client reach the Tuxedo application servers running on the Exalogic machine via the 10 GB Ethernet over InfiniBand network of Exalogic Machine (EoIB) as shown in Figure 1.
- Scenario 2: Oracle Exalogic machine connected to Oracle Exadata Database Machine; the requests from workstation which acts as a remote client reach the Tuxedo application servers running on the Exalogic machine via the InfiniBand fabric of Exalogic Machine as shown in Figure 2.

Figure 1 Oracle Tuxedo Enterprise Deployment Reference Topology with Oracle Database Over Ethernet

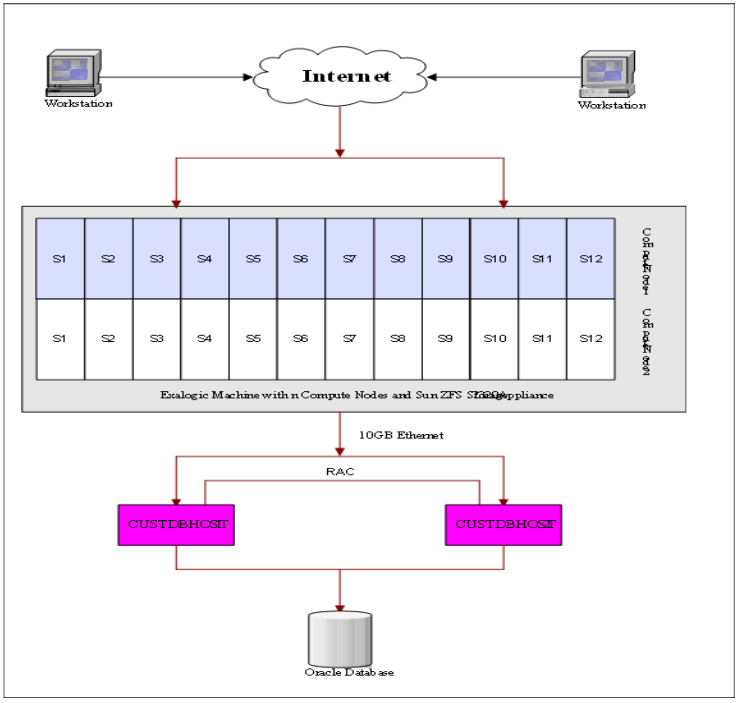
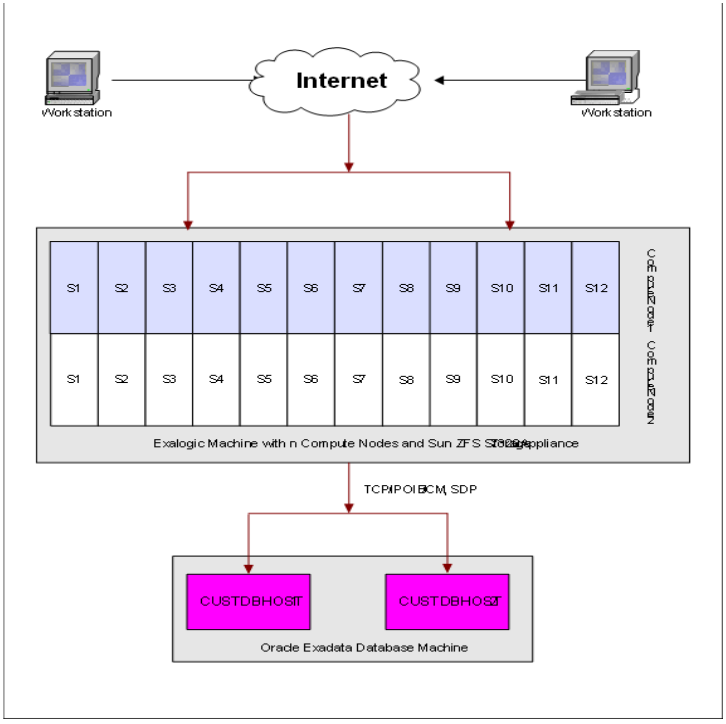


Figure 2 Oracle Tuxedo Enterprise Deployment Reference Topology with Oracle Exadata Database Machine

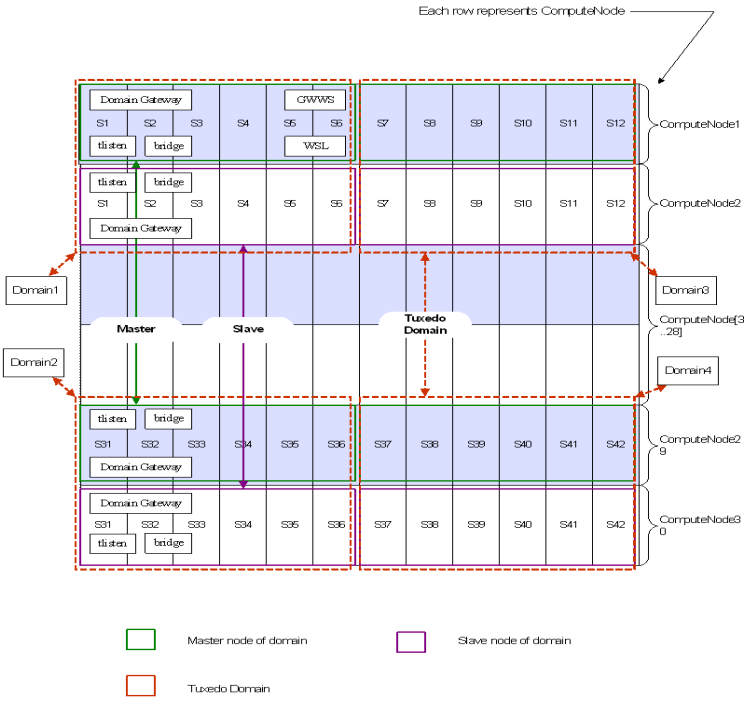


Note: The number of compute nodes in Oracle Exalogic machine depends on your purchased hardware configuration, such as Oracle Exalogic machine full rack (30 compute nodes), Oracle Exalogic machine half rack (16 compute nodes), and Oracle Exalogic Machine quarter rack (8 compute nodes).

Horizontal Slicing Within an Exalogic Machine

Figure 3 illustrates horizontal slicing of the Oracle Tuxedo enterprise deployment reference topology within an Oracle Exalogic machine full rack.

Figure 3 Horizontal Slicing Within Exalogic Machine Full Rack



Based on the enterprise deployment reference topology, you can create your own topology.

Configuration Scenario Used in This Guide

The configuration examples described in this guide are based on a simple scenario, including the following:

- Two Tuxedo domains: `Domain1` and `Domain2`. Each domain is in MP mode which spanning across two compute nodes.
- `ComputeNode1` is the master node for `Domain1`; `ComputeNode2` is the slave node. Accordingly, `ComputeNode29` is the master node for `Domain2`; ComputeNode30 is the slave node.
- For each domain, there are six application servers running on the master node and another six servers on slave nodes as the backup. All the servers provide the same service.

- All the application servers are associated with a database instance.
- Every node of the domain is configured with GWTDOMAIN gateway. So the gateway from one domain will establish the connection with all the gateways in another domain.
- GWWS and WSL are configured on Domain1's master node, which are used to accept the remote request or access from outside of the Exalogic machine.
- One of the call paths from a remote /WS client is: Workstation client request -> WSL on `Domain1` -> `Domain1` server -> `Domain2` server.
- The network traffic from remote workstation clients comes in via the 10 Gb Ethernet over InfiniBand network of Exalogic Machine (EoIB). The internal network traffic is through IP over InfiniBand (IPoIB). So the WSL and GWWS are configured listening on `BOND1` interface for EoIB; and all the other Tuxedo components which are used for internal communication, like `tlisten`, bridge, GWTDOMAIN, are configured on `BOND0` interface for IPoIB.

Installing the Software

This section describes the software installation required for the enterprise deployment reference topology.

This section contains the following topics:

Downloading the Oracle Tuxedo Installer

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

You must download the Oracle Tuxedo Linux x86-64 (64-bit) installer, as follows:

- Download the BIN file from the Oracle OTN Web site at: http://www.oracle.com/technology/software/index.html
- Copy the Tuxedo`111120_64_Linux_01_x86.bin` to a local directory on `ComputeNode1`.

Installing Oracle Tuxedo System on Sun Storage 7000 Unified Shared Storage System

You can install the Oracle Tuxedo product binaries in one of the shares on Sun ZFS Storage 7320 appliance locations. Note that the share, which is a shared file system, must be accessible by all compute nodes. And considering the access permission problem, we recommend that when you create a user account, ensure it has the same uid and gid on all the Exalogic compute nodes (to avoid permission access issues). For example, create NIS accounts for users.

You must run the Oracle Tuxedo installer on `ComputeNode1`, as follows:

1. Before installing Oracle Tuxedo System, make sure the compute nodes meet the general installation requirements for the hardware and software. For Tuxedo 11g Release 1, please refer to the Oracle Tuxedo 11g Release 1 (11.1.1.2.0) Platform Data Sheets.

2. Log in to `ComputeNode1` as the Oracle Tuxedo administrator.

   The Oracle Installation program uses a temporary directory in which it extracts the files from the archive that are needed to install Oracle Tuxedo on the target system. By default, the temporary directory is `/tmp`. Enter the following command at the shell prompt:

   ```
   export IATEMPDIR=tmpdirname
   ```

   to replace the default temporary directory `/tmp`.

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

   ```
   prompt>sh ./tuxedo111120_64_Linux_01_x86.bin –i console
   ```

   The **Choose Locale** screen is displayed.

3. In the **Choose Locale** screen, enter 1, which is associated with **English**.

   The **Introduction** screen is displayed.

4. In the **Introduction** screen, press `<ENTER>` to continue.

   The **Choose Install Set** screen is displayed.

5. In the **Choose Install Set** screen, enter 1, which is associated with **Full Install**.

   The **Choose Oracle Home** screen is displayed.

6. In the **Choose Oracle Home** screen, enter 1, which is associated with **Create new Oracle Home**.

The **Specify a new Oracle Home directory** screen is prompted.

7. Enter your Oracle Home directory. Be sure to use the full pathname when specifying the new Oracle Home directory.

   The Oracle Home should be on the shared file system on the Sun Storage 7000 Unified Storage System, and can be accessible by all compute nodes in the Oracle Exalogic machine.

   The **Choose Product Directory** screen is displayed.

8. In the **Choose Product Directory** screen, enter 2, which is associated with **Use Current Selection**.

   The **Install Samples (Y/N)** is prompted.

9. Enter **Y** to install the samples.

   The **Pre- Installation Summary** screen is displayed.

10. In the **Pre- Installation Summary** screen, press `<ENTER>` to continue.

    The **Ready To Install** screen is displayed.

11. In the **Ready To Install** screen, press `<ENTER>` to install.

    The **Installing** screen is displayed.

12. In the Installing screen, no user input is required.

    When it finishes, the **Configure tlisten Service** screen is displayed.

13. In the **Configure tlisten Service** screen, enter a `tlisten` password of your choice. Your password must be a string of alphanumeric characters in clear-text format that is no more than 80 characters in length. Then **Verify** your password.

    Note: A common password is required for all the nodes in an Oracle Tuxedo application (domain) to communicate successfully. For this reason, you must use the same password when you install Oracle Tuxedo on more than one compute nodes for a single application.

    The **SSL Installation Choice** screen is displayed.

14. In the **SSL Installation Choice** screen, you can choose to enter 1, which is associated with **YES (This is not mandatory for the installation)**.

    The **Enter Your LDAP Settings for SSL Support** screen is displayed.

15. In the **Enter Your LDAP Settings for SSL Support** screen, input your **LDAP Service Name**, **LDAP PortID**, **LDAP BaseObject and LDAP Filter File Location (This is not mandatory for the installation)**.

    The **Installation Complete** screen is displayed.

16. In the **Installation Complete** screen, press `<ENTER>` to exit the installer.

File and Disk Space Allocation

In an Oracle Tuxedo application, all system files might be stored together on the same raw disk slice or OS file system. While it is possible to use regular OS filesystem files for the configuration files, we strongly recommend that you store the transaction log, TLOG, on a raw disk device. Because the TLOG seldom needs to be larger than 100 blocks (51200 bytes assuming 512-byte blocks), and because disk partitions are always substantially larger than 100 blocks, it may make sense to use the same device for both the configuration files and the TLOG.

Space outside the OS file system is usually referred to as *raw disk space*. Not only is I/O faster when done by system calls reading directly from and writing directly to device special files on raw disks, but a physical `write()` occurs right away. When using an OS file system, Oracle Tuxedo cannot predict or control the precise moment at which a `write()` is done. When using raw disk space, however, Oracle Tuxedo has accurate control of the write operation, which is particularly important for entries in the Oracle Tuxedo transaction log. Also, when multiple users are accessing the system, being able to control the write operation is important for assuring database consistency.

If you decide to use raw disk space for your Oracle Tuxedo application, you may find that the hard disk devices on your system are fully allocated to file systems such as `/(root)` and `/usr`. If that is the case, you must repartition your hard disk device in order to set aside some partitions for use as non-OS file systems. For repartitioning instructions, refer to the system administration documentation for your platform.

If you decide to use OS filesystem, we recommend that you store the configuration files and the TLOG on compute node's solid-state disks (SSDs) which can shorten the XA transaction latency.

Configuring Tuxedo Applications

After installing the Oracle Tuxedo System, you must configure it for the Oracle Exalogic enterprise deployment topology.

This section shows how to configure the Oracle Tuxedo System for two domains, as illustrated in Figure 4. There are two domains (`domain1` and `domain2`), each of them are configured across

two compute nodes in MP mode. They provide different services and communicate with each other. The relationship between the two domains is illustrated in Figure 5. Besides this, domain1 has /WS and SALT configured, that means the remote request from outside of the Exalogic machine can access the services provided by `domain1`.

You can follow this example configuration to create other Oracle Tuxedo domains on the remaining compute nodes on the Exalogic Machine, based on your application deployment and isolation requirements.

