

ORACLE SERVICE ARCHITECTURE LEVERAGING TUXEDO

KEY FEATURES AND BENEFITS

FEATURES

- Web services calls to and from Oracle Tuxedo with extensible data mapping
- SCA programming model support and container
- SCA components in Python and Ruby
- Metadata driven application development model
- MTOM support for binary data
- WS-Security implementation
- WS-AtomicTransaction support
- XML schema for custom buffer types
- Multiple encoding support with SOAP messages
- Enhanced error reporting

BENEFITS

- Enable interoperability among distributed applications that span diverse hardware and software platforms.
- SCA programming model simplifies component re-use and seamless communications between components
- Rapid SCA component development using dynamic scripting languages
- Simplify access to Oracle Tuxedo applications using Web services protocols.
- Atomic transactions ensure data consistency
- Increase performance by 75 percent versus Java-based Web services offerings.
- Facilitate the development of heterogeneous distributed applications with a cross-platform, cross-language data model.
- Lower your TCO by leveraging existing IT assets without the need for costly rewrites.

Oracle Service Architecture Leveraging Tuxedo (SALT) is a standards-based, bi-directional Web services gateway and SCA based services composition environment supporting enterprise applications written in C, C++, COBOL, Python, and Ruby. SALT provides transparent and reliable access to existing Oracle Tuxedo services as standard Web services with a high-performance, easy-to-use, and configuration-driven model. In addition, Oracle Tuxedo applications can call external Web services as Tuxedo services, with the gateway transforming the call using SOAP over HTTP protocol. The result is widespread access to applications using Web services without any Web services specific coding, along with interoperability among distributed applications that span diverse hardware and software platforms.

SOA-Enable Oracle Tuxedo Applications Without Re-Coding

Oracle SALT allows organizations to integrate mission-critical applications written in C, C++, COBOL, Python, and Ruby into service-oriented architectures (SOA) without costly rewrites to Java or Microsoft .NET. Leveraging a configuration-driven approach eliminates the need for programming changes and allows companies to use existing enterprise assets and to lower their total cost of ownership (TCO).

Oracle SALT provides the ability to support heterogeneous composite application development without complex enterprise application-integration (EAI) products. It supports the new Service Component Architecture (SCA) programming model to improve component re-use and focus on business logic rather than system specific programming, connectivity, or transformation. This removes the need to change Tuxedo applications frequently to adapt to other enterprise applications, freeing IT departments to shift resources to more-strategic, business-driven projects.

As a native Oracle Tuxedo Web services implementation, Oracle SALT reduces the run-time conversions that might exist with other solutions for accessing Oracle Tuxedo services.

Create an Extensible Web Services Architecture

Oracle SALT and Oracle Tuxedo allow enterprises to develop composite (or hybrid) end-to-end solutions that leverage the scalability and robustness of Oracle Tuxedo combined with the extensibility of SOA. Uniting these two environments simplifies a broad spectrum of transaction-processing solutions via powerful, open standards-

based APIs. With a high-performance, easy-to-use, configuration-driven model for bidirectional Web services integration, developers can publish existing Oracle Tuxedo services as standard Web services using the SOAP/HTTP(S) protocol.

With Oracle SALT, Oracle Tuxedo services can transparently call external Web services as if calling another native service. In addition to basic Web services protocols, the application complies with most primary Web services specifications, including SOAP, Web Services Description Language (WSDL), WS-Addressing, WS-Security, WS-AtomicTransaction, and WS-ReliableMessaging. This improves TCO by leveraging existing skill sets, tools, and information assets.

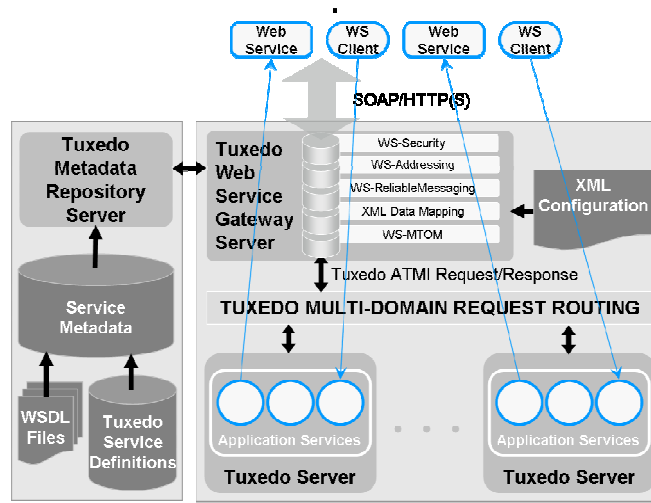


Figure 1. Oracle SALT in an enterprise services network

Oracle SALT can be integrated with Oracle Service Registry and Oracle Enterprise Repository to publish Tuxedo services metadata for broad access within the enterprise, enabling their use in Oracle BPEL PM, Oracle Business Rules, and Oracle Service Bus as well as any 3rd party users of the Registry and Repository.

