

Oracle Service Bus Statement of Direction *August 2008*

Market-leading ESB offers unmatched flexibility and capabilities

Strategy fully preserves development investments of both BEA and Oracle customers.

INTRODUCTION

Oracle's place in the Enterprise Service Bus market has been significantly strengthened through the acquisition of BEA and the incorporation of AquaLogic Service Bus into the Oracle SOA Suite product family alongside the Oracle ESB. This document details how Oracle's go-forward ESB strategy offers complete, cohesive ESB capabilities of unmatched flexibility and functionality. And equally important, this strategy fully preserves all development investments of both BEA and Oracle customers by providing seamless, fully automated upgrade paths from the current versions to the unified platform in 11g and beyond.

WHAT IS AN ENTERPRISE SERVICE BUS?

An Enterprise Service Bus (ESB) is a key component of modern IT infrastructures that provides the foundation for Service-Oriented Architectures (SOA). The core features of an ESB are:

Connectivity

While newer applications may expose services over standard, easily consumable interfaces such as SOAP, an overwhelming majority of the services available in enterprises are still locked in legacy systems, from mainframes to custom applications. The first task of an ESB is to ensure that all these existing assets can be easily accessed and integrated. An ESB must offer rich and comprehensive connectivity options to standard interfaces (SOAP, messaging, etc.) but also to packaged applications and legacy systems via adapters.

Messaging & Routing

Once connectivity is established, data and messages need to flow reliably. An ESB needs to offer efficient, secure and reliable transport options. In addition, the ESB needs to incorporate advanced data routing facilities to determine where the data needs to go. Such routing can be done based on headers, content or other external rules.

Transformation

As data flows between services, it needs to be transformed to match the format that target services expect to receive. Data transformations often are the single most expensive operations in data flows so an ESB needs to offer rich but also high-performance transformation capabilities.

Event Sensors

Monitoring and auditing routing rules and instance data require an instrumentation framework of event sensors native to an ESB. Messages must be correlated, aggregated and audited across varying protocols and formats.

Policy Management

Security and policies in general are critical to successful operations, governance and compliance. A policy management model needs to be able to enforce authentication, authorization, and encryption in addition to more complex business-level policies, independent of the service provider implementation. An ESB needs to incorporate a global policy management framework for consistency and agility.

Virtualization

A core principle of SOA is to ensure that any service consumer can access any service provider - and from any platform. This implies that the appropriate protocol and data formats are supported to insulate consumers and providers. This enables target services to be protected from overload through such techniques as throttling but also decouples business concerns from technology concerns via abstraction. Such separation of concerns ensures that an IT-level change such as a database relocation will not require any change to business logic. The level of sophistication in virtualization capabilities is often a differentiating factor between ESBs.

Value-Added Features

A market-leading ESB will also offer additional value-added features such as domain-value maps and cross-references for ease of integration with packaged applications. Another example would be support for specific message formats and protocols such as SWIFT and FIX in the financial services domain.

OVERVIEW OF ORACLE'S CURRENT ESB OFFERING

Oracle Enterprise Service Bus

Introduced with the 10gR3 release of the SOA Suite in 2006, Oracle Enterprise Service Bus has seen one major and two minor releases.

Flow definitions are captured as metadata and loaded in an easily clusterable stateless engine. The product follows the strong emphasis on standards from the Fusion Middleware stack: XML, XSD, XPath and XSLT for data definition,

manipulation and transformation, JCA for adapters, WSDL for interface definition, JMS for messaging and Java EE as the runtime platform. This commitment to standards allows the product to be hot-pluggable and it runs today in a variety of platforms including Oracle Application Server, BEA WebLogic, IBM WebSphere and JBoss.

The primary audience for the Oracle Enterprise Service Bus was application developers and integrators who need an intuitive and declarative way to achieve service mediation as they built SOA-based composite applications. As such the primary design-time environment is JDeveloper, complemented with the ability to modify routing rules at runtime using the web-based management console.

