

WHITE PAPER

Oracle Builds Comprehensive SOA Platform

Sponsored by: Oracle

Sandra Rogers

Stephen D. Hendrick

January 2005

Architecting IT Solutions Today to Address Tomorrow's Requirements

Historically, most systems and applications have been designed to address explicit business requirements. However, by the time these solutions are implemented, the demands placed on the business often have changed. The challenge posed in today's complex and volatile business atmosphere is to create an IT environment designed for the unknown, with detailed business policies and rules abstracted from fixed functional and operational capabilities. It is the ability to deliver these abstracted capabilities that underlies the strength of service-oriented architecture (SOA).

A well-constructed, standards-based SOA can empower a business environment with a flexible infrastructure and processing environment by provisioning independent, reusable automated business and systems functions as services and providing a robust and secure foundation for leveraging these services. Efficiencies in the design, implementation, and operation of SOA-based systems can allow organizations to adapt far more readily to a changing environment.

Increased Productivity and Extensibility

Unlike the large, monolithic batch applications of the past, SOA-based solutions consist of autonomous services and processes that can be invoked dynamically. The ability to execute just the required services and eliminate spurious processing steps is a key benefit.

Many processes can be boiled down to similar, if not same, actions. If an action that can be used within varied processes and environments is defined generically, it need be codified only once. The ability to reuse and only extend for localized requirements can lead to simplified maintenance and mitigate risk.

Governing the Enterprise

Enterprises are held accountable to a variety of internal and external parties — whether the federal government, a board of directors, or a self-defined business policy. Many organizations today are reviewing and adjusting procedures and enhancing audit and reporting capabilities to address fiduciary and industry regulations such as Sarbanes-Oxley, HIPAA, Patriot Act, and BASEL II.

Whether enterprises are working to achieve internal productivity and business goals or to honor external commitments and contracts, they face increasing demand for not only concise and consistent metrics but also reliable control of operations. Within the

constructs of an SOA, utilizing centralized registry and metadata management can improve systems and business visibility and provide greater control and flexibility in defining and adjusting business rules and parameters.

SOA Design Requirements and Principles

SOA represents a significant step forward in "abstracted development." It is based on the combination of and interaction between services, associated with messages, and governed by policies. Although SOA can be implemented without the use of Web services, these should be seen as a primary example of a message delivery model for this architecture. Standards are key to enabling interoperability as well as key issues including quality of service (QoS), system semantics, security, management, and reliable messaging. However, SOA should be differentiated from the use of Web services because it is far more than just a convenient means of making components interoperate. SOA also is the precursor to introducing more technically sophisticated approaches to IT such as event-driven architecture (EDA), where dynamically triggered messaging is a key focus.

SOA Design Requirements

SOA represents a structured approach to application development, which results in higher levels of application scalability, maintainability, and extensibility. IDC believes that enterprises must adhere to the following design requirements in order to capture these benefits. The following do not define *how* to implement an SOA but rather speak to the basic functionality that must be resident in a system that is SOA-compliant:

- ☒ **Loosely coupled.** Services are designed to be autonomous and independent of implementation. This enables greater flexibility in introducing new services, aggregating existing services into composite solutions, and changing the deployment of services versus having to address inherited levels of dependence.
- ☒ **Service encapsulation.** This involves the separation of the interface to the service from the detailed internal specification and implementation of the service.
- ☒ **Interface standardization.** A standardized vocabulary to service interoperability ensures interservice communication.
- ☒ **Shared semantic framework.** This is a method to ensure the comprehension of the content of messages communicated between services.

SOA Design Principles

Within an SOA foundation, core infrastructure and support elements themselves should be designed and leveraged as services. These address such critical aspects as security; management and monitoring; as well as data and access services, event and message processing services, and the like. Because these functions need to be "always on," it is best to build in robust runtime capabilities.

A number of characteristics, while not necessarily requirements, significantly improve the operation of an SOA. These architectural features, which are referred to as design principles, are as follows:

- ☒ **Service discovery.** Provides a mechanism to locate and provision available services and facilitate service administration
- ☒ **Service security.** Manages identity and security policy and enforcement, ensuring proper access to data, messages, and services
- ☒ **Service-oriented management.** A facility to define, enforce, and monitor systems policy in a distributed, network-based environment in adherence to business rules and service level agreements (SLAs)
- ☒ **Service integration.** Facilitates the processing of messages, service-enabling interfaces, and the transformation of information to common formats
- ☒ **Event-driven architecture (EDA).** The use of messaging as the catalyst for service invocation (Events are a type of message and signal a change of state. Business processes, governed by business rules, respond to events. Events may also be routed to other systems or users through alerts.)
- ☒ **Service provisioning.** Core systems and application deployment infrastructure for service provisioning, load balancing, and managed runtime services moving beyond the device or server to address a dynamically networked environment
- ☒ **Service rendering.** Handles transformation for presentation to various "clients," with a typical access point being an enterprise portal or mobile device
- ☒ **Service development and life-cycle management.** An environment to design, develop, and manage service artifacts for an SOA environment

