

Oracle® Tuxedo System and Application Monitor (TSAM)

Administration Guide

11g Release 1 (11.1.1.2)

July 2011

Copyright © 2007, 2011, Oracle and/or its affiliates. All rights reserved.

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

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

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

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

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

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

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

Introduction to Oracle TSAM

Overview	1-1
Oracle TSAM Features	1-2
Oracle TSAM Components	1-3
Oracle TSAM Agent	1-3
Oracle TSAM Manager	1-4
Oracle TSAM Architecture	1-4
Oracle TSAM Concepts	1-5
Call Path Monitoring	1-5
Service Monitoring	1-7
System Server Monitoring	1-7
Transaction Monitoring	1-7
Oracle Tuxedo Application Runtime for CICS and Batch Monitoring	1-7
Policy Monitoring	1-8
Performance Metrics	1-8
Oracle TSAM Use Cases	1-10
Optimizing Application Development	1-11
Understanding Your Applications	1-11
Solving Application Performance Problems	1-12
Improving Application Performance	1-13
Quick Start	1-14
See Also	1-16

Oracle TSAM Agent

Prerequisites	2-1
Oracle TSAM Agent APIs	2-1
tpgetcallinfo	2-2
Local Monitor Server (LMS)	2-2

Oracle Tuxedo Application Runtime for CICS and Batch Configuration.	2-2
Turning Oracle TSAM On and Off	2-3
See Also	2-3

Oracle TSAM Manager

Overview	3-1
Oracle TSAM Data Server.	3-2
Oracle TSAM Console	3-2
Application Server.	3-2
Database.	3-2
Configuring Oracle TSAM Manager.	3-3
See Also	3-3

Introduction to Oracle TSAM

This topic contains the following sections:

- [Overview](#)
- [Oracle TSAM Features](#)
- [Oracle TSAM Components](#)
- [Oracle TSAM Concepts](#)
- [Oracle TSAM Use Cases](#)
- [Quick Start](#)

Overview

Oracle Tuxedo System and Application Monitor (TSAM), is an Oracle Tuxedo add-on product. Tuxedo is widely used by enterprises that develop and use in mission-critical applications. It acts as the infrastructure layer in distributed computing environments. The complexity of Tuxedo and the applications running on top of it makes performance measurement extremely complex.

Oracle TSAM monitors the major performance sensitive areas of a Tuxedo-supported enterprise computing environment. It can be used to monitor *real-time* performance bottlenecks and business data fluctuations, determine service models, and provide notification when pre-defined thresholds are violated.

Oracle TSAM Features

The following is a list of Oracle TSAM features:

- Tracking a Tuxedo system call transmissions. Each monitored call is assigned a unique ID and is propagated along a call path tree. TSAM is able to track calls across multiple machines and domains.
- *Real-time* call path tree tracking of a monitored request is displayed and the performance metrics for each step are available.
- Call pattern summarization based on historical call tracking data.
- Monitoring a particular Tuxedo service, checking its response time, IPC queue length and execution status. The data can be queried using recent or historical data.
- Monitoring CORBA interface metrics
- Gathers Oracle Tuxedo GWTDOMAIN, BRIDGE and GWWS overall throughput, graphically displaying the business data flow curve.
- Tracking transactions with XA API specifications. Displays execution status and time used on each XA call. TSAM helps diagnose global distributed transactions. Correlate transactions across multiple domains in tree style. TSAM supports the transaction monitoring propagation that is if monitoring enabled for the transaction initiator, the whole transaction path will be monitored.
- Monitoring Oracle Tuxedo Application Runtime for CICS and Batch specific metrics (including transactions and terminal data).
- Flexible monitoring controls. Powerful monitoring policy can achieve the exact monitoring result and reduce the performance impact. The sampling can be based on interval, ratio and runtime data. The monitoring can be turned on or off dynamically without restarting application.
- Comprehensive SLA alert configurations based on monitoring metrics. Alert evaluation is based on Tuxedo FML Boolean expression. The event generated by Alert can be post to Tuxedo Event Broker. Some alert type can also drop the staled service request.
- Programming APIs that retrieve metadata packaged in a monitored call. Helps developers make application decisions dynamically.
- Plug-in mechanism for performance metrics collection at the Tuxedo infrastructure level. It provides great integration capability between TSAM and other third-party products.