Web Services Architecture Features and Benefits	
Bi-directional calls to Web services	Allows Oracle Tuxedo application services to be accessed via Web services, and Tuxedo services to call external Web services
Configuration-driven deployment	Supports a configuration-driven style of deployment allowing you to specify a list of Oracle Tuxedo services that can be exposed as Web services
Metadata repository	Maintains Tuxedo service contract metadata and automates the creation of WSDLs for publishing Tuxedo services as Web services.
Web services gateway (GWWS)	Handles Web services protocols and performs automatic SOAP message conversion to and from Tuxedo buffers; acts as an Oracle Tuxedo gateway process and can be managed in the same manner as general Oracle Tuxedo servers
WSDL document generation and publishing	Automatically generates WSDL documents that can be integrated using Web services development tools or published to a Universal Description, Discovery, and Integration (UDDI) server
Seamless upgrade	Enables use of existing WSDL, configuration files, and service metadata without any manual changes to upgrade to newer versions
MTOM support for binary data	Implements optimized MIME multiple/related serialization of SOAP messages for Oracle Tuxedo and X_OCTET CARRAY buffers
Multiple encodings for inbound requests	Supports multiple character encodings in addition to previously supported UTF-8 encoding for SOAP messages

Forward user data with TPFALL	Provides capability to send user data back to the Web services client in a TPFALL instance by way of a SOAP fault message
WS-Security implementation	Supports WS-Security 1.0 and 1.1 for inbound requests. Supports username and X.509 certificates, security tokens, Web services client authentication, and WS-SecurityPolicy
WS-AtomicTransaction implementation	Supports WS-AtomicTransaction and WS-Coordination to allow Tuxedo applications to participate in or coordinate atomic Web services transactions

Improve Efficiency with a Service Component Architecture Programming Model

Oracle SALT SCA container provides metadata driven, SCA-based application development. The inherent characteristics of SCA enable programmers to focus on business logic by greatly facilitating reuse and automating rewiring of existing components. Although components may run in multiple container types and use different protocols, these are compatible and accessible through SCA composition tools. Support of SCA programming model on top of Tuxedo renders the ability to more effectively manage service lifecycle, including systematic re-use of existing services in SCA-based composite applications, as well as runtime discovery of service signature. SCA support renders Tuxedo-based applications (those commonly written in C/C++, COBOL, Python and Ruby) interoperable with other enterprise applications written in Java, .NET, etc, a significant benefit to customers. The SCA container in SALT also supports the development of SCA components in Python and Ruby. As with other SCA components, Python and Ruby components can utilize existing ATMI services and be used by ATMI clients.

