

Oracle SaaS Platform: Building On-Demand Applications

*An Oracle White Paper
June 2007*

TABLE OF CONTENTS

Introduction 3
 SaaS Drivers 3
 Business Requirements 4
 Technical Requirements 5
 The Next Generation of SaaS 7
Oracle SaaS Platform 8
 Scalable Database for SaaS architecture 9
 Shared Services, SOA and Integration 12
 Customizable Business Processes 13
 Metadata-driven User-Interface 13
 Comprehensive Development Platform 14
 Real-time In-process Analytics 15
 Data Security and Privacy 15
 Identity Management 16
 Data Security 16
 Service-Level Compliance and Configuration Management 17
Conclusion 19

INTRODUCTION

The adoption of Software as a Service (SaaS) or On-Demand presents several technical and business challenges for all players involved – independent software vendors (ISVs), hosting services providers and the customers. The technical challenges for SaaS providers include support for multi-tenancy, security, usability and service-level management. Oracle SaaS Platform— consisting of database, middleware and systems management software— provides a comprehensive platform that addresses these challenges and runs on grid.

In simplest terms, SaaS refers to hosted on-demand delivery of applications, as opposed to on premise. However, software as a service is not a singular model but encompasses a spectrum of choices ranging from hosting traditional packaged applications at a remote site to multi-tenant shared applications.

The Software as a Service delivery model is rapidly gaining adoption. A leading analyst firm predicts that by year 2011, 25% of all net new software revenue will come from SaaS. In order to leverage this growth, ISVs need a platform that can meet the needs of next generation SaaS.

SaaS Drivers

The move to Software as a Service is driven by customer requirements and preferences. Customers are attracted to the promise of faster adoption, minimal impact and cost to internal IT and familiar web-based interfaces similar to consumer web applications resulting in faster adoption and minimal training costs. Although initially limited to small and medium businesses, there is increasing evidence that large enterprises are also evaluating and adopting SaaS for certain applications.

This demand for SaaS delivery is leading new ISVs to consider adoption of SaaS; and many existing ISVs are looking to offer their existing applications in the hosted model – either by hosting existing applications or by re-writing to a new architecture for SaaS.

ISVs can also leverage the following benefits of a SaaS delivery model:

- Access to new accounts/markets: With Software as a Service model, ISVs can target smaller customers who can well afford a subscription price and appreciate the minimal IT footprint.

- **Customer Insight:** This is perhaps the most strategic competitive advantage from SaaS for ISVs. With on-premise delivery, the ISV rarely gets a close look at how the software is actually used by customers- what features are most used, what kind of data is stored etc. With SaaS, the ISV can monitor how the applications are used and develop deep knowledge of usage patterns, usability issues, etc. This insight can help in improved product roadmaps and also provide ideas for what new applications would be valuable to the customers.
- **Value-added services:** The shorter product lifecycle and ability to make frequent upgrades and fixes makes it much easier to bring new and incremental services (applications) to market and test them with real-life users. This combined with better insight makes SaaS an ideal platform for continuous experimentation and innovation.

Business Requirements

Software as a Service differs from traditional on-premise delivery where customers typically have to purchase the software, install, configure and then support it. The ISV developers are only responsible for building and testing the software; and for on-going support and bug fixing. This division of responsibilities is significantly altered in case of SaaS: the SaaS provider is now responsible for day to day operations and ongoing maintenance, including such tasks as capacity planning, patch management and upgrades. The SaaS provider must do all this while meeting its service level agreements for uptime and performance. This imposes significant challenges to ISVs that are used to traditional software delivery where the majority of the effort is in building and shipping quality software.

While imposing new burdens on the ISVs, the SaaS model also simplifies the problem in some ways. If the software is to be offered only as a hosted service, the SaaS provider can now build and optimize the software for a single delivery environment, significantly reducing the burden of porting and testing on multiple platforms. Since the customer is interested in the service and not in how exactly it is built, the SaaS provider can build the software on a platform and architecture of its choosing to optimize for cost and performance.

There are many challenges facing the SaaS provider as they look to build, deploy and manage these services. The following summarizes the key representative challenges and drivers for these ISVs:

- **Service-level Management:** Customers are increasingly asking for well-defined Service-level Agreements (SLAs) around performance, unplanned downtime, data backup and recovery, etc. The application design and the underlying platform must provide the necessary functionality and tools to meet these promises to the customer.
- **Customizable Solutions:** One of the most common concerns expressed by customers considering the SaaS alternatives to traditional on-premise

applications is the ability to customize. Ability to customize data, process and UI can help ISVs sell to more customers and also partner with other vendors to offer vertical solutions.

