Oracle® Tuxedo

Release Notes

12*c* Release 2 (12.2.2)

April 2016

ORACLE

Oracle Tuxedo Release Notes, 12*c* Release 2 (12.2.2)

Oracle Tuxedo 12*c* Release 2 (12.2.2) Release Notes

Date: April 2016

Table 1 Revision History

Revision Date	Summary of Change
April 2016	12c Release 2 (12.2.2) GA

This document contains release notes for the Oracle Tuxedo 12*c* Release 2 (12.2.2). It includes the following topics:

About This Oracle Tuxedo Release

What's New and Improved

Oracle Tuxedo 12*c* Release 2 (12.2.2) includes the following new major features and enhancements:

Distributed Caching

Distributed caching feature of Tuxedo provides access to a distributed cache to Tuxedo applications. This feature leverages Oracle Coherence as the distributed cache and provides new APIs for cache access. With this feature, one can take advantage of all the benefits that Oracle Coherence has to offer for distributed caching. This feature enables the following use cases:

- Data caching for Tuxedo applications

  When data caching is enabled, one can store Tuxedo typed buffer in the distributed cache. Tuxedo applications, running anywhere in Tuxedo domain, can now retrieve the data from the cache. This offers a new way of sharing data between clients and servers as well as a way to cache frequently accessed data, without having to go to the database every time. Most Tuxedo buffer types are supported.

- Result caching for Tuxedo services

  When result caching is enabled, Tuxedo first checks the cache to see if response for the request buffer exists in the cache. This is done based on a criterion, which consists of configurable fields in the request buffer. If cache hit succeeds, response is returned from the cache and service is not invoked. Service is invoked if a current cache entry does not exist. Once service is invoked, results are placed in to the cache.

For more information, see *Using Oracle Tuxedo Distributed Caching (TDC)*.

Faster and Flexible Startup of Applications

With this feature, one can significantly improve the start-up time of large applications. Servers configured in an application can be started in parallel. The feature enables specifying startup dependencies of boot sequence at the group and server levels.

For more information, see `tmboot(1)`.

Integrating Audit with Oracle Platform Security Services (OPSS)

This feature enables auditing of Tuxedo services based on Oracle Platform Security Services (OPSS) audit component. Based on the configuration, specific events with relevant data are generated, which can eventually be stored on a file system or eventually in the Database to further analysis using BI tools. Straightforward XML based configuration makes it easy to add/change audit policies without any impact to the application.

For more information, see *Integrating Audit with Oracle Platform Security Services (OPSS)* and *Implementing Custom Auditing*.

Integration with Oracle Access Manager (OAM)

This release allows Tuxedo applications to leverage rich security features of Oracle Access Manager. Tuxedo applications can authenticate and authorize using the credentials and resource authorization policies stored in OAM. This feature also allows single-sign-on of mobile and other applications when accessing Tuxedo services using SALT Web services.

Note: OAM integration is not certified on IBM AIX 64/32 bit platforms.

For more information, see, *Setting up OAUTHSVR as the Authentication Server*, and `OAUTHSVR(5)`

Java Server Modules and Dynamic Configuration Reload

Oracle Tuxedo Java server introduces a module entity to provide better isolation among different applications running within a Tuxedo Java server. Tuxedo Java server also provides dynamic configuration reload capability to allow users to add, remove and update applications with no need to restart Tuxedo Java server.

For more information, see *Oracle Tuxedo Java Server*.

XA Transaction Enhancements

Following XA related enhancements are included in this release:

- Logging Last Resource: With Logging Last Resource feature, Tuxedo allows one non-XA resource to participate in an XA global transaction. For more information, see *Logging Last Resource Transaction Optimization*.
- In the event of a Resource Manager failure, Tuxedo servers can be configured to suspend the impacted services, and keep try reconnecting to the RM, and resume the related services when the reconnection to RM succeeds. For more information, see `RM_ERR_THRESHOLD` `UBBCONFIG(5)` `SERVERS` section, and `TA_RM_ERR_INTERVAL` `TM_MIB(5)` `T_SERVER` Class Definition.
- In the event a TMS process goes down, the XA transaction aborts immediately, rather than waiting for transaction time out. Reduces response time.

SNMPv3 Support

In this release, SNMP agent has been upgraded to support SNMPv3. As a result of this upgrade, two important security features are introduced:

- USM (User-based Security Model)

Provides authentication and privacy (encryption) functions and operates at the message level.

- VACM (View-based Access Control Model)

  Determines whether a given principal is allowed access to a particular MIB object to perform specific functions and operates at the PDU level.

For more information, see *Using SNMPv3*.

New Service Management Console

This release of Tuxedo introduces a new console for managing Tuxedo services. The easy-to-use and cool looking UI provides following functionality in this release:

- Tuxedo metadata repository editor: allows to add/edit/delete Tuxedo service definitions to be used by Jolt, SALT or other Tuxedo components.
- Web services (SOAP and REST) configuration: enables Tuxedo services to be accessed as SOAP or REST services and enables Tuxedo applications to access external SOAP/REST Web services.
- Mainframe transaction integrator: enables mainframe transactions to be accessed as Web services (SOAP or REST) or enables mainframe transactions to access external Web services.