- Scalable Tuxedo-side server monitoring design to meet small, middle and large Tuxedo runtime environments.
- J2EE based solution. Easy to deploy, configure and use. It is a pure Web-based solution with WEB 2.0 technologies. The TSAM Console can be accessed anywhere using a compatible Web browser.
- Easy-to-understand metrics database schema. The metrics can be used for data mining or further business analysis.

Oracle TSAM Components

Oracle TSAM includes two components:

- [Oracle TSAM Agent](#)
Performs Tuxedo-side data collection, policy enforcement and alert evaluation at run time.
- [Oracle TSAM Manager](#)
Performs data storage, aggregation, computing and representation.

Oracle TSAM Agent

The Oracle TSAM Agent handles all Tuxedo-side back-end logic. It works in conjunction with the Oracle TSAM Manager, and includes the following sub-components:

- **Oracle TSAM Framework:** The framework is the data collection engine. It is an independent layer working between Tuxedo infrastructure and other TSAM components. This module is responsible for run time metrics collection, alert evaluation and monitoring policy enforcement.
- **Oracle TSAM Plug-in:** An extensible mechanism invoked by the Oracle TSAM Framework. The Oracle TSAM Agent provides default plug-ins to send data to the LMS (Local Monitor Server), and then to the Oracle TSAM Manager. The plug-in allows custom plug-in to be hooked to intercept the metrics. The default plug-in communicates with LMS with share memory. Application will not be blocked at metrics collection point.

You can develop your own plug-ins for additional data processing. A customized plug-in can be linked to an existing plug-in chain, or replace the default plug-in.
- **Local Monitor Server (LMS):** The LMS is an Oracle Tuxedo system server. The Oracle TSAM default plug-in sends data to the LMS. The LMS then passes the data to the Oracle

TSAM Manager in HTTP protocol. LMS is required on each Tuxedo machine if the node need to be monitored.

For more information, see [Oracle TSAM Agent](#).

Oracle TSAM Manager

The Oracle TSAM Manager is built on J2EE technology. It includes following components:

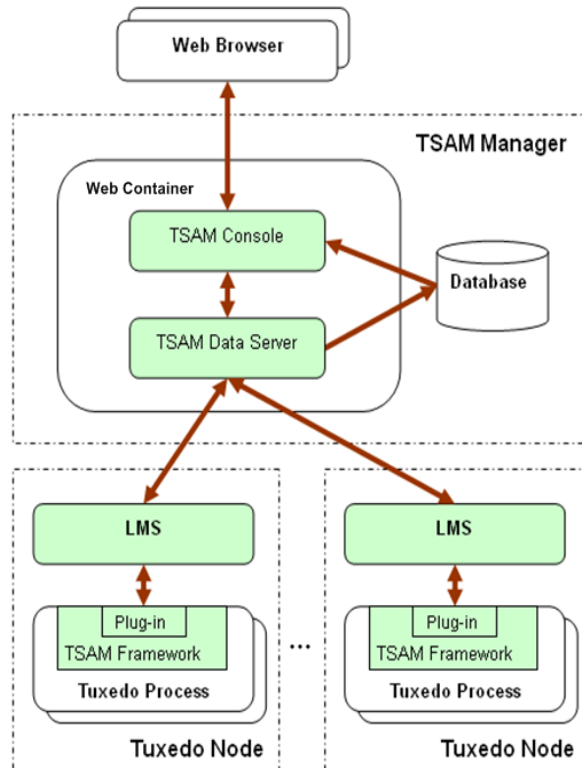
- **Oracle TSAM Data Server:** The data server is responsible for:
 - accepting data from the LMS and store them into database
 - accepting requests from representation layer and does data processing
 - communicating with LMS for configuration instructions.
- **Oracle TSAM Console:** The Oracle TSAM presentation layer. It is a J2EE Web application and can be accessed via a compatible Web browser. After logging on to the Oracle TSAM Console, you have access to full Oracle TSAM functionality.

For more information, see [Oracle TSAM Manager](#).

Oracle TSAM Architecture

[Figure 1-1](#) shows the Oracle TSAM architecture.

