Oracle® Tuxedo

Upgrading to Oracle Tuxedo 22c Release 1 (22.1.0.0.0 and 22.1.1.0.0) Technical Guidelines

22c Release 1 (22.1.0.0.0 and 22.1.1.0.0)

August 2025

Table of Contents

Upgrading to Oracle Tuxedo 22*c* Release 1 Technical Guidelines

This document introduces new features and enhancements accumulated in Oracle Tuxedo 22*c* Release 1 (22.1.0.0.0 and 22.1.1.0.0) from Oracle Tuxedo 6.5. It provides guidelines for upgrading applications based on Oracle Tuxedo lower releases to Oracle Tuxedo 22*c* Release 1. It also provides an overview of Oracle Tuxedo 22*c* Release 1 integration with other Oracle products.

- Release Levels

- Platform Support and 64-bit Migration

- Security Enforcement

- Compatibility

- Interoperability

- Oracle Tuxedo 22c Release 1 Upgrade

- Performance Impact and Memory Usage Increase

Release Levels

This document discusses Oracle Tuxedo 22.1.0.0.0 and Oracle Tuxedo 21.1.1.0.0. The only difference between these releases is that Oracle Tuxedo 22.1.1.0.0 adds support for TLS 1.3. (On some operating systems, the underlying TLS provider requires the presence of OS libraries that are not available on some supported versions of the OS. This necessitates the use of a separate release to provide TLS 1.3 support.)

All information in this document applies to both Oracle Tuxedo 21.1.0.0.0 and to Oracle Tuxedo 21.1.1.0.0 unless specifically noted.

Platform Support and 64-bit Migration

Some hardware platforms have become obsolete between the release of Oracle Tuxedo 12.2.2 and the release of Oracle Tuxedo 22c. Also, Oracle Tuxedo 22c provides only 64-bit binaries (with the exception that 32-bit binaries are still available for Microsoft Windows client-only functionality.)

Some customers will need to port their applications to a different hardware platform or will need to port their 32-bit application to 64-bit Tuxedo.

Resources to help with application porting are available on the internet. These are some key points to remember:

- The effort involved in porting an application will greatly depend on how well the application was originally coded.
- If an application includes different hardware platforms or includes both 32-bit nodes and 64-bit nodes, ensure that XDR encoding is turned on by specifying different machine types where needed. In the UBBCONFIG file (but not in other configuration files), the default is not to perform XDR encoding when the *MACHINES section does not specify TYPE for a pair of nodes.
- Pay close attention to any variables of type int. On 32-bit platforms using an int where a long should have been used will work, but this will not work when sizeof(long) > sizeof(int).
- If moving an application to a platform with different endianness, make sure that the code does not depend on the byte order of numeric values.
- Some C compilers allow calling an external function even if no function prototype is defined for that function. The compiler will output a warning, and the function return type will default to int. If the actual function return type is a pointer, then the

pointer value will be truncated when a 32-bit application is ported to 64-bits. Problems can also happen if the actual function return type is a long.

- If porting to 64-bit Windows with Microsoft Visual Studio, remember that sizeof(long) is 4 on this platform even though it is a 64-bit platform.

Security Enforcement

Oracle Tuxedo 22c enforces a secure Tuxedo deployment by default. Further details are available at https://docs.oracle.com/en/database/oracle/tuxedo/22/oxtrn/whats-new.html#GUID-40438DF1-68E0-441E-9F1D-DA9345B1F125 .

Customers upgrading to Oracle Tuxedo 22c are encouraged to choose strong security parameters. Customers are still allowed to use weaker security, but must set environment variables or configuration file parameters to explicitly acknowledge any such choices.

Compatibility

Applications built with Oracle Tuxedo lower releases are upwardly compatible with Oracle Tuxedo $22c$ on both the configuration file level and source code level. You do not need to change source code or configuration source files unless a new feature is required.

If you are using Tuxedo on Exalogic platform, the option RDMA supported in Oracle Tuxedo 12c Release 1 (12.1.1) and Oracle Tuxedo 11g Release 1 (11.1.1.3.0) is not supported in Oracle Tuxedo 12c Release 2 (12.1.3) and later releases

In many cases, Oracle Tuxedo 12.2.2 executables are binary compatible with Oracle Tuxedo 22c. Exceptions where application code does need to be recompiled are as follows:

- If the application is being upgraded from Tuxedo 12.1.3 or earlier.
- If the application is being moved from 32-bit Tuxedo to 64-bit Tuxedo, or if moving to a different hardware platform.
- If the compiler used to compile the application with Tuxedo 12.2.2 is not binary compatible with the compiler used with Tuxedo 22c.

The detailed platform and compiler information can be found at https://docs.oracle.com/en/database/oracle/tuxedo/22/otxig/oracle-tuxedo-22c-release-22.1.0.0.0-platform-data-sheets.html for Release 22.1.0.0.0 and at https://docs.oracle.com/en/database/oracle/tuxedo/22/otxig/oracle-tuxedo-22c-release-22.1.1.0.0-platform-data-sheets.html for Release 22.1.1.0.0.

All existing binary configuration files need to be regenerated on Oracle Tuxedo $22c$.

There should be no unfinished transactions in the TLOG or DMTLOG when upgrading to Oracle Tuxedo $22c$.The TLOG and DMTLOG need to be regenerated.

For some APIs default behavior in Oracle Tuxedo $12c$ Release 2 (12.2.2) and later releases has changed to meet new requirements. Wherever there is a change in default behavior, switches are introduced to revert back to the behavior in earlier Oracle Tuxedo versions. The change in default behavior is listed below:

- TM_CBL_IGNORE_CONTEXT

  This switch applies to COBOL applications only. CONTEXT-FLAG is introduced in Oracle Tuxedo 8.1, which allows Oracle Tuxedo 6.5 COBOL programs to call TPINITIALIZE fail with TPEINVAL.

  If it is set to "Y", existing Oracle Tuxedo 6.5 COBOL programs will run correctly in Oracle Tuxedo $22c$. The consequence is they cannot get multiple-context feature. If you decide to use multiple-context in an Oracle Tuxedo $22c$ COBOL program, this should be disabled.

- `TM_TUX65_COMPATIBILITY` (introduced in Oracle Tuxedo 8.0)

  This switch applies to `$APPDIR` directory access permission. In Oracle Tuxedo 6.5, if the Oracle Tuxedo security configuration is `NONE` (even though the Oracle Tuxedo administrator has no write access to `$APPDIR`), they can generate the `BDMCONFIG` file using `tmloadcf`, even with `CMDTUX_CAT:1278` reported. In Oracle Tuxedo 8.0 and later, `CMDTUX_CAT:1278` and `CMDTUX_CAT:1275` are reported, `tmloadcf` fails.

  This may address some requirements where the Oracle Tuxedo administrator has no Oracle Tuxedo application binary access to `$APPDIR`. If `TM_TUX65_COMPATIBILITY` is set to `1`, Oracle Tuxedo 6.5 behavior is restored.

- `TPNOTRAN_PASS`

  This switch applies to transactional applications that need to maintain the same behavior of Oracle Tuxedo 8.1 inOracle Tuxedo 22c for the case where the current transaction fails but the native client is still trying to send requests via `tpcall` with the `TPNOTRAN` flag.

  In Oracle Tuxedo 8.1, if a transaction has failed or timed out, for native client, `tpcall()` or `tpacall()` with `TPNOTRAN` flag can still send a request message to a server. However, from Oracle Tuxedo 9.0 onwards, only `tpacall()` with `TPNOBLOCK|TPNOTRAN|TPNOREPLY` flags can succeed. Otherwise, the `tpcall` is not able to send any request to the server.

  With this environment variable set to "`YES`", the native client can send `tpcall()` requests with `TPNOTRAN` flag to server when current transaction has failed or timed out.

- `GWTDOMAIN` and `TMNOTHREADS`

  This switch applies to enhance performance for single thread applications.

  From Oracle Tuxedo 9.0 onwards, `GWTDOMAIN` works in multi-thread mode by default. Setting `TMNOTHEADS=Y` to `GWTDOMAIN` server may cause undefined behavior. Either unset this environment variable for `GWTDOMAIN`, or use "`-s`" `CLOPT` for `GWTDOMAIN` in the `UBBCONFIG` file so that `GWTDOMAIN` can operate in simple thread mode.

Interoperability

Interoperability permits Oracle Tuxedo 22c to communicate over a network connection with Oracle Tuxedo 22c or earlier releases. Oracle Tuxedo supports intra-domain and inter-domain interoperability.