A new Tuxedo system server, `TMADMSVR`, is configured in `UBBCONFIG` in order to use this console.

For more information, see `TMADMSVR`, `MTP`, and *Using Oracle Tuxedo Services Console*.

Enhancements to RECORD Buffer Type

This release includes the following enhancements to the RECORD buffer type:

- Support for conversion from RECORD to VIEW/32 and FML/32 and vice-versa
- Support for `REDEFINES` in the RECORD buffer
- Support for RECORD buffer type in Jolt and SALT

For more information, see *RECORD Features*, *RECORD Functions*, *File Formats, Data Descriptions, MIBs, And System Processes Reference*, and *Command Reference*.

MSSQ Notification

With this feature enabled, a Tuxedo request can be sent to a specific MSSQ server instance. This is made possible via a new API `tpadvertisex(3c)`, which allows to advertise a singleton service, a service which can be advertised by one and only one server instance.

With this feature enabled, XA affinity feature can work in MSSQ configurations.

For more information, see *Advertising and Unadvertising Services* and `tpadvertisex(3c)`.

Installer Enhancements

- Console mode installation: Tuxedo installer now supports console install/deinstall mode.
- Clone mode installation: Tuxedo installer also supports clone mode to copy an existing Oracle Tuxedo installation to a different location and update the copied bits to work in the new environment.

For more information, see *Installing the Oracle Tuxedo System*.

Statistical Trace in GWTDOMAIN

This feature gathers statistics for the remote Oracle Tuxedo domain service calls, and flushes the statistics trace to the audit log file in a specific interval time. This feature makes it easier for you to track cross domain activities.

For more information, see `dmconfig(5)`.

TMTRACE Enhancement

This release of Tuxedo allows `tmtrace` to be enabled at the service level. You can now trace one particular service in a server, which may have many services.

For more information, see `tmtrace(5)`and `tmadmin(1)`.

Jolt Enhancements

This release of Tuxedo Jolt includes `tmtrace` for Jolt clients, providing workstation client equivalent tracing. This release also includes ECID propagation via Jolt clients, if running in WebLogic Server.

For more information, see `tmtrace(5)`and `tmadmin(1)`.

Deprecated Functionality

Jolt Repository

Jolt Repository is deprecated and removed in this release. All service definitions stored in Jolt repository can be loaded in the Tuxedo metadata repository using the bulk loader tool.

The original Jolt repository server `JREPSVR` is also deprecated, and all services that `JREPSVR` provided are now provided by `TMMETADATA`. If `tmloadcf` detects presence of `JREPSR` in `UBBCONFIG`, it automatically removes `JREPSVR` and adds `TMETADATA` if not already configured.

Using one repository (Tuxedo metadata repository), and one server (`TMMETADATA`), improve operational effectiveness and reduces the risk of service definitions getting out of sync.

For more information, see *Using Oracle Jolt* and *Managing The Oracle Tuxedo Service Metadata Repository*.

Oracle Tuxedo Administration Console

Oracle Tuxedo Administration Console is deprecated and removed in this release.

Upgrade Considerations

You need to rebuild all application clients and servers that you want to execute on a system upgraded to Oracle Tuxedo 12c Release 2 (12.2.2) from a previous release of Oracle Tuxedo.

Some Oracle Tuxedo functions are delivered in object files in Oracle Tuxedo release 12.1.3 or below. In this release of Oracle Tuxedo, these object files are packaged in libraries. When rebuilding the corresponding executables, you need to link with the related libraries instead of the object files.

For complete information on upgrading to Oracle Tuxedo 12*c* Release 2 (12.2.2), see *Upgrading the Oracle Tuxedo System to 12c Release 2 (12.2.2)* in Installing the Oracle Tuxedo System.

Supported Platforms

Oracle Tuxedo software runs on the platforms listed in *Oracle Tuxedo 12c Release 2 (12.2.2) Platform Data Sheets*. Oracle has certified these platforms for development and production use with the Oracle Tuxedo 12*c* Release 2 (12.2.2) product. Oracle can provide customer support only for these platforms.

Note: Although Oracle has attempted to implement the Oracle Tuxedo software in a manner that conforms to industry-standards, it is not feasible for Oracle to certify its use with all third-party databases, ORBs, and other products.

Additional software ports and certifications will be made available after the initial release of 12*c* Release 2 (12.2.2). For information regarding subsequent ports and certifications, please refer to the Platform Support information on the Oracle Web site at the following link: *Oracle Tuxedo 12c Release 2 (12.2.2) Platform Data Sheets*.

Note: The required stack size for an Oracle Tuxedo server thread has slightly increased; you may need to adjust the thread stack size accordingly in case of stack overflow issues.