Figure 1-1 Oracle TSAM Architecture



Oracle TSAM Concepts

Call Path Monitoring

Tuxedo is typically used by a client program (not necessarily a Tuxedo client process) that calls a service to perform a business computing logic scenario. The service implementation is completely transparent to the caller. This type of middleware transparency provides many benefits for development, deployment, and system administration. However, from a monitoring perspective, it is difficult for the end user or administrator to figure out what happens “behind the scene”. Oracle TSAM call path monitoring helps to alleviate this problem.

Call Path Tree Definition

A simple Tuxedo application call triggers a set of service invocations. The involved services constitute a tree ("call path tree"). A call path tree strictly defines the following factors:

- What type of services are involved to perform the initial service request.
- The service invocation depth (that is, the depth of the call path tree).
- The service invocation sequence. For example, client A calls SVC1. SVC1 calls SVC2 and SVC3.
- Call transportation. The *edge* (how information is sent and received) of a call path tree represents the transportation information from caller to service provider. It could be an IPC queue, BRIDGE connection or DOMAIN connection. The elapsed time used for each transportation is also recorded.
- Call path metrics. Lots of metrics are available during the message propagation within Tuxedo system, such as the message size, execution status, transaction and CPU consumption etc.

Monitoring Initiator

A "monitoring initiator" is a process that "initiates" tracking a call path tree. The process can be a Tuxedo client, application server, client proxy server (WSH/JSH), the Tuxedo domain gateway server or web services proxy serve GWWS. A typical scenario is when a `tpcall/tpacall` is invoked by the monitoring initiator; call path monitoring begins. All the back-end services involved in this call are displayed on the call path tree representation in the Oracle TSAM Console.

Note: Currently only `tpcall/tpacall` can trigger a call path monitoring. Other communication models are not supported.

A Tuxedo application server performs two functions:

- All sub-calls made in the service implementation are a part of the call path tree started by the original monitoring initiator (if the incoming request is already monitored).
- It is a monitoring initiator with calls made in the service routine according to the monitoring policy definition.

Service Monitoring

Service monitoring focuses on Tuxedo service execution status. It does not care about call correlation, as call path monitoring does. Service monitoring can be used with call path monitoring together or performed independently. Tuxedo CORBA is also based on the service infrastructure, so in this release the CORBA interfaces can also be monitored.

System Server Monitoring

Tuxedo has several important system servers: BRIDGE, GWTDOMAIN and GWWS. BRIDGE connects multiple Tuxedo machines within a Tuxedo domain. GWTDOMAIN connects one Tuxedo domain with others. GWWS is the web services gateway. The system server monitoring tracks message throughput, pending sent messages and awaiting reply messages on each network link for BRIDGE and GWTDOMAIN. For GWWS, the web service requests statistics will be collected.

Transaction Monitoring

A critical use of Tuxedo is transaction monitoring. Tuxedo coordinates activities in a distributed transaction with an XA compliant resource manager, such as a database. Oracle TSAM transaction monitoring tracks each XA call triggered in a transaction allowing you to clearly identify where a global distributed transaction is bottle necked. TSAM supports the transaction monitoring capability propagation. That is if the transaction initiator is monitored, all XA calls on the transaction path will be monitored. The propagation supports transaction across domains.

Oracle Tuxedo Application Runtime for CICS and Batch Monitoring

Oracle TSAM allows you to monitor the following Oracle Tuxedo Application Runtime for CICS and Batch components.

- CICS Region representation (status and components).
- CICS Transaction metrics monitoring.
- CICS Terminal related metrics monitoring.
- ARTJES component monitoring

For more information, see [Oracle Tuxedo Application Runtime for CICS and Batch documentation](#).

Policy Monitoring

Oracle TSAM provides a comprehensive policy monitoring mechanism. When the proper policy monitoring settings are created, you can collect the exact metrics needed with minimum application performance impact. You define a policy using the TSAM Web console and apply it to an Oracle Tuxedo application automatically. Oracle TSAM Policy Monitoring has the following characteristics:

- **Monitoring Category.** One monitoring policy can focus on one kind of monitoring, such as call path. It can also cover multiple interested areas.
- **Enable or Disable.** Oracle TSAM monitoring can be dynamically turned on or off. Monitoring policy can be predefined and enabled when the monitoring is needed. All enabled monitoring policy will be applied to Tuxedo applications automatically while application is running. Non-started application will get the policy while it is started.
- **Interval-Based Monitoring.** Monitoring is initiated based on specific time intervals. For example, call path monitoring. An interval-based monitoring policy can specify that the call path is tracked in 60-second intervals.
- **Ratio-Based Monitoring.** Monitoring is initiated by the number of executions. For example, service monitoring. A ratio that is set to 5 indicates that every 5 executed services are monitored. For call path monitoring, a ratio set to 5 indicates that every 5 `tpcall/tpacall` calls are monitored.
- **Runtime Condition Filtering.** TSAM monitoring policy supports some run time filters. Customer can monitor a particular service, a request from specific client and some kind of process type. The filter supports regular expression format.
- **Flexibility to Reduce Monitoring Performance Impact.** Oracle TSAM monitoring control enables you to configure the monitoring policy based on your application size, load and network activity. The monitoring policy can support only alert triggering without raw metrics storage.

Performance Metrics

Oracle TSAM performance metrics are listed as follows:

- **Correlation ID:** A unique identifier that represents a call path tree. It is generated by the monitoring initiator plug-in. It uses the following format:

```
DOMAINID:MASTERHOSTNAME:IPCKEY LMID PROCESSNAME PID TID COUNTER  
TIMESTAMP
```

[Listing 1](#) shows an example of a Correlation ID. The monitored call is started by the program “bankclient” with process **ID 8089** and thread ID 1 on machine “SITE1” on Tuxedo domain “TUXDOM1”. The master is “bjso118” and IPCKEY in TUXCONFIG is “72854”.

Listing 1 Correlation ID Example

```
TUXDOM1:bjso118:72854 SITE1 bankclient 8089 1 99 1259309025
```

- **Service Name:** The name of an Oracle Tuxedo Service.
- **Location:** The set of metrics to identify the process who sends out the performance metrics. It includes information about domain, machine, group and process name etc.
- **IPC Queue Length:** The message number in an IPC queue.
- **IPC Queue ID:** Oracle Tuxedo identifier of an IPC queue.
- **Execution Time:** The time used in an Oracle Tuxedo service or XA call execution in milliseconds.
- **Wait Time:** The time used of a message in the transportation stage.
- **CPU Time:** The CPU time consumed by the service request processing. It only applies to single threaded server.
- **Message Size:** The Oracle Tuxedo message size.
- **Execution Status:** The `tpreturn` service return code. It is defined by the Oracle Tuxedo ATMI interface.
- **Call Flags:** The flags passed to `tpcall/tpacall` in the Oracle Tuxedo ATMI interface.
- **Call Type:** `tpcall`, `tpacall`, or `tpforward`.
- **Elapse Time:** The time elapsed time a call is monitored.
- **GTRID:** Oracle Tuxedo global transaction ID.
- **Pending Message Number:** The number of messages which are delivered to the Oracle Tuxedo network layer and waiting for being sent.

- **Message Throughput:** The total message number and volume accumulated in system server monitoring intervals.
- **Waiting Reply Message Number:** The number of requests in GWTDOMAIN awaiting a reply from the remote domain.
- **XA Code:** The XA call return code in transaction monitoring.
- **XA Name:** The XA call name.
- **GWWS Metrics:** A set of metrics used to measure GWWS throughput, including:
 - Inbound Message Throughput
 - Inbound Message Processing Time
 - Outbound Message Throughput
 - Outbound Message Processing Time
- Oracle Tuxedo Application Runtime for CICS and Batch Metrics
 - Transaction metrics
 - TCP terminal throughput

Oracle TSAM Use Cases

Oracle TSAM is built on top of Oracle Tuxedo and has unique service, call, and transaction tracking capabilities. Enterprise organization usually have many widely distributed services deployed and one client request that requires complex back-end service coordination to perform the processes.

It can be difficult for an administrator to figure out what exactly is happening during these interactions. Oracle TSAM call path monitoring helps to alleviate this problem.

The followings are FAQs will help you to better understand how Oracle TSAM works with your applications:

- [Optimizing Application Development](#)
- [Understanding Your Applications](#)
- [Solving Application Performance Problems](#)
- [Improving Application Performance](#)

Optimizing Application Development

Is the message routing working as expected?

Message routing is an important concept in Tuxedo. It impacts the system's performance, business logic process correctness and application reliability and high availability. Lots of factors can affect Tuxedo message routing, such as:

- Load balance
- Data dependent routing (DDR)
- Transaction Affinity (Oracle RAC support)

In development stage, huge of tests needed to verify your settings take effect. Without TSAM, it is hard to tell the exact request going to the correct service desired. TSAM call path can depict each call path clearly and speed your development process.