- **Low TCO to Scale:** The ISVs need to be able to predict and manage the cost of operating SaaS solutions. Unlike traditional software vendors, SaaS providers must take into account the cost of delivering the service, up keep and ongoing maintenance and upgrades. These costs can rarely be passed on to the customer and directly impact the SaaS provider's bottom-line.
- **Time to Market:** One key advantage SaaS ISVs have over traditional software vendors is the ability to expedite time to market. It is therefore important to have infrastructure and methodologies in place that allow the ISV to bring new functions and features to market in much less time than traditional software.
- **Ability to Monetize:** SaaS delivery model allows for incremental upsell at relatively low cost. ISVs want to be able to offer enhancements and complementary functionality or higher service-level guarantees without significant changes in the underlying application or infrastructure. It should be technically feasible to monitor and bill based on metrics such as usage.
- **Security:** The customers are often entrusting critical data to the SaaS provider and must be protected from theft and misuse. In addition, several regulations and laws apply to certain types of data (health records, financial records, government records) and in certain geographies. Breaches in security can not only hurt the end customer but could also damage the SaaS provider's reputation and expose him to the risk of litigation. Therefore, holistic security at all tiers is a prime business requirement.

Technical Requirements

As with traditionally packaged software, developers and architects of SaaS providers must design and architect their applications to meet customers' functional business requirements and build applications that are easy to use, deploy and manage. The on-demand delivery adds several new business requirements and constraints that must be met as discussed above. This translates into technical requirements for the choice of the platform technologies.

The requirements that are new or require special emphasis for SaaS are as follows:

- **Manageability:** ISVs realize that service down time, poor performance and the inability of end users to complete critical business processes can be costly to their business. Service availability and performance problems can lead to high administrative costs spent on problem isolation and resolution – this is especially true if the process is slowed by traditional disparate management tools that do not provide an end-to-end fully integrated view and administration of services and the underlying infrastructure stack. It is

“A pre-integrated SaaS platform means we spend more time on building new services and less time on infrastructure.”

**—VP of Engineering
Leading EDI SaaS vendor**

important to have the ability to model real world services from end-user's perspective where business services are mapped to IT resources; monitor key service indicators for availability, performance, usage and service level compliance; and provide centralized reporting and viewing of service levels from a realtime and historical perspective. In the context of SaaS, where a service provider may have multiple instances of the application and underlying infrastructure running, it is even more imperative that the service level management tools can scale and support hundreds, even thousands of instances – in order to sustain and grow the business profitably.

- **Security and Privacy:** One of the key concerns of customers looking at adopting SaaS is security and identity management. The security infrastructure must ensure that customers' sensitive data and credentials are secure even in a multi-tenant environment. Effective monitoring and auditing tools should be available to track access and usage. Identity management also can allow the SaaS provider to interface with a customer's existing directory and provisioning infrastructure, and delegate everyday tasks like user management to the end user or the customer's IT department. This has the added benefit of lowering costs for the SaaS provider as it is no longer responsible for mundane tasks (such as password resets) that can often result in high help desk costs.
- **Shared services and SOA:** A standards-based service oriented architecture enables the SaaS provider to build the application out as a set of services that can be re-used rather than a single monolithic application. The many benefits of SOA include faster time to market for new applications that can be composed rather than developed from scratch, greater interoperability with existing services within and outside the firewall, increased reuse, business processes that are easy to configure and evolve, and the ability to define and measure response-time and availability for specific services.
- **Meta-data driven Customization:** Customers often have business requirements and processes that may differ from those of other co-hosted customers, and may also not exactly fit the industry standards, if they exist. The on-demand applications must allow for easy customization and extension of user interfaces, data and business processes. This should require minimal to no coding on the part of the end user.
- **Environment-centric design:** Traditional software application developers are often forced to code to and take advantage of only the least common denominator features of various infrastructure software in order to be able to port and deploy on a wide variety of systems, databases and middleware. This *late binding* of the application to the deployment environment eliminates the ability to take advantage of latest advances until all vendors catch up. With SaaS, the service provider can optimize the architecture taking into consideration its choice of operating system, databases and middleware. This

reduces development costs, time to market and cost of maintenance over the life of the application, and can provide higher performance as well.