SOA Considerations

To deliver on the promises that SOA offers, many enterprises will require intense reviews of their respective IT environments to iron out predefined and undefined business processes and specifications. In many cases, this review will induce and, at times, require institutional change such as reorganizing resources and imposing a behavioral shift that cuts across organizational boundaries.

SOAs are highly network-intensive solutions that require complex transport services for routing and monitoring. Advances beyond agent technologies for noninvasive service-oriented security and management technologies are evolving. These are used for defining, managing, and monitoring how services and systems assets interact, bringing levels of visibility and control beyond the container to the service, message exchange, and document levels. Other areas still to be addressed are the complex nature of life-cycle management and support for services.

From a technical point of view, one must consider that industry standards (and the products based on those standards) are still evolving, as is the experience of implementers. Although there has been tempered cooperation thus far, as standards

move up the complexity stack within the QoS and business process domains, fragmentation of vendor support is a looming risk. The realities and complexities of segueing between SOA and non-SOA environments will persist and need to be readily accommodated. Fortunately, SOA can be approached incrementally once the infrastructure and core capabilities are in place.

The involvement of business industry leaders in helping define generic process and industry-specific definitions becomes ever more critical as a first step to ensuring that solutions are created that can facilitate automation and pervasive acceptance.

SOA Platform Reference Model

IDC has established a reference model to graphically represent system "services" that compose an SOA platform and should adhere to SOA design requirements and principles previously noted. Core services include integration, process orchestration, state management, policy, and deployment, and they are surrounded by other services that determine how solutions are built (development), accessed by people and other environments (access), secured and managed (security and management), and utilized (application and data).

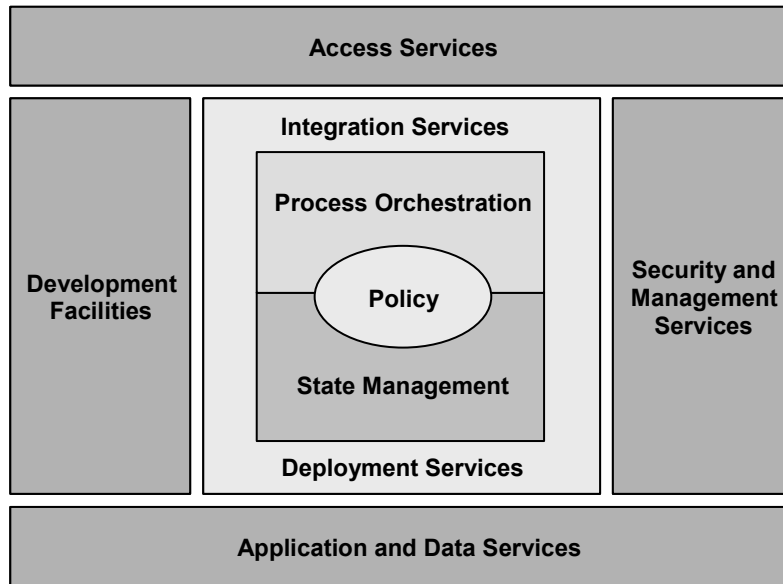
Figure 1 shows the high-level services necessary to support an SOA initiative. It is important to note that these building blocks do not account for all IT activities but simply those that are germane to implementing an SOA.

The following items describe *core* SOA platform services:

- ☒ **Deployment services.** To host and manage the operational, runtime functions
- ☒ **Integration services.** Data and process-style integration across operating environments and platforms with support for demand-driven (request/reply) or event-driven interoperation
- ☒ **Process orchestration.** Organizes and aggregates services into flows to automate system and business processes
- ☒ **Policy.** A business policy or systems rule or condition that governs an action (Business rules are the foundation of business processes.)
- ☒ **State management.** The ability to recognize, support, and manage entity state, thereby providing support for processing governed by state transitions

FIGURE 1

SOA Platform Reference Model



Source: IDC, 2005

The following items describe *supporting* SOA platform services:

- ☒ **Access services.** Reliable and secure system and people-based access to services and necessary system artifacts within the SOA
- ☒ **Development facilities.** Full life-cycle support and versioning of services and messages, including modeling, coding, debugging, testing, deployment, and change control
- ☒ **Security and management services.** Service and process monitoring, management, security, and ID management
- ☒ **Application and data services.** Services built around the supporting data persistence and data semantics

Today, there is a tremendous amount of overlap in some technologies as they converge to address these needed infrastructural services. For example, application services typically address limited aspects of provisioning, management, and integration. Some integration and deployment technologies also embed rudimentary orchestration and messaging capabilities. Policies are actually defined at all levels of the SOA computing landscape and typically reside in multiple technologies.

