BEA SALT™

Product Overview

Version 1.1
Document Released: August 16, 2006

Copyright

Restricted Rights Legend

Trademarks and Service Marks

Contents

BEA SALT Overview

BEA SALT Overview

This section contains the following topics:

Understanding BEA SALT

BEA SALT (Service Architecture Leveraging Tuxedo) is a separately licensed product that runs on top of Tuxedo. BEA SALT exposes existing Tuxedo services as standard Web services and provides access points to Tuxedo services through SOAP over HTTP/S protocol.

In addition to basic Web service protocols, BEA SALT complies with most primary Web services specifications: WS-ReliableMessaging and WS-Addressing, SOAP 1.1, SOAP 1.2, and WSDL 1.1, allowing BEA SALT to interoperate with other Web service products and development toolkits. With BEA SALT, you can easily export existing Tuxedo services as Web services without having to perform any programming tasks.

What Are Web Services?

Web services are a set of functions packaged into a single entity that is available to other systems on a network, and can be shared by and used as a component of distributed Web-based applications. The network can be a corporate intranet or the Internet. Other systems, such as customer relationship management (CRM) systems, order-processing systems, and other existing

back-end applications, can call these functions to request data or perform an operation. Because Web services rely on basic, standard technologies which most systems provide, they are an excellent means for connecting distributed systems together.

The software industry has evolved toward loosely coupled service-oriented applications that interact dynamically over the Web. The applications break down the larger software system into smaller modular components, or shared services. These services can reside on different computers and can be implemented by vastly different technologies, but they are packaged and accessible using standard Web protocols, such as XML and HTTP, thus making them easily accessible by any user on the Web.

Web services are defined to share the following properties that make them easily accessible from heterogeneous environments:

- Web services are accessed using widely supported Web protocols such as HTTP.
- Web services describe themselves using an XML-based description language.
- Web services communicate with clients (both end-user applications or other Web services) through simple XML messages that can be produced or parsed by virtually any programming environment or manually, if necessary.

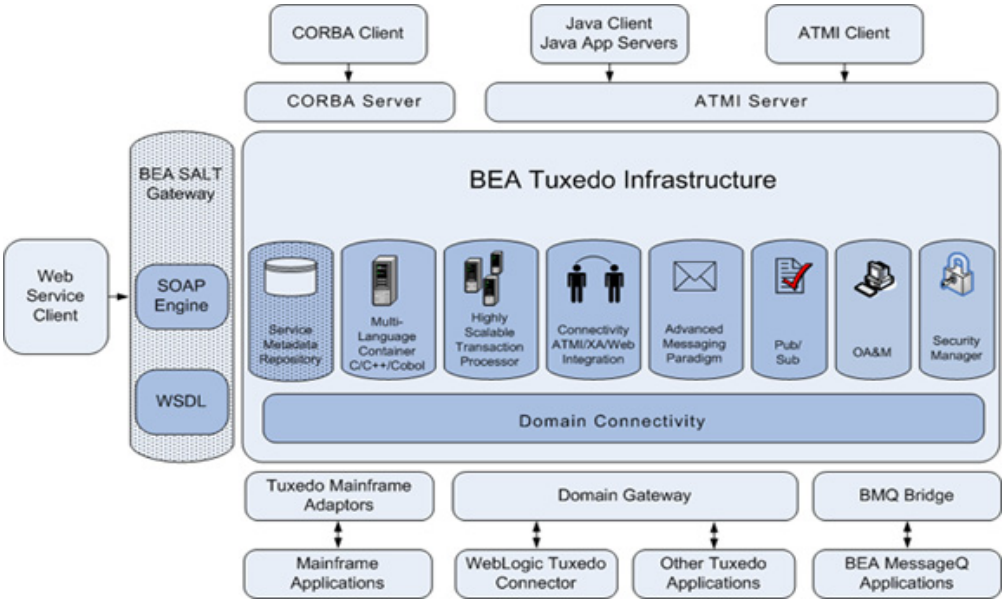
Why Use BEA SALT?