- **Provisioning, patching and systems management:** The SaaS provider is now responsible for bug fixes, upgrades and patching. With traditional delivery the customer could afford to schedule some downtime but this becomes increasingly harder or impossible to do if the software is being used by multiple customers often across different continents and time zones. The underlying infrastructure and application architecture should allow patches and upgrades with minimal disruption in service.
- **Scalability, high availability and reliability:** The underlying platform must provide for easy scale-out and scale-up of applications and shared services, so that more tenants and more per-tenant users can be added without degradation in service levels. As with traditional on-premise software, the customers want scalability, high availability and reliability from the SaaS provider for mission critical applications. But, the SaaS model means that even for applications that are not very mission-critical it is important to design for scalability, high availability and reliability as hundreds or thousands of customers could now be impacted by planned or unplanned downtime. Furthermore, a multi-tenant architecture and shared services add new complexity to problems like restoring data. For example, how do you (or do you) plan to restore data for one single customer without impacting others; and how do you provide varying levels of service (SLAs).
- **Multi-tenancy:** Single instance multi-tenant architecture is used by many SaaS providers in order to amortize costs across multiple tenants. However, while this approach is suitable for many applications and customers, especially when targeting small and medium businesses, it is not the only architecture for SaaS. In fact, architectures for SaaS vary from hosted applications with multiple instances (ASP) to multi-tenant shared services models. The choice of the appropriate tenancy model will depend on customer requirements such as privacy, security, high availability, customizability, and comfort with the shared model. They also vary based on ISV's business assessments in terms of ability to monetize customer-specific services.

The Next Generation of SaaS

The initial SaaS providers have focused on building web-based applications that seek to duplicate the functionality offered by on-premise vendors over the internet. This move to on-demand coupled with changes in how software is sold has resulted in the first wave of SaaS. But the next generation of SaaS applications must evolve, in several different ways.

Next generation SaaS must allow for customizable processes, embedded analytics, secure multi-tenancy and standards-based integration using SOA

First, just as on-premise applications are evolving and expanding in functionality, so SaaS applications also need to evolve to stay competitive, both with on-premise and with other SaaS options. In particular they must meet customer demands for increased productivity and ease of use, better manageability, and more flexibility. To enable this, the next generation of applications increasingly include such technical features as *embedded business processes*, web 2.0 style user interaction (*Rich Internet Applications*) and *embedded in-process real-time analytics*.

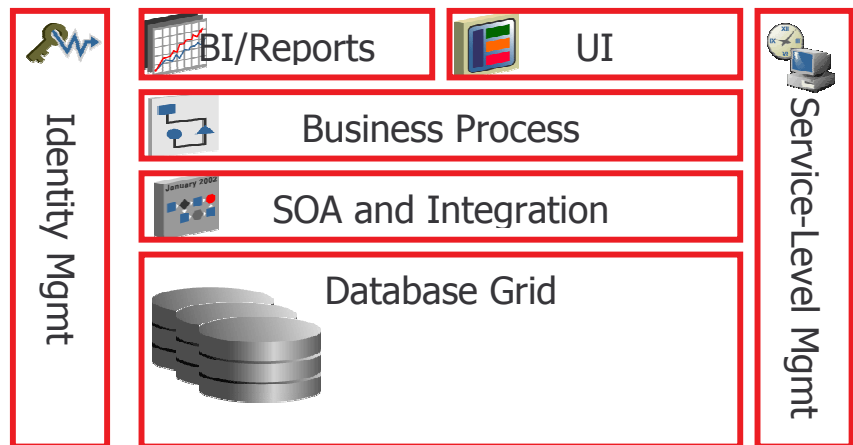
Second, a SaaS application needs strong support for the hosted delivery model. SaaS applications require a security model to ensure customers' security is not compromised by remote hosting and co-tenancy. They also require a scaleable hosting platform and management tools that can handle the extra size and complexity of a multi-tenant installation. In other words, the platform must enable *secure multi-tenancy*.

Third, SaaS applications must often co-exist and interoperate with on-premise applications and systems, as well as other external entities including customers, suppliers and partners. To enable this, conformance to industry standards is important; so are good tools and technology for *integration*.

The Oracle SaaS platform is well suited to serve as the foundation for the next generation of SaaS applications, as we shall illustrate in the following sections.

ORACLE SAAS PLATFORM

A comprehensive Software-as-a-Service platform must provide all the key capabilities outlined above to build and deliver an on-demand application. Oracle SaaS Platform consists of three major components – database, middleware and systems management – that together provide the necessary functionality for a complete and pre-integrated platform.



[Figure 1: Oracle SaaS Platform technologies]

The components of Oracle SaaS Platform are designed to work with each other and the broader ecosystem by adhering to standards and best practices. It is important to point out here that even though there are several advantages to using the entire platform, the platform technologies can be used in a best-of-breed environment;

the service provider can pick and choose the components of the Oracle stack that best meet its needs.

Next, we look at some of the key components and how they meet various requirements for software as a service.