Intra-Domain Interoperability

Oracle Tuxedo 22c can coexist in the same domain with Oracle Tuxedo 12cR2 (12.2.2 or 12.1.3), 12cR1, 11gR1, 10gR3, 10.0, 9.1, and 9.0. In this environment, the propagation of transaction context (transactional state information) and security context (user identity) between application clients and servers is fully supported. Administration is fully supported in this environment.

Note: Tuxedo 12cR2 (12.2.2) and later releases take TLSv1.2 by default if SSL is enabled. If a Tuxedo 22c master connects to a Tuxedo 12.1.3 or earlier release node in MP mode, `tlisten` must be started on the master before running the `tmboot` command.

From Oracle Tuxedo 11g Release 1 (11.1.1.1.0), service name length is increased from 15 to 127. Resource name and remote service name length have also increased from 15 to 127 in domain configuration (for TDomain only). You can also configure a local service name or a remote service with a name less than or equal to 127 characters for `TDOMAIN`. Note the following:

- Long service names are not permitted when Oracle Tuxedo 10gR3 or earlier coexists in, and joins the same Oracle Tuxedo 22c domain.

- In the `UBBCONFIG` file, the value of the `AUTHSVC` keyword in the `*RESOURCE` section and the `SVCNM` keyword in the `*SERVICES` section are not allowed to use long service names when Oracle Tuxedo 10gR3 or earlier software coexists in the same Oracle Tuxedo 22c domain. If long service names are used, the earlier Oracle Tuxedo release site will not boot.

- Oracle Tuxedo 22c application servers with one or more services using long service names will not boot when Oracle Tuxedo 10gR3 or earlier coexists in same the Oracle Tuxedo domain.

- Any dynamic addition of services with long names fails when Oracle Tuxedo 10gR3 or earlier coexists in the same Oracle Tuxedo domain.

- Oracle Tuxedo 10gR3 or earlier software is not allowed join a domain if one or more services with long names has already booted in the current Oracle Tuxedo domain.

- You cannot configure a long local resource and remote service name for `TDOMAIN` gateway servers that are deployed in running Oracle Tuxedo 10gR3 or earlier in the same Oracle Tuxedo domain.

- If users specify XPath in `ROUTING` section, Oracle Tuxedo 11g or earlier release sites will not be booted.

- In MP mode, when the following features are enabled, another node fails to be booted up if its Tuxedo version is lower than Tuxedo 12gR1 or if the other node is running on the platforms that those features do not support.

    - XA Affinity

    - Common XID

    - Single Group Multiple Branches (SGMB)

    - FAN Integration

    - Direct Cross Domain Communication Leveraging RDMA

    The following features are enabled for Tuxedo12gR1 RP073 or later if the non-master node is running on IBM AIX (64-bit), HP-UX (64-bit), Oracle Solaris(64-bit) on SPARC, Linux x86-64, and Linux x86:

    - XA Affinity

    - Common XID

    - Single Group Multiple Branches (SGMB)

    - FAN Integration

    Note: When the option `EECS` (in `OPTIONS` of `UBBCONFIG` `*RESOURCES` section) is specified, the following four features (XA Affinity, Common XID, Single Group Multiple Branches, and FAN Integration) are enabled by default.

Limitations/Actions

- The master machine (and master backup machine if so configured) must run an Oracle Tuxedo higher release.

- It must be specified in `servopts` of server (e.g., `BRIDGE` and application servers), to interoperate with Oracle Tuxedo 6.5. For more information, see *Oracle Tuxedo File Formats, Data Descriptions, MIBs, and System Processes Reference*.

Inter-Domain Interoperability

Inter-domain capabilities (available through a pair of communicating TDomain processes), are limited to the capabilities available to the TDomain process running in the earlier Oracle release.

In all of these scenarios, administration, transaction context propagation, and security context propagation between domains is fully supported.

Oracle Tuxedo 22*c* supports inter-domain interoperability with Oracle Tuxedo 12.2.2, 12.1.3, 12.1.1, 11.1.1.x, 10.3, 10.0, 9.1, 9.0, and 8.1 CORBA domains. This capability includes the ability to advertise CORBA C++ factories across domain boundaries.

Note: Tuxedo 22c takes TLSv1.2 by default if SSL is enabled. If a Tuxedo 22c domain connects to Tuxedo domain that is earlier than Tuxedo 12.1.3 RP070, you must explicitly specify `TLSversion =  TLSv1.0` in the `DMCONFIG` file `*DM_TDOMAIN` section (see `DMCONGIG(5)`) for every `GWTDOMAIN` connection.

Limitations/Actions

- The inter-domain capabilities are limited to the capabilities available in lower release Oracle Tuxedo domains.

Client-Server Interoperability

The capabilities available to a client for a particular client-server pair depend on the release of both the application client and the server application. For example, if you have an Oracle Tuxedo 22c ATMI client interoperating with an Oracle Tuxedo 11.1.1.1.0 server application, only Oracle Tuxedo 11.1.1.1.0 functionality is available to the client.

Client/Server Affinity Interoperability

Oracle Tuxedo Client/Server Affinity interoperability is supported as follows:

- The Client/Server Affinity feature does not work with Oracle Tuxedo 10.3 or earlier native clients; however, it does work with older /WS or Jolt client versions connecting to Oracle Tuxedo 11.1.1.1.0 and later.

  - MP mode

If Client/Server Affinity is configured in the `UBBCONFIG` file and there are non-master nodes that use Oracle Tuxedo 10.3 or earlier, the master node and only nodes installed with Oracle Tuxedo 11.1.1.1.0 or later are affected.

  - Domain mode

To use Client/Server Affinity in a multiple-domain session, Oracle Tuxedo 11.1.1.1.0 or later must be installed on all domains involved in the session.

Feature-Specific Interoperability

The Oracle Tuxedo $22c$ feature-specific interoperability is supported as follows:

- XA Affinity for Transactions

  In an MP domain, the master node is Oracle Tuxedo $22c$ with "`XA_AFFINITY`" enabled and a non-master node is prior to Tuxedo 12.1.1 RP073, or is going to run on a non-supported platform. In this case, the non-master node will fail to boot.

- Common XID

  In an MP domain, the master node is Oracle Tuxedo $22c$ with `COMMON_XID` enabled while and a non-master node is prior to Tuxedo 12.1.1 RP073, or is going to run on a non-supported platform. In this case, the non-master node will fail to boot.ingle Group Multiple Branches (SGMB)

  In an MP domain, the master node is Oracle Tuxedo $22c$ with `SGMB` enabled and a non-master node is prior to Tuxedo 12.1.1 RP073, or is going to run on a non-supported platform. In this case, the non-master node will fail to boot.

- FAN Integration

  In an MP domain, the master node is Oracle Tuxedo $22c$ with `FAN` enabled and a non-master node is prior to Tuxedo 12.1.1 RP073, or is going to run on a non-supported platform. In this case, the non-master node will fail to boot..

- Direct Cross Domain Communication Leveraging RDMA

  In an MP domain, the master node is Oracle Tuxedo $22c$ with "Direct Cross Domain Communication Leveraging RDMA" enabled and a non-master node is of prior release of Tuxedo 12.1.3, or is going to run on a non-supported platform. In this case, the non-master node will fail to boot.

- TLSv1.2 Support

Earlier Tuxedo releases only supported TLSv1.0. Tuxedo 12.2.2 and Tuxedo 22.1.0.0.0 support TLS1.2/1.1/1.0 and use TLSv1.2 by default. (Tuxedo 22.1.1.0.0 also supports TLSv1.3.) Tuxedo 12.2.2 and later releases either self-adapt or use TLS1.2 on the SSL client side to inter-operate with a component that acts as an SSL server on top of earlier Tuxedo release(s). You must explicitly specify TLS1.0 on the SSL client side (including GWTDOMAIN, COBRA client, GWWS outbound HTTPS), for connection to a Tuxedo 12.1.3 RP070 or earlier release.

Interoperability With Other Oracle Products

WebLogic Server

Oracle Tuxedo 22*c* can interoperate WebLogic Server releases. There are three tools to support it: Jolt for WebLogic Server, WebLogic Tuxedo Connector and Oracle Tuxedo JCA adaptor.

- Jolt for WebLogic Server:

  Oracle Tuxedo 22*c* Jolt client can interoperate with Oracle Tuxedo ATMI servers via Jolt 1.2 and later releases.

  Jolt for WebLogic Server can provide unidirectional communication from WebLogic Server to Oracle Tuxedo 22*c*. It provides a Java-based client API that manages requests to Oracle Tuxedo services via a Jolt Service Listener (JSL) running on the Oracle Tuxedo server. The Jolt API is embedded within the WebLogic API and is accessible from a servlet or any other Oracle WebLogic application.