Core Differentiation:

- Advanced ERP integration with specific features such as:
 - Domain Value Maps (DVM) to execute on the fly data substitution
 - Embedded Cross-Reference engine to adapt business objects (ex: customer object) to the multiple source and target formats (ex: Siebel CRM to SAP, PeopleSoft, etc.) as they flow through the bus.
 - Very rich, standards-based (JCA) adapters offering, benefiting from the deep expertise of Oracle in ERP
- Built-in global policy management leveraging the advanced capabilities of Oracle Web Services Management, allowing designers and operators to attach policies (security or others) to endpoints at any step of the lifecycle of a project: design, deployment or monitoring.
- Evolution into the 11g Service Infrastructure, with the ability to implement routing rules and transformations directly within SOA composite applications

Oracle Service Bus (formerly AquaLogic Service Bus)

AquaLogic Service Bus (ALSB) is a very mature product and a market-leading standalone Enterprise Service Bus. It was introduced in 2005 and the latest release in February 2008 marked its 2nd major version. After the acquisition of BEA by Oracle the product has been rebranded Oracle Service Bus (OSB).

Oracle Service Bus is also entirely driven by standards and applications are based on XML metadata. In addition to being able to work with XML-based data, the engine can propagate payloads in their native formats. This capability is key to the superior performance of the engine in high-volume environments.

AquaLogic Service Bus' main user base has traditionally been the operations team in charge of SOA infrastructure and, as such, the ideal design environment for this audience was a web-based console. As a result, end-to-end ESB flows can be designed in the thin-client web UI of ALSB. Any change is tracked and audited

using sessions – this ensures complete accountability as well as the ability to revert any operation. ALSB also supported a full IDE-based design environment in its latest major release.

Core Differentiation:

- **Virtualized Endpoint Management:** Ability to specify multiple endpoints for a business service allowing service pooling, load-balancing and failover. Support is also provided for throttling service invocations to protect them from overload.
- **Configuration framework:** ALSB checks the integrity and semantics of all artifacts to ensure runtime integrity. If new artifacts are introduced to break the application, references are given to find and eliminate any issues
- **Advanced Mediation patterns:** ALSB provides support for patterns such as Split-Join allowing true parallel service invocation aggregating results across parallel split. This pattern, often called “service chaining” is especially helpful in low-latency, high-volume scenarios.

Complementary Service Mediation Use-Cases

As described, Oracle and BEA have focused on different use cases and audiences with their respective Enterprise Service Buses. Oracle ESB’s primary mission was to provide service mediation and application integration capabilities to SOA composite application developers. This use-case can be described as “in-line mediation” since it is used for transformation, routing and event delivery inside of what is logically a composite application. BEA AquaLogic Service Bus on the other hand focused on solving the operations and scalability problems in a standalone service bus. Here the use-cases that it focused on were those of service virtualization, on-ramping and aggregation of external services and gateway functionality. This is what makes Oracle and BEA’s approaches so complementary and enables the evolution of both Oracle ESB and AquaLogic Service Bus into the Oracle strategic roadmap.

THE ROAD FORWARD: A NO-RISK CONVERGENCE

Convergence Strategy Overview

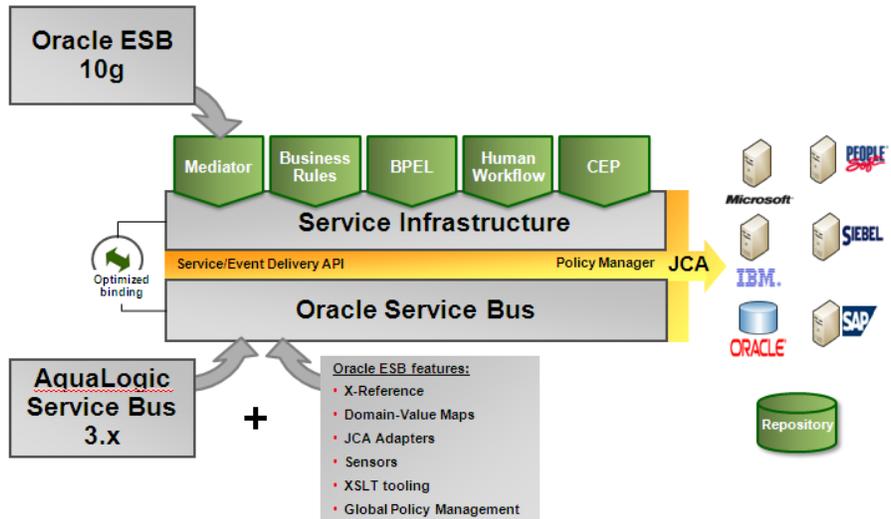
By combining the respective strengths of Oracle ESB and AquaLogic Service Bus in the Oracle SOA Suite platform, Oracle will enable all the common ESB use-cases and provide unmatched flexibility and functional capabilities.