A platform that does not leverage shared registry and repository capabilities, cross-technology metadata management, and end-to-end service-level monitoring can result in not only tremendous waste because of redundancy and extra transference steps but also increased risk to the semantic integrity of the environment.

The Role of Standards

Adopting a core set of industry-accepted standards is critical to realizing the benefits of an SOA and should be considered a design requirement. First and foremost, these standards facilitate interoperability between components of the SOA, tackling network protocols, message processing, and basic system-level semantics concerning services and processes. They provide a foundation for ubiquitous connectivity for self-describing and secure messages, documents, and services.

For all practical purposes, Web services standards, including XML, SOAP, WSDL, UDDI, and WS-X (where X is a placeholder for standards in process in the areas of security, event routing, message reliability, policy, addressing, federated identity, etc.), continue to be adopted en masse by vendors and users and are quickly becoming the de facto standards around SOA.

Tools are evolving that facilitate composition of complex workflows and dynamic service invocation. These capabilities combine enterprise-level runtime support for asynchronous data and message/event exchanges that are critical for solutions requiring state management and message correlation. Business Process Execution Language (BPEL) has emerged as the leading specification to standardize service orchestration and process automation. Many proprietary products address this aspect of the computing environment; however, when integrating with external processes, the use of such a standard will be increasingly important. The BPEL specification is currently backed by Microsoft, IBM, SAP, BEA, Oracle, and a cadre of other software vendors, and it builds upon the WSDL and XML Schema Definition (XSD) developed by the W3C.

Grid Computing

Although not technically required to fulfill SOA requirements, the grid concept is so symbiotic with the objectives of SOA that it demands inclusion in this discussion. This synergy is so well recognized that the communities addressing standards for Web services and grid computing are beginning to converge.

Grid computing involves the leveraging of a virtual pool of resources (servers, storage devices, databases, network devices, etc.) to support enterprise workloads. These resources can be allocated to different, and parallel, workloads depending on priorities. Grid computing is enabled by a collection of software services that automatically manages this workload supply and demand.

The abstracted benefits from this approach include high flexibility to meet changing business needs and more optimized resources for lower costs. System and functional services that follow SOA design tenets are best positioned to leverage grid services and capitalize on these added benefits.

Oracle's Grid and SOA Strategy

Oracle's grid computing infrastructure, announced in 2003, was a highly significant step by Oracle to promote foundational asset services that both service and are optimized by SOAs. In addition, Oracle DB10g is the first commercial database to be grid-enabled. Oracle's grid infrastructure provides centralized virtualization, dynamic provisioning, resource pooling, automated tuning, and unified management across layers of the computing stack (processors, storage, databases, application servers, and applications).

Above Oracle's grid infrastructure resides a relatively complete array of development and deployment tools including, most notably, Oracle Database 10g, Oracle Application Server 10g (AS10g), and Oracle JDeveloper 10g. A variety of recent acquisitions and licensing arrangements has helped Oracle build out its SOA portfolio, especially in the areas of orchestration, business rules engines, and complex event processing. Oracle's focus on supporting EDA is demonstrated in multiple products such as Oracle/Streams JMS, Oracle Integration Business Activity Monitoring (BAM), and Edge Server/RFID. Oracle has also been systematically SOA-enabling the Oracle E-Business Suite (EBS). This task is largely complete with the new 11i.10 release, which is now also supported by the Oracle Application Server 10g, further unifying the development and deployment platform with its application offerings. This means that the integration features of AS10g can be used to effectively build additional discrete and composite services with EBS as well as integrate with external systems.