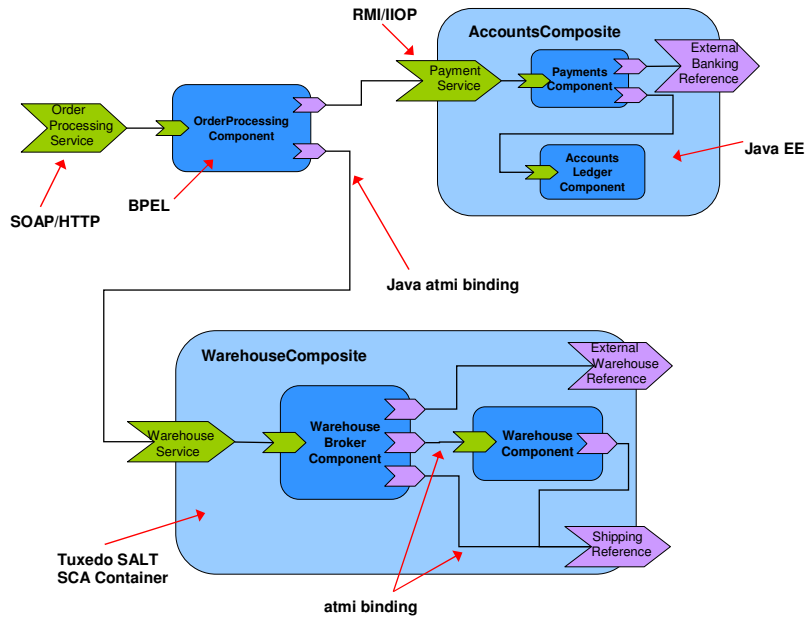


Figure 2. Process flow for a SCA programming model

SCA Programming Model Features and Benefits	
Client as well as Server side binding for ATMI, SOAP, Java	Out of the box multi-protocol support
Development and Runtime tools	Improved efficiency in developing and deploying clients and servers as well as for runtime administration
Support for Simple and Complex data types	Provides customers flexibility in programming with complex data types
Automatic data transformation and wiring	Improved developer efficiency by not having to develop additional code
Support for multi-byte characters	Support for internationalization and localization
Support for global transactions	Ensures transactional integrity by way of XA support in Tuxedo
Python and Ruby support	Allows SCA components to be developed in the dynamic object oriented scripting languages Python and Ruby thus reducing the time to develop components

Optimize Performance, Scalability, and Interoperability

Oracle SALT improves performance with significantly faster response times while offering full support for multiple interaction styles, including inbound/outbound, synchronous/asynchronous, and request/response. SALT supports XML and key WS standards, and interoperates with multiple Web services platforms, including Oracle WebLogic Server, Oracle Service Bus, Apache Axis, and .NET.

Performance and Interoperability Features and Benefits	
Asynchronous and reliable messaging	Provides asynchronous communication model and complies with WS-Addressing specification; also supports reliable message delivery conforming to WS-ReliableMessaging specification
Interoperability	Provides interoperability with Microsoft .NET 3.0/Indigo release, with AXIS 2.0 and with Oracle WebLogic Server and Oracle Service Bus
XML schema for custom buffer types	Supports the use of XML schema to describe the content of custom buffer types to convert custom buffers to and from XML
XML validation	Delivers the mechanism to describe format/content of XML fragments
Scalability	A single Oracle Tuxedo domains can support thousands of concurrent clients using services that generate external Web services calls
Overlapped requests	Handles multiple concurrent requests with high throughput – sufficient to support all application services in an Oracle Tuxedo domain

RELATED PRODUCTS AND SERVICES

Oracle Tuxedo delivers a robust platform to run high-volume applications across distributed, heterogeneous computing environments, enabling transactions that stretch from customer-facing, business-critical applications to back-office processes, across any system, anywhere in the world.

RELATED PRODUCTS:

- Oracle Tuxedo
- Oracle Tuxedo System and Applications Monitor (TSAM)
- Oracle Tuxedo Mainframe Adapters
- Oracle Tuxedo JCA Adapter
- Oracle Tuxedo Application Runtime for CICS and Batch
- Oracle Tuxedo Application Rehosting Workbench
- Oracle Tuxedo Jolt
- Oracle Enterprise Repository
- Oracle Service Bus
- Oracle BPEL Process Manager
- Oracle WebLogic Server

Security and Platform Support

Oracle SALT uses the Oracle Tuxedo security framework for authentication and protects point-to-point transactions using Secure Socket Layer (SSL) protocols. The application supports 32-bit and 64-bit binary platforms and can be extended to other platforms based upon customer need.

Security and Platform Support Features and Benefits	
HTTP basic authentication	The gateway server supports Oracle Tuxedo user profiles passed from the Web services client via HTTP basic authentication protocol.
WS-Security support	The gateway server supports the WS-Security and WS-SecurityPolicy protocols to allow inbound authentication by username and X509 tokens, and digitally signing messages.
SSL link-level security	The gateway server supports HTTPS requests and link-level security, enabled by specifying an SSL host and port number in the Oracle SALT configuration file.
32-bit binary support	AIX 5.3 on PPC, Solaris 9 on SPARC, Windows 2003 on Intel x86, HP-UX 11.23 on PA-RISC, HP-UX 11.23 Itanium, SLES 9 on x86.
64-bit binary support	HP-UX 11.23 Itanium.

Contact Us

For more information about Oracle SALT, please visit oracle.com or call +1.800.ORACLE1 to speak to an Oracle representative.



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2010, Oracle and/or its affiliates. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners. 0109