Figure 4 Oracle Tuxedo Configuration

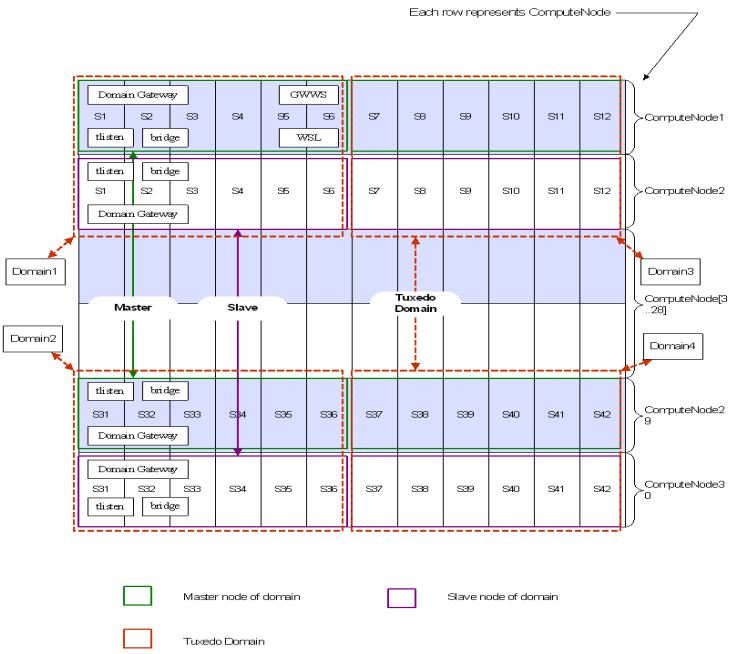
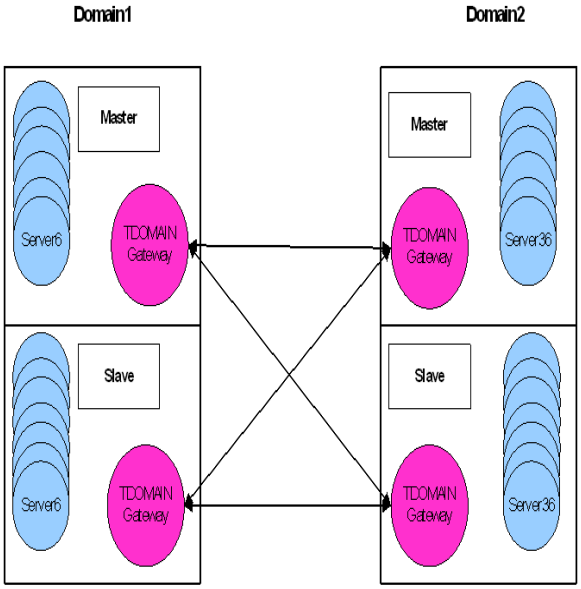


Figure 5 Oracle Fusion Middleware Configuration



In this sample configuration, two Oracle Tuxedo domains are created including the following:

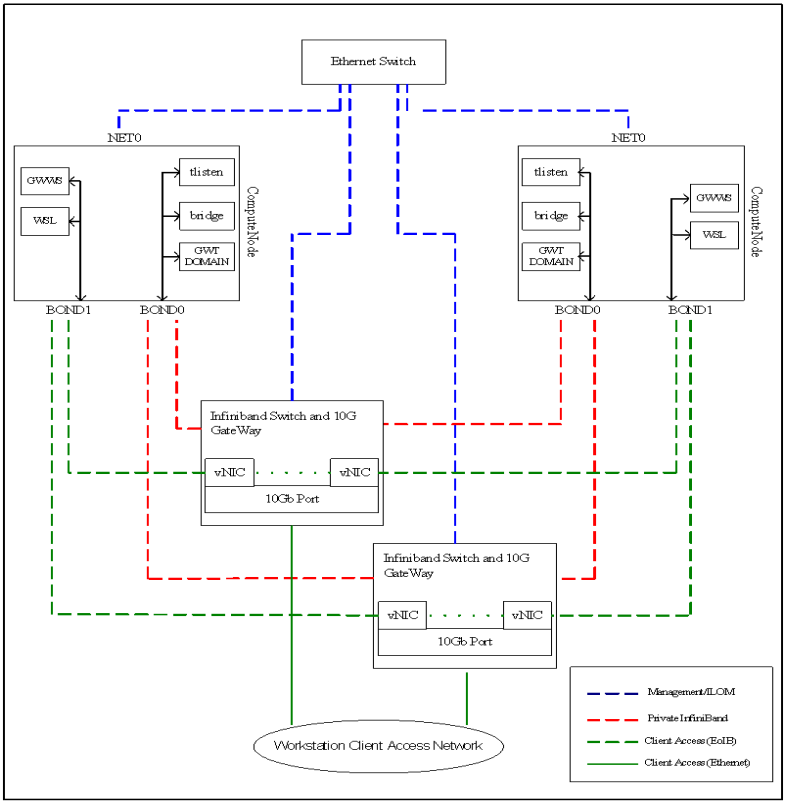
- Every domain is in MP mode which spans across two compute nodes. Domain1 is configured across `ComputeNode1` and `ComputeNode2`. Domain2 is configured across `ComputeNode29` and `ComputeNode30`.
- `ComputeNode1` is the master node for Domain1 , and `ComputeNode2` is the slave node. Accordingly `ComputeNode29` is the Domain2 master node, and `ComputeNode30` is the slave node.
- For each domain, there are six application servers running on the master node which provide the same service (and as the master backup), there are also another six application servers running on the slave node.
- All the application servers are associated with a database instance.
- The service provided by Domain1 is `TOUPPER1` and Domain2 provides the `TOUPPER2`. When a client sends a request to `TOUPPER1`, within it, `TOUPPER2` is invoked accordingly.
- Every node of the domain is configured with `GWTDOMAIN` gateway. So the gateway from one domain establishes connections with all the gateways in another domain. The

advantage is that when the application running on one compute node has problem, the client can still access the service through another gateway. For example, when a client sends a request to `TOUPPER1`, in `TOUPPER1`, it invokes `TOUPPER2` in Domain2. But now the `GWTDOMAIN` and application servers on `ComputeNode29` are not responding for some reason, then the request from `TOUPPER1` to `TOUPPER2` is picked up by the GWTDOMAIN and `ComputeNode30` application servers. The client can still get the correct response. This ensures the high availability of the services.

- GWWS and WSL are configured on Domain1's master node, which are used to accept the remote request or access from outside of the Exalogic machine.
- One of the call paths from a remote /WS client is: Workstation client's request -> WSL on Domain1 -> Domain1's server -> Domain2's server.

In this enterprise deployment topology, the request from remote workstation clients comes in via the 10 GB Ethernet over InfiniBand network of Exalogic Machine (EoIB). The internal network traffic is through IP over InfiniBand (IPoIB). So the WSL and GWWS are configured listening on BOND1 interface for EoIB; and all the other Tuxedo components which are used for internal communication, like `tlisten`, bridge, GWTDOMAIN, are configured on BOND0 interface for IPoIB as shown in Figure 6.

Figure 6 Exalogic Machine Network Overview



The schematic representation of Oracle Exalogic machine's network connectivity includes the following:

- Default `BOND0` interface, which is the private InfiniBand fabric including the compute nodes connected via Sun Network QDR InfiniBand Gateway Switches
- Default `BOND1` interface, which is the Ethernet over InfiniBand (EoIB) link
- `NET0` interface, which is associated with the host Ethernet port 0 IP address for every compute node and storage server head
- tlisten, bridge and GWTDOMAIN are listening on the IP address which is bound to BOND0

- WSL and GWWS are listening on the IP address which is bound to BOND1

Important Notes Before You Begin

Read the following notes before you start configuring Oracle Tuxedo components:

- The configuration example used in this section describes how to configure the environment for `Domain1` and `Domain2` that cross two compute nodes in MP mode.
- Every domain has a Master node and a Slave node. You can extrapolate the information included in these procedures to set up and configure the environment for your remaining Slave node, as necessary. This configuration depends on your specific application deployment and isolation requirements as well as the Exalogic Machine configuration.
- Exalogic Machine Full Rack includes 30 compute nodes, Exalogic Machine Half Rack includes 16 compute nodes, and Exalogic Machine Quarter Rack includes 8 compute nodes. You should plan your application deployment and infrastructure accordingly.

You must complete the following procedures to configure Oracle Tuxedo Middleware:

- Prerequisites
- Setting Up Oracle Tuxedo System Build and Runtime Environments
- Editing the UBBCONFIG File
- Editing the UBBCONFIG File
- Creating the Universal Device List and the Transaction Log
- Starting the tlisten Process
- Running buildtms for Oracle Tuxedo Applications That Use XA Resource Managers
- Running buildserver for Oracle Applications
- Booting Oracle Tuxedo Applications
- Configuring SDP Protocol Support for Infiniband Network Communication to the Database Server

Prerequisites

The following are the prerequisites for configuring Oracle Tuxedo 11*g* Release 1 PS 1 products for Oracle Exalogic:

- Preconfigure the environment, including database, storage, and network.
- Install Oracle Tuxedo System 11g Release 1 PS 1 as described in "Installing the Software".

Setting Up Oracle Tuxedo System Build and Runtime Environments

You need to set several environment variables before using Oracle Tuxedo to build and run Oracle Tuxedo applications. Table 1 lists those environment variables.

Table 1 Oracle Tuxedo Core Environment Variables

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

Besides the Tuxedo Core Environment Variables listed above, you also need to set and export the following environment Variables as shown in Table 2

Table 2 Environment Variables

PATH=$APPDIR:$TUXDIR/bin:/bin:$PATH
LD_LIBRARY_PATH=$APPDIR:$TUXDIR/lib:/lib:/usr/lib:$LD_LIBRARY_PATH
BDMCONFIG=$APPDIR/bdmconfig

You must do the following steps:

1. On ComputeNode1 and ComputeNode2, open the terminal window. At the command prompt, export all the environment variables (or you can set these environments variables in a file and source it like tux.env which is generated by Tuxedo installation):

```
export TUXDIR=<OracleHome>/ tuxedo11gR1

export APPDIR=<Application Directory>

export TUXCONFIG= $APPDIR/tuxconfig

export PATH=$APPDIR:$TUXDIR/bin:/bin:$PATH

export
LD_LIBRARY_PATH=$APPDIR:$TUXDIR/lib:/lib:/usr/lib:$LD_LIBRARY_PATH

export LANG=C

export BDMCONFIG=$APPDIR/bdmconfig

…
```

   Because the application servers on ComputeNode1/ComputeNode2 need to access the Oracle Database, so you also need to export the environment variables for it.

```
export ORACLE_HOME=<Where you install the Oracle Database>

export PATH= $ORACLE_HOME/bin:$PATH

export LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH

export TUXWA4ORACLE=1
```

The TUXDIR, APPDIR, and TUXCONFIG environment variables must match the values of the TUXDIR, APPDIR, and TUXCONFIG parameters in the MACHINES section of the UBBCONFIG file.

2. Repeat Step 1 above on ComputeNode29 and ComputeNode30 for Domain2.

Editing the UBBCONFIG File

Please refer to Section 5 File Formats, Data Description, MIBs, and System Processes Reference UBBCONFIG in Oracle Tuxedo 11g Release 1 (11.1.1.2.0) documentation.

Notes: Remember to set the tlisten and bridge address to the IP addresses which are bound to BOND0 interface.

Remember to set the WSL and GWWS address of to the IP addresses which are bound to BOND1 interface.

Editing the DMCONFIG File

Please refer to Section 5 File Formats, Data Description, MIBs, and System Processes Reference DMCONFIG in Oracle Tuxedo 11g Release 1 (11.1.1.2.0) documentation.

Note: Remember to set the address of GWTDOMAIN to the IP addresses which are bound to BOND0 interface.

Creating the Universal Device List and the Transaction Log

You must create the Universal Device List (UDL) and define a UDL entry for the global transaction log (TLOG) on each compute node in your application that uses global transactions. The TLOG is a log file in which information about transactions is kept until the transaction is completed. You must do the following:

1. Define the TLOG

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

2. Create the UDL and UDL Entries for TLOG

   You must manually create a UDL entry for the `TLOGDEVICE` on each machine where a TLOG is needed. You may create these entries either before or after you have loaded `TUXCONFIG`, but you must create these entries before booting the application. In this configuration example, you only need to create a UDL entry on ComputeNode1.

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

   To create the UDL and a UDL entry for TLOG on ComputeNode1/ComputeNode2 in your application that will use global transactions, follow these steps:

   In ComputeNode1/ComputeNode2's terminal window, enter the following command:

   ```
   tmadmin -c
   ```

   ```
   crdl -z config -b blocks
   ```

   Here `-z config` specifies the full pathname of the device on which the UDL should be created (that is, where the TLOG will reside), and `-b blocks` specifies the number of blocks to be allocated on the device. The value of `config` should match the value of the `TLOGDEVICE` parameter in the `MACHINES` section of the `UBBCONFIG` file. The blocks must be larger than the value of `TLOGSIZE`.

For example:

```
tmadmin -c
crdl -z <APPDIR>/TLOG -b 200
```

3. Do the same steps as above for Domain2.

Starting the tlisten Process

You must start a `tlisten` process on each compute node of a networked Oracle Tuxedo application before the application is booted. The `tlisten` process enables you and the Oracle Tuxedo software running on the `MASTER` node to start, shut down, and administer Oracle Tuxedo processes running on the non-`MASTER` nodes.

Manually starting a `tlisten` process from a command-line shell:

- `ComputeNode1`:

  ```
  $TUXDIR/bin/tlisten -l //IP address bound to BOND0:32001
  ```

- ComputeNode2:

  ```
  $TUXDIR/bin/tlisten -l //IP address bound to BOND0:32007
  ```

The `-l` option must match the value of the `NLSADDR` parameter in the `*NETWORK` section of the `UBBCONFIG` file.

Do the same for `Domain2`.

Running buildtms for Oracle Tuxedo Applications That Use XA Resource Managers

For Oracle Tuxedo applications that use distributed transactions and XA-compliant resource managers, you must use the `buildtms` command to construct a transaction manager server load module.