- WebLogic Tuxedo Connector

  The WebLogic Tuxedo Connector (WTC) is included with WebLogic Server. The WTC gateway supports the TDomain gateway protocol, working with Oracle Tuxedo TDomain gateway. It enables bidirectional interoperability between the WebLogic Server and Oracle Tuxedo ATMI and CORBA environments.

  For more information, see *Oracle Tuxedo Interoperability*.

- Oracle Tuxedo JCA Adapter

  The Oracle Tuxedo JCA Adapter is a JCA-based resource adapter that provides bi-directional service invocation between JCA 1.5 compliant application servers and the Oracle Tuxedo system. Oracle Tuxedo JCA Adapter supports global and local transactions conforming to JCA transaction standards. It supports connection management, transaction infestation, identity propagation, and link-level security. Oracle Tuxedo JCA Adapter (with RP008) is certified with Oracle WebLogic Server 12.1.3.

For more information, see *Oracle Tuxedo JCA Adapter documentation*.

Oracle Service Bus

Oracle Tuxedo 22*c* can interoperate with OSB 10gR3 (10.3), OSB 11.1.1.3.0, OSB 11.1.1.5.0, and OSB 12.1.3.0.0.

From Oracle Service Bus 10.3, Oracle Tuxedo Transport provides sufficient configuration of WTC artifacts from the OSB console instead of configuring WTC artifacts in WebLogic Server Administration Console. This release also supports JATMI queuing capabilities.

For more information, see *Oracle Service Bus Interoperability Solution for Tuxedo*.

Interoperability with Third-Party ORBs

Bootstrapping an Oracle Tuxedo CORBA domain establishes communication between a CORBA application client and the domain. Two bootstrapping mechanisms are available:

- The Oracle mechanism using the Bootstrap object

- The CORBA Interoperable Naming Service (INS) bootstrapping mechanism specified by the OMG

Support for INS was added in Oracle Tuxedo release 8.0. With the addition of INS, third-party ORBs that use INS are able to interoperate with the Oracle Tuxedo CORBA server ORB.

Note: The Oracle Tuxedo CORBA client environmental objects continue to be supported in the current Oracle Tuxedo release, just as they were supported in Oracle Tuxedo 8.1, 9.0, 9.1, 10.0, 10.3, 11.1.1.x, and 12.x.

A CORBA application client uses the Oracle Tuxedo Bootstrap object or the INS bootstrapping mechanism to obtain references to the objects in an Oracle Tuxedo CORBA domain. Oracle client ORBs use the Oracle mechanism, and third-party client ORBs use the CORBA INS mechanism. For more information about bootstrapping an Oracle Tuxedo domain, see *Oracle Tuxedo CORBA Programming Reference*.

A JAVA client can leverage JAVA naming mechanism to get initial context and look up Oracle Tuxedo COBRA service, if `IIOP`/`SSLIOP` (IIOP over SSL) is supported by third-party vendor, for instance, WebLogic thin client and/or full client.

Oracle Tuxedo 22*c* Release 1 (21.1.x) Upgrade

Two kinds of procedures are provided to upgrade from an Oracle Tuxedo lower release to Oracle Tuxedo 22*c*:

- Simple upgrade

- Hot upgrade

Simple upgrade is for the configuration that the whole Oracle Tuxedo domain can be shutdown during upgrade. If the whole Oracle Tuxedo lower release domain still needs to serve as normal during upgrade, hot upgrade can be used, where Oracle Tuxedo 22*c* can be added without interrupting the service operation, in order to do this, the original Oracle Tuxedo domain need to be one multiple-host domain with backup master node configured.

The following sections describe the steps to upgrade to Oracle Tuxedo 22*c* Release from a lower Oracle Tuxedo release. In the text we take upgrading from Oracle Tuxedo 8.1 to Oracle Tuxedo 22*c* as example. Actually, the same steps are generally applicable when upgrading from a lower Oracle Tuxedo release to a higher release.

The procedures for upgrading to Oracle Tuxedo Release 22c are quite similar to the procedures for upgrading to Oracle Tuxedo Release 12.2.2, and further information on the upgrade procedure is available at *Upgrading the Oracle Tuxedo System to 12c Release 2 (12.2.2)*.
The following sections describe how an administrator can upgrade an Oracle Tuxedo 8.1 domain to Oracle Tuxedo 22*c*:

- Simple Upgrade

- Hot Upgrade

Simple Upgrade

To perform a simple upgrade, do the following steps:

1. Shutdown the Oracle Tuxedo 8.1 domain.

2. Backup required files in the Oracle Tuxedo 8.1 domain.

   There are certain files (e.g., RM file, `CATLOG` configuration file, Jolt configuration file, etc.), that you need to back up prior to the installation of Oracle Tuxedo 22*c* (22.1.x) and then restore after the installation is complete.

3. Get Oracle Tuxedo configuration files in Oracle Tuxedo 8.1 domain.

   The text version Oracle Tuxedo 8.1 configuration files are required in Oracle Tuxedo 22*c* . You can get them from an earlier backup or generated by commands (e.g., `tmunloadcf`).

4. Backup binary version configuration.

   This is just for backup purpose.

5. Prepare machines and required software.

   If the original Oracle Tuxedo 8.1 machine is qualified for Oracle Tuxedo 22*c* installation, which mainly addresses the operating system and compiler installed, just uninstall Oracle Tuxedo 8.1. If the machine is a Windows system, the relevant Oracle Tuxedo windows services should be stopped before the uninstall. Before installing the Oracle Tuxedo 22*c* software, make sure that the required versions of the operating system and the compiler have been installed. For operating system and compiler requirements for each supported platform, see *Oracle Tuxedo 22c Release (22.1.0.0.0) Platform Data Sheets* at https://docs.oracle.com/en/database/oracle/tuxedo/22/otxig/oracle-tuxedo-22c-release-22.1.0.0.0-platform-data-sheets.html or Oracle Tuxedo 22c Release 2 (22.1.1.0.0) Platform Data Sheets at https://docs.oracle.com/en/database/oracle/tuxedo/22/otxig/oracle-tuxedo-22c-release-22.1.1.0.0-platform-data-sheets.html

6. Install Oracle Tuxedo 22*c*

For multiple-host Oracle Tuxedo 8.1 domains, Oracle Tuxedo 22*c* must at least be installed on the master and backup node. Other nodes can be upgraded to Oracle

7. Tuxedo 22*c* at the same time, but they can also co-exist with Oracle Tuxedo 22*c* and upgrade later.Reload Oracle Tuxedo 22*c*configuration files.

The original text version Oracle Tuxedo 8.1 configuration files, such as `UBBCONFIG`, `DMCONFIG`, device files for /Q and transaction log, should be regenerated on the master machine. The configuration might have some difference. Text version files should be updated first before regeneration. Possible changes include:

– `TUXCONFIG` and `TUXDIR` need to be changed if they are different from the original setting.

– Upgrade security settings to the latest recommendations and/or set environment variables to acknowledge any insecure settings. Refer to the "Security Enforcement" section earlier in this document for more information.If you use Event Trigger in Oracle TSAM Plus 12.2.2 (Oracle TSAM supports Oracle Tuxedo 8.1/9.1/10.0/10.3), in `UBBCONFIG`, `MAXSPDATA` must be set. Its value should be bigger than `(MAXQUEUES * 257 * 2 + 8224)` or `(MAXSERVERS * 257 * 2 + 8224)` if `MAXQUEUES` is not defined in the `UBBCONFIG` file.

8. Recompile and relink all your Oracle Tuxedo application code using Oracle Tuxedo 22*c* build utilities.

9. Reboot the Oracle Tuxedo 22*c* domain.

10. Shutdown and upgrade other nodes in your multiple-host configuration.

This is the same as the above steps; you could also upgrade all machines before rebooting the domain.

Hot Upgrade

1. Shutdown the backup master machine.

2. Backup required files in Oracle Tuxedo 8.1 domain.

This is same as simple upgrade, you can get the backup files from backup master node.

3. Prepare machines and required software.

Same as simple upgrade, select either the original backup master machine or one new machine meeting the software requirements of Oracle Tuxedo 22*c*.

4. Install Oracle Tuxedo 22*c* on the backup master machine.

5. Generate upgrade indicator files.

Generate one empty file `UPGRADE` in the `$APPDIR` directory of master node and backup master node.