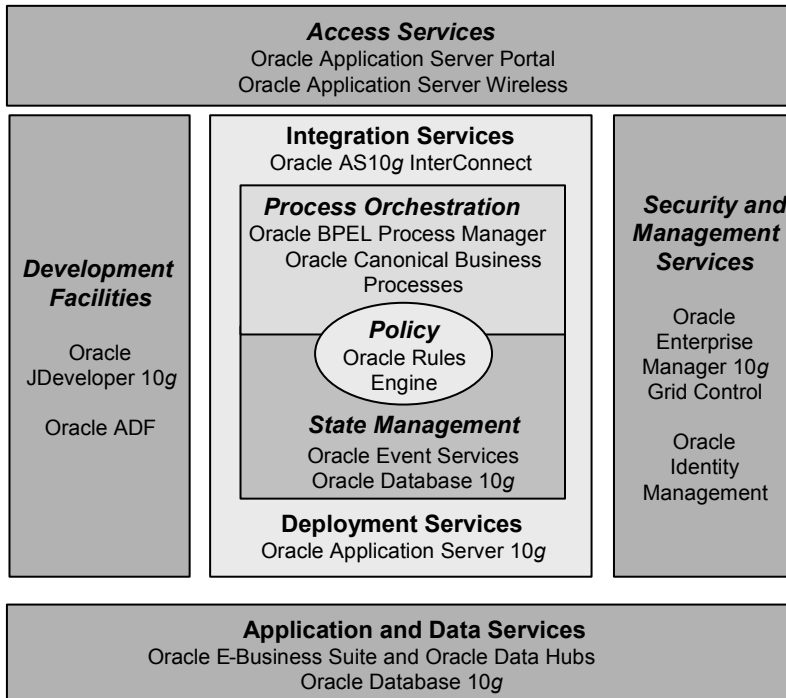
The company positions this technology to aid in creating a single "source of truth" for key enterprise data assets such as information regarding customers and products.

Oracle appears to be well-aligned with the SOA platform reference model and many of the SOA design requirements, providing one or more capabilities for each of the core containers. This is important because it means that there are no serious omissions in Oracle's approach to SOA. In addition, the company is embracing its role in key standards bodies such as the WS-I, W3C, OASIS, and Liberty.

Figure 2 outlines Oracle products and technologies as of Q4 2004 overlaid on the SOA platform reference model.

FIGURE 2

Oracle's SOA Products as of Q4 2004



Source: IDC, 2005

Oracle's Deployment Services

Oracle Application Server 10g (AS10g)

AS10g provides a standards-based deployment platform with J2EE 1.4 runtime services and support for Web services. Oracle has built its portal infrastructure technology, wireless capabilities, content management workflow, integration services, and a set of business intelligence offerings on top of the AS10g infrastructure. These attributes are complemented by grid infrastructure support, which is chartered to provide reliability, high availability, and scalability with fundamental server-level management and security, load balancing, and caching services:

- ☒ AS10g has multiple ways to address scalability through resource pooling. Each hardware instance can run multiple AS10g instances, and each AS10g instance can run multiple JVMs. Furthermore, each JVM can run multiple threads, with each thread running a single J2EE application module.
- ☒ Load balancing can be addressed a number of ways: between HTTP server processes, within OC4J for Web and EJB instances, at the "component" level, or by leveraging a third-party IP load balancing appliance. Clustering is also supported by coordinating a group of AS10g instances.

Oracle's ongoing support of standards is important to ensure portability and interoperability. Some of the critical Web services standards the company indicates it supports include security (WS-Security), reliable messaging (WS-Reliability), policy, and addressing.

Oracle will offer Web services management capabilities, called the WS-Management Gateway, in the AS10g R2 time frame. According to Oracle, this will provide a centralized place to monitor, manage, and enforce policies on services.

Oracle's Integration Services

Oracle AS10g Integration

The integration features of AS10g are of critical importance to Oracle's SOA strategy because they enable interoperability of technologies within and outside the Oracle technology stack and provide the core message processing engine fueling such interactions. To place an increased emphasis on this capability, Oracle recently introduced a standalone integration offering under the name Oracle Integration, complementing the Oracle Business Process Execution Language (BPEL) Process Manager.

In the AS10g R2 time frame, Oracle BPEL Process Manager will add advanced workflow capabilities to the integration offering. Oracle Workflow, which is currently used extensively within Oracle E-Business Suite (packaged applications), can also be leveraged. Oracle indicates that a bridge between the BPEL Process Manager and Oracle Workflow can allow customers who have extended EBS applications with Oracle Workflow to take advantage of BPEL processing for SOA deployments.