On `ComputeNode1/ComputeNode2`, under the `$APPDIR` directory, type the following command:

```
buildtms -r Oracle_XA -o ORATMS
```

Oracle_XA is the published name of the Oracle XA interface. `ORATMS` is the Transaction Manager name defined in your UBBCONFIG file.

Do the similar steps for `Domain2`.

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

Running buildserver for Oracle Applications

You need to use `buildserver` to build the Oracle Tuxedo applications servers.

Assume your application source code for `simpserv1` is named `simpserv1.pc`, the following is an example about how to build your application servers.

`On ComputeNode1/ComputeNode2, type the command:`

a. `$(ORACLE_HOME)/bin/proc MODE=ORACLE def_sqlcode=yes code=ansi_c include=$(ORACLE_HOME)/precomp/public iname=simpserv1.pc`

b. `$(CC) -c -I$(ORACLE_HOME)/precomp/public  simpserv1.c`

c. `buildserver -o simpserv1-f simpserv1.o -r Oracle_XA -s TOUPPER1`

Do the same steps for Domain2 (note the service name is `TOUPPER2)`.

You can adjust your compilation parameters according to your application servers.

Booting Oracle Tuxedo Applications

You need to use the command `tmboot` to boot all the Tuxedo Applications.

On the master node `ComputeNode1/ComputeNode29`, type the following command:

`tmboot -y`

Then all the administration and application servers are booted.

Configuring SDP Protocol Support for Infiniband Network Communication to the Database Server

Oracle Tuxedo specializes in managing transactions, on behalf of ATMI and CORBA applications, from their point of origin—typically on the client—across one or more server machines, and then back to the originating client. When a transaction ends, Tuxedo ensures that all the systems involved in the transaction are left in a consistent state.

The Oracle Tuxedo system uses the X/Open XA interface for communicating with the various resource managers. The XA Standard is widely supported in all the major database vendor products.

You can also use SDP (Sockets Direct Protocol) for Oracle Database invocations.

Please configure the database to support InfiniBand, as described in the "Configuring SDP Protocol Support for Infiniband Network Communication to the Database Server" topic in the *Oracle Database Net Services Administrator's Guide*.

Managing Oracle Tuxedo Applications

Prerequisites

The following are prerequisites for managing and monitoring your enterprise deployment:

- Ensure that Tuxedo software is installed in all of exalogic machine nodes where your applications are deployed.
- Ensure that application specific directories and files are propagated into all of Exalogic machine nodes for your networked applications. They include the `APPDIR` environment variable, executables, FML/FML32 field tables, VIEW/VIEW32 tables.
- If the applications involve distributed transaction processing, you must have created a global transaction log (TLOG) on each participating exalogic machine node.
- If you want to use TSAM to monitor your applications you must also ensure that the TSAM software is installed on all of exalogic machine nodes. TSAM is installed on top of Tuxedo and must be compatible with current Tuxedo release. But the TSAM manager is not manadatory to install in these of exalogic machine nodes. TSAM Manager must run in an existsing Java application server.
- Ensure that the tlisten process are running on all of exalogic machine nodes for a networked application that runs on more than one machine, as established by the MODEL MP parameter in the RESOURCE section of the application's UBBCONFIG file.

Monitoring

You can monitor your application in two ways: Tuxedo command line monitoring tools and TSAM monitor management console.

Monitoring Application with Tuxedo Methods

The detail description can refer to the Tuxedo document of *Administering an Oracle Tuxedo Application at Run Time*.

Although Tuxedo already provides rich means to monitor and administrate applications, but from centralizing monitoring perspective, Oracle TSAM should be better as the selection.

Monitoring Application with Oracle TSAM

Oracle TSAM provides comprehensive monitoring and reporting for Oracle Tuxedo system and applications.

The Oracle TSAM agent enables collection of various application performance metrics (including call path, transactions, service, system servers). The Oracle TSAM Manager provides graphical user interface that correlates and aggregrate performance metrics collected from one or more Tuxedo domains. Below, several typical monitoring metrics are listed to demonstrate the monitoring function of Oracle TSAM, more Oracle TSAM features and functions can be found in the Oracle TSAM documentation.

Accessing Oracle TSAM Manager

TSAM manager can be installed inside or outside of Exalogic machine nodes. Before accessing TSAM manager you must do the following prerequisites:

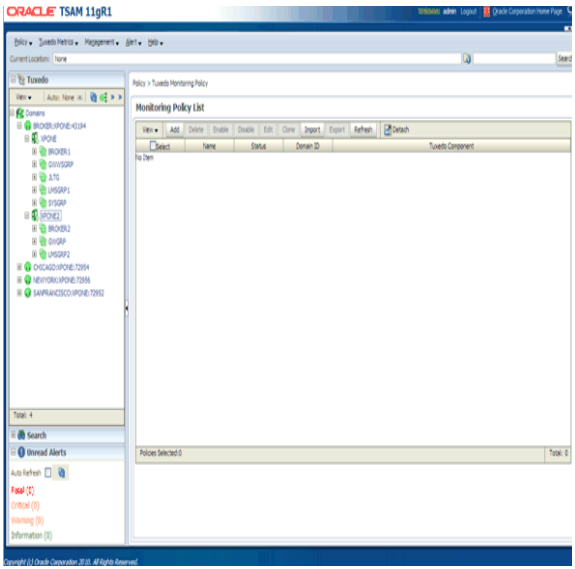
- Make sure the LMS process in each of monitor application nodes is running. LMS is a server shipped within Oracle TSAM and it must already be deployed each monitored Exalogic node.
- TSAM Manager is already deployed in a Java application server and the required database must be already configured.

For more information, see the *Oracle Tuxedo System and Application Monitor (TSAM) Deployment Guide*.

You can access Oracle TSAM by navigating to `http://<hostname>:<port>/tsam`.

The TSAM home page show in Figure 7 gives you an at-a-glance view of the overall status of your monitored environment.

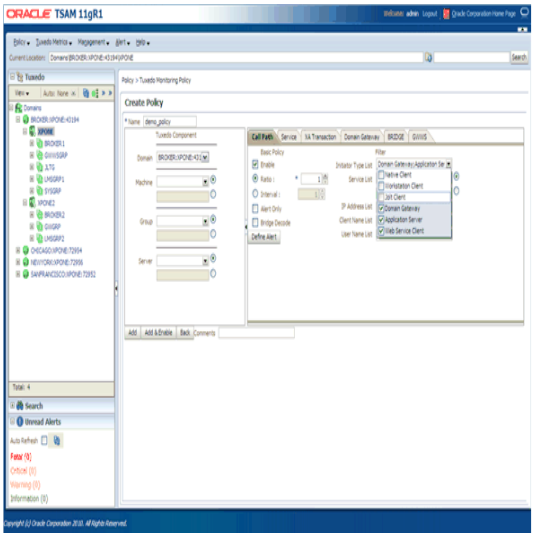
Figure 7 Oracle TSAM Manager Home Page



Monitoring Policy Definition

Oracle TSAM provides a comprehensive policy monitoring mechanism. You can define specific monitoring policy for any monitoring point in each Tuxedo domain that deployed in Exalogic machine. Figure 8 shows a sample page demonstrating how to define a policy.

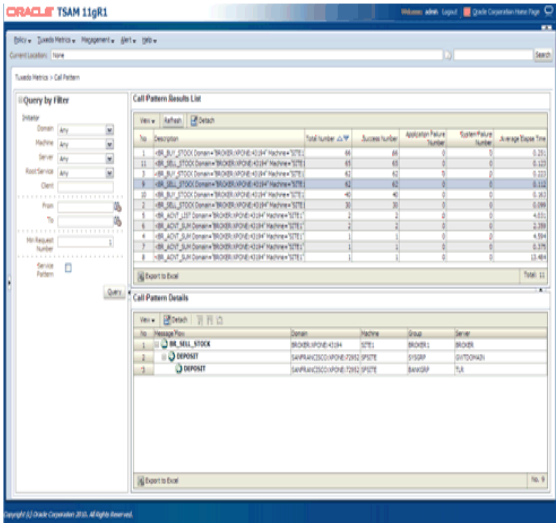
Figure 8 Oracle TSAM Policy Definition



Service Call Path

Oracle TSAM provides a capability named **Call Path Monitoring** used for end user or administrator to penetrate the service propagation path and to see what happens "behind the scene". Figure 9 shows a sample page demonstrating the **Call Path Monitoring**.

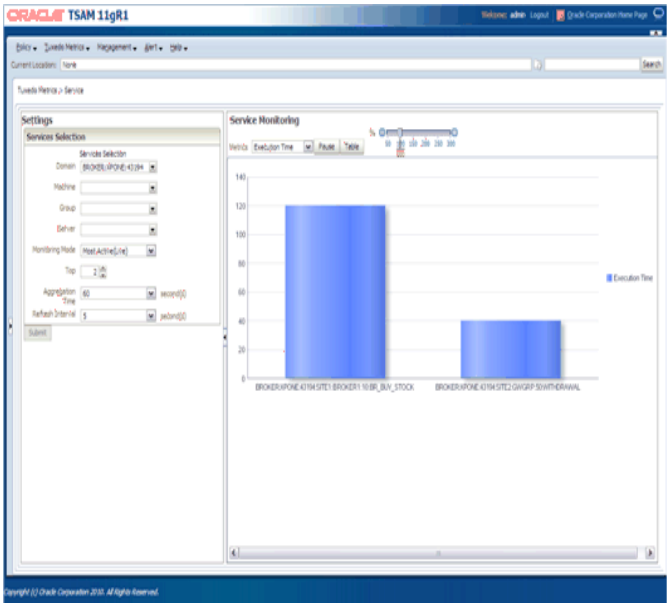
Figure 9 Call Path Monitoring Page



Service Monitoring

Oracle TSAM provides rich graphical presentation to show service activity statistic information. Figure 10 shows a sample page demonstrating Service Monitoring.

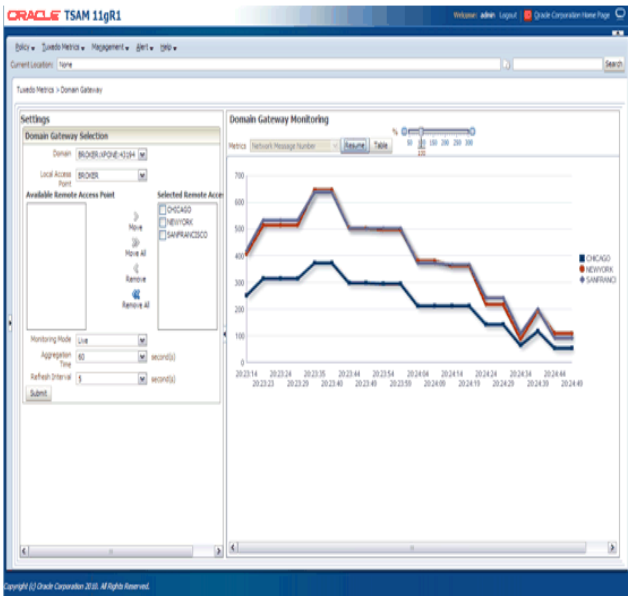
Figure 10 Service Monitoring Page



Server Monitoring

Oracle TSAM also provides the capability to monitor current system servers deployed in an Exalogic machine node. Figure 11 shows a sample page demonstrating the /TDomain gateway server monitoring.

Figure 11 GWTDOAM Server Monitoring



Scaling Out the Topology

As an administrator, you must ensure that once an application is up and running, it continues to meet the performance, availability, and security requirements set by your company. The Oracle Tuxedo system allows you to make changes to your configuration without shutting it down. To help you dynamically modify your application, the Oracle Tuxedo system provides the following three methods: the Oracle Administration Console, command-line utilities (`tmadmin`, `tmconfig`), and the Management Information Base (MIB) API. By these three methods, you can add, change and remove an application including adding a new Exalogic machine node, new server group, new server and activating this server.

Exalogic/Oracle Tuxedo Application Examples

The following two examples show how to add a new Exalogic machine node to a running Oracle Tuxedo application.

Example 1

To add a new node in MP configuration using MIB, create a file named addnode.dat as shown in Listing 1.

Listing 1 addnode.dat File

```
$cat addnode.dat
TA_OPERATION  SET
TA_CLASS      T_MACHINE
TA_LMID       simple3
TA_PMID       node's physical name
TA_TUXCONFIG  absolute pathname of the TUXCONFIG file
TA_TUXDIR     absolute pathname of the directory where the Tuxedo is
installed
TA_APPDIR     absolute pathname of application directory
TA_STATE      NEW
<cr>
```

Then type:

```
$ ud32 < addnode.dat
```

The `LMID simple3` node is added. You can also do the same to add GROUPS, SERVERS to this new node using T_GROUP or T_SERVER classes.

Example 2

To add a new MP configuration node using `tmconfig`, do the following steps:

```
1. $ tmconfig

2. Section:
   1) RESOURCES,
   2) MACHINES,
   3) GROUPS
   4) SERVERS
   5)SERVICES
   6) NETWORK
   7) ROUTING
   8) QUIT
   9) WSL
   10) NETGROUPS
   11) NETMAPS
   12) INTERFACES [1]: 2

3. Operation:
   1) FIRST
```

```
   2) NEXT
   3) RETRIEVE
   4) ADD
   5) UPDATE
   6) CLEAR BUFFER
   7) QUIT [1]:4

4. Enter editor to add/modify fields [n]?y
   TA_OPERATION  SET
   TA_LMID simple3
   TA_PMID node physical name
   TA_TUXCONFIG absolute pathname of the TUXCONFIG file
   TA_TUXDIR absolute pathname of the directory where the Tuxedo is
   installed
   TA_APPDIR absolute pathname of application directory
   TA_STATE NEW

   <cr>
```

Meanwhile, you can use the similar ways to dynamically remove an Oracle Tuxedo node, server group, server via MIB or `tmconfig` command.

For more information, see *Dynamic Modifying an Application* in *Administering an Oracle Tuxedo Application at Run Time*.

Oracle Tuxedo Application Runtime for CICS and Batch Deployment

This section provides explanations and instructions for installing, configuring and managing Oracle Tuxedo Application Runtime for CICS and Batch (ART) on an Oracle Exalogic machine.