6. Recompile and relink all your Oracle Tuxedo application code on backup master machine using Oracle Tuxedo 22*c* build utilities.

7. Boot up the backup master Oracle Tuxedo 22*c* node from the master node.

8. Migrate the master machine from Oracle Tuxedo 8.1 to the new Oracle Tuxedo 22*c* node.

9. Shutdown the acting backup master node running Oracle Tuxedo 8.1.

The original text version Oracle Tuxedo 8.1 configuration files should be regenerated on the master machine.

Note: Configuration is slightly different. Before regeneration, you have to change the text version files accordingly first.

10. Install Oracle Tuxedo 22*c* on the acting backup master machine.

    This is similar to Oracle Tuxedo 22*c* installation on the original master backup node. The same software requirements should be met.

11. Recompile and relink all your Oracle Tuxedo application code on the acting backup master machine using Oracle Tuxedo 22*c* build utilities.

12. Remove upgrade indicator files.

13. Boot up the acting backup master Oracle Tuxedo 22*c* node from the master node.

14. Shutdown and upgrade other nodes in your multiple-host configuration.

This is same as simple upgrade.

Performance Impact and Memory Usage Increase

You can disable some of the features if not needed to get better performance.

- If multi-threading is not needed, you can set the `TMNOTHREADS=Y` environment variable to disable multi-threading in an Oracle Tuxedo 22*c* application.

- If distributed transactions and Plug-in Security mechanisms are not required in your applications, you can define `OPTIONS NO_XA`, `NO_AA` in the `UBBCONFIG` file to disable these two features for better performance.

Compared with Oracle Tuxedo 8.1/9.0/9.1/10.0/10.3/11.1.1.x/12.1.1/12.1.3/12.2.2, the memory usage of each server is similar. Compared with even older releases the increase might be slightly bigger, but still should not be a major concern. Compared with Oracle Tuxedo 12.1.1, the existing

`EXALOGIC` option and/or the new option of `EECS` will enable Exalogic-related features or SuperCluster-related features by default, which may consume more memory.

Several factors contributed to the memory increase, such as OS library size, Oracle Tuxedo library size and the scale of Oracle Tuxedo configuration. In Solaris 9, compared with Oracle Tuxedo 6.5, the increase might be at least is 6M for each server. It is recommended to run tests using the actual configuration to verify how much memory is needed.

Appendix A: What's New in Oracle Tuxedo 22*c* (22.1.0.0.0 and 22.1.1.0.0)

This section introduces major Oracle Tuxedo features of which are accumulated from Oracle Tuxedo 7.1 through 22.1.1.0.0 release.

- Oracle Tuxedo 7.1
- Oracle Tuxedo 8.0
- Oracle Tuxedo 8.1
- Oracle Tuxedo 9.0
- Oracle Tuxedo 9.1
- Oracle Tuxedo 10.0
- Oracle Tuxedo 10.3
- Oracle Tuxedo 11gR1 (11.1.1.1.0)
- Oracle Tuxedo 11gR1 (11.1.1.2.0)
- Oracle Tuxedo 11gR1 (11.1.1.3.0)
- Oracle Tuxedo 12.1.1
- Oracle Tuxedo 12.1.3
- Oracle Tuxedo 12.2.2
- Oracle Tuxedo 21.1.0.0.0
- Oracle Tuxedo 21.1.1.0.0

Oracle Tuxedo 7.1

Security for E-commerce

- Support for Security Systems Provided by Third-party Vendors.

  With the availability of the plug-in architecture, Oracle Tuxedo customers who want to customize authentication, authorization, auditing, or public key security can contact a third-party security vendor to procure the appropriate plug-ins (code modules).
  Alternatively, customers can use the Oracle Tuxedo default plug-ins provided for each of these security capabilities.

- Message-based Digital Signature.

  Message-based digital signatures enhance Oracle Tuxedo security by allowing a message originator to prove its identity, and binding that proof to a specific message buffer. The scope of protection is end-to-end: a message buffer is protected from the time it leaves the originating process until the time it is received by the destination process.

- Message-based Encryption.

  Message-based encryption keeps data private, which is essential for most applications that transport data over the Internet. The scope of protection is end-to-end: a message buffer is protected from the time it leaves the originating process until the time it is received by the destination process.

Multithreading and Multicontexting

In this release, the Oracle Tuxedo system provides new ATMI functions that enable programmers to write multithreaded and/or multicontexted clients and servers. In a multithreaded application, a single client can initiate multiple, concurrent (parallel) calls. In a multicontexted application, a client can establish multiple associations with multiple domains. On the server side, multithreading provides the ability to process multiple service requests simultaneously within one server process.

Embedded FML (EFML) Support

This release provides the ability to embed FML records within other FML records. By using this functionality, you can manage a single EFML buffer as a single structure rather than a series of discrete data fields.

XML Buffer Support

XML buffers enable Oracle Tuxedo applications to use XML for exchanging data within and between applications. Oracle Tuxedo applications can send and receive simple XML buffers, and route those buffers to the appropriate servers. All logic for dealing with XML documents, including parsing, resides in the application.

IPC Tool

The IPC tool allows system administrators to determine and clean up the IPC resources used by the Oracle Tuxedo system after an unexpected shutdown.

Domains MIB

Provides programmatic and dynamic administration of the Domains configuration.

Jolt and Web-based Administration Console

Bundles Oracle Jolt and the Web-based Administration Console with the Oracle Tuxedo software.

Oracle Tuxedo 8.0

CORBA integration

Bundles CORBA ORB from WebLogic Enterprise 5.1 with Oracle Tuxedo software and provide more functionality.

Support for Multithreaded CORBA C++ Applications

Release 8.0 of Oracle Tuxedo supports multithreaded C++ client and server applications. While there are many advantages to writing multithreaded applications, there are some disadvantages such as application complexity. Oracle Tuxedo 8.0 supports two threading models: thread per object and thread per request.

Support for CORBA C++ Unicode Character Set

Release 8.0 of the Oracle Tuxedo software supports international (multi-byte) characters, or wide characters. Support for wide characters enables the Unicode character set to be used in the CORBA Interface Definition Language (IDL) and in CORBA client and server applications. As part of this new functionality, the C++ IDL compiler now supports `wchar` and `wstring` data types, recursive sequences, indirect typecodes, and forward references.

Support for C++ Object-by-Value Types

Support for C++ object-by-value types enhances the ability of Oracle Tuxedo CORBA C++ applications to access Oracle WebLogic Servers 6.0 via RMI/IIOP.

Load Balancing Features

Load Balancing that Reduces Bridge Traffic

Release 8.0 of Oracle Tuxedo includes new load balancing features. In Oracle WebLogic Enterprise 5.1, in a multiprocessor configuration, load balancing randomly distributed workload across all machines supporting a particular interface within a domain. In Oracle Tuxedo 8.0, load balancing favors the local machine when distributing workload and, thus, reduces bridge traffic.

Support for Parallel Objects

Support for parallel objects was added to Oracle Tuxedo CORBA in release 8.0 as a performance enhancement. The parallel objects feature enables you to designate all business objects in a particular application as stateless objects. The effect is that, unlike stated business objects (which can only run on one server in a single domain), stateless business objects can run on all servers in a single domain.

Support for Single Sign-On Across ATMI and CORBA Domains

Support for the T-Engine security plug-in framework was provided in Oracle Tuxedo 7.1 and Oracle WebLogic Enterprise 5.1 (CORBA-only), but security context could not be propagated across these environments. In Oracle Tuxedo 8.0, both the ATMI and CORBA environments support the same plug-in framework, and therefore, common, interoperable security is achieved.

Server Performance Enhancements

Oracle Tuxedo 8.0 includes the following server options that enable you to customize the server for optimal performance:

- Service and Interface Caching
  - Turning off Authorization and auditing security
  - Turning off Multithreaded Processing

– Turning off XA transactions

Domain Performance Enhancements

Enhancements have been made to the Oracle Tuxedo Domains software to improve performance in the following areas:

- Reduce boot time in large domain configurations

- Enable cross-domain assignment of message priorities

- Domain transaction auditing

Oracle WebLogic Server Integration Enhancements

In release 8.0, the Jolt software includes the following enhancements that improve the integration of Oracle WebLogic Server with Oracle Tuxedo services:

- Propagation of Oracle WebLogic Server user authentication

- Connection pools reset option

- XML buffer type support

- Buffer allocation enhancements

Oracle Tuxedo 8.1

XML Parser Integration

Release 8.1 of Oracle Tuxedo incorporates the Apache Xerces C++ parser into the Oracle Tuxedo system for use by customer applications to read and write XML data. The Xerces C++ parser is enhanced to cache DTD, XML schema, and entity files.