Other key technologies within Oracle Integration include:

- ☒ **Oracle InterConnect.** Using Oracle InterConnect (Oracle's Enterprise Service Bus product), which is based on an open messaging backbone, customers can choose to process JMS, SOAP, Oracle Advanced Queuing (AQ), or other third-party messaging structures. Oracle is committed to supporting multiple standards, especially for Web services and Java, and in some instances contributing to their development as in the case of the JSR 208 (Java Business Integration) spec.
- ☒ **Oracle Integration Business Activity Monitoring (BAM).** Oracle's BAM offering, based on XML to define activities and instrument monitoring points, provides visibility of processes and events for reporting, analytics, or alerting. The product provides a dashboard to visualize data in real time and to define critical business metrics and KPIs associated with complex events.
- ☒ **Application and system adapters.** Oracle Integration includes a JCA-based adapter framework with numerous adapters for packaged applications (SAP, PeopleSoft, Oracle, Siebel, J.D. Edwards), messaging systems (MQSeries, Oracle AQ, JMS), Internet transports (ftp/s, http/s, smtp), databases (JDBC, Oracle, DB2, Sybase, Informix, Microsoft SQL Server, files), and mainframes (CICS, DB2, VSAM, IMS).
- ☒ **B2B capabilities.** These capabilities provide support for EDI, RosettaNet, and other B2B standards. Oracle also offers capabilities for trading partner management to automate and manage communications between disparate partners.

Oracle's Process Orchestration Services

Oracle BPEL Process Manager

Through its acquisition of Collaxa Inc. in June 2004, Oracle gained technology (now called Oracle BPEL Process Manager) for coordinating Web services and orchestrating business processes.

Oracle BPEL Process Manager provides native support for the BPEL standard and acts as a Web services orchestrator by specifying the order in which it is meaningful to call a collection of services and assigning responsibilities for each of the services to participants ("partners") that act as service providers or consumers in a one-way or two-way interaction.

This technology is a critical element of Oracle's SOA offering because the orchestration of Web service processes is a necessary and key evolutionary step in transforming its application development environment from being component-oriented to service-oriented.

Oracle Canonical Business Processes

Targeted at accelerating time to market, Oracle Canonical Business Processes are prepackaged horizontal and industry-specific business applications that encapsulate workflows and elements of the Oracle E-Business Suite. They have been built out of what the company had termed "Business Flow Accelerators," initially developed by Oracle Consulting and systems integrator partners.

While we know little about the degree of standardization around the data and process models supporting these business processes, we would expect alignment with Oracle Data Hubs strategy (explored in further detail later in this document).

Although many unanswered questions remain concerning the degree of needed customization and extent of indirection that will be built into these processes, the strategy is desirable for many enterprises that would prefer to buy versus build such solutions from scratch. However, this road is not yet easily traveled. SAP, Siebel, and others have made significant investments in this space, yet the market has been slow to pick up on these offerings. While we can attribute much of the apparent ambivalence of customers to the newness of this technology, Oracle stands to benefit from the efforts of others to perform important evangelism for these next-generation applications.

Oracle's State Management Services

Oracle Event Services

There is a symbiotic relationship between events, state management, and business rules. Simply put, events trigger a change in state, and business rules then dictate the state transition. Therefore, an event server becomes a critical enabling technology in support of business rules engines, business process orchestration, and business process automation. Based on licensed technology, Oracle's event server monitors distributed sources of data and events, correlates and detects events in real time, and alters and/or triggers application flows upon detecting an event and managing event scenarios. Oracle Integration BAM has event processing capabilities at present.

Oracle has plans to make available a separate event server component that not only will be leveraged in the Oracle Integration BAM product but also will function independently.

Oracle Database 10g (DB10g)

While DB10g is well-known for many industry-leading features, its inclusion here is primarily in recognition of the key role that state and persistence play in an SOA. Because databases already address data persistence and state, they are likely to serve as a foundation for overall SOA state management. Naturally, performance demands will necessitate additional state management constructs, which will either be extensions to the database or perhaps materialize as finite state machines. Either way, these capabilities will be part of the SOA core capabilities.

Oracle's Policy Services

Oracle Rules Engine

The Oracle Rules Engine is a Java-based rules engine that was developed during the 1990s at Sandia National Laboratories. Oracle has licensed this technology and intends to deliver it in Oracle Application Server 10g Release 2. A JDeveloper authoring environment will follow.

Using business rule engines (BREs) is an appealing approach to building highly complex and scalable business processes, and although IDC is very optimistic about the future of BREs, adoption of this type of technology has been somewhat measured to date. The level of sophistication required to bring together all of the supporting facilities, such as rules-based development and management as well as design specification skills, is still evolving in the marketplace.

Oracle appears to be off to a good start with extensions being developed for JDeveloper and its acquisition of an event server, but building a complete event-driven business rules management system will be a time-consuming effort. However, the payoff should be a technology that not only provides ideal support for SOA but also positions Oracle beyond SOA for event-driven processing.