Understanding Your Applications

What happens behind a simple call?

Enabling call path monitoring for a Tuxedo client or application server allows you to find out all the information behind a simple `tpcall/tpacall`. The tracking points span multiple machines and multiple domains. You can clearly see the following information in the call path tree:

- The service invocation hierarchy that supports your call
- The transmission cost for each message flow step, from IPC Queue to Network
- The execution status of each service involved
- The call type and call flags of all the intermediate calls
- The waiting time in queue and response time for each service
- The end-to-end response time

What about my services?

Service monitoring enables you to measure your service response time, IPC queue length, and execution status. Service monitoring provides the following information:

- Service Execution Status Summary.

Oracle TSAM tells you how many service executions succeeded or failed recently or during a period of time. Oracle TSAM also computes the average response time. These are important factors in measuring the quality of your services.

- Service Activity Trends.

Oracle TSAM also displays your services activity trends. It tells you what the peak time is and when the services requests are low.

Is my network busy?

Oracle TSAM allows you to monitor the network connection attached to your local domain gateways. You can easily find which link is busy and its data fluctuation trend. You have more in-depth understanding of the business data flow model between departments and organizations.

Who participates in my transaction?

Oracle TSAM monitors the transaction XA calls. Transaction participants are listed on the transaction monitoring page. For a large distributed transaction, a slow branch can result in the entire transaction being slowly completed. Oracle TSAM lets you know who the transaction participants are, and how much time is used during XA calls. The transaction monitoring can also help find the bottleneck in a two phase commit stage if multiple resource managers involved.

Solving Application Performance Problems

Why is the service response time slow recently?

Turn on the call path monitoring for a particular call to investigate the following:

- How much network-side time is used
- Which services are the most time-consuming point in the call path tree
- Is the service routed to a remote machine or a domain
- Is client wait time a reply problem?

My back-end services failed, but I don't know which one.

Turn on call path monitoring. You can find the service execution status for this call.

How many kinds of call paths are in my application?

Turn on call path monitoring using an adequate sampling policy. Oracle TSAM will tell you how many call paths (a “call pattern”) exist in your application.

Why is my global distributed transaction completed slowly?

Turn on Oracle TSAM transaction monitoring. You can see the execution time used by the transaction participants.

I want to correlate local transactions with remote transactions.

Turn on Oracle TSAM transaction monitoring for all involved processes and GWTDOMAIN. The Oracle TSAM Console shows you the transaction mapping between local and remote transactions.

I want to know what is the peak time that my local domain uses resources from the remote domain, and how busy it is.

Use Oracle TSAM system server monitoring on the GWTDOMAIN. Oracle TSAM records the information for you, and shows you the throughput trends.

Can I check program request information?

Turn on call path monitoring with the proper monitoring policy and then use “`tpgetcallinfo`”. The following information is provided.

- The timestamp when the request leaves the caller
- The timestamp when the request comes into to the server IPC queue
- The client IP address (workstation client, GWWS client)

The monitoring initiator process, `tpgetcallinfo()`, can also tell you the total time used.

Improving Application Performance

Are my services too fine grained?

In some cases, too many services supporting a request may add to performance overhead. Use call path tree to investigate. The service number and the tree depth are key analysis factors.

Are my services deployed properly?

Some services are called more frequently than others. Use call path monitoring to gather the information, and re-consider the service deployment. It is best to have the most used services located on the local machine and LAN. Services across domain services should be used carefully.

Do I have too many servers configured?

Oracle TSAM provides a central view of your Tuxedo applications with multiple domain support. Using Oracle TSAM Console allows you to easily see how many domains, machines, servers and services are configured.

I want to be notified when something is wrong with my system

Oracle TSAM provides comprehensive alert configuration based on the metrics collected. The base technology of TSAM alert evaluation is Tuxedo FML Boolean expression, so you can combine complex conditions to compose an alert, such as:

- Report an alert while the execution time greater than 30 seconds and execution is failed.
- Report an alert if heuristic commit happens
- Report an alert if the CPU time consumption is too high in my service

How to drop stale requests?