Multibyte Character Encoding

Release 8.1 of the Oracle Tuxedo software supports the multibyte coded character sets required by Chinese, Japanese, Korean, and other Asian Pacific languages. Oracle Tuxedo 8.1 includes a new multibyte string data typed buffer named `MBSTRING` for transport of multibyte character user data.

Localized Install for Japan

Release 8.1 of Oracle Tuxedo includes an installation enhancement enabling customers to install and interface with the Oracle Tuxedo system in English or Japanese.

TUXCONFIG Parameter Length Enhancement

This enhancement increases the maximum allowable length of certain Oracle Tuxedo configuration parameters from 64 or 78 to 256 characters. This enhancement is to increase the maximum length of pathname-holding parameters to hold longer pathname strings.

Domain Keepalive Function

The Domains keepalive functionality keeps Oracle Tuxedo interdomain connections open during periods of inactivity, but also enables TDomain gateways to quickly detect interdomain connection failures.

Global Maximum Transaction Timeout Enhancement

This enhancement adds a global maximum transaction timeout parameter to cap ATMI transaction timeout values that are excessively long.

Single Point Security Administration Option

This feature allows Oracle Tuxedo and Oracle WebLogic Server applications to share the WebLogic Server database. This feature applies to deployments involving an Oracle Tuxedo 8.1 and WebLogic Server 7.0 or later application.

Domain Gateway Performance Improvements

The Domain gateway performance improvement requires no change in the user interface. Performance of the TDomain gateway improved by:

- Improving various internal algorithms for scheduler and message receivers
  - Using cache whenever feasible to look up configuration information

Remote Domain Connection Policy Enhancement

This feature allows users to selectively establish connection on a per remote domain basis.

Multithreaded Bridge

This feature introduces an Oracle Tuxedo Bridge server process that has both single-threaded and multithreaded execution capabilities. A configuration setting determines whether the Bridge process is configured for single-threaded execution or multithreaded execution.

Oracle Tuxedo 9.0

Service Metadata Repository

The Oracle Tuxedo Service Metadata Repository is a repository similar to the Jolt repository. It provides Oracle Tuxedo application developers and administrators the ability to store and retrieve detailed services parameter information on any or all Oracle Tuxedo application services.

The Service Metadata Repository will provide enhanced Web services support and additional integration between Oracle Tuxedo and WebLogic.

XML To and From FML/FML32

To better enable integration of XML into Oracle products, XML conversion to and from FML/FML32 in an Oracle Tuxedo environment is provided.

Xerces Parser Upgrade

Oracle Tuxedo 9.0 delivers the Xerces parser 2.5.0 as part of the installation package. This upgraded parser will provide support for XML parser validation as well as continued XML support comparable to that of Oracle Tuxedo 8.1.

Domain Gateway Performance Enhancement

This feature improves the Oracle Tuxedo domain GWTDOMAIN performance by introducing multithreaded execution capability. It also allows other types of domain gateways to take advantage of this feature in the enhanced Common Gateway Architecture with simple program changes.

Domain Gateway Session Connection Policy

This feature allows configuring a specific TDOMAIN session from a local GWTDOMAIN gateway to a remote GWTDOMAIN gateway.

Domain Gateway Connection Events

Oracle Tuxedo 9.0 reports predefined events for Domain Gateway connections. Events primarily report issues such as connection success, failure and dropped connections.

IIOP Client Failover

Oracle Tuxedo CORBA remote clients invokes object methods via an IIOP gateway. In some cases, the IIOP gateway connection may fail due to reasons such as a shutdown of the IIOP gateway or network disconnection. This feature provides a transparent mechanism for a CORBA remote client to automatically connect to an alternative ISL and then retry the request.

Service-Level Blocktime

This feature provides applications with greater control over blocktime, providing capabilities to specify blocktimes for individual services, calls made by a particular Oracle Tuxedo context, or for one particular blocking call.

Kerberos Authentication

Oracle Tuxedo 9.0 provides a Kerberos security plug-in which allows Oracle Tuxedo native clients via an authentication server, KAUTHSVR, to validate the credentials passed by the security plug-in and return results. The KAUTHSVR will also take over what the default AUTHSVR does to provide Oracle Tuxedo security ACL checking.

Cert-C PKI Plug-in

Oracle Tuxedo 9.0 provides a Cert-C PKI plug-in which allows users to sign, seal, and envelope Oracle Tuxedo typed message buffers by using the public key encryption algorithm. The plug-in allows the typed buffer to be encrypted and then put a signature on it.

Oracle Tuxedo 9.1

Oracle RAC Support

Oracle Tuxedo 9.1 provides Transaction Monitor support for Oracle RAC by allowing an administrator to specify lists of groups associated with different RAC instances, thereby allowing Oracle Tuxedo to ensure that groups associated with different instances of the same RAC database do not participate in the same transaction.

Oracle Tuxedo .NET Workstation Client

The Oracle Tuxedo .NET Workstation Client enables you to write Oracle Tuxedo client applications using .NET programming languages to access Oracle Tuxedo services. It also provides connectivity between .NET workstation applications and Oracle Tuxedo services.

Remote Desktop Enhancement

Allows Oracle Tuxedo to start up, be accessed and shut down using MS Windows Remote Desktop.

Domain Transaction Performance Enhancement

Improved performance when switching from Jolt or WLEC to WTC.

Oracle Tuxedo 10.0

SSL Support for ATMI Applications

This feature provides support for SSL encryption over all network links in Oracle Tuxedo where LLE encryption is available.

Generic AUTHSVR

Generic AUTHSVR is a new Oracle Tuxedo system server (GAUTHSVR) that enables Oracle Tuxedo users to be authenticated with LDAP based directory servers without need to write custom code.

MQ Adapter

The MQ Adapter provides bi-directional, transactional connectivity to and from WebSphere MQSeries.

DoS

Provides Oracle Tuxedo TDomain domain gateway features used to defend against DoS attacks, and Oracle Tuxedo Domain improved password pair configuration flexibility.

Integrating ACUCOBOL in buildclient/buildserver

In Oracle Tuxedo 10.0, buildclient/buildserver can accept COBOL source files and generate C stub code automatically using ACUCOBOL compiler version 6.2.0 or above.

Integrating IBM COBOL v2.0 32-bit in buildclient/buildserver on AIX

In Oracle Tuxedo 10.0, buildclient/buildserver can accept COBOL source files and generate C stub code automatically using IBM COBOL v2.0.

OpenLDAP for X.509 Certificate Lookup

Oracle Tuxedo 10.0 PKI plug-in added support for OpenLDAP for X.509 certificate lookup.

Extended MAXSERVICES Range

The before Oracle Tuxedo 10.0 release the `MAXSERVICES` is limited to `32766`. In Oracle Tuxedo 10.0, the upper limit of `MAXSERVICES` has been extended to `1048574`.

Oracle Tuxedo 10.3

IPv6 Support

Before Oracle Tuxedo 10.3, only IPv4 is supported. In Oracle Tuxedo 10.3, IPv6 is supported. The most obvious improvement in IPv6 over the IPv4 is that IP addresses are lengthened from 32 bits to 128 bits.

Application-Created Context in ATMI Server

Two new APIs, `tpappthrinit(3c)` and `tpappthrterm(3c)`, are provided for application-created thread in ATMI server to create and terminate separate Oracle Tuxedo context. In context created using `tpappthrinit(3c)`, the application-created server thread can initiate service requests and define transactions.

Oracle Tuxedo Access Log

Assists Oracle Tuxedo client administrators to monitor application validity at runtime. You can record application high water client count, current client count, and named users.

Enhancements

CLOPT Length

The `CLOPT` length of an Oracle Tuxedo ATMI server in `UBBCONFIG` is increased from `256` to `1024`.

FML/FML32 Field Name Length

The field name length of an FML/FML32 buffer is increased from `30` to `254`.

tlisten(1) password encryption

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

Dynamic DMIB Update

Allow re-configuring the listening address of the remote domain gateway without shutting down the local domain.

Domain Gateway Persistent Disconnect

Local domain with a `PERSISTENT_DISCONNECT` connection policy will neither connect to nor accept connect request from any remote domain.

IBM COBOL v3.1 support on AIX

Oracle Tuxedo supports both 32-bit and 64-bit IBM COBOL v3.1 in `buildclient/buildserver`.