This section contains the following topics:

- Installing ART Runtime
- Configuring ART CICS Runtime
- Configuring ART Batch Runtime
- Managing ART Applications

Installing ART Runtime

Downloading ART Installer

You must download the ART Linux x86-64 (64-bit) installer, as follows:

1. Download the BIN file from the Oracle Tuxedo OTN Web site at: http://www.oracle.com/technetwork/middleware/tuxedo/downloads/index.html
2. Copy the `art111121_64_linux_x86_64.bin` to a local directory on `ComputeNode1`.

Installing ART Software

You can install the Oracle ART product binaries in one of the shares on Sun ZFS Storage 7320 appliance locations. Note that the share, which is a shared file system, must be accessible by all compute nodes. And considering the access permission problem, it's recommended to create users with the same uid and gid on each Exalogic compute node. For example, create NIS accounts for users.

You must run the Oracle ART installer on a compute node (e.g., server named ComputeNode1), as follows:

1. Before installing Oracle ART, make sure the compute nodes meet the general installation requirements for the hardware and software.
2. Log in to `ComputeNode1` as the Oracle Tuxedo administrator.

   The Oracle Installation program uses a temporary directory in which it extracts the files from the archive that are needed to install Oracle Tuxedo on the target system. By default, the temporary directory is `/tmp`, you can enter the following command at the shell prompt:

   ```
   export IATEMPDIR=tmpdirname
   ```

   to replace the default temporary directory /tmp.

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

   ```
   prompt>sh ./ art111121_64_linux_x86_64.bin –i console
   ```

   The **Introduction & Install Set** screen is displayed.
3. In the **Choose Install Set** screen, enter 1, which is associated with **Full Install**.

   The **Choose Oracle Home** screen is displayed.

4. In the **Choose Oracle Home** screen, enter 1, which is associated with **Create new Oracle Home**.

   The **Specify a new Oracle Home directory** screen is prompted.

5. Enter your Oracle Home directory. Be sure to use the full pathname when specifying the new Oracle Home directory.

   The Oracle Home should be on the shared file system on the Sun Storage 7000 Unified Storage System, and can be accessible by all compute nodes in the Oracle Exalogic machine.

   The **Install Samples** screen is displayed.

6. Enter **Y** to install the samples.

   The **Pre- Installation Summary** screen is displayed.

7. In the **Pre- Installation Summary** screen, press `<ENTER>` to continue.

   The **Installing** screen is displayed.

8. In the **Installing** screen, no user input is required.

   The **Installation Complete** screen is displayed.

9. In the **Installation Complete** screen, press `<ENTER>` to exit the installer.

Configuring ART CICS Runtime

Important Notes Before You Begin

Before you start configuring Oracle ART components, note the following:

- The configuration example describes how to configure the environment for a single node in SHM mode. Multi-node CICS deployment using Tuxedo MP mode is described in High Availability Deployment for Oracle Tuxedo ART CICS Application
- You need to install a tn3270 terminal emulator which is used to connect with ART CICS runtime, such as IBM PCOMM, x3270 series.
- Exalogic Machine Full Rack includes 30 compute nodes, Exalogic Machine Half Rack includes 16 compute nodes, and Exalogic Machine Quarter Rack includes 8 compute nodes. You can deploy migrated CICS applications on multiple nodes using either:
  - Oracle Tuxedo MP configuration for all required nodes in a single Tuxedo domain, or

  - Multiple Tuxedo domains, each consisting of either a single node in SHM mode, or multiple nodes in MP configuration

- You should plan your application deployment and infrastructure based on the expected workload characteristics and required processing capacity.

Initial Configuration of the ART CICS Runtime

Before configuring an ART CICS application, certain environment variables and paths must be defined in order to create the ART CICS Runtime environment as listed in Table 1.

Table 1 ART CICS Runtime Environment Variables Usage

Variable	Value	Usage	Variable usage
`TUXDIR`	Set up at Installation time	Mandatory. Directory containing the Installed Tuxedo product.	TUXEDO
`TUXCONFIG`	Set up at Installation time	Mandatory. Full path name of the Tuxedo `tuxconfig` file	TUXEDO
`KIXDIR`	Set up at Installation time	Mandatory. Absolute path of the directory containing the CICS Runtime product	ART CICS Runtime
`APPDIR`	${KIXDIR}/bin	Mandatory. Directory containing the CICS Runtime Servers Binaries	ART CICS Runtime
`KIXCONFIG`	Set up at Installation time	Mandatory. Directory where the Resources Configuration Files of the ART CICS Runtime are located	ART CICS Runtime
`LC_MESSAGES`	C	Formats of informative and diagnostic messages	Tuxedo
`OBJECT_MODE`	64	64-bit architecture	ART CICS Runtime

Server Configuration of the ART CICS Runtime

The ART CICS Runtime includes several servers which act as Oracle Tuxedo application servers. These include:

– The Terminal Connection servers (TCP servers: ARTTCPH and ARTTCPL servers): manage user connections and sessions to ART CICS applications through 3270 terminals or emulators.

```
ARTTCPL SRVGRP="identifier" SRVID="number" CLOPT="[servopts options]
-- -n netaddr -L pnetaddr [-m minh] [-M maxh] [-x session-per-handler]
[-p profile-name] [-D] [+H trace-level]"
```

– The Connection Server ARTCNX: manages the user session and some system transactions relative to security (CSGM: Good Morning Screen, CESN: Sign On, CESF: Sign off).

```
ARTCNX SRVGRP="identifier" SRVID="number" CONV=Y MIN=1 MAX=1
RQADDR=QKIX110 REPLYQ=Y CLOPT="[servopts]"
```

– The Synchronous Transaction server ARTSTRN: manages standard synchronous CICS transactions that can run simultaneously.

```
ARTSTRN SRVGRP="identifier" SRVID="number" CONV=Y MIN=minn MAX=maxn
RQADDR=queueaddr REPLYQ=Y CLOPT="[servopts] -- -s TEST-l grp1:group2"
```

– The Synchronous Transaction servers ARTSTR1: manages CICS synchronous transaction applications that can not run simultaneously but only sequentially.

```
ARTSTR1 SRVGRP="identifier" SRVID="number" CONV=Y MIN=1 MAX=1
CLOPT="[servopts] -- [-s TEST] [-l group1:group2:…]"
```

– The Asynchronous Transaction servers ARTATRN and ARTATR1: are similar to the ARTSTRN and ARTSTR1 but for asynchronous transactions started by EXEC CICS START TRANSID statements.

```
ARTATRN SRVGRP="identifier" SRVID="number" CONV=N MIN=minn MAX=maxn
RQADDR=QKIXATR REPLYQ=Y CLOPT="[servopts] -- [-s TEST] [-l
group1:group2:…]"
```

```
ARTATR1 SRVGRP="identifier" SRVID="number" CONV=N MIN=1 MAX=1
CLOPT="[servopts] -- [-s TEST] [-l group1:group2,…]"
```

– TS Queue servers ARTTSQ: manage the use of CICS Temporary Storage Queues.

```
ARTTSQ SRVGRP="identifier" SRVID="number" MIN=1 MAX=1
CLOPT"[servopts] -- [-l group1:group2,…]"
```

– TD Queue servers ARTTDQ: centralizes the TD Queue operations management requested by applications It publishes one service per declared queue in the configuration file, and affects all CICS TD operations offering TD QUEUE services for each queue.

```
ARTTDQ SRVGRP="identifier" SRVID="number" MIN=1 MAX=1
CLOPT"[servopts] -- [-l group1:group2,…]"
```

– The DPL server ARTDPL: runs DPL programs.

```
ARTDPL SRVGRP="identifier" SRVID="number" MIN=minn MAX=maxn
CLOPT="[servopts] -- -s TEST -l grp1:group2"
```

– The Administration server ARTADM: manages the administration of ART CICS Runtime application resources.

```
ARTADM SRVGRP="identifier" SRVID="number"
```

Resource Configuration of the ART CICS Runtime

CICS Runtime manages a subset of the resource types previously defined in the CICS CSD file on z/OS. Each resource type definition is stored in the dedicated resource configuration file. All these files are located in the `${KIXCONFIG}` directory.

ART CICS Runtime manages the following resources:

- Tranclasses (`transclasses.desc` file)

  This file contains all the distinct Transaction classes (Tranclasses) referenced by the CICS Transactions.

- Transactions (`transactions.desc` file)

  A transaction is a CICS feature allowing a program to be run indirectly through a transaction code either manually from a 3270 screen or from another COBOL CICS program.

- Programs (`programs.desc` file)

  This file contains a list of all COBOL or C programs invoked through EXEC CICS START, LINK or XCTL statements.

- TS Queue Model (`tsqmodel.desc` file)

  This file contains all the TS Queue models referenced by TS Queues used in the CICS programs.

- Mapsets (`mapsets.desc` file)

  This file contains all the mapsets referenced by the CICS applications.

- Typeterms (`typeterms.desc` file)

  This file contains all of the 3270 terminal types supported by the ART CICS Runtime TCP servers.

Configuring ART Batch Runtime

This section will help you to understand ART Batch Runtime configuration requirements and how to user Tuxedo Job Enqueueing Service (TuxJES) to submit and manage batch jobs. For more information, see Oracle Tuxedo ART Runtime documentation.

Using the Batch Runtime EJR

Setting Environment Variables

Table 2 lists the environment variables called in the KSH scripts and must be defined before using the software.

Table 2 KSH Script Environment Variables

Variable	Usage
DATA	Directory where the data is stored
SPOOL	Sysout file directory
TMP	Temporary directory for application files

Table 3 lists the environment variables used by the ART Batch Runtime and must be defined before using the software.

Table 3 Oracle Tuxedo Application Runtime for Batch Environment Variables

Variable	Usage
PROCLIB	PROC and INCLUDE files directory.
MT_ACC_FILEPATH	File-to-file concurrency access.
MT_DB_LOGIN	Database connection user credentials.
MT_LOG	Logs directory.
MT_TMP	Temporary directory for internal files.

Submitting a Job Using EJR

When using the ART Batch Runtime, TuxJES can be used to submit jobs which are started by initiators (ARTJESINITIATOR servers), but a job can also be executed directly using the EJR (Execute Job Request) launcher.

Before performing this type of execution, ensure that the entire context is correctly set. This includes environment variables and directories required by the ART Batch Runtime.

Example of launching a job with EJR:

```
# EJR DEFVCUST.ksh
```

For a complete description of the EJR launcher, see the Oracle Tuxedo Application Runtime for Batch Reference Guide.

Using Tuxedo Job Enqueueing Service (TuxJES)

Overview

TuxJES implements a subset of the mainframe JES2 functions (for example, submit a job, display a job, hold a job, release a job, and cancel a job).

TuxJES is an Oracle Tuxedo application; Oracle Tuxedo is required in order to run TuxJES.

TuxJES includes the following key components:

- genappprofile

  Generates the security profile for Oracle Tuxedo applications

- artjesadmin

  TuxJES command interface. It is an Oracle Tuxedo client

- ARTJESADM

  TuxJES administration server. It is an Oracle Tuxedo server.

- ARTJESCONV

  TuxJES conversion server. It is an Oracle Tuxedo server.

- ARTJESINITIATOR

  TuxJES Job Initiator. It is an Oracle Tuxedo server.

- ARTJESPURGE

TuxJES purge server. It is an Oracle Tuxedo server.

For more information, see the Oracle Tuxedo Application Runtime for Batch Reference Guide.

Configuring a TuxJES System

TuxJES is an Oracle Tuxedo application. Most of the TuxJES components are Oracle Tuxedo client or Oracle Tuxedo servers. You must first configure TuxJES as an Oracle Tuxedo application. The environment variable JESDIR must be configured correctly which points to the directory where TuxJES installed.

The following TuxJES servers should be included in the Oracle Tuxedo configuration file (UBBCONFIG):

- ARTJESADM
- ARTJESCONV
- ARTJESINITIATOR
- ARTJESPURGE

For the TuxJES administration server ARTJESADM, a TuxJES configuration file should be specified using the -i option. In the Oracle Tuxedo configuration file (UBBCONFIG), ARTJESADM should be configured before the ARTJESCONV, ARTJESINITIATOR, or ARTJESPURGE servers.

For more information, see the Oracle Tuxedo Application Runtime for Batch Reference Guide.

TuxJES uses the Oracle Tuxedo /Q component, therefore an Oracle Tuxedo group with an Oracle Tuxedo messaging server TMQUEUE with TMS_QM configured is required in the UBBCONFIG file. The name of the /Q queue space should be configured as JES2QSPACE.

TuxJES uses the Oracle Tuxedo Event component, therefore an Oracle Tuxedo user event server, TMUSREVT is required in the UBBCONFIG file.

A TuxJES system can be either an Oracle Tuxedo SHM application which runs on a single compute node, or an Oracle Tuxedo MP application which runs on multiple Exalogic compute nodes and uses local initiators to control the workload on each node.