Oracle's Access Services

Oracle Application Server Portal (OracleAS Portal)

The OracleAS Portal component of AS10g provides access to information and application services and facilitates user interaction with business processes. OracleAS offers a standard array of portal development tools, content management and publishing features, and deployment support, with the key SOA capability focused on the rendering of services. Through Oracle AS10g Integration capabilities, sources can include both Oracle and third-party technologies such as packaged applications, transaction and messaging middleware, and databases.

Oracle Application Server Wireless (OracleAS Wireless)

OracleAS Wireless enables applications written in XML to be accessed from any browser or voice-enabled device. The OracleAS Wireless architecture is intended to provide independence from the underlying networks, protocols, devices, and gateways by abstracting these wireless complexities to one protocol (HTTP) and one language (XML). OracleAS Wireless also includes a set of location, alerting, and content syndication services to support faster application development.

One of the key SOA-based features of OracleAS Wireless is its messaging service. This messaging service is bidirectional and is based on open technologies, including Web services and XHTML. Its capabilities include many types of "push" technologies, such as alerts, notifications, subscriptions, and provisioning.

Another interesting feature of OracleAS Wireless is support for RFID. Oracle will deliver RFID software capabilities as part of OracleAS Wireless (10g R2), which provides "sensor-based services."

Oracle's Development Facilities

Oracle JDeveloper 10g

Oracle JDeveloper 10g is an integrated development environment (IDE) for building applications and Web services using the latest industry standards for Java, XML, and SQL. Thus it provides a mechanism to:

- Build standards-based solutions with rich user interfaces (UIs)
- Service-enable existing applications, including legacy applications and PL/SQL stored procedures
- Define service orchestration rules (although how JDeveloper interfaces and integrates with the Oracle Rules Engine will evolve over the next 12 months).

JDeveloper generates the necessary WSDL code to ensure that any Java class or PL/SQL package can be Web service-compliant and can facilitate development activities by enabling the consumption of Web services at design time. The product also supports the creation and deployment of UDDI repositories and generates code stubs that can be used to activate Web services. Oracle is committed to ensuring that JDeveloper code conforms to WS-I standards as they evolve and continues to support J2EE, core XML, and an XML Schema editor.

Oracle is developing a plug-in for orchestrating services, which would provide a unified interface for building both business processes and J2EE applications within JDeveloper. This plug-in is an outgrowth of the Collaxa Design Time Environment, adding BPEL capabilities to existing JDeveloper UML Modeling and Data Modeling.

While there is no doubt that JDeveloper provides numerous SOA constructs, Oracle still has some distance to travel before JDeveloper can be considered a full life-cycle IDE. The rationale for this statement is the limited degree of model-driven development functionality and the continued reliance on third-party support for industrial strength features in the areas of team development, change control, and testing.

Oracle Application Development Framework (ADF)

Oracle ADF is a toolset within JDeveloper for building service-oriented applications based on a three-layer model:

1. **The view layer** provides the interface to the application or service and can comprise HTML, Java components, or XML to render the interface. The view target can be Web-based, client/server, or wireless.
2. **The controller layer** encapsulates applications or process workflow. The controller is tasked with managing the flow between pages for Web-based systems. ADF uses the Apache Jakarta Struts open source controller for Java-based Web applications.
3. **The model layer** connects services to the objects that use them in the other layers. Consequently, this is where components of business services are bound to the controller and view layers. Oracle's TopLink Mapper provides relational-to-object data mapping capabilities; however, alternative approaches can also be applied, including the use of Java classes, Web services, or Oracle ADF business components.

The objective of ADF is to enable developer productivity through a visual and declarative environment with a choice of technologies and development styles, including models (Web services, EJBs, or TopLink), controllers (ADF controller or Struts controller), and view technologies (JavaServer Faces [JSF], UIX, or Swing).

Oracle supports JSF — a standard UI framework for Java Web applications. JSF helps abstract the business logic and presentation layers, thus separating components from their appearance. This also allows for UI rendering on different devices. Oracle supports this feature by adding metadata descriptions so no interface coding is necessary. Instead, the company indicates that developers only need to incorporate an XML metadata description to achieve a custom or standardized rendering. Along with support for Web services standards, ADF enables many different interface classes and the decoupling of these interfaces with the underlying service or component.

Oracle's Security and Management Services

Oracle Identity Management (IdM)

Due to the highly distributed nature of SOA, a common model for centralized and role-based security and identity management is needed. Additional capabilities for handling aggregated service security for complex composite solutions are also warranted.

Oracle's IdM is targeted to provide a consolidated infrastructure to define identities, roles, and security policies, leveraging capabilities from Oracle's Phaos acquisition. Based on LDAP and PKI standards, the core IdM functionality revolves around directory services to manage user identities and access, Web single sign-on, provisioning services, digital certificates, delegated administration, and certificate authority. These capabilities also extend into legacy environments.