Oracle Tuxedo 11gR1 (11.1.1.1.0)

Client/Server Affinity

The Oracle Tuxedo Client/Server Affinity feature provides the flexibility to set up a simple session-aware application model. It creates a "virtual" request routing scope using the Oracle Tuxedo ATMI RPC infrastructure. When a session is established, all subsequent calls are impacted by the routing scope until the session is terminated (explicitly or implicitly). With Client/Server Affinity, you can retain session context resources inside the client/server affinity scope.

Extended ATMI Service Name Length

The maximum Oracle Tuxedo ATMI service name length is increased to 127 characters.

Enhancements

Domain Retries Connection Establishment After Incorrect Password Failure

When configured, `ON_STARTUP` domain gateway continuously re-tries to establish connection to remote domain when domain password pair validation fails.

Flexibility to Run Oracle Tuxedo as User Other than Administrator on Microsoft Windows

If configured in Microsoft Windows, processes booted in an Oracle Tuxedo domain are owned by the user who executed the `tmboot` command, instead of the user who starts the `TUXIPC` system service.

Oracle Tuxedo 11gR1 (11.1.1.2.0)

Support Multiple Resource Managers in one Oracle Tuxedo Group

In this release, Oracle Tuxedo supports multiple Resource Managers (RMs) in one group. Every group application server has the ability to communicate with multiple RMs in one global transaction. For more information, see *Oracle Tuxedo ATMI COBOL Function Reference*, *Oracle Tuxedo ATMI C Function Reference*, and *Oracle Tuxedo File Formats, and Data Descriptions, MIBs, and System Processes Reference*.

Nested Views

Previous Oracle Tuxedo releases have supported views. In Release 11.1.1.2.0 view functionality has been extended to support nested views. For more information, see *Managing Typed Buffers* in *Programming an Oracle Tuxedo ATMI Application Using C*.

Domain-Level AUTOTRAN

Previous Oracle Tuxedo releases support AUTOTRAN for individual services. Domain-level AUTOTRAN configuration is added in this release.

Oracle Tuxedo 11gR1 (11.1.1.3.0)

Oracle Tuxedo Exalogic Improvements

Self-tuning Locking Mechanism

Allows you to dynamically tune SPINCOUNT while taking the runtime environment into consideration, thus improve performance especially when there are heavy load on the system using XA without administrator configure SPINCOUNT with a static value.

For more information, see *Oracle Tuxedo/Oracle Exalogic Users Guide* and *File Formats, Data Descriptions, MIBs, and System Processes Reference*.

Direct Cross Node Communication Leveraging RDMA

In Oracle Tuxedo applications, processes on separate machines communicate with each other through bridge processes using a socket. The communication between bridges can be considerably slow.

This feature utilizes RMDA through which processes on separate Exalogic machines can communicate with each other directly. If Oracle Tuxedo application processes use RDMA instead of bridge processes overall performance is improved.

For more information, see *Oracle Tuxedo/Oracle Exalogic Users Guide*.

Oracle Tuxedo SDP Support

This feature allows Oracle Tuxedo user to configure which protocol to use, either SDP or normal TCP between Oracle Tuxedo components including Domain gateway, bridge, work station client and WSH, Jolt client and JSH t to leverage the advantages Exalogic provides such as high bandwidth, low latency as well as reduced CPU involvement.

For more information, see *Oracle Tuxedo/Oracle Exalogic Users Guide*, *File Formats, Data Descriptions, MIBs, and System Processes Reference, Command Reference*, and *ATMI C Function Reference*. Also, *Configuring the Oracle Jolt System* in *Using Oracle Jolt*.

TLOG Information To Oracle Database

Allows you the flexibility of using an Oracle database instead of file system to store the TLOG. You can also leverage various high availability Oracle database features in disaster recovery as needed.

For more information, see *About Transactions* in *Setting Up an Oracle Tuxedo Application*, and *File Formats, Data Descriptions, MIBs, and System Processes Reference and Command Reference*.

Oracle Tuxedo 12.1.1

Optimizations for Oracle Exalogic

Use of Shared Memory for Inter Process Communication

Oracle Tuxedo 12.1.1 significantly enhances performance of Oracle Tuxedo applications on Exalogic with use of shared memory queues instead of IPC Message Queues for inter process communication on the same Oracle Tuxedo node. With the use of shared memory queues, the sender and receiver processes can exchange pre-allocated messages in shared memory, thus eliminating the need to copy messages several times before message reaches its intended target and resulting in much better throughput and lower latency.

For more information, see *How to Create the TUXCONFIG File* in *Administering an Oracle Tuxedo Application at Run Time*, *ATMI C Function Reference*, *File Formats, Data Descriptions, MIBs, and System Processes Reference*, and *Command Reference*.

Shared Applications Staging

With Oracle Tuxedo 12.1.1, one can share application directory (APPDIR) among many compute nodes of the storage appliance on an Exalogic system, making it easier to manage application deployment.

For more information, see UBBCONFIG in *File Formats, Data Descriptions, MIBs, and System Processes Reference*.

Read-Only Optimization for XA

Optimized Distributed Transaction processing within and across Oracle Tuxedo domains for read-only transaction, including global transaction across Oracle Tuxedo domain and WTC (in WLS 12.1.1 - Contact Oracle Support for a patch, or higher release of WLS). One of the typical scenarios is every branch of the global transaction access the same Oracle Database instance.

In order to use this feature with WebLogic Server, minimum patch requirement for WebLogic Server must be met. This feature is not supported for CORBA applications.

For more information, see the RESOURCES and T_DOMAIN sections in *File Formats, Data Descriptions, MIBs, and System Processes Reference*.

Application Packaging and Deployment

Oracle Tuxedo 12.1.1 introduces a new concept of application packages. An application package is self-contained deployable unit. Application packages contain application binaries and required configuration artifacts, and can be automatically deployed/undeployed to an already running Oracle Tuxedo application domain. Application packaging and deployment feature includes in this release provide infrastructure for private cloud applications.

For more information, see *Oracle Tuxedo Application Packaging and Deployment* in *Setting up an Oracle Tuxedo Application*.

Developing New Applications Using Java

With this feature, one can develop new Oracle Tuxedo services using Java programming language in order to extend existing C/C++/COBOL applications. Java services are deployed and coexist in the same container as C/C++/COBOL services, allowing one to manage and monitor applications written in different programming languages using same set of tools.

Coexistence in the same container also optimizes transaction coordination across services written in different languages. Java services development includes following major features in the 12.1.1 release:

- POJO programming model

- JATMI based API

- XA transactions

- Monitoring and management - equivalent to C/C++/COBOL services

For more information, see *Managing ATMI Java Server* in *Setting up an Oracle Tuxedo Application*, and *Programming an Oracle Tuxedo Application Using Java*.

IBM WebSphere MQSeries Adapter Features

Following new features are added to the IBM WebSphere MQSeries Adapter:

- Reduced CPU usage, better throughput and higher scalability through multithreaded and event driven architecture of `TM_MQI` server

- Automatic connection failure recovery for `TM_MQI` and `TM_MQO` servers

- Support for clustered queues

- Ability to connect to remote MQSeries Manager

- Access to MQSeries message headers

- Recoverable messages in case security failure

For more information, see *File Formats, Data Descriptions, MIBs, and System Processes Reference, Command Reference*, and *Oracle Tuxedo MQ Adapter*.

Service Versioning

This feature provides a configuration-driven way to deploy different versions of Oracle Tuxedo services in an application domain or across domains without changing the existing code. By use of version, one can logically partition the existing Oracle Tuxedo applications into different virtual application domains, machines, and server groups based on current Oracle Tuxedo management hierarchy, so as to respond to several of special business access logics and on the other hand satisfy upgrade requirements in non-stop mode.

For more information, see *Applying Service Version to Tuxedo Applications* in *Setting up an Oracle Tuxedo Application*.

High Availability Configuration for Data Dependent Routing

With this feature, multiple server groups can be configured for the same data range, thus allowing incoming requests to failover to an alternate group, if primary group is not available, increasing the availability of the application. This feature can be used either within a domain or across the domains.

For more information, see `ROUTING` Section in *File Formats, Data Descriptions, MIBs, and System Processes Reference*.

Use of XPath for XML based Data Dependent Routing

With this feature, one can use XPath for much greater flexibility in specifying routing criteria if XML buffer type is used for data dependent routing.

For more information, see `UBBCONFIG` in *File Formats, Data Descriptions, MIBs, and System Processes Reference*.

Generic LDAP Authentication/Authorization Framework

Oracle Tuxedo 12.1.1 provides a flexible authentication and authorization framework that can be used to store credentials and Access Control Lists (ACLs) in LDAP or another 3rd party framework.