Managing ART Applications

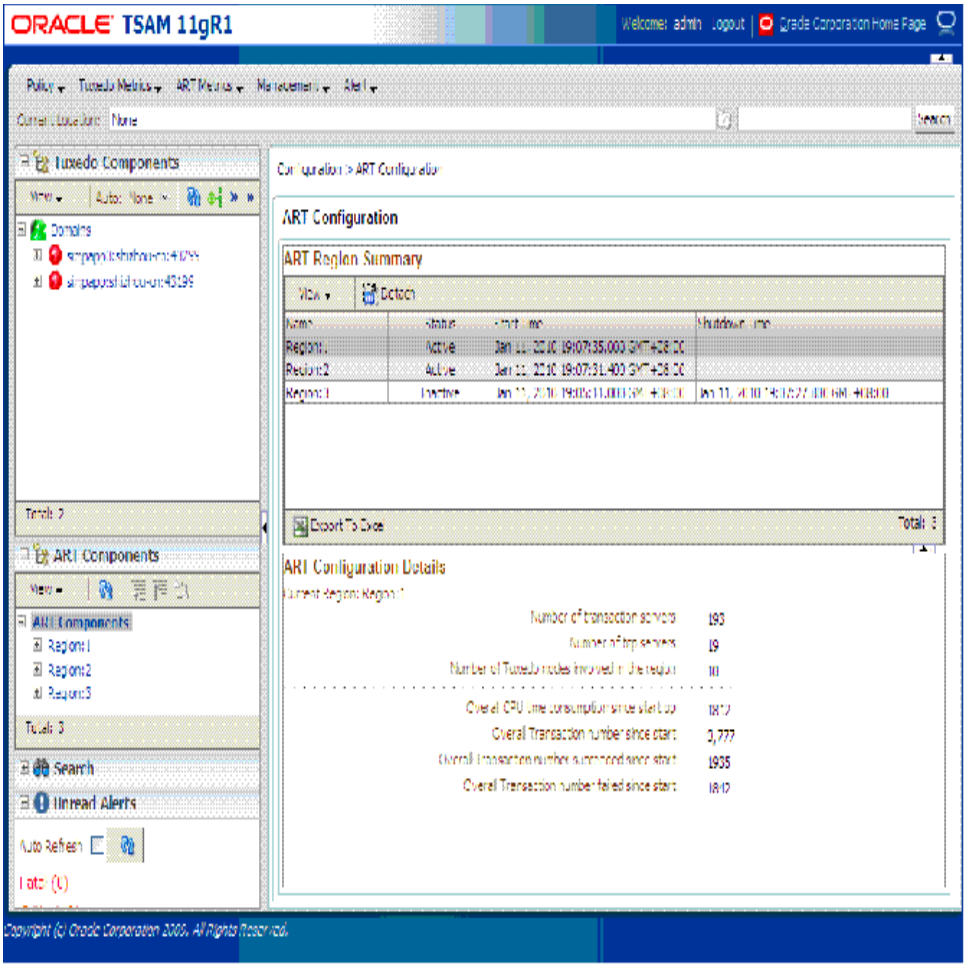
You can manage and monitor ART Runtime by using Oracle Tuxedo System and Application Monitor (TSAM), which is an Oracle Tuxedo add-on product.

Monitoring ART CICS Runtime

Oracle TSAM can be used to monitor the ART CICS transactions and terminals.

TSAM ART CICS Transaction monitoring provides an overview for each CICS region which is monitored by TSAM as shown in Figure 12. The information includes CICS region components, status, and overall statistics metrics.

Figure 12 TSAM ART CICS Transaction Monitoring

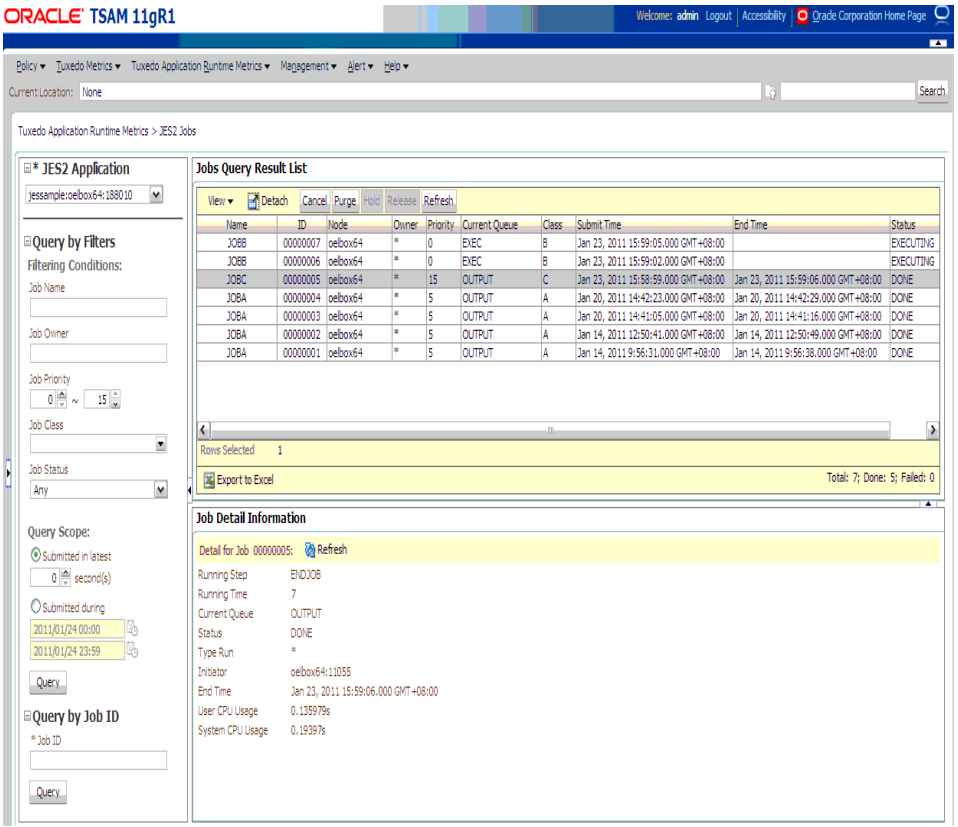


Oracle TSAM ART CICS Transaction monitoring provides live and historical metrics data (including number of transaction calls, execution time, and CPU consumption time).

Managing ART Batch Runtime

Oracle TSAM can also be used to manage ART TuxJES system as shown in Figure 12. From the Oracle TSAM console, you can display job information, cancel a job, or purge a job.

Figure 13 Batch Runtime Monitoring



High Availability and Scalability Deployment for an Oracle Tuxedo ART Batch Environment

After installing the Oracle Tuxedo and ART Batch software, you must configure Oracle ART Batch for the Oracle Exalogic enterprise deployment topology. This configuration uses the following Oracle Fusion Middleware components:

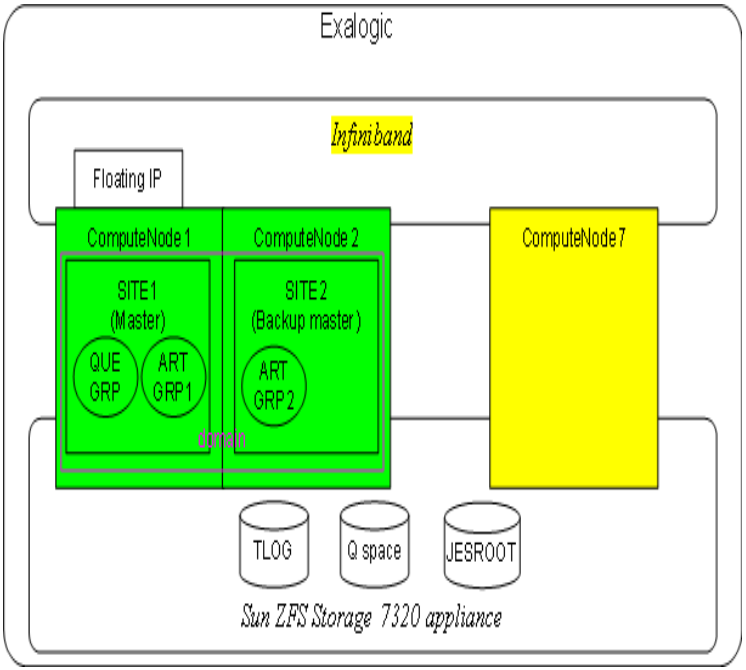
- Oracle Tuxedo 11g Release 1 PS1 (11.1.1.2.0)
- Oracle ART 11g Release1 PS1 (11.1.1.2.0)

The main dependency for high availability of the ART Batch environment is high availability configuration of the /Q system servers (such as TMQUEUE).

This section provides a configuration scenario including two machines ComputeNode1 and ComputeNode2, as well as ComputeNode7 as a standby machine, which supports failover and failback of ART Batch environment, as shown in Figure 14.

You can follow this example configuration to create ART Batch environment on the Exalogic machine, based on your application deployment and management requirements.

Figure 14 Art Batch Application Deployment Configuration Scenario



In this example configuration, you are creating an Oracle Tuxedo domain including the following:

- Oracle Tuxedo and Oracle ART Batch are installed on `ComputeNode1`, `ComputerNode2` and `ComputeNode7` respectively.
- An Oracle Tuxedo domain comprises a master node on `ComputeNode1` and a slave (backup master) node on `ComputeNode2`. `ComputeNode7` is a standby node.
- Oracle Tuxedo system servers TMQUEUE and TMS_QM in QUEGRP group run on the master node.
- A group of ART JES servers run on each node respectively.
- Oracle Tuxedo /Q Queue Space, TLOG and some ART Batch configuration files located on Sun ZFS Storage 7320 appliance, so that each of three nodes is able to access these resources with identical paths.

Note: All three nodes have their own end point (`BOND0 IP` addresses of the machines as the host addresses). In this example:

```
ComputeNode1 BOND0 IP =192.168.10.1.
ComputeNode2 BOND0 IP=192.168.10.2
ComputeNode7 BOND0 IP=192.168.10.7
```

Before You Begin

Note the following before you start configuring Oracle ART Batch components:

- The configuration example used in this guide describes how to configure the environment using three compute nodes (`Dept_1` using `ComputeNode1`, `ComputeNode2` and `ComputeNode7`). In this example, an ART JES Batch environment runs on ComputeNode1 and `ComputeNode2`, `ComputeNode1` runs a master Oracle Tuxedo node `SITE1`, and `ComputeNode2` runs a slave oracle Tuxedo node `SITE2`. `ComputeNode7` is a standby machine.
- You can extrapolate the information included in these procedures to set up and configure the environment for a larger batch grid environment, or multiple batch grid environments on an Exalogic machine. These would depend on your specific batch workload requirements, including parallelization possible within and across batch streams.

  Oracle Exalogic machine full rack includes 30 compute nodes, an Oracle Exalogic machine half rack includes 16 compute nodes, and Oracle Exalogic machine quarter rack includes 8 compute nodes. You should plan your application deployment
- You can deploy migrated Batch applications on multiple nodes using either
  - Oracle Tuxedo MP configuration for all required nodes in a single Tuxedo domain, or
  - Multiple Tuxedo domains, each consisting of either a single node in SHM mode, or multiple nodes in MP configuration

Prerequisites

Note the following prerequisites for configuring Oracle ART Batch products for Oracle Exalogic:

- Your environment (including storage, and network), must be preconfigured.
- Oracle Tuxedo 11g Release 1 PS1 (11.1.1.2.0) and Oracle ART 11g Release1 PS1(11.1.1.2.0) must be already installed.
- An Oracle Home directory must be created on a Sun ZFS Storage 7320 appliance shared file system and accessible by `ComputeNode1, ComputeNode2 and ComputeNode7.`

Enabling a Floating IP for Oracle Tuxedo Node SITE1 on ComputeNode1

Oracle Tuxedo tlisten and bridge on the Tuxedo node SITE1 must be configured to listen on a floating IP Address to enable it to seamlessly failover from one host to another. In case of a failure, the Tuxedo node SITE1, along with the virtual IP Address, can be migrated from one compute node to others.

You are associating `ComputerNode1` with a virtual hostname (virtualhost1). This Virtual Host Name must be mapped to the appropriate floating IP (e.g., 10.0.0.17) by a custom `/etc/hosts` entry. Check that the floating IP is available per your name resolution system, (`/etc/hosts`), in the required nodes in your enterprise deployment reference topology. The floating IP (10.0.0.17) that is associated with this Virtual Host Name (virtualhost1) must be enabled on `ComputeNode1`. On `ComputeNode2` and `ComputeNode7`, `hostname virtualhost1` also have to be associated into this floating I address.

To enable the floating IP on `ComputeNode1`, complete the following steps:

1. On `ComputeNode1`, run the `ifconfig` command as the root user to get the value of the netmask. as shown in Listing 1

Listing 1 Netmask Value

```
[root@ComputeNode1 ~] # /sbin/ifconfig

bond0  Link encap:Ethernet  HWaddr 00:11:43:D7:5B:06

inet addr:139.185.140.51  Bcast:139.185.140.255  Mask:255.255.255.224 inet6
addr: fe80::211:43ff:fed7:5b06/64 Scope:Link

UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1

RX packets:10626133 errors:0 dropped:0 overruns:0 frame:0

TX packets:10951629 errors:0 dropped:0 overruns:0 carrier:0 collisions:0
txqueuelen:1000

RX bytes:4036851474 (3.7 GiB) TX bytes:2770209798 (2.5 GiB) Base
address:0xecc0 Memory:dfae0000-dfb00000
```

2. On ComputeNode1, bind the floating IP Address to the network interface card using ifconfig command as the root user. Use a netmask value that was obtained in Step 1.

   ```
   /sbin/ifconfig networkCardInterface Virtual_IP_Address netmask netMask
   ```

   For example:

   ```
   /sbin/ifconfig bond0:1 10.0.0.17 netmask 255.255.255.224
   ```

   In this example, bond0:1 is the virtual network interface created for internal, fabric-level InfiniBand traffic.

3. Run the arping command as the root user to update the routing table:
   `/sbin/arping -q -U -c 3 -I networkCardInterface Floating_IP_Address`

   For example: `/sbin/arping -q -U -c 3 -I bond0 10.0.0.17`

   Note: It is recommended that you run this command several times to update the routing table.

4. Verify the floating IP address of the ComputeNode1 by pinging it from ComputeNode2.

   For example: `/bin/ping 10.0.0.17`

   Note: In this enterprise deployment topology, example IP addresses are used. You must replace them with your own IP addresses that you reconfigured using Oracle OneCommand. Even if the master Tuxedo node does not require a floating IP, it is recommended that you assign a floating IP if you want to migrate the Tuxedo node SITE1 manually from ComputeNode1 to ComputeNode7.

Configuring an ART Batch Environment on an Oracle Tuxedo Domain (ComputeNode1 and ComputeNode2)

To configure an ART Batch application on an Oracle Tuxedo Domain (comprised of ComputeNode1 and ComputerNode2), you must do the following steps:

1. Specify an Oracle Home on Sun ZFS Storage 7320 appliance, for instance, in our sample environment /u01/app/Oracle. Then install Oracle Tuxedo and Oracle Tuxedo Application Runtime for CICS and Batch under the Oracle Home directory.

   When you finish installation, two child directories under oracle: tuxedo11gR1 and art11gR1 are created (which contain Tuxedo and ART installation respectively).