Oracle IdM supports federated identity management (Liberty) and FIPS-certified capabilities; meanwhile, the company is in the process of building Security Assertion Markup Language (SAML) and WS-Security functionality to support service-level security, which will be available in later releases. Oracle is focused on also providing additional standards support in its IdM offering, including WS-SLC, XML Dig Sig, XKMS, and XACML, all of which are geared to propagate identity between services.

Oracle Enterprise Manager 10g Grid Control (EM10g)

Enterprise Manager is perhaps Oracle's single most important product because it enables the management of all applications and systems that are based on the Oracle technology stack, including AS10g and DB10g, as well as overall control of grid resources. EM10g provides monitoring (application and system), administration (target and user), and configuration (inventory, patch, deployment, and certification) of as well as information (alerts, reports, portals, warehouses) on a wide range of software and nonsoftware resources, including Web sites, applications, databases, application servers, host systems, storage systems, and networks. Therefore, EM10g manages system assets that can operate within an SOA but does not provide direct service-oriented management. Oracle indicates that with the R2 release a Web Services Management Gateway will be included with both AS10g and EM10g.

It should be mentioned that EM10g does use SOA and EDA design guidance internally in the configuration of events to support its monitoring capabilities, policy definition at the service and service aggregation level, and predictive alerting (evaluating rate of change).

Oracle's Application and Data Services

Oracle E-Business Suite (EBS)

Oracle E-Business Suite continues to undergo a significant degree of reengineering to support SOA, some of which is reflected in its 11i.10 release. Some key design changes involve separating service interfaces from their implementation and leveraging AS10g to provide core infrastructure with its J2EE runtime, registry, and repository services. Oracle indicates that its EBS also includes an integration repository for access to all of its service interfaces, along with over 800 preconfigured "business events," 150 OAG "messages," and Web services and industry-specific protocol support. These changes are to help facilitate service discovery and easier implementation and integration of added modules and other user-defined services.

Oracle Data Hubs and Oracle Database 10g

A harmonized and consistent view of enterprise data across services is important to deploying composite applications. For instance, when pulling together a cross-marketing solution built using customer order data from a Siebel application, customer shipping information from an SAP application, and financials data from an Oracle application, these three versions of customer data need to be in sync for the application to process accurately. The use of an abstracted data layer in an SOA promotes a more centralized, structured, and maintainable data environment for operations, analytics, and decision making.

The purpose of an Oracle Data Hub is to establish a centralized reference point for varied data sources with a unique, canonical data model, to which other data schemas may be mapped and reconciled. These hubs are oriented around general use scenarios, such as the Customer Data Hub and the Product Data Hub. According to Oracle, these products also include data integration and quality services that can synchronize with many of the leading ERP applications (SAP, PeopleSoft, and Siebel) as well as legacy applications. Oracle positions Data Hubs as a way to allow SOA developers to focus less time on data inconsistencies across applications and their interfaces and spend more time on addressing the ultimate business process.

Beyond data persistence, metadata management plays a key role in an SOA environment. DB10g is positioned to provide these capabilities as well as act as the foundation for technologies such as the Data Hubs, therefore establishing it as a core component of Oracle's SOA offering.

Conclusion and Recommendations

SOA is garnering a tremendous amount of interest for a variety of reasons, but primarily because of its near-perfect alignment with the core principles that govern how people and the constructs created by people, such as business processes, operate. On a more tangible level, SOA provides an efficient model for the development, deployment, use, and integration of IT assets.

We have watched the software industry over the years grasp and master first reusable functions and then reusable objects. Now it is poised on the brink of reusable services. A number of key issues have converged to enable this development:

- ☒ An unprecedented level of industry acceptance of standards for interoperability and system-level semantics
- ☒ Availability of software enabling higher levels of developmental productivity utilizing code generation, analysis, modeling, and orchestration methods
- ☒ Introduction and familiarization with technologies focused on more granular identity-based management of IT activities and processes characterized by higher levels of abstraction

SOA is also a stepping-stone and foundational infrastructure on which to support grid and event-driven processing, which releases IT from being merely reactive (as in a classic demand-driven request/reply architecture) to being proactive, leveraging automated triggers for closed-loop processes and faster responses to set conditions. It is also a core element of the embryonic "on-demand" vision being espoused across the IT industry. Looking out on the horizon, we expect advances in information-based search, analytics, and correlation that will push the envelope further to realizing this ideal.