BEA SALT is configuration-driven and does not require programming changes to access Tuxedo services. Because it is a native Tuxedo Web service, it reduces the conversion process that may exist with other solutions for accessing Tuxedo services.

Major benefits of Web services include:

- Interoperability among distributed applications that span diverse hardware and software platforms
- Easy, widespread access to applications using Web protocols
- A cross-platform, cross-language data model (XML) that facilitates developing heterogeneous distributed applications

Figure 1 BEA SALT Gateway on Top of Tuxedo Infrastructure



Invoking Tuxedo Services Using BEA SALT

In order to deploy and invoke a Tuxedo service using BEA SALT, you must do the following:

1. Define Tuxedo application services using the Tuxedo Service Metadata Repository.
2. Compose one or more BEA SALT configuration files.
3. Add the TMMETADATA and GWWS servers to your Tuxedo UBBCONFIG file.
4. Boot the Tuxedo application.
5. Client-end user downloads the WSDL document file from the GWWS server specified URL.
6. Client-end user generates client side stub-code from the WSDL document file with a SOAP development kit.
7. Generate client-side program.
8. Run the client to invoke the Web service with SOAP messages.

The GWWS server translates the SOAP message into Tuxedo typed buffer, dispatches the translation to the Tuxedo service, and gets the reply. The GWWS server then translates the reply into a SOAP message and sends it back to the client.

BEA SALT Feature Summary

BEA SALT consist of the following features:

- Configuration-Driven Deployment
- GWWS Gateway
- WSDL Document Generation and Publishing
- Asynchronous and Reliable Messaging
- Security

Configuration-Driven Deployment

BEA SALT supports a configuration-driven style of deployment and management. The BEA SALT configuration file is a single root XML file. The BEA SALT configuration file allows you to specify a list of Tuxedo services that can be exposed as Web Services, policy information, and other system level information. One configuration file is used to represent a particular GWWS process or a group of failover GWWS processes.

Using the Tuxedo Service Metadata Repository

Service contract information of all listed Tuxedo services must be accessible by accessing the Tuxedo Service Metadata Repository service provided by the local Tuxedo domain. BEA SALT leverages the Tuxedo Service Metadata Repository for Tuxedo Web services exposure.

GWWS Gateway

BEA SALT provides a Tuxedo system server, GWWS, which can handle Web service SOAP messages over HTTP/S protocol. The GWWS server acts as a Tuxedo gateway process and can be managed in the same manner as general Tuxedo system servers. SOAP request messages are parsed by the GWWS server, converted into Tuxedo typed buffers, and dispatched to corresponding Tuxedo services. The Tuxedo response typed buffers returned by the Tuxedo services are converted to SOAP response messages by GWWS and sent back to the client side. A typed buffer plug-in mechanism in GWWS handles custom defined buffer types.

You can have multiple GWWS instances in one Tuxedo domain. The same functionality for multiple GWWS instances is provided by specifying the same BEA SALT configuration file to improve throughput and failover protection. You can also group multiple GWWS instances in different configuration files for different purposes.

When the GWWS server boots, it loads the specified configuration, which includes reading the configuration XML file, validating the XML file, loading corresponding Tuxedo service contract information from the Tuxedo Service Metadata Repository, and loading the WS-ReliableMessaging policy definition file. The GWWS server can reload the configuration dynamically at runtime. You can also download the WSDL document file from the GWWS server (which is based on the latest BEA SALT configuration file being used by the GWWS process).

WSDL Document Generation and Publishing

Web Services Description Language (WSDL) is an XML-based specification that describes a Web service. A WSDL document describes Web service operations, input and output parameters, and how a client application connects to the Web service.

You do not need to compose the WSDL document manually; it is automatically generated as part of the SALT Web services development process. The generated WSDL document can be integrated using Web services development tools or can be published to a UDDI server.