2. Create a directory as the JES root directory: /u01/app/simpjob/jesroot

3. Download and build pdksh-5.2.14, and put the executable ksh under the /u01/app/simpjob/ directory

4. On ComputerNode1, create a directory as the working directory for ART Batch application: /u01/app/simpjob/site1
5. On ComputerNode2, create a directory as the working directory for ART Batch application: /u01/app/simpjob/site2
6. Set environment variables as listed in Table 1on ComputeNode1, ComputeNode2 and ComputeNode7 respectively:

Table 1 Environment Variables

Environment Variable	Usage	Value on Tuxedo node SITE1 ComputeNode1 ComputeNode7	Value on Tuxedo node SITE2 ComputeNode2
TUXDIR	Tuxedo installation directory	/u01/app/Oracle/tuxedo11gR1	
JESDIR	ART Batch installation directory	/u01/app/Oracle/art11gR1/Batch_RT	
MT_KSH	pdksh executable absolute path	/u01/app/simpjob/ksh	
MT_ROOT	EJR directory	$JESDIR/ejr	
APPDIR	Tuxedo application working directory	/u01/app/simpjob/site1	/u01/app/simpjob/site2
QMCONFIG	/Q queue space device name	/u01/app/simpjob/QUE	
TUXCONFIG	Tuxedo application configuration file	/u01/app/simpjob/site1/tuxconfig	Not applicable
PATH		${TUXDIR}/bin:${JESDIR}/bin:${MT_ROOT}:${APPDIR}:${PATH};	
LD_LIBRARY_PATH		$TUXDIR/lib:$LD_LIBRARY_PATH	
DATA	Directory where the data is stored	/u01/app/simpjob/data	

Table 1 Environment Variables

Environment Variable	Usage	Value on Tuxedo node SITE1 ComputeNode1 ComputeNode7	Value on Tuxedo node SITE2 ComputeNode2
MT_LOG	The contents of this file provide the production team with detailed information about the execution of a job.	/u01/app/simpjob/Logs/log	
SPOOL	Sysout files directory	/u01/app/simpjob/Logs/sysout	
MT_TMP	Temporary directory for internal files	/u01/app/simpjob/tmp	
MT_ACC_FILEPATH	File-to-file concurrency access	/u01/app/simpjob/acc	

7. Create UBBCONFIG file under $APPDIR on the master node as shown in Listing 2

Listing 2 UBBCONFIG File on Master Node

```
*RESOURCES

IPCKEY          132540
DOMAINID        jessample
MASTER          SITE1,SITE2
MODEL           MP
MAXACCESSERS    200
MAXSERVERS      50
NOTIFY          SIGNAL
LDBAL           Y
OPTIONS         LAN,NO_AA ,MIGRATE

*MACHINES

virtualhost1
LMID = SITE1
```

```
        TUXDIR ="/u01/app/Oracle/tuxedo11gR1"
        TUXCONFIG = "/u01/app/simpjob/site1 /tuxconfig"
        TLOGDEVICE =/u01/app/simpjob/TLOG"
        TLOGSIZE=10
        APPDIR = "/u01/app/simpjob/site1 "
        ULOGPFX = "/u01/app/simpjob/site1 /ULOG"

ComputeNode2
        LMID = SITE2
        TUXDIR ="/u01/app/Oracle/tuxedo11gR1"
        TUXCONFIG = "/u01/app/simpjob/site2/tuxconfig"
        TLOGDEVICE =/u01/app/simpjob/ TLOG"
        TLOGSIZE=10
        APPDIR = "/u01/app/simpjob/site2 "
        ULOGPFX = /u01/app/simpjob/site2 /ULOG"

*NETWORK
SITE1
                NADDR="// virtualhost1:9103"
                NLSADDR="// virtualhost1:3150"

SITE2
                NADDR="// ComputeNode2:9103"
                NLSADDR="// ComputeNode2:3150"

*GROUPS

QUEGRP
        LMID = SITE1,SITE2  GRPNO = 1
        TMSNAME = TMS_QM TMSCOUNT = 2
        OPENINFO = "TUXEDO/QM:/u01/app/simpjob/QUE:JES2QSPACE"

EVTGRP
```

```
                LMID= SITE1,SITE2  GRPNO=2

ARTGRP1
        LMID = SITE1  GRPNO = 3
ARTGRP2
        LMID = SITE2  GRPNO = 4

#
*SERVERS
#
DEFAULT:        CLOPT="-A"
TMUSREVT        SRVGRP=EVTGRP SRVID=1 CLOPT="-A"
TMQUEUE
        SRVGRP = QUEGRP  SRVID = 1
        GRACE = 0  RESTART = Y CONV = N MAXGEN=10
        CLOPT = "-s JES2QSPACE:TMQUEUE -- -t 5 "

ARTJESADM        SRVGRP =ARTGRP1  SRVID = 1 MIN=1 MAX=1
                CLOPT = "-A -- -i /u01/app/simpjob/jesconfig"
ARTJESCONV       SRVGRP =ARTGRP1  SRVID = 20 MIN=1 MAX=1
                CLOPT = "-A --"
ARTJESINITIATOR       SRVGRP =ARTGRP1  SRVID =30
                CLOPT = "-A -- -n 20 -d"
ARTJESPURGE       SRVGRP =ARTGRP1  SRVID = 100
                CLOPT = "-A --"

ARTJESADM        SRVGRP =ARTGRP2  SRVID = 1 MIN=1 MAX=1
                CLOPT = "-A -- -i /u01/app/simpjob/jesconfig"
ARTJESCONV       SRVGRP =ARTGRP2  SRVID = 20 MIN=1 MAX=1
                CLOPT = "-A --"
ARTJESINITIATOR       SRVGRP =ARTGRP2  SRVID =30
```

```
                CLOPT = "-A -- -n 20 -d"
ARTJESPURGE       SRVGRP =ARTGRP2  SRVID = 100
                CLOPT = "-A --"

*SERVICES
```

The key points in this configuration files are:

- In order to leverage Tuxedo ART failover and failback capabilities on an Exalogic machine, we use the virtual machine name `ComputeNode1` in the configuration file, which is parsed into the floating IP address "10.0.0.17" in the Art Batch environment at run time.
- SITE1 is a master node and SITE2 is a backup master node.
- In a Tuxedo MP configuration, /Q related system servers for a specific queue space must run in the same Tuxedo node. In order to provide high availability, group QUEGRP must have an alternate location for fail-over. When the master node fails, you can migrate group QUEGRP to the alternate node.
- The queue space and TLOG device are built on Sun ZFS Storage 7320 appliance and will be used by the master node only (in normal operation). When the master Tuxedo node fails, the backup master node can take over these resources.
- To leverage ART Batch and Exalogic scale out, ART Batch dedicated servers are deployed on the master node and the slave node respectively. Additional slave nodes can be easily added to this configuration.
- ART Batch servers on the master node and the slave node should share the same copy of Batch resources on the Sun ZFS Storage 7320 appliance, including `jesconfig` file, `jesroot` directory, spooling device and logging facility.

8. Create a `jesconfig` file under the `/u01/app/simpjob` directory. The content of this file includes:

   ```
   JESROOT=/u01/app/simpjob/jesroot
   DEFAULTJOBCLASS=B
   DEFAULTJOBPRIORITY=9
   DUPL_JOB=NODELAY
   ```

9. Execute `tmloadcf -y` UBBCONFIG file on `ComputeNode1`.

10. On `ComputeNode1`, create TLOG file using script file `crlog` as follows:

```
tmadmin <<!
echo
crdl -b 200 -z /u01/app/simpjob/TLOG
crlog -m SITE1
q!
```

11. On ComputeNode1, create /Q queue space device using script file `jesinint` shipped with ART Batch samples. Before executing this script, please modify the beginning of lines as below:

```
#!/bin/ksh
qmadmin  /u01/app/simpjob/QUE <<!
echo
crdl  /u01/app/simpjob/QUE 0 10000
qspacecreate
JES2QSPACE
22839
5000
50
1000
1000
10000
errque
y
16
….
```

12. On the ComputeNode1, start `tlisten` using the command:

```
ComputeNode1> tlisten -l  // virtualhost1:3150
```

13. On the ComputeNode2, start `tlisten` using the command:

```
ComputeNode2> tlisten -l  //localhost:3150
```

14. Run `tmboot -y` on `ComputeNode1` to boot up Oracle Tuxedo and the ART Batch environment

Verify Job Execution

After the ART batch application starts on the ComputeNode1 and ComputeNode2, you can execute jobs with `artjesadmin` for verifications.

You can run `artjesadmin` on either the master node or the slave node. No matter where you issue the submit job command, the job may be executed either on the master node or on the slave

node depending on its class and the configuration of the initiators. In order to make sure the ART Batch related servers on each Tuxedo node are able to access the job script files correctly, it is strongly recommended that you do the following:

1. Put all JOB scripts in a repository on Sun ZFS Storage 7320 appliance.

   For example: `/u01/app/jobs`.

2. Submit job using the full job script path.

   For example:

```
[root@ComputeNode1 ~] # artjesadmin
artjesadmin - Copyright (c) 2010 Oracle
All Rights Reserved.
> smj -i /u01/app/jobs/JOBA
Job 00000035 is submitted successfully
```

You can submit a number of jobs and execute system utility `ps` on the master node and the slave node respectively to observe "EJR" processes executing jobs on each node.You can also see the status of the jobs using Oracle TSAM (as described in Managing ART Batch Runtime).

Master Node and Group QUEGRP to ComputeNode2 Failover

In case the compute node (ComputeNode1) hosting the master node fails, users can migrate master node into the backup master node, using `tmadmin` command `pclean` to clean partitioned SITE1 and reboot the /Q related group on the SITE2 as shown in Listing 3.

Note the following:

1. Migrate the master node into the backup master node, on `ComputeNode2`:

Listing 3 Master Node/backup Node Migration

```
ComputeNode2> tmadmin
tmadmin - Copyright (c) 1996-2010 Oracle.
Portions * Copyright 1986-1997 RSA Data Security, Inc.
All Rights Reserved.
Distributed under license by Oracle.
Tuxedo is a registered trademark.
TMADMIN_CAT:199: WARN: Cannot become administrator.Limited set of
commands available.

> master
```

```
Are you sure? [y, n] y

Creating new DBBL on SITE2, please wait ...
New DBBL created on SITE2
> q
```

2. Clean the partitioned node and fail over /Q related group into ComputeNode2 as shown in Listing 4

Listing 4 /Q Related Group into ComputeNode2

```
ComputeNode2> tmadmin
tmadmin - Copyright (c) 1996-2010 Oracle.
Portions * Copyright 1986-1997 RSA Data Security, Inc.
All Rights Reserved.
Distributed under license by Oracle.
Tuxedo is a registered trademark.
TMADMIN_CAT:199: WARN: Cannot become administrator.Limited set of commands
available.
> psr
Prog Name      Queue Name  Grp Name      ID RqDone Load Done Current Service
---------      ----------  --------      -- ------ --------- ---------------
ARTJESPURGE    00004.00100 ARTGRP2       100      -         - ( - )
ARTJESPURGE    00003.00100 ARTGRP1      100      -         - ( PARTITIONED )
BBL            30003.00000 SITE2          0      -         - ( - )
BBL           30002.00000 SITE1          0      -         - ( PARTITIONED )
DBBL           132540      SITE2          0     12       600 ..MASTERBB
ARTJESADM      00004.00001 ARTGRP2        1      -         - ( - )
ARTJESADM      00003.00001 ARTGRP1        1      -         - ( PARTITIONED )
TMQUEUE        00001.00001 QUEGRP         1      -         - ( PARTITIONED )
TMS_QM         QUEGRP_TMS  QUEGRP     30001      -         - ( PARTITIONED )
TMUSREVT       00002.00001 EVTGRP         1      -         - ( PARTITIONED )
```

```
BRIDGE          656828      SITE2           1       -          - ( - )
BRIDGE          394684      SITE1           1       -          - ( PARTITIONED )
TMS_QM          QUEGRP_TMS  QUEGRP      30002       -          - ( PARTITIONED )
ARTJESCONV      00004.00020 ARTGRP2        20       -          - ( - )
ARTJESCONV      00003.00020 ARTGRP1        20       -          - ( PARTITIONED )
ARTJESINITIATO 00004.00030 ARTGRP2        30       -          - ( - )
ARTJESINITIATO 00003.00030 ARTGRP1        30       -          - ( PARTITIONED )

> pclean SITE1

Cleaning the DBBL.
        Pausing 10 seconds waiting for system to stabilize.
        10 SITE1 servers removed from bulletin board

> boot -g QUEGRP
INFO: Oracle Tuxedo, Version 11.1.1.2.0, 64-bit, Patch Level (none)

Booting server processes ...

exec TMS_QM -A :
        on SITE1 -> CMDTUX_CAT:822: ERROR: No BBL available, cannot boot

tmboot: WARN: No BBL available on site SITE1.
        Will not attempt to boot server processes on that site.

        on SITE2 -> process id=14713 ... Started.
exec TMS_QM -A :
        on SITE2 -> process id=14714 ... Started.
exec TMQUEUE -s JES2QSPACE:TMQUEUE -- -t 5  :
```

```
        on SITE2 -> process id=14715 ... Started.
3 processes started.
TMADMIN_CAT:1295: ERROR: Failure return from tmboot - 0x1, errno 0.
```

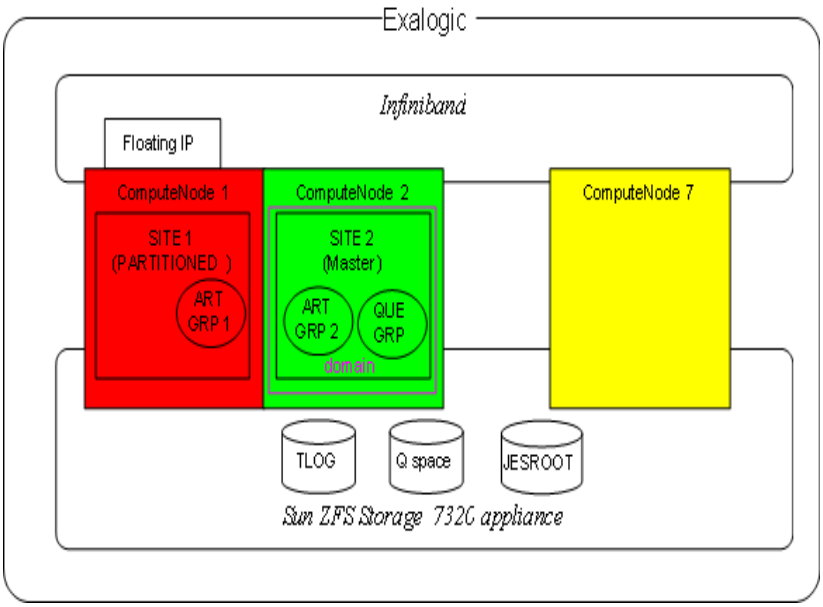
When boot group QUEGRP, oracle Tuxedo tries to boot it on SITE1 first. If it fails, it tries to boot group QUEGRP on alternative location SITE2.