Generic LDAP authentication and authorization framework includes following major features:

- LDAP based authentication and authorization
- Flexible LDAP schema support
- Nested group support for authorization

For more information, see *How to Enable Generic LDAP Based Security* in *Using Security in ATMI Applications*.

Expedited Diagnostics through ECID Propagation

With this feature, an ECID (Execution Context ID) is propagated with each request within Oracle Tuxedo and across various products in Oracle stack. Propagation of ECID allows easy correlation of requests across Oracle Tuxedo domains and Oracle products, such as WebLogic Server, Database and so on, making it faster easier to diagnose application problems.

For more information, see *Configuring Tuxedo for Propagating ECID* in *Setting up an Oracle Tuxedo Application*.

Automatic Master Node and Server Group Migration

This feature enables automatic migration of the master node to a designated back up without any manual intervention, thus minimizing the application downtime and increasing the availability. Similarly this feature also enables automatic migration of server groups.

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

Millisecond Granularity for Timeouts

This feature introduces millisecond granularity for various Oracle Tuxedo timeouts, and other configuration parameters, such as for `SCANUNIT`. Millisecond granularity allows faster cleanup, restart, and migration of failed servers and nodes as well as faster transaction timeouts, enabling enforcement of tighter service level agreements, such as in algorithmic trading applications.

For more information, see *ATMI C Function Reference*, *ATMI COBOL Function Reference*, and *File Formats, Data Descriptions, MIBs, and System Processes Reference*.

Cross Domain Event Broker

This feature allows subscribe, unsubscribe, and post of brokered events across Oracle Tuxedo domains as can be done in a local Oracle Tuxedo domain.

For more information, see *Subscribing to Events* in *Administering an Oracle Tuxedo Application at Run Time*.

Server-Side Pseudo Code from Service Definition

If a service definition is in Oracle Tuxedo Metadata Repository, this definition can be used to generate server pseudo code in 'C' programming language using tmunloadrepos command. Server side pseudo code is generated in addition to client pseudo code, which can be done in prior releases.

For more information, see `tmunloadrepos` in *Command Reference*.

Nested Views for Jolt

Nested Views for Jolt are now supported in the 12.1.1 release.

For more information, see *Creating the Oracle Tuxedo Service Metadata Repository* in *Setting up an Oracle Tuxedo Application*, and *Using Oracle Jolt*.

SCA Programming Model and Support for PHP, Python, and Ruby

Oracle Tuxedo 12.1.1 included support for the SCA programming model that was originally released in Oracle SALT 10.3. As part of SCA, it also included support for the PHP, Python, and Ruby dynamic programming languages. However, SCA support has since been deprecated and remove from Oraclel Tuxedo.

For more information, see *Service Component Architecture*.

Oracle Tuxedo 12.1.3

For new features of Oracle Tuxedo 12*c* Release 2 (12.2.2), see *Oracle Tuxedo 12c Release 2 (12.1.3) Release Notes*.

Oracle Tuxedo 12.2.2

For new features of Oracle Tuxedo 12*c* Release 2 (12.2.2), see *Oracle Tuxedo 12c Release 2 (12.2.2) Release Notes*.

Oracle Tuxedo 22.1.0.0.0

For new features of Oracle Tuxedo 12*c* Release 2 (12.2.2), see *What's New and Improved in Oracle Tuxedo (22.1.0.0.0)* at https://docs.oracle.com/en/database/oracle/tuxedo/22/oxtrn/whats-new.html .

Oracle Tuxedo 22.1.1.0.0

The only new feature in Oracle Tuxedo 21.1.1.0.0 is support for TLS 1.3. For more information, see *What's New and Improved in Oracle Tuxedo 22.1.1.0.0* at https://docs.oracle.com/en/database/oracle/tuxedo/22/oxtrn/whats-new-and-improved-22.1.1.0.01.html

Appendix B: Integration

There are several add-on product options which can work with Oracle Tuxedo together to provide add-on services.

- Oracle SALT (Service Architecture Leveraging Tuxedo)
- Oracle TMA (Tuxedo Mainframe Adapter)
- Oracle TSAM (Tuxedo System and Application Monitor)

Oracle SALT (Service Architecture Leveraging Tuxedo)

Oracle SALT (Service Architecture Leveraging Tuxedo) is a separate product that runs on top of Oracle Tuxedo. Currently Oracle has released SALT 1.1, SALT 2.0, SALT 10.3, SALT 11.1.1.1.0, SALT 11.1.1.2, SALT 12.1.1, SALT 12.1.3, SALT 12.2.2, and SALT 22.1. Oracle SALT exposes existing Oracle Tuxedo services as standard Web services and provides access points to Oracle Tuxedo services through SOAP and REST over HTTP/S protocol.

In addition to basic Web service protocols, Oracle SALT complies with most primary Web services specifications: WS-ReliableMessaging and WS-Addressing, SOAP 1.1, SOAP 1.2, WSDL 1.1, WSSE 1.0 and 1.1 (from SALT 2.0), and MTOM (from SALT 2.0), allowing Oracle SALT to interoperate with other Web service products and development toolkits. With Oracle SALT, you can easily export existing Oracle Tuxedo services as Web services without having to perform any programming tasks. In SALT 2.0, external Web service access capability from Oracle Tuxedo applications is provided, and you can easily extend Oracle Tuxedo integration capability without having an extensive understanding of Web services.

Oracle SALT 10.3

Provides Service Component Architecture (SCA) Programming and Service Contract Discovery.

Oracle SALT 11.1.1

Provides new features including:

- Python and Ruby support,, SCA Structure support, and scatuxgen Metadata Generation ToolThis release provided several new features related to Service Component Architecture (SCA). However SCA has subsequently been deprecated and removed from Oracle SALT.

- WS-TX Support

Provides bi-directional transactional interoperability between Web Services and Oracle Tuxedo applications. Applications transparently make use of the GWWS system server transactional capabilities.

Oracle SALT 12.1.1

Provides new features including:

- Web Services Configuration Tool

Oracle SALT 12.1.1 provides an HTTP-based configuration tool to expose existing Oracle Tuxedo services as Web services without manually editing configuration files. It allows you to learn service definitions for existing Oracle Tuxedo services, edit service definitions in metadata repository, and create Web services definitions and `SALTDEPLOY` files through an easy to use graphical user interface.

For more information, see *Enabling the SALT Configuration Tool* in *Oracle SALT Configuration Guide*.

- Security Assertion Markup Language (SAML) Single Sign-On (SSO) Support

Provides the ability to recognize a SAML token inside a SOAP message request received by the Oracle SALT Web Services Gateway (GWWS). Based on the token contents, GWWS makes the decision to grant or deny access to Oracle Tuxedo resources.

For more information, see *Configuring Oracle Tuxedo Web Services*/*Configuring Security Features* in *Oracle SALT Configuration Guide*.

- New Data Type Support

Supports nested View32 data types. It also supports mapping of additional View and View32 primitive types to and from XML.

For more information, see *Data Type Mapping and Message Conversion* in *Oracle SALT Programming Guide*.

For more information, see *Oracle Service Architecture Leveraging Tuxedo (SALT) 12c Release 1 (12.1.1) documentation*.

Oracle SALT 12.1.3

Provides new features including:

- RESTful Web API

Existing Oracle Tuxedo services can be made available to and accessed by http clients as RESTful Web services eliminating the need to use SOAP/http for lightweight applications and expediting integration with other applications. RESTful web API can use XML or JSON payload for data transfer.

In addition to accessing Oracle Tuxedo services as RESTful Web services, Oracle Tuxedo applications can also access external RESTful services without having to write any code. Oracle Tuxedo applications can invoke RESTful services the same as if invoking Oracle Tuxedo services. The SALT gateway acts as the proxy for RESTful Web services.

For more information, see *Enabling the SALT Configuration Tool* and *Representational State Transfer (REST) Option* sections in *SALT Configuration Guide*.

- Custom HTTP Headers

   HTTP headers can pass relevant application control information to or from Oracle Tuxedo services. For incoming RESTful Web services, any custom HTTP header is attached to an Oracle Tuxedo buffer and passed to the invoked Oracle Tuxedo service. An Oracle Tuxedo service header can be read using a provided API. Similarly, an Oracle Tuxedo application can set HTTP headers in an Oracle Tuxedo buffer using a provided API, which in turn is converted to an HTTP header by the SALT gateway.

   For more information, see Enabling the SALT Configuration Tool, Custom HTTP Headers in the SALT Configuration Guide.