Oracle Database enables secure multi-tenancy with features like RAC, Virtual Private Databases and Database Vault.

Scalable Database for SaaS architecture

The data tier represents a significant challenge in the design, implementation and deployment of SaaS applications. The ISV must decide whether to implement a shared everything multi-tenant data architecture or shared nothing separate databases or other choices between these two extremes. The decision depends on a wide variety of factors including:

- Customization requirements
- Security and data privacy requirements
- Ability to monetize separate instances/business model requirements
- Scale of number of customers and users
- Performance & service-level requirements

The pros and cons of various degrees of multi-tenancy have been studied in quite some detail in the SaaS literature. In this paper, we will focus on the technical issues of design and implementation for the various choices but do not prescribe any particular style as the decision is best made by taking various business and technical requirements into consideration.

Oracle Database provides the ideal platform for multi-tenancy due to its support for grid computing which in conjunction with choice of low-cost commodity servers allows the ISV to scale out at lower cost. Oracle Database exploits the two core tenets of grid computing: virtualization and provisioning.

With *virtualization*, individual resources (e.g. computers, disks, application components and information sources) are pooled together by type then made available to consumers (e.g. people or software programs) through an abstraction. Virtualization means breaking hard-coded connections between providers and consumers of resources, and preparing a resource to serve a particular need without the consumer caring how that is accomplished.

With *provisioning*, when consumers request resources through a virtualization layer, behind the scenes a specific resource is identified to fulfill the request and then it is allocated to the consumer. Provisioning as part of grid computing means that the system determines how to meet the specific need of the consumer, while optimizing operation of the system as a whole.

Oracle Database provides the industry-standard functionality of a relational database plus support for a grid of databases based on Oracle Real Application Clusters, managed through a single console (Oracle Enterprise Manager Grid

Control). The key capabilities of Oracle Database most pertinent to software as a service include:

- Oracle Real Application Clusters for database grid
- Automated Storage Management for storage grid
- DataGuard for disaster recovery
- Tablespaces for ease of data management
- Automatic Workload Management (support for “Services”)
- Virtual Private Database for data privacy
- Partitioning

Oracle Real Application Clusters enable a single database to run across multiple clustered nodes in a grid, pooling the processing resources of several standard machines. Oracle is uniquely flexible in its ability to add nodes on demand to allow the SaaS provider to be able to add resources as more customers and users sign up. Contrast this with having to plan a priori and purchase either a server that is too large by implementing a solution for anticipated peak capacity in the future, or a server that may need to be replaced and upgraded to add additional capacity with change in demand.

In addition to computing resources, SaaS providers also need to manage large amounts of storage. Oracle Automatic Storage Management (ASM), a feature of Oracle Database, provides a virtualization layer between the database and storage so that multiple disks can be treated as a single disk group and disks can be dynamically added or removed while keeping databases online. Existing data will automatically be spread across available disks for performance and optimal utilization. This includes support for multiple databases, which could be at different software version levels, accessing the same storage pool. This flexibility allows the SaaS provider to support several (or all) customers on a single group of disks and can add (or remove) disks without bringing down the database. Due to support for multiple databases, the SaaS provider can take advantage of this solution irrespective of the degree of multi-tenancy including both multi-tenant architecture as well as single-tenant architecture.

Additionally, the Oracle provides Oracle Data Guard, a disaster-recovery solution for Oracle databases that is easy to install and maintain. Oracle Data Guard is unique in that it can be configured to provide maximum availability and performance with efficient zero data loss via synchronous linking over long (200+ miles) distances and automatic failover for recovery within seconds. Oracle Data Guard also maximizes the ROI of your standby hardware by letting you make use of that hardware to offload reporting and testing, and use your standby hardware for online upgrades of your Oracle Database software and your underlying hardware and operating system.

New partitioning features in Oracle Database allow partitioning of tables with parent-child relationships, and optimize for multi-tenancy by using tenant id as partitioning key

SaaS providers sometimes need to move customers from one system to another for various reasons such as a customer may decide to move from a shared database to its own database, say for greater performance or security. Traditionally this requires extraction of data from one database and then insertion into the other, a time consuming process. Oracle Database supports transportable tablespaces providing a rapid mechanism for data movement across databases and even across platforms e.g., Windows to Solaris.

Oracle Database allows data to be partitioned for improved performance and manageability of very large datasets. Oracle Database 11g is unique in its ability to partition multiple tables together following the parent / child relationships between tables. This referential partitioning enabled shared everything multi-tenant data architectures where multiple tenants share the same set of tables to use partitioning. Performance is enhanced because the data for each tenant is stored together automatically and transparently in their own data blocks. SQL operations that require scans of data need only scan the data for the particular tenant. Indexes are also more efficient because they are local to each tenant's partitions. Management is also enhanced. For example, a tenant's partition can be moved to a different system using transportable partitions which functions much like transportable tablespaces described above.