After successfully booting the QUEGRP group on the `ComputeNode2`, all "WAITING" jobs are processed successively.

Note: You can also reboot EVTGRP on the ComputeNode2.

Listing 15 illustrates the application deployment after failing over the master node and group QUEGRP to ComputerNode2.

Figure 15 Art Batch Application Deployment after Master Failover



SITE1 to ComputeNode7 Failover

After the master node was migrated into ComputeNode2, the ART Batch application can go on executing jobs, but SITE2 is the only active node. In order to guarantee high availability and high scalability, users may intend to fail over SITE1 into ComputeNode7.

Assumptions:

- tlisten and bridge on the ComputeNode1 are configured to listen on 10.0.0.17. This address is the floating IP assigned to the node on the ComputeNode1 using the BOND0 interface.
- The SITE1 failed over from ComputeNode1 to ComputeNode7, and the two nodes have these IPs (BOND0):
  - ComputeNode1: 192.168.10.1
  - ComputeNode7: 192.168.10.7
  - -0.0.0.17: This is the floating IP where Oracle Tuxedo node SITE1 is running, assigned to bond0:Y.
- The working directory of ComputeNode1 where the Tuxedo node SITE1 is running is on shared storage. In the sample configuration, the directory /u01/app/simpjob/site1 is used.

The following procedure shows how to fail over the Tuxedo node SITE1 to a different machine (ComputeNode7), meanwhile the SITE1 node will still use the same Oracle Tuxedo machine (which is a logical machine, not a physical machine).

1. Migrate IP to the second node.
   a. Run the following command as root on ComputeNode1 (where bond0:Y is the current interface used by ADMINVHN1):

      ```
      ComputeNode1> /sbin/ifconfig bond0:Y down
      ```

      Note: this step can be omitted if ComputeNode1 has experienced a hardware crash.

   b. Run the following command as root on ComputeNode7:

      ```
      ComputeNode7> /sbin/ifconfig <interface:index> <IP_Address> netmask
      <netmask>
      ```

      For example:

      ```
      /sbin/ifconfig bond0:1 10.0.0.17 netmask 255.255.255.0
      ```

Note: Ensure that the netmask and interface to be used match the available network configuration in `ComputeNode2`.

2. Update routing tables through `arping`. Run the following command as root on `ComputeNode7:`

```
ComputeNode7> /sbin/arping -b -A -c 3 -I bond0 10.0.0.17 netmask
255.255.255.224
```

3. Start the `SITE1` on `ComputeNode7` from `ComputeNode2`:

   a. On `ComputeNode7`, add `hostname virtualhost1` into file `/etc/host` and point it to floating IP address 10.0.0.17

   b. Start `tlisten` on `ComputeNode7`

```
ComputeNode7> tlisten -l  // virtualhost1:3150
```

   c. b. Start BBL and bridge first:

```
ComputeNode2> tmboot -B SITE1
INFO: Oracle Tuxedo, Version 11.1.1.2.0, 64-bit, Patch Level (none)
Booting admin processes ...
exec BBL -A :
        on SITE1 -> CMDTUX_CAT:821: INFO: Duplicate server.
0 processes started.
```

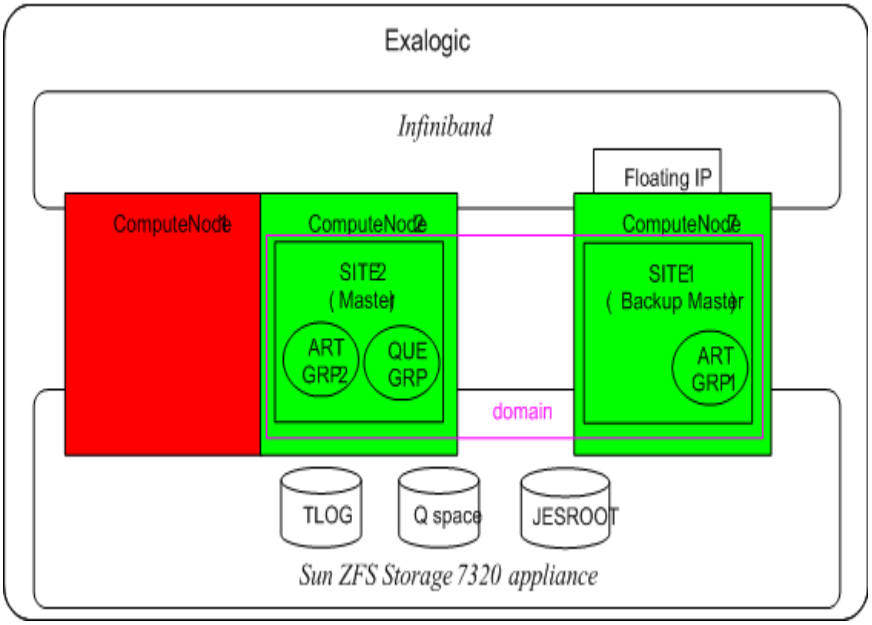
   d. b. Start `ARTGRP1`

```
ComputeNode2> tmboot -g ARTGRP1
INFO: Oracle Tuxedo, Version 11.1.1.2.0, 64-bit, Patch Level (none)
Booting server processes ...
exec ARTJESADM -A -- -i /u01/app/simpjob/jesconfig :
        on SITE1 -> process id=24347 ... Started.
exec ARTJESCONV -A -- :
        on SITE1 -> process id=24348 ... Started.
exec ARTJESINITIATOR -A -- -n 20 -d :
        on SITE1 -> process id=24350 ... Started.
exec ARTJESPURGE -A -- :
        on SITE1 -> process id=24352 ... Started.
```

4. processes started.

Figure 16 illustrates application deployment after failing over `SITE1` to `ComputerNode7`.

Figure 16 Art Batch Application Deployment After Failing Over SITE1



Failing Over SITE1 Back to ComputeNode1

After fixing `ComputeNode1`, you may (optionally) failover SITE1 back to `ComputeNode1`; do the following steps:

1. Shutdown `SITE1` first from `ComputeNode2`:

```
ComputeNode2> tmshutdown -l SITE1

Shutting down server processes ...

   Server Id = 100 Group Id = ARTGRP1 Machine = SITE1:      shutdown
   succeeded

   Server Id = 30 Group Id = ARTGRP1 Machine = SITE1:       shutdown
   succeeded

   Server Id = 20 Group Id = ARTGRP1 Machine = SITE1:       shutdown
   succeeded
```

```
Server Id = 1 Group Id = ARTGRP1 Machine = SITE1:       shutdown
succeeded

4 processes stopped.

ComputeNode2> tmshutdown -B SITE1

Shutting down admin processes ...

        Server Id = 0 Group Id = SITE1 Machine = SITE1: shutdown
succeeded

1 process stopped.
```

2. Disable the floating IP on `ComputeNode7`.

```
ComputeNode7> /sbin/ifconfig bond0:N down
```

3. Recover the floating IP on `ComputeNode1`.Run the following command on `ComputeNode1`:

```
ComputeNode1> /sbin/ifconfig bond0:Y 192.168.10.1 netmask
255.255.255.0
```

Note: Ensure that the netmask and interface to be used match the available network configuration in ComputeNode1.

Update routing tables through `arping`. Run the following command from `ComputeNode1`:

```
ComputeNode1> /sbin/arping -b -A -c 3 -I bond0:1 10.0.0.17
```

4. Start `tlisten` on `ComputeNode1`

```
ComputeNode1> tlisten -l  //virtualhost1:3150
```

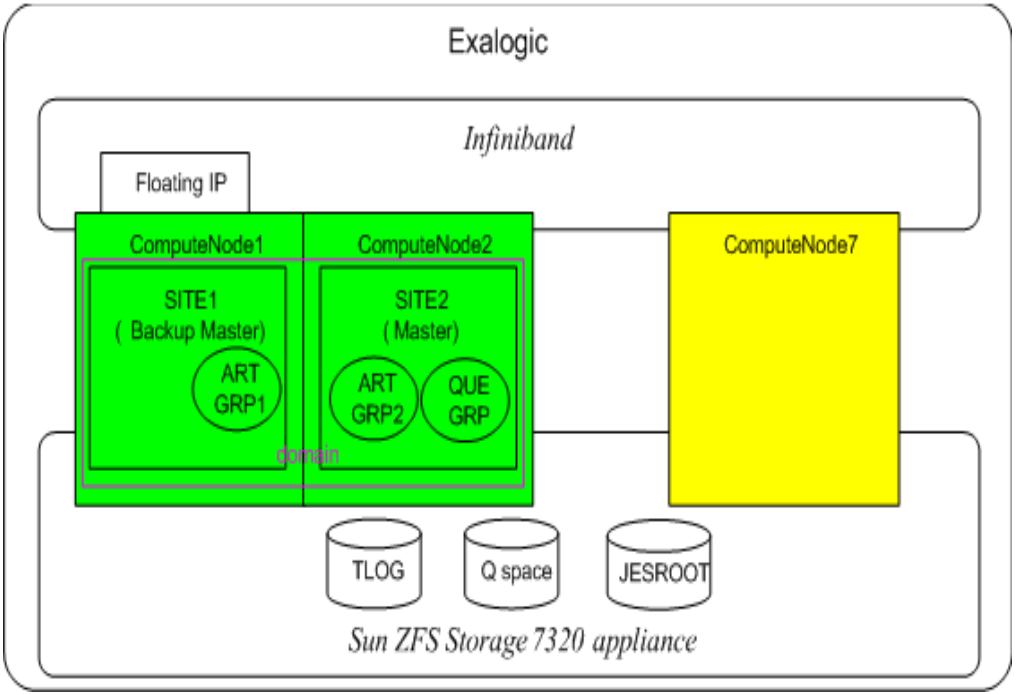
5. Reboot `SITE1` on `ComputeNode1` from `ComputeNode2`:

```
ComputeNode2> tmboot -B SITE1

INFO: Oracle Tuxedo, Version 11.1.1.2.0, 64-bit, Patch Level (none)

Booting admin processes ...

exec BBL -A :

        on SITE1 -> process id=24811 ... Started.

1 process started.

ComputeNode2>  tmboot -l SITE1

INFO: Oracle Tuxedo, Version 11.1.1.2.0, 64-bit, Patch Level (none)

Booting server processes ...

exec TMUSREVT -A :

        on SITE1 -> process id=24816 ... Started.
```

```
exec ARTJESADM -A -- -i /u01/app/simpjob/jesconfig :
        on SITE1 -> process id=24817 ... Started.
exec ARTJESCONV -A -- :
        on SITE1 -> process id=24818 ... Started.
exec ARTJESINITIATOR -A -- -n 20 -d :
        on SITE1 -> process id=24820 ... Started.
exec ARTJESPURGE -A -- :
        on SITE1 -> process id=24822 ... Started.
5 processes started.
```

Listing 16 illustrates the application deployment after failing `SITE1` back to `ComputeNode1`.

Figure 17 Art Batch Application Deployment After Failing Back SITE1



High Availability Deployment for Oracle Tuxedo ART CICS Application

High Availability with ART CICS Runtime

The Oracle Tuxedo ART CICS runtime deployment is highly available. Because ART CICS runtime servers are Tuxedo application servers, these servers and Tuxedo system servers can be replicated on different nodes in a Tuxedo MP domain, so that they can provide services even after some of the nodes are down.

For more information about Tuxedo high availability, see Achieving High Availability with Oracle Tuxedo at http://www.oracle.com/technocrats/middle ware/tuxedo/overview/index.html.

Initial Configuration of the ART CICS Runtime MP Domain

All environment variables used by ART CICS runtime are listed in Table 1.

- On MP non-master nodes, set these environment variables before starting the `tlisten` process.
- On MP master node, set these environment variables before loading UBBCONFIG file, and booting the application.

Server Configuration of the ART CICS Runtime MP Domain

All ART CICS runtime servers are listed in Server Configuration of the ART CICS Runtime. For functions that require high availability, the corresponding server should be replicated on different nodes in the MP domain. MP domain configuration for ART CICS runtime on Exalogic should follow common rules for Tuxedo MP application as described in Configuring Tuxedo Applications, as well as the rules for ART CICS runtime MP application as following,

- Tuxedo `tlisten` and BRIDGE processes are for internal communication between different nodes of an MP domain, so they should listen on the network address which is bound to IP over InfiniBand (IPoIB) interface. The network address used by `tlisten` and BRIDGE processes are specified by the `NLSADDR` and `NADDR` parameters respectively in NETWORK section of UBBCONFIG file.
- ART terminal server ARTTCPL is used to accept connections through 3270 terminal emulators from outside or inside of the Exalogic machine, so it should listen on the network address which is bound to Ethernet over InfiniBand (EoIB) interface. The network address used by ARTTCPL server is specified by its command line option '-n' for normal

connection, or '-S' for SSL connection. You may use floating IP for the network address of ARTTCPL server, so that switching ARTTCPL server to another node will be transparent to 3270 terminal emulator.