Some times the service execution is slow and quite some request messages are waiting for a long time in the IPC queue. The client who issues the request might be already got timeout notification, but the service still continually to process the request. Oracle TSAM Alert can allow you to configure an "drop request" action with some metrics. For example, you can drop a request if the waiting time of a message greater an large value. You can also configure an alert with drop request action under the condition that the message number in the request queue exceeding some threshold.

Quick Start

To add Oracle TSAM functionality to an existing Oracle Tuxedo application, do the following steps:

1. Install Oracle TSAM

Install the Oracle TSAM Agent and the Oracle TSAM Manager. For more information, see the [Oracle TSAM Installation Guide](#).

2. Deploy Oracle TSAM Agent and Oracle TSAM Manager

For more information, see the [Oracle TSAM Deployment Guide](#).

3. Find Your Oracle Tuxedo Configuration

Login to the Oracle TSAM Manager Console (for example: **http://localhost:8080/tsam/faces**).

Your Oracle Tuxedo configuration can be found in the Oracle Tuxedo component tree panel. For more information, see the [Oracle TSAM Console User Guide](#).

4. Configure Monitoring Policy

Monitoring policy defines what and how you want to monitor.

For more information, see [Monitoring Policy Management](#) in the “*Oracle TSAM Agent User Guide*.”

For more information, see [Policy Management](#) in “*Oracle TSAM Console User Guide*.”

• Typical Monitoring Policy

Go to Policy > Monitoring Policy page, click "Add" to enter the "Create Policy" page. Input "tsampolicy" in the "Name:" input field.

• Monitor call path initiated from a particular client

- a. In the left panel "Tuxedo Component", select the "Domain" in the drop down lists
- b. In the right panel "Call Path" tab, select the "Enable" checkbox
- c. Select the "Filter" check box with "Workstation Client"
- d. Input a client name in the "Client Name" input box
- e. Click "Add & Enable" button

• Monitor services of a particular server

Similar steps as call path monitoring policy, except you must select options at the "Service" tab. In the "Filter", you can select service from a list or input it manually.

Other monitoring policy set have similar steps. You can group several monitoring categories in one policy.

5. Start to Monitor Tuxedo

Login to TSAM Console, and start to monitor Tuxedo system and application.

Go to Metrics > Call Path to monitor call path.

Go to Metrics > Service to monitor service.

Go to Metrics > Domain Gateway to monitor GWTDOMAIN.

For more information, see the [Oracle TSAM Console User Guide](#).

See Also

- [Oracle TSAM Deployment Guide](#)
- [Oracle TSAM Console User Guide](#)
- [ART Runtime Documentation](#)
- [ART Workbench Documents](#)

Oracle TSAM Agent

This chapter contains the following sections:

- [Prerequisites](#)
- [Oracle TSAM Agent APIs](#)
- [Local Monitor Server \(LMS\)](#)
- [Oracle Tuxedo Application Runtime for CICS and Batch Configuration](#)
- [Turning Oracle TSAM On and Off](#)

Prerequisites

To effectively and correctly use the Oracle TSAM Agent, please note the following prerequisites:

- The system clocks for all monitored Tuxedo machines and the Oracle TSAM Manager are synchronized. A uniform time server is recommended.
- Set each Tuxedo domain to a unique DOMAINID in the UBBCONFIG file.

Oracle TSAM Agent APIs

The following APIs are used in conjunction with Oracle TSAM:

- [tpgetcallinfo](#)

tpgetcallinfo

User applications can take advantage of the monitoring attributes by using `tpgetcallinfo`. `tpgetcallinfo` is designed for call path monitoring. Using `tpgetcallinfo`, allows applications to make dynamic decisions based on application performance metrics.

For more information, see `tpgetcallinfo()` in the *Oracle TSAM Reference Guide*.

Local Monitor Server (LMS)

Local Monitor Server (LMS) is a component of Oracle TSAM Agent. It performs the following tasks:

- Act as a data proxy server between Oracle TSAM agent framework and Oracle TSAM Manager
- Accept policy and alert configuration request from Oracle TSAM Manager and apply to Tuxedo domain
- Send Tuxedo domain configuration to Oracle TSAM Manager
- Other help functionality

For more information, see [LMS](#) in the *Oracle TSAM Reference Guide*.