It is important to re-iterate that features such as Real Application Clusters (RAC), DataGuard and Automatic Storage Management can be leveraged without altering the application code. This ability to run any application on a grid without requiring any changes to code is a key Oracle differentiator that can be leveraged for building a SaaS environment that is easy to scale and supports high availability.

As an example, one SaaS provider moved to Oracle Database from another windows based relational database and was as a result, able to consolidate and lower the number of servers required to service its customers in a multi-tenant environment using Oracle RAC.

Caching for Scalability

Major e-commerce and software-as-a-service sites on the Web need to find the right information for delivery to users, quickly and reliably. Many of these sites use Oracle Database as the backend repository for their transaction processing systems, including order management and billing. However, they often use Berkeley DB as a high-performance read/write cache to deliver data to Web browsers quickly.

In this case, Berkeley DB acts as a cache for information stored in the Oracle Database database. That cache may be augmented with other data – user preferences and identity, for example. When a user visits the Web site to make a purchase or fetch an article, the Web server generates the page if it does not already exist, and then caches the content in Berkeley DB for fast delivery the next time it is requested.

Shared Services, SOA and Integration

One of the key enablers of Software as a Service is Web Services. Greater re-use and lower cost of operations can be achieved by designing key components of the on-demand application and platform as standards-based web services which can then be accessed over a services bus. In addition to connecting with internal services, the SaaS provider needs to provide mechanisms for interfacing with customer and partner services. Customers are no longer satisfied with first generation silo-ed approach to SaaS but look to integrate them with their existing business processes and IT infrastructure. Similarly, it is important to build an ecosystem around your services and integrate with third-party services to provide a more comprehensive solution to your customers. The use of standards-based web services allows easier integration with customers and partners.

Oracle SOA Suite provides a comprehensive suite of key best of breed SOA technologies that plugs into heterogeneous IT infrastructures and enables enterprises to incrementally adopt SOA. The components of the suite benefit from common capabilities including a single deployment and management model and tooling, end-to-end security and unified metadata management. Oracle SOA Suite is unique in that it provides a set of integrated capabilities – messaging, service discovery, orchestration / BPM, activity monitoring, Web services management and security, business rules, services directory and development tool, service enabled portal, yet, at the same time, provides support for existing middleware technologies – such as third party J2EE application servers, development tools, and message queues and ESBs. Hence, SaaS ISVs can adopt the whole suite, which benefits from an integrated set of capabilities, or, adopt pieces a la carte. The use of common metadata repository allows applications and services to be configured and customized without writing code.

Clustered Caching for Reliability and Availability

The use of SOA introduces the challenge of maintaining conversational state in high performance environments. The conversational state is managed by business services and processes and corresponds to currently executing operations, processes, sessions, and workflows. This contrasts with persistent data that is managed by data services and is usually stored in databases.

Oracle Coherence is a trusted in-memory data management solution for ensuring reliability and high availability for Java-based service hosts, such as Java Platform, Enterprise Edition (Java EE) application servers. It makes sharing and managing data in a cluster as simple as on a single server. It accomplishes this by coordinating updates to the data by using clusterwide concurrency control, replicating and distributing data modifications across the cluster by using the highest-performing clustered protocol available, and delivering notifications of data modifications to any servers that request them.

Oracle Coherence, which provides replicated and distributed (partitioned) data management and caching services on top of a reliable, highly scalable peer-to-peer

BPEL has become an industry-standard for definition of executable business processes. Oracle BPEL PM allows multiple versions of the same business process to run in a single container.

clustering protocol, has no single point of failures. It automatically and transparently fails over and redistributes its clustered data management services when a server becomes inoperative or is disconnected from the network. When a new server is added or when a failed server is restarted, it automatically joins the cluster and Oracle Coherence fails services back to it, transparently redistributing the cluster load. Oracle Coherence includes network-level fault-tolerance features and transparent soft-restart capabilities to enable servers to self-heal.

Customizable Business Processes

SaaS applications can leverage existing shared services located within or outside the firewall of the SaaS provider to provide new functionality using services orchestration. Business Process Execution Language (BPEL) is the industry standard for orchestration of web services. BPEL processes encode stateful, multi-step processes in a standards based XML language. These processes can call and interact with web services and traditional interfaces including databases, applications and messaging systems.