- ART connection server ARTCNX offers internal services needed by ART terminal handlers during user connection and disconnection phases. It generates terminal ID for each connection. When replicating ARTCNX server, command line option '-t' should be specified for the ARTCNX server to use unique terminal ID prefix within the MP domain.
- When ART CICS runtime resource definition directory, specified by environment variable KIXCONFIG, is not shared between different nodes of the MP domain, configure ART administration server ARTADM on all nodes to propagate resource definition files from MP master node to non-master nodes.
- ART CICS runtime transaction servers ARTSTRN and ARTATRN, and DPL (Distributed Program Link) server ARTDPL, can be replicated on different nodes of an MP domain using active/active configuration.
- ART CICS runtime transaction servers ARTSTR1 and ARTATR1 should be replicated on different nodes of an MP domain using active/passive configuration. Because theses servers handle the transactions that cannot run concurrently, only one such server for a particular transaction can be booted, and the alternate server on the backup node should be configured and automatically started to take over the server on the failing node.
- ART CICS TSQ (Temporary Storage Queue) server ARTTSQ should be replicated on different nodes of an MP domain using active/passive configuration. For a particular TS queue model only one ARTTSQ server can be booted within an Oracle Tuxedo domain. In order to achieve high availability, the alternate server for this TS queue model on the backup node should be configured and automatically started to take over the server on the failing node. Simultaneously, the directory holding the TS queue files (specified by `KIX_TS_DIR` environment variable), should be on the shared file system which is accessible to all nodes where the primary and alternate ARTTSQ servers reside.
- ART CICS TDQ (Transient Data Queue) server ARTTDQ should be replicated on different nodes of an MP domain using active/passive configuration. Because for a particular extra partition TD queue only one ARTTDQ server can be booted within a Tuxedo domain, so to achieve high availability, the alternate server for this TD queue on the backup node should be configured and automatically started to take over the server on the failing node. At the same time, the directory holding the TD queue files, specified by environment variable KIX_TD_DIR, should be on the shared file system which is accessible to all nodes that primary and alternate ARTTDQ servers reside.

Resource Configuration of the ART CICS Runtime MP Domain

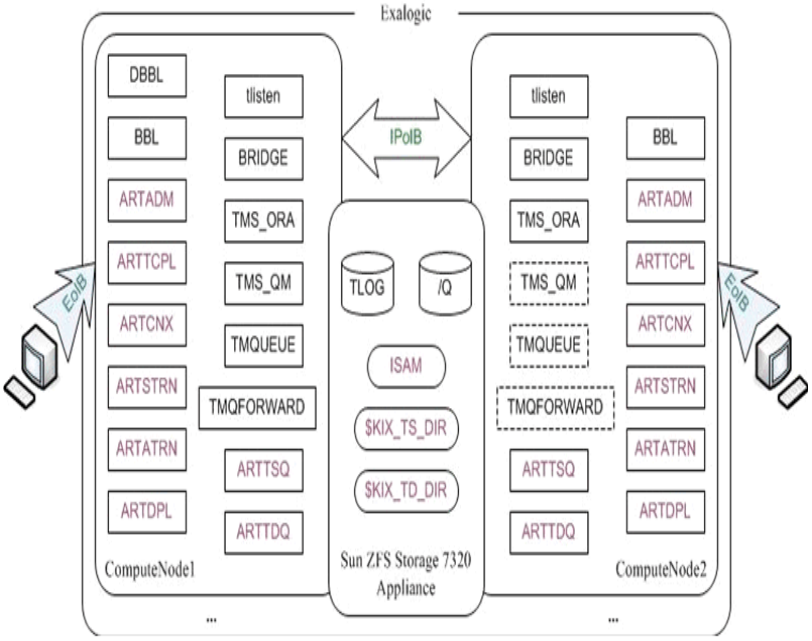
All ART CICS runtime resource definition files are listed in Resource Configuration of the ART CICS Runtime. You can share these resource definition files among different nodes of the MP domain, or have these files propagated from MP master node to non-master nodes automatically by the ARTADM server. To achieve high availability, all resources, including converted VSAM files and Tuxedo /Q should be on the shared file system which is accessible to all nodes.

- Tuxedo transaction log (TLOG) device should be configured on the shared file system which is accessible to all nodes.
- ART CICS resource definition files are stored under the `${KIXCONFIG}`directory. To share resource definition files among different nodes of an MP domain, use common `${KIXCONFIG}` directory on the shared file system which is accessible to all nodes. If the resource definition directory `${KIXCONFIG}` is not shared, configure ART administration server ARTADM on all nodes to propagate resource definition files from MP master node to non-master nodes.
- For mainframe VSAM file, you may choose to convert it to database table or indexed ISAM file. If the mainframe VSAM file is converted to database table, its data can be accessed by multiple ART CICS transactions concurrently, and the data is recoverable. If the mainframe VSAM file is converted to indexed ISAM file, its data can not be accessed concurrently, and the data is not recoverable. The directory used by ART CICS runtime to locate the ISAM files is specified by environment variable DATA. So the `${DATA}` directory should be on the shared file system which is accessible to all nodes.
- Delayed asynchronous transactions are stored temporarily in Tuxedo /Q queue space ASYNC_QSPACE. And inter partition TD queue data is also stored in Tuxedo /Q. You need to configure the /Q device on the shared file system which is accessible to all nodes.

ART CICS Runtime MP Domain Deployment Example

Following is a sample deployment of the ART CICS runtime MP domain.

Figure 18 ART CICS runtime MP Domain Deployment Example



Listing 1 shows an ART CICS runtime MP domain deployment UBBCONFIG file example.

Listing 1 ART CICS Runtime MP Domain Deployment UBBCONFIG File Example

```
#--------------------------------------------------------------------
*RESOURCES
IPCKEY          <Base IPC key>
DOMAINID        KIXD
MODEL           MP
MASTER          NODE1,NODE2
OPTIONS         LAN,MIGRATE
MAXACCESSERS    200
```

```
MAXSERVERS    100
MAXSERVICES   1000

#------------------------------------------------------------------
*MACHINES
"<Hostname of NODE1>"
    LMID=NODE1
    TUXDIR="<Tuxedo installation directory>"
    APPDIR="<Application directory on NODE1>"
    TUXCONFIG="<TUXCONFIG absolute path on NODE1>"
    TLOGDEVICE="<TLOG device on shared file system>"

"<Hostname of NODE2>"
    LMID=NODE2
    TUXDIR="<Tuxedo installation directory>"
    APPDIR="<Application directory on NODE2>"
    TUXCONFIG="<TUXCONFIG absolute path on NODE2>"
    TLOGDEVICE="<TLOG device on shared file system>"

#------------------------------------------------------------------
*NETWORK
NODE1 NLSADDR="//<IPoIB address>:<Port for tlisten>"
      NADDR="//<IPoIB address>:<Port for BRIDGE>"
NODE2 NLSADDR="//<IPoIB address>:<Port for tlisten>"
      NADDR="//<IPoIB address>:<Port for BRIDGE>"

#------------------------------------------------------------------
```

```
*GROUPS
NODE1_NOTRN
    LMID=NODE1 GRPNO=1

NODE2_NOTRN
    LMID=NODE2 GRPNO=2

NODE1_ORA
    LMID=NODE1 GRPNO=3
    TMSNAME="TMS_ORA" TMSCOUNT=2
    OPENINFO="Oracle_XA:Oracle_XA+Acc=P/<Oracle user>/<Oracle
password>+SqlNet=<Oracle SID>+SesTm=600"

NODE2_ORA
    LMID=NODE2 GRPNO=4
    TMSNAME="TMS_ORA" TMSCOUNT=2
    OPENINFO="Oracle_XA:Oracle_XA+Acc=P/<Oracle user>/<Oracle
password>+SqlNet=<Oracle SID>+SesTm=600"

QUEGRP
    LMID=NODE1,NODE2 GRPNO=5
    TMSNAME="TMS_QM" TMSCOUNT=2
    OPENINFO="TUXEDO/QM:<Q device on shared file system>:ASYNC_QSPACE"

#--------------------------------------------------------------------
*SERVERS
ARTADM
    SRVGRP=NODE1_NOTRN SRVID=10
```

```
ARTADM
    SRVGRP=NODE2_NOTRN SRVID=10

ARTTCPL
    SRVGRP=NODE1_NOTRN SRVID=20
    CLOPT=" -- -n //<EoIB address>:<Connection port> -L //<EoIB
address>:<Internal port> -m 1 -M 4"

ARTTCPL
    SRVGRP=NODE2_NOTRN SRVID=20
    CLOPT=" -- -n //<EoIB address>:<Connection port> -L //<EoIB
address>:<Internal port> -m 1 -M 4"

ARTCNX
    SRVGRP=NODE1_NOTRN SRVID=30
    MIN=1 MAX=1
    CONV=Y
    CLOPT="-o <Stdout for ARTCNX on NODE1> -e <Stderr for ARTCNX on NODE1>
-- -t A"

ARTCNX
    SRVGRP=NODE2_NOTRN SRVID=30
    MIN=1 MAX=1
    CONV=Y
    CLOPT="-o <Stdout for ARTCNX on NODE2> -e <Stderr for ARTCNX on NODE2>
-- -t B"
```

```
ARTSTRN
    SRVGRP=NODE1_ORA SRVID=10
    CONV=Y
   CLOPT="-o <Stdout for ARTSTRN on NODE1> -e <Stderr for ARTSTRN on NODE1>
-- -s <System ID> -l <Resource groups>"

ARTSTRN
    SRVGRP=NODE2_ORA SRVID=10
    CONV=Y
   CLOPT="-o <Stdout for ARTSTRN on NODE2> -e <Stderr for ARTSTRN on NODE2>
-- -s <System ID> -l <Resource groups>"

ARTATRN
    SRVGRP=NODE1_ORA SRVID=20
   CLOPT="-o <Stdout for ARTATRN on NODE1> -e <Stderr for ARTATRN on NODE1>
-- -s <System ID> -l <Resource groups>"

ARTATRN
    SRVGRP=NODE2_ORA SRVID=20
   CLOPT="-o <Stdout for ARTATRN on NODE2> -e <Stderr for ARTATRN on NODE2>
-- -s <System ID> -l <Resource groups>"

ARTDPL
    SRVGRP=NODE1_ORA SRVID=30
    CLOPT="-o <Stdout for ARTDPL on NODE1> -e <Stderr for ARTDPL on NODE1>
-- -s <System ID> -l <Resource groups>"

ARTDPL
    SRVGRP=NODE2_ORA SRVID=30
```

```
    CLOPT="-o <Stdout for ARTDPL on NODE2> -e <Stderr for ARTDPL on NODE2>
-- -s <System ID> -l <Resource groups>"

ARTSTR1
    SRVGRP=NODE1_ORA SRVID=40
    MIN=1 MAX=1
    CONV=Y
   CLOPT="-o <Stdout for ARTSTR1 on NODE1> -e <Stderr for ARTSTR1 on NODE1>
-- -s <System ID> -l <Resource groups>"

ARTSTR1
    SRVGRP=NODE2_ORA SRVID=40
    MIN=1 MAX=1
    CONV=Y
   CLOPT="-o <Stdout for ARTSTR1 on NODE2> -e <Stderr for ARTSTR1 on NODE2>
-- -s <System ID> -l <Resource groups>"

ARTATR1
    SRVGRP=NODE1_ORA SRVID=50
    MIN=1 MAX=1
   CLOPT="-o <Stdout for ARTATR1 on NODE1> -e <Stderr for ARTATR1 on NODE1>
-- -s <System ID> -l <Resource groups>"

ARTATR1
    SRVGRP=NODE2_ORA SRVID=50
    MIN=1 MAX=1
   CLOPT="-o <Stdout for ARTATR1 on NODE2> -e <Stderr for ARTATR1 on NODE2>
-- -s <System ID> -l <Resource groups>"
```

```
ARTTSQ
    SRVGRP=NODE1_ORA SRVID=60
    MIN=1 MAX=1
    CLOPT="-o <Stdout for ARTTSQ on NODE1> -e <Stderr for ARTTSQ on NODE1>
-- -s <System ID> -l <Resource groups>"

ARTTSQ
    SRVGRP=NODE2_ORA SRVID=60
    MIN=1 MAX=1
    CLOPT="-o <Stdout for ARTTSQ on NODE2> -e <Stderr for ARTTSQ on NODE2>
-- -s <System ID> -l <Resource groups>"

ARTTDQ
    SRVGRP=NODE1_ORA SRVID=70
    MIN=1 MAX=1
    CLOPT="-o <Stdout for ARTTDQ on NODE1> -e <Stderr for ARTTDQ on NODE1>
-- -s <System ID> -l <Resource groups>"

ARTTDQ
    SRVGRP=NODE2_ORA SRVID=70
    MIN=1 MAX=1
    CLOPT="-o <Stdout for ARTTDQ on NODE2> -e <Stderr for ARTTDQ on NODE2>
-- -s <System ID> -l <Resource groups>"

TMQUEUE
    SRVGRP=QUEGRP SRVID=10
    CLOPT="-s ASYNC_QSPACE:TMQUEUE -- "
    RESTART=Y GRACE=120
```

```
TMQFORWARD
    SRVGRP=QUEGRP SRVID=20
    CLOPT="-- -q ASYNC_QUEUE"
    RESTART=Y GRACE=120
```

Scale Out ART CICS Deployment at Runtime

As with other Tuxedo applications, to meet the requirement of performance, availability and security, ART CICS deployment can be changed without shutting it down. To help you dynamically modify your application, Tuxedo provides the following three methods: Administration Console, command-line utilities (`tmadmin`, `tmconfig`), and Management Information Base (MIB) API. Using any one of these three methods, you can add, change and remove an application including adding a new machine node, new server group, or new server and activating the server, etc.

To add a new ARTTCPL server via MIB, do the following steps:

1. Create a file named addserver.dat as shown in Listing 2.

Listing 2 addserver.dat File

```
SRVCNM  .TMIB
TA_OPERATION    SET
TA_CLASS        T_SERVER
TA_SERVERNAME   ARTTCPL
TA_SRVGRP       <Server group name>
TA_SRVID        <Server ID>
TA_CLOPT        <Server CLOPT>
TA_STATE        NEW
```

2. Set FML32 environment variables:

```
export FLDTBLDIR32=$TUXDIR/udataobj
```

```
export FIELDTBLS32=tpadm,Usysfl32
```

3. Add new ARTTCPL server using the `ud32` utility

   ```
   ud32 -C tpsysadm < addserver.dat
   ```

4. Boot the server using the `tmboot` command

   ```
   tmboot -g <Server group name> -i <Server ID>
   ```