The strategic platform going forward will consist of two main components:

- **The Mediator** is an intra-composite mediation component that is deployed within an application. It is responsible for brokering communications between components that make up a composite, enabling transformation, routing, event delivery and payload validation inside the composite. The main users of the Mediator are application developers

assembling multiple components into a composite application which can be deployed, versioned and managed as a single deployment unit (all based on the SCA, or Service Component Architecture, standard). This is the component that all the Oracle Fusion Applications are leveraging, as well as AIA – the Oracle Application Integration Architecture. Oracle ESB seamlessly evolves into the Mediator in the 11gR1 release.

- **The Oracle Service Bus** provides standalone service bus capabilities, enabling separation of concerns between application developers and target systems or services. The main users of the Oracle Service Bus are integration developers and operations personnel. Their mission is to shield application developers from changes in the endpoint services or systems and to prevent those systems from being overloaded with requests from upstream applications. In the 10g and 11g releases, the Oracle Service Bus will also be augmented with key features from the current Oracle ESB, such as domain-value-map, cross-references, JCA adapters and a JDeveloper-based design-time. The Oracle Service Bus is based on Aqualogic Service Bus, with the incorporation of some of the advanced features from the Oracle SOA platform.

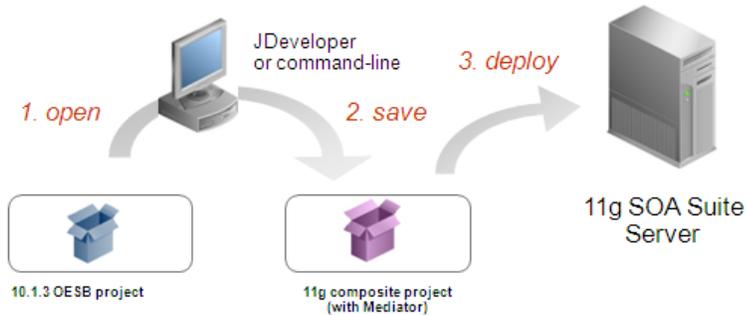


Overview of the Oracle ESB strategy and roadmap

Seamless Upgrade

Upgrade paths to 11g are fully automated, for both Oracle Enterprise Service Bus and AquaLogic Service Bus customers

Oracle Enterprise Service Bus



Overview of the Oracle ESB upgrade path from 10g to 11g

The upgrade path for Oracle Enterprise Service Bus users from 10g to 11g is fully automated. Oracle Enterprise Service Bus users simply open 10g ESB routing rules in JDeveloper 11g. At this point, JDeveloper converts them to Mediator components that can be deployed to the 11g service infrastructure. All adapter configurations, XSLT transformation and routing rules are preserved.

AquaLogic Service Bus

For AquaLogic Service Bus users the upgrade path is similarly seamless: moving from 10g to 11g is a simple upgrade and the backward compatibility of metadata and projects is honored as it has been with past product versions, when moving

from version 2.6 to 3.0 for instance. Again, no manual changes are anticipated for current users to upgrade to 10g and 11g based versions of the Oracle Service Bus.

TECHNICAL DETAILS OF THE CONVERGENCE

Incorporating Oracle Service Bus (formerly AquaLogic Service Bus) to the Oracle Platform

Since the 10g release of the SOA Suite, Oracle has successfully been executing on the plan of providing a rational and cohesive platform with a single design-time environment, single runtime environment and single management & monitoring environment. The net result for customers is increased usability and greatly simplified management.

In future releases, the Oracle Service Bus will run on this common platform and the following sections describe the various tasks involved.

Connectivity & Adapters

Oracle has consolidated all connectivity strategies of the integration products using a standards-based, in-memory and transactional, adapter framework leveraging JCA (Java Connector Architecture). This JCA framework is being integrated in the Oracle Service Bus in the short-term. This will open a wealth of new connectivity options to new customers as well as greatly enhancing the connectivity solutions available to existing BEA customers. The Oracle industry-leading adapters will be certified for use within OSB in a staggered fashion, starting with AQ (Advanced Queuing), RDBMS and Oracle Applications. Other adapters, including adapters for other packaged applications, mainframes, etc will follow in short order.