There are two approaches to obtaining the WSDL document, you can:

- use the WSDL document file generating utility, `tmwsdlgen`.
- download the WSDL document provided by the GWWS server via HTTP.

Asynchronous and Reliable Messaging

BEA SALT provides an asynchronous communication model complies with WS-Addressing (WS-Addressing 1.0, Aug., 2004) specification. BEA SALT also supports reliable message delivery between client and server conforming to WS-ReliableMessaging (WS-ReliableMessaging 1.0, Feb., 2005) specification.

The later specification allows Web service instances to reliably communicate with the GWWS server in case of network failure or other scenarios. In particular, this feature provides for an interoperable protocol in which a message sent from a source endpoint to a destination endpoint is guaranteed with a defined delivery policy. The WS-Policy specification is supported to define the delivery policy for WS-ReliableMessaging.

Security

BEA SALT uses the Tuxedo security framework for authentication and passes a user profile via the HTTP basic authentication specification. Point-to-point security is carried out using SSL link-level security.

HTTP Basic Authentication

BEA SALT uses Tuxedo security framework for system authentication. The GWWS server supports Tuxedo user profiles passed from the Web service client via HTTP basic authentication protocol.

SSL Link-Level Security

The GWWS server handles HTTPS requests. You can enable link-level security by specifying an SSL host and port number in the BEA SALT configuration file. Parameters necessary for setting up an SSL connection are also specified in the BEA SALT configuration file.

Standards Supported by BEA SALT

BEA SALT supports the following Web service standards:

- Standards for transmitting data and Web service invocation calls between the Web service and the user of the Web service.
  - SOAP 1.1

    For more information, see SOAP 1.1 specification
    http://www.w3.org/TR/2000/NOTE-SOAP-20000508/
  - SOAP 1.2

    For more information, see SOAP 1.2 specification
    http://www.w3.org/TR/soap12-part0/
  - SOAP with Attachment

    The SOAP with Attachment specification describes a standard way to associate a SOAP message with an attachment in its native format in a multipart MIME structure for transport. BEA SALT allows you to represent a MIME-attached Tuxedo CARRAY typed buffer.

    For more information, see SOAP Messages with Attachments
    http://www.w3.org/TR/SOAP-attachments

- A standard for client applications to find a registered Web service and to register a Web service.
  - UDDI 2.0

    For more information, see UDDI 2.0 specification
    http://uddi.org/pubs/ProgrammersAPI-V2.04-Published-20020719.htm
- Standards for describing the Web service to clients so they can invoke it.
  - WSDL 1.1

    For more information, see WSDL1.1 specification
    http://www.w3.org/TR/2001/NOTE-wsdl-20010315
  - WS-Policy

    For more information, see WS-Policy specification
    http://specs.xmlsoap.org/ws/2004/09/policy/ws-policy.pdf and
    WS-PolicyAttachment specification
    http://specs.xmlsoap.org/ws/2004/09/policy/ws-policyattachment.pdf
- Standards for Web service infrastructure.
  - WS-Addressing

    For more information, see WS-Addressing specification
    http://www.w3.org/Submission/2004/SUBM-ws-addressing-20040810/
  - WS-ReliableMessaging

    For more information, see WS-ReliableMessaging specifications
    http://specs.xmlsoap.org/ws/2005/02/rm/ws-reliablemessaging.pdf and
    http://specs.xmlsoap.org/ws/2005/02/rm/WS-RMPolicy.pdf

What Next?

After becoming familiar with the BEA SALT Product Overview, refer to the following topics for installing, configuring and running Web services using the SALT product.

- Install the BEA SALT product.

  For an explanation of how to install the product, refer to *BEA SALT Installation Guide*.
- Configure the BEA SALT product.

  For an explanation of how to configure and administer the product, refer to *BEA SALT Administration Guide* and *BEA SALT Reference Guide*.

- Invoke Web services using the BEA SALT product.

  For an explanation of how to invoke Web services using SALT, refer to *BEA SALT Programming Web Services*.