Business processes can be integrated into applications—this is often a key feature enabling application flexibility, because significant parts of the application behavior can be moved out of hard-coded logic and into an XML based process definition that is easy to view, monitor, and change. The ability to customize business processes is one of the key requirements for next generation SaaS applications as they are move beyond simple use cases.

By way of example, a SaaS provider that caters to the B2B market was writing code to support any customization requests by the trading partners such as a change of one of the web services invoked as part of the process. This was a cumbersome task requiring the systems to be re-started to apply the new patches. The SaaS provider now uses Oracle BPEL Process Manger's ability to dynamically alter connections at runtime without any downtime. This means that the SaaS provider can be responsive to the changing needs of their customers at a lower cost and implement the changes over much shorter periods of time- hours as opposed to days previously.

Metadata-driven User-Interface

Customers are increasingly used to rich customizable user interfaces and collaboration features when interacting with web-based applications in the consumer world. SaaS providers can provide these features to their customers using Oracle technology and rich application frameworks, without having to build from scratch while adhering to industry standards.

Oracle Fusion Middleware combines the standards-based, declarative development of JavaServer Faces, the flexibility and power of Portlets and runtime customization, and a set of integrated Web 2.0 services into a new generation of context-centric, composite applications. Oracle ADF is an SOA framework that automates and manages business and data services and provides a standard data and service binding layer based on industry standards (JSR 227) that can be used

with process flows, page flows and service invocations. Using Oracle WebCenter Suite capabilities, you can build applications that allow users to communicate effectively through multiple channels: web, wireless, instant messaging, and even voice over IP. Built-in services also include document sharing and search, discussion forums, and Wikis. The user doesn't merely pass through the application to get to these services; the services are integrated into the very fabric of the application. The ability to support communities and multi-channel interaction will become increasingly important with the next generation of SaaS as users demand features that they find useful in consumer oriented web offerings.

Comprehensive Development Platform

Since Oracle database and middleware are standards-based, the developers have a choice of development tools to work with from third-party vendors, open source and Oracle. Oracle tools for building applications include Oracle JDeveloper for Java and SOA applications and Oracle Application Express for rapid web-based database applications.

Oracle JDeveloper is a comprehensive and integrated environment for creating, composing, and maintaining applications. Oracle JDeveloper is not only a Java development environment, but also encompasses tools for database, XML schema, and object design; layout of pages and user interface; and design of BPEL Process Manager process definitions, Oracle Enterprise Service Bus service routing rules and transformations, and Oracle Business Rules rule definitions. Visual editors in many cases reduce or eliminate the need to write custom code.

Oracle JDeveloper also supports remote debugging and integrates with major source code control systems. It also includes Oracle SQL Developer – a graphical tool for database development. With SQL Developer, you can browse database objects, run SQL statements and SQL scripts, and edit and debug PL/SQL statements.

Oracle JDeveloper is packaged with Oracle Application Development Framework (ADF), which provides a rich set of templates for developing business applications. ADF is standards-based and can work with a wide choice of frameworks and technologies, including Struts, JSP, JSF, and Spring.

Oracle ADF Faces is a set of pre-built user interface components that include tables, selectors, buttons, and charts. These significantly cut application development time by allowing UIs to be built out of highly functional components. ADF Faces components also provide a consistent look and feel for your application. For end users, they allow much of the rich experience and interactivity of a dedicated desktop application in a thin-client, zero-download Web application. Oracle JDeveloper strongly supports Service Oriented Architecture. With modeling tools and wizards that greatly facilitate developing business logic and exposing this logic as services. The ability to build, test and debug new applications and services from a single tool can help reduce time to market for new SaaS services.

Oracle Application Express (Oracle APEX), formerly called HTML DB, is a rapid web application development tool for the Oracle database. Using only a web browser and basic knowledge of programming, one can develop and deploy on-demand applications that are both fast and secure. Oracle Application Express combines the qualities of a private database, productivity, ease of use, and flexibility with the qualities of an enterprise database, security, integrity, scalability, availability and built for the web. Application Express is a tool to build web-based applications and the application development environment is also conveniently web-based. Application Express is ideally suited for data-driven web applications targeted at SMBs or departments where complexity of N-tier application architecture may not be suitable. Oracle Application Express is standard feature included with Oracle Database and does not require a separate middle tier.

Customers do not want to log into separate systems to access reports that can help them do their jobs. Oracle BI is used by many ISVs to support in-process real-time analytics.

Real-time In-process Analytics

Business Intelligence has traditionally been an add-on to applications, made available only to a small set of users and not to all application users. Oracle Business Intelligence is designed to allow reports, charts, and analytics to be embedded in applications. By making these capabilities available in the context of the on-demand application, a SaaS provider can allow all users to access actionable information and make better decisions. For example, a call center manager can chart response times and customer satisfaction trends and take corrective action if needed; a manager using a financial application can see reports on budgets and spending; and so on.