   For more information, see *Enabling the SALT Configuration Tool* and *Custom HTTP Header Option* sections in *SALT Configuration Guide*.

- WS-Security for External Web Services

   Message-level authentication is provided using an X.509 certificate to sign messages. Oracle Tuxedo can invoke an external Web service using SOAP/http with the principal identity of the X.509 certificate.

   For more information, see *Configuring Oracle Tuxedo Web Services*/*Configuring Security Features* in *SALT Configuration Guide*.

- Data Transformation Tracing

   Tracing all incoming and outgoing messages is enabled (including RESTful Web services, SOAP/http Web services, and all data transformation from XML to Oracle Tuxedo buffers and vice-versa).

   For more information, see *XML-to-Tuxedo Data Type Mapping for External Web Services* in *SALT Programming Guide*.

- ECID Propagation

   ECID (Execution Context ID) is propagated with each request within Oracle Tuxedo and across various products in an Oracle stack. ECID propagation enables request correlation across Oracle Tuxedo domains and Oracle products (such as Oracle WebLogic Server, Oracle Database, etc.), making it quicker to diagnose application problems.

   For more information, see *Configuring Tuxedo for Propagating ECID*.

- Dynamic Configuration and MIB

   You can dynamically reload configuration file changes without any downtime. An MIB interface is provided, which enables reading Web services configuration and runtime statistics.

   For more information, see *Enabling the SALT Configuration Tool* and *MIB Class Interface* sections in *SALT Configuration Guide*.

- XML Complex Attribute Mapping

   Enhanced usability is provided by using an attribute field within a complexType element in a WSDL for accessing external Web services. complexType attribute fields are mapped "one-by-one" to corresponding FML32 fields.

   For more information, see *Data Type Mapping and Message Conversion* in *SALT Programming Guide*.

- Configuration Tool Enhancements

   The configuration tool has been enhanced to provide the following features:

   – Support for RESTful Web services

   – Enables import of external Web services

   – Support for FireFox and Safari Web browsers

   – A test client for RESTful Web services

   For more information, see *Enabling the SALT Configuration Tool* in *SALT Configuration Guide*.

Oracle SALT 12.2.2

Provides new features including:

- Custom SOAP Headers

    This feature enables use application specific headers in SOAP messages when accessing Tuxedo services using SOAP/http. This feature leverages the existing tpsetcallinfo() and tpgetcallinfo() APIs to place/retrieve data into/from the SOAP header for inbound/outbound directions.

    – Inbound (Oracle Tuxedo services exposed as web services, or native web services).

    – Outbound (invoking web services as Oracle Tuxedo services, or external web services).

        For more information, see `tpsetcallinfo()`, `tpgetcallinfo()`, Enabling the SALT Configuration Tool, Custom SOAP Headers in the *SALT Configuration Guide*.

- RECORD Buffer Type Support

    RECORD buffer type is now supported in SALT. RECORD buffer type is a flexible way to correctly and completely represent COBOL copybook records.

    RECORD buffers are used in SALT to expose mainframe transactions and ART Transactions as Web services and to access Web services from these transactions.

    For more information, see *Using Oracle Tuxedo Service Metadata Repository for SALT*, and *Tuxedo-to-XML Data Type Mapping for Oracle Tuxedo Services*.

- Mainframe Transaction Publisher for SALT

    This release of SALT makes it easier for transactions running in IBM mainframe¡¯s CICS/IMS environments to be exported as Web services (SOAP/http or RESTful Web services). In addition, such mainframe transactions can also access external Web services using SOAP/http or RESTful API. Such integration is completely configuration driven and no code development is needed.

    SALT includes tools, such as wscobolcvt, which allow to import COBOL copybook and create metadata repository artifacts, WSDL and other required configuration to expose mainframe transactions as Web services. Similarly, tools allow import of a WSDL for external Web service and create required artifacts for mainframe transactions to access external Web services.

    For more information see, `wscobolcvt`, *SALT Mainframe Transaction Publisher*, and *XML-to-Tuxedo Data Type Mapping for External Web Services*.

- New Tuxedo Services Console

    – This release of SALT replaces existing SALT configuration tool with a new Tuxedo Services Console to export, import Tuxedo services. This new console provides following major functionality:

    – Services Management: add/edit/delete Tuxedo service definitions in the Tuxedo metadata repository.

    – Configure Web services (SOAP and REST): enables Oracle Tuxedo services to be accessed as SOAP or REST services and enables Tuxedo applications to access external SOAP/REST Web services.

    – Integrate with mainframe transactions: enables mainframe transactions to be accessed as Web services (SOAP or REST), or enables mainframe transactions to access external Web services.

    For more information, see `TMADMSVR` and `MTP`

- Single-Sign-on with Oracle Access Manager

    This release of SALT integrates with Oracle Access Manager (OAM) seamlessly, with automatic detection of the OAM authentication server configuration and handling of OAM tokens. This achieves single sign-on at the HTTP level for support of secure SOAP and REST access.

For more information see *Configuring Oracle Tuxedo Web Services*, Configuring Security Features.

Oracle SALT 22.1

For information about new features in Oracle SALT 22.1, see *Major Enhancements Post Oracle SALT Release 12.2.2* at https://docs.oracle.com/en/database/oracle/salt/22/salrn/major-enhancements.html .

Oracle TMA (Tuxedo Mainframe Adapter)

Oracle Tuxedo Mainframe Adapter is a domains-based connectivity product that provides bidirectional connectivity between Oracle Tuxedo clients or servers and application programs residing on different mainframe platforms, it can achieve this connectivity through SNA, OSI TP and TCP protocol.

After upgrading to Oracle Tuxedo 22c, Oracle Tuxedo Mainframe Adapter should be upgraded too. TMA 22c (22.1.0.0.0) is required for Oracle Tuxedo 22c Please contact your Oracle representative to find out about availability of TMA 22c .

Oracle TSAM (Tuxedo System and Application Monitor)

Oracle TSAM (Tuxedo System and Application Monitor) and TSAM Plus are two separate products that run on top of Oracle Tuxedo. Currently Oracle has released TSAM Plus 12.1.1.1, TSAM Plus 12.1.3, and TSAM Plus 12.2.2. Oracle TSAM Plus 12c Release 2 (12.2.2) can monitor Oracle Tuxedo 9.1, 10.3, 11gR1PS1, 11gR1PS2, 12.1.1, 12.1.3, 12.2.2, and 22.1. There are separate installer binaries for TSAM Plus. Customers should install TSAM Plus 12c Release 2 (12.2.2) on top of Oracle Tuxedo 22c. TSAM Plus 12c Release 2 (12.2.2) only supports Oracle Tuxedo 12c Release 2 (12.2.2) and Oracle Tuxedo 22c (22.1.x). Oracle TSAM Plus provides comprehensive monitoring and reporting for Oracle Tuxedo system and applications. It includes two components: Oracle TSAM Plus Agent and Oracle TSAM Plus Manager.

The Oracle TSAM Plus agent enables collection of various applications performance metrics (including XA and non-XA transactions, services, system servers). The Oracle TSAM Plus Manager provides graphical user interface that correlates and aggregates performance metrics collected from one or more Oracle Tuxedo domains. It displays this information in real time.

Oracle TSAM Plus is designed for detailed end-to-end transaction and services monitoring. It enables users to set and monitor response-time service level agreements (SLAs), investigate the performance and behavior of live application services, and improve capacity planning by using comprehensive utilization metrics for all components of the Oracle Tuxedo infrastructure. Furthermore Oracle TSAM Plus helps customers optimize resources to maximize IT hardware in complex, mission-critical application environments.

As of the original publication date of this document, separate Oracle TSAM Plus 22c binaries had not been released.

Appendix C: Reference

This document provides consolidated guidelines for upgrading to Oracle Tuxedo 22*c* (22.1.0.0.0 and 22.1.1.0.0). The detailed instructions should be referenced at the following Oracle Tuxedo 22*c* and other related product online documentation.

- *Oracle Tuxedo Online Documentation Main Page*

- *Oracle Tuxedo 22c Release (22.1.0.0.0) Platform Data Sheets*

- *Oracle Tuxedo 22c Release (22.1.1.0.0) Platform Data Sheets*

- *Oracle Tuxedo Family Platform Datasheets for multiple Releases*

- *Oracle SALT 22.1 Documentation*

- *Oracle Tuxedo 12c Release 2 (12.2.2) Interoperability*

- *Upgrading the Oracle Tuxedo System to 22c*

- *Oracle Tuxedo Interoperability with Oracle Service Bus 2.5*

- *Oracle TSAM 12.2.2 Documentation*