Oracle Tuxedo Application Runtime for CICS and Batch Configuration

To use Oracle Tuxedo Application Runtime for CICS monitoring, an additional configuration file (`Region-Group.mapping`) is required. The file must be put in the directory specified by the `KIXCONFIG` environment variable. For more information, see [Oracle Tuxedo Application Runtime for CICS Users Guide](#).

Note: The `KIXCONFIG` environment variable is required for all ART applications.

[Listing 2-1](#) shows a `Region-Group.mapping` file format example.

Listing 2-1 Region-Group.mapping File Format Example

```
[region]
name= REG1
```

```
tuxgroups=APPGRP1,APPGRP2
resources_group=resgroup1,resgroup2

[region]
name= REG2
tuxgroups=APPGRP3,APPGRP4
resources_group=resgroup3,resgroup4
```

Each region has a section in this file. The CICS region is called "name". The `tuxgroups` parameter is followed by the Oracle Tuxedo group name. The `resources_group` parameter is the CICS resource group in this region.

For more information, see the [Oracle Tuxedo Application Runtime for CICS Reference Guide](#).

Note: For Oracle Tuxedo Application Runtime for CICS 11.1.1.2.2 running on Linux, the ARTMON server must be configured in order to enable TSAM monitoring.

If you need to monitor Oracle Tuxedo Application Runtime for Batch ARTJES component, set the `JESMONITOR` environment variable to `yes` before you start the LMS server.

Turning Oracle TSAM On and Off

Oracle TSAM Agent provides a `tmadmin` command to turn on/off. If you want to turn off Oracle TSAM temporarily, this command can be used. The format is as follows:

```
changemonitor (chmo) [-m machine] on|off
```

The `-m` parameter specifies the logic machine name where the Oracle TSAM collection is disabled. Without this option, monitoring on all machines is disabled. By default, monitoring is turned on. If monitoring is turned off, all data collection is stopped even if there is a monitoring policy defined.

See Also

- [Oracle TSAM Deployment Guide](#)
- [Oracle TSAM Console User Guide](#)
- [Oracle Tuxedo Application Runtime for CICS and Batch documentation](#)

Oracle TSAM Manager

This chapter contains the following sections:

- [Overview](#)
- [Oracle TSAM Data Server](#)
- [Oracle TSAM Console](#)
- [Application Server](#)
- [Database](#)
- [Configuring Oracle TSAM Manager](#)

Overview

The Oracle TSAM Manager is the data manipulation and representation component of Oracle TSAM. It is a J2EE application. The Oracle TSAM Manager provides the following functionality:

Communicates with the Oracle TSAM Agent for performance metrics, configuration information and other utility messages.

Provides a Web console for Oracle TSAM administration, monitored data presentation and alerts management.

Oracle TSAM Data Server

The Oracle TSAM Data Server is the communication interface to Oracle TSAM. It accepts requests from LMS and metrics query requests from web browser. For each LMS, the URL of the data server must be configured correctly. The format is as follows:

```
CLOPT="-A -- -l host:port/tsam"
```

- Host is the box where the web application deployed.
- Port is the port number of the java server.
- tsam is the Oracle TSAM URL.

Note: From an HTTP perspective, the Oracle TSAM Agent LMS is the HTTP client, and the Oracle TSAM Manager is the HTTP server. If a firewall is deployed between the Oracle TSAM Manager and Tuxedo applications, the firewall must allow the LMS to issue HTTP requests to the Oracle TSAM Manager.

Oracle TSAM Console

Oracle TSAM Console is the Web application which provides a GUI interface for administration and data presentation. For more information, see the [Oracle TSAM Console Users Guide](#).

Application Server

The Oracle TSAM Manager runs in a Java application server. The supported Java application servers include Oracle WebLogic Server and Apache Tomcat.

Database

Oracle TSAM uses a relational database to store the following information:

- Performance metrics collected by the Oracle TSAM Agent
- Tuxedo component information from the Oracle TSAM Agent
- User account information
- Alerts and events

The Oracle TSAM Manager supports Oracle Database and Apache Derby. For more information, see the [Oracle TSAM Deployment Guide](#).

Configuring Oracle TSAM Manager

Oracle TSAM Manager provides some global parameters for tuning purpose. They are available at the Management/Global Parameters page. Please refer the console guide for detail usage.

See Also

- [Oracle TSAM Deployment Guide](#)
- [Oracle TSAM Console Users Guide](#)

See Also