The Oracle BI Suite includes ad-hoc query and analysis, OLAP analysis, interactive dashboards, reporting, proactive intelligence and alerts, mobile analytics, and more. It offers a unified view of information; empowers business users and analysts over the web; and combines real time and historical intelligence with alerting to drive actions to improve business performance. Another important requirement is that customers often need to print their reports to specific formats or export data. The Oracle BI Suite includes a reporting tool that allows users to publish data to a variety of formats: PDF, XML, HTML, HTMLCSS, Postscript, PCL, Delimited text, and RTF. The reports and dashboards can be customized based on the role of the user providing them with useful actionable intelligence. By providing real-time, actionable business intelligence through interactive dashboards, custom reports, and historical trending, users are able to uncover new patterns and identify issues before they impact their business.

Data Security and Privacy

One of the key concerns raised by customers of Software as a Service is data security and privacy. An effective security strategy requires a holistic approach based on a framework that extends across applications, middleware, and data stores. The major components of such a framework fall into two primary areas. These are Identity Management and Data Security and Privacy.

Most data breaches involve current or former employees of the organization. Automation of provisioning and de-provisioning tasks can significantly reduce risk while reducing cost for the SaaS ISV by delegating day-to-day tasks to users.

Identity Management

The enforcement of security policies restricting who has access to what, when, and from where constitutes the primary goal of any security architecture. A centralized framework ensures such policies are consistently applied across all applications and systems. Oracle Identity Management provides a complete solution including support for:

- **Authentication & Authorization:** This could take several forms from the simplest username/password combination to stronger authentication schemes such as token cards and bio-metrics. SaaS providers sometimes need to interact with other third-party service providers or with services behind the corporate firewall of the user. This requires cross-domain authentication. Oracle Identity Management includes federation capabilities that enable SaaS providers to securely link accounts and identities across security boundaries without a central user repository or endless synchronizations of data stores. With implementations of standards-based protocols, Oracle Identity Federation provides an interoperable way to provide cross domain single sign-on for customers, SaaS providers and any other 3rd-party without the exorbitant costs of managing, maintaining and administering their identities and credentials.
- **User Provisioning & Identity Administration:** Managing accounts and security policies for thousands of users across hundreds of applications is a daunting task. Manual administration is not just costly, but also presents security challenges as human errors are inevitable. Easing administration through automated user provisioning and de-provisioning, self-service user functions (such as password resets), and delegated administration to the end user or customer's IT or helpdesk dramatically reduces costs and improves security.

Data Security

One of the challenges of Software as a Service is that customers rely on the service provider to store and manage their data. Securing the perimeter is no longer sufficient as data theft often occurs with help from insiders. Database Vault simplifies the protection of existing applications, preventing access to application data by the DBA or other highly privileged users within the database. Oracle Database Vault introduces multi-factor authorization checks and a rules engine to further protect the database and application. Factor checks and rules enforcement are in addition to existing privileges on access to application data. Oracle Database Vault enforces separation of duty, a critical requirement when applications and databases are consolidated by ensuring that the privileged DBAs are limited to managing the database but not allowed to access the application data. Finally, Oracle Database Vault provides auditing and a comprehensive set of out-of-the-box security reports that provide the detection and reporting component necessary for compliance and auditors.

Another key feature for data security and privacy is Virtual Private Database (VPD). The VPD enables data access control by user or by customer with the assurance of physical data separation. For SaaS, the Virtual Private Database can ensure that customers can access only their own data. The SaaS provider can maintain data of multiple customers in the same Oracle database, while permitting each customer to see only its own data.

This is how VPD works – it is enabled by associating one or more security policies with tables or views. Direct or indirect access to a table with an attached security policy causes the database to consult a function that implements the policy. The policy function returns an access condition known as a predicate (a WHERE clause), which the database appends to the user's SQL statement, thus dynamically modifying the user's data access. For SaaS, the WHERE clause would include the tenant id thereby ensuring that it is impossible for another tenant to access a tenant's data even if the SaaS developer made a mistake or the customer made inappropriate customizations.

The Virtual Private Database results in lower costs of ownership in deploying services. Security can be built once, in the data server, rather than in each service that accesses data. Security is stronger, because it is enforced by the database, no matter how a user accesses data. Security is no longer bypassed by a user customization be it via code, accessing an ad hoc query tool or new report writer. Indeed, several SaaS ISVs and many Oracle applications themselves use VPD to enforce data separation for hosting, including Oracle SalesOnline.com and Oracle Portal.