Hot-Pluggability

AquaLogic Service Bus had until now only been certified on BEA WebLogic Server. Oracle is committed to make this key product available on 3rd-party Java EE platforms, like the rest of the SOA Suite and as such the Oracle Service Bus will be “hot-pluggable” in a future release.

SCA integration

The Oracle SOA Suite 11g platform is built on a key standard called Service Component Architecture (SCA). BEA, a founding member of the Open SOA initiative behind SCA, had also made strong commitments to SCA. Moving forward, OSB application packaging and tooling will be enhanced to fully participate in the Oracle SCA model unifying the runtime and the design-time.

Security & Policies

Both Oracle and ESB had chosen WS-Policy as the standard for advertising and specifying policy requirements such as security or Quality of Service (QoS). In addition, Oracle had extended the Oracle Web Services Manager (OWSM) capabilities and made it the foundation for global management of policies. The

ability to attach, detach, author and monitor policies in a central fashion will be extended to the Oracle Service Bus (as it has been extended to all other components in the SOA Suite 11g).

Design-Time Environment

JDeveloper is the IDE of choice for SOA development at Oracle.

AquaLogic Service Bus users have traditionally been using the web-based console to design, manage and monitor ESB flows. The web console is a key differentiator for ALSB (now Oracle Service Bus) and will therefore remain a primary design environment moving forward.

The web-based console was complemented in the 3.0 release with full IDE support in Eclipse. The Eclipse capabilities will be ported to Oracle JDeveloper to provide a one-stop IDE for all SOA design needs. Until that port is complete, customers will still be able to use the Eclipse environment as IDE (in addition to the web-based console).

Data Transformation

There are two main languages for designing XML transformations: XQuery and XSLT. Both of them are widely adopted, each presenting advantages in specific areas. Remarkably, BEA and Oracle had taken very similar but complementary approaches: at runtime both could execute XSLT or XQuery, but on the tooling side BEA had invested in XQuery while Oracle had invested in XSLT. Moving forward, customers will have the option to define transformations using either language, based on their personal preferences or use cases – a flexibility often requested in the past.

Business Event Monitoring

Oracle Service Bus (formerly AquaLogic Service Bus) is by default a stateless engine, to guarantee speed and concurrency. However it has a functionality called message reporting, allowing administrators to save very specific portions of messages traveling through the infrastructure (for instance an order id). This feature is very similar to the Oracle concept of trackable field and sensors – these sensors being also the mechanism to push data to the real-time Business Activity Monitoring (BAM) dashboards. The concepts of message reports and sensors will be merged moving forward, allowing BEA customers to push instance data not just to a database but also to BAM dashboards.

Closed Loop Governance

Enterprise Manager provides end-to-end management & monitoring across Oracle products (database, applications and middleware) and beyond (via extensible management packs). Prior to the acquisition, BEA was executing on a strategy to provide closed-loop governance between the design-time and run-time environments with the Governance suite comprised of Enterprise Repository,

Registry and OEM'd product from Amberpoint re-branded ALSM. This vision is now being realized with a much more mature set of features provided by Enterprise Manager.

The first step is integrating Oracle Service Bus service metrics and configuration into Enterprise Manager and this work is already underway.

CONCLUSION

The converged Oracle Service Bus and SOA platform will continue to offer all current Oracle ESB and AquaLogic Service Bus features

Upgrade is fully automated for both Oracle and BEA customers

Oracle's ESB strategy after the BEA acquisition is simple: evolve the two complementary service bus products into our best-of-class SOA platform. Because the companies' respective products were addressing different use-cases and targeting different user bases, this strategic roadmap can be implemented cleanly and rapidly. And best of all, the upgrade is seamless for both Oracle and BEA customers, with fully automated upgrade paths to the SOA Suite 11g platform already available in developer preview today. The resulting Oracle Service Bus and SOA Suite will offer an unparalleled range of features, superset of the two individual products' features, and tighter integration to the Oracle FMW platform.

Purpose

This document provides an overview of features and enhancements included in future releases of Oracle Service Bus and the Oracle SOA Suite. It is intended solely to help you assess the business benefits of leveraging the Oracle Service Bus and Oracle SOA Suite and planning for the implementation and upgrade of the product features described.

Disclaimer

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle Software License and Service Agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.



Oracle Service Bus Statement of Direction
August 2008
Author: Demed L'Her
Contributing Authors: Deb Ayers, David Shaffer

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Copyright © 2008, 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. 0408