Conclusion

Currently, Oracle's greatest strength in SOA is derived from the scope of its Enterprise Manager and Application Server products. The sheer volume of codesets in Oracle AS10g speaks to its functionality, and the company's commitment to J2EE and Web services standards is obviously critical. The acquisition and integration of

the Collaxa BPEL Process Manager elevate AS10g's orchestration capabilities beyond those of most other competing application servers. However, the application server market is a fiercely contested battleground, and volatility in technical leadership remains high. Oracle's acquisition of a business rule engine, its licensing of an event server, and its own internal innovation focused on supporting canonical business processes impart key differentiating SOA support and EDA capabilities.

Oracle ADF, with a three-tier approach to combining business services with models, controllers, and views, recognizes the needed scope and flexibility in decoupling service interfaces from service implementations. When JDeveloper's support for conventional Java, XML, and Web services development is added to the mix, the result is a creditable, SOA-enabling IDE.

The combination of Oracle's EBS, Data Hubs, and Application Server is targeted at bridging the gap between packaged and custom solutions. The added capabilities of its BPEL Process Manager and Integration offerings help enable the development of composite applications that can leverage existing applications as services. With the expected eventual integration of the Process Manager into the portal codeset, Oracle should have significantly more SOA dimensionality for data and service access and B2B process support than it does with its current Process Connect solution.

The net result creates a strikingly positive accounting of the current and proposed ability of Oracle products to support SOA.

Challenges and Opportunities for Oracle

A significant amount of Oracle's existing user base leverages its core application server and infrastructure services as a default for Oracle applications or data environments, positioning the company already as a leader in deployment software. However, the application integration and deployment arena continues to be highly competitive, and a perspective may exist that Oracle infrastructure will support only its own portfolio; thus, much market education and reference cases from highly heterogeneous scenarios will be required to influence buyers to evaluate these products for their technological merits.

Although Oracle seems to have paid careful attention to selecting product acquisitions that complement existing products and support SOA, how well it continues to extend the capabilities of these technologies will determine how robust and differentiated an SOA foundation is ultimately offered. In addition, Oracle will need to provide an appropriate tool to help developers integrate events, rules, and services within service-oriented applications.

Oracle must also acknowledge that many organizations looking to adopt SOAs have already installed varied components of the infrastructure and would look to potentially acquire particular capabilities versus a bundled package. This is where flexible pricing, packaging, as well as an open technology model supporting accepted industry standards will remain critical. Oracle already offers targeted packages of Oracle AS10g, explicitly around integration, identity management, and business intelligence. Oracle's packaging of the BPEL Process Manager for IBM WebSphere or BEA WebLogic is a further indicator of its commitment to an open technology model.

Companies with complex computing environments are drawn to Web services and SOA as a key IT strategy to address interoperability, integration, and high leveragability of existing systems. It will be critical for Oracle to maintain significant investment in and support of Web services standards, including open participation in committees and interoperability testing initiatives. Oracle also must extend the scope of its technology and business partnerships, especially with best-of-breed vendors.

Customers will demand greater degrees of exposure of Oracle's applications as services that can be accessed and utilized with a shared SOA development toolset and repository, along with overall support from core infrastructure services. Organizations will also seek a common development environment to build composite applications, with less reliance on specialized toolsets for various elements of the SOA. Oracle is aware of these challenges and will be focused on addressing them as its overall offering matures. Moreover, with regard to full life-cycle development and management technologies, organizations should look to Oracle's partners to fill out those very important SOA capabilities.

Recommendations

Most enterprises will adopt SOA in a phased approach, and core standards will continue to evolve. SOA is the architectural paradigm into which vendors continue to pour significant R&D efforts. Consequently, the competitive landscape and products available to address explicit SOA requirements are likely to change rapidly over the next 12 to 24 months.

Our evaluation of Oracle's current product and technology road map with that of a robust SOA framework shows many valuable capabilities, and we believe that Oracle is well-positioned to support the development of complex solutions in this venue. Although late to the game, the company has effectively pulled together a nearly complete set of functionality for SOA environments through a series of pointed acquisitions and significant R&D efforts. Oracle may not have previously been considered a viable player in the Web services and SOA arena, but users would be wise to closely evaluate the next few iterations of the vendor's product line as it continues to integrate new robust capabilities to support SOA and grid computing.

Copyright Notice

External Publication of IDC Information and Data — Any IDC information that is to be used in advertising, press releases, or promotional materials requires prior written approval from the appropriate IDC Vice President or Country Manager. A draft of the proposed document should accompany any such request. IDC reserves the right to deny approval of external usage for any reason.

Copyright 2005 IDC. Reproduction without written permission is completely forbidden.