Service-Level Compliance and Configuration Management

As discussed earlier, the shift to SaaS means that ISVs (service providers) are now responsible for day-to-day operations, including monitoring and management of applications, infrastructure software and systems. Historically, the cost of managing large sets of systems has increased linearly – or more – with each new system added to the environment. It's simply not possible to maintain management costs in the grid-based SaaS environment with traditional managements solutions that do not scale. Furthermore, CIOs want visibility into the performance of the SaaS providers against the SLAs. This can be turned into a competitive edge by providing visibility into historical trends to address one of the common customer concerns when it comes to outsourcing a business function.

Oracle Enterprise Manager Grid Control provides several capabilities that provide visibility and are especially key to managing large hosted environments:

- Management of Databases, Application Servers, Web Servers and other components from a single console. Management of Oracle and non-Oracle components, including load balancers, packaged applications, and directories.

Service Level Management and providing visibility into how well a service provider is meeting these goals can help alleviate customer concerns and help in the pre-sales cycle.

- Ability to set and monitor policies for configuration and security settings, compare existing configurations to known good or recommended configurations.
- Enables easy provisioning of software by creation of ‘gold images’ and deployment to multiple servers using cloning, including support for RAC clusters.
- Management of “many as one:” ability to apply changes to groups of systems and applications.
- Configurable reporting on service levels and performance
- Strong diagnostic tools for pinpointing errors and performance issues, including the specific tier, component, application or even database statement that is causing an issue.
- Support for rolling patch and upgrade, to minimize application downtime.
- Ability to automate and schedule administrative tasks.

Oracle Enterprise Manager Grid Control *configuration management* solution is designed to eliminate the costs and errors that plague traditional, manual configuration management solutions and can be a significant impediment to growth for a SaaS provider. Grid Control centrally tracks hardware, software installation – including patch levels – and software configuration data for every service and system that it manages. This detailed information is regularly and automatically collected and updated as changes occur. This information repository is then used to proactively detect configuration policy violations, e.g., for an Oracle database the Grid Control can evaluate the configuration against best practices and issue alerts or apply corrective actions based on rules.

Oracle Enterprise Manager Grid Control also provides extensive *service-level management* capabilities. The SaaS provider can leverage these features to provide visibility into service-level compliance to the customer. Enterprise Manager enables administrators to model real world services by mapping the end user’s view of a service to its underlying IT infrastructure. Enterprise Manager supports services with many different underlying protocols and related ‘service tests’. In addition, the services can also be represented graphically via the topology view. The service topology feature enables visual modeling and viewing of dependencies.

Enterprise Manager allows SaaS providers to define service availability for each service. The availability of a service could be defined in terms of the successful execution of one or more ‘service tests’ representing an end user task. For example, for a CRM application availability is defined as ability to successfully login and access a sales report. Enterprise Manager also allows availability to be defined at the component level such as database, operating system, etc. It can measure response metrics by use of beacons that execute service tests remotely reflecting the real-world usage pattern where the customer and SaaS provider could be in two very different parts of the world.

All this information can be rendered via a Services Dashboard for the IT administrators and management, and select reports can also be provided to end-users to give them visibility into how well the SaaS provider is meeting its service level goals.

CONCLUSION

The move to Software as a Service requires software vendors and hosting providers to deal with several new business and technical challenges. The key business challenges include the ability to deliver services and manage growth while containing costs. The technical challenges for SaaS providers include support for multi-tenancy, security, service-level management, and customizable data, processes and user-interface. A pre-integrated platform that provides the necessary building blocks needed to rapidly build out application services using a service oriented approach can help accelerate time to market and save on cost.

Oracle SaaS platform technologies including the database, middleware and enterprise management are engineered to work together and can run on a grid of low-cost servers. The platform provides end-to-end security for customer data even in the context of multi-tenancy. It also allows you to include real-time in-process analytics and actionable dashboards in your SaaS applications. The rich user-interface components allow for Web2.0 style applications to be built using industry-standard components. The model-driven metadata for data, processes and user interface enables customization through configuration rather than code.

In conclusion, Oracle is a complete, standards-based, pre-integrated platform that runs on a grid and provides the following key capabilities for a SaaS provider:

- Secure Multi-Tenancy
- Metadata-driven Customizable Processes, Data and UI
- Real-time In-Process Analytics
- Rich User Interface
- Integration with On-premise and SaaS applications
- Service Level Management



Oracle SaaS Platform
November 2007 (Version 1.3)
Author: Anshu Sharma
Contributing Authors: Gordon Smith, Jon Dart

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 © 2007, Oracle. 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, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.