

An Oracle White Paper
October 2009

Platform-as-a-Service Private Cloud with Oracle Fusion Middleware

Introduction	1
Overview: Cloud Computing and the Enterprise	2
The Siloed World: Laborious Set-up and Manual Expansion.....	2
Breaking Down the Silos	3
Cloud Computing	3
Types of Cloud Computing.....	4
Public Cloud vs. Private Cloud	5
PaaS Private Cloud – Natural for Enterprises	6
Organizational Support for PaaS Private Cloud.....	6
Requirements and Reference Architecture for PaaS Private Cloud	7
The Basic PaaS Private Cloud Life-Cycle	8
PaaS Private Cloud with Oracle Fusion Middleware.....	9
Application Grid.....	9
SOA and BPM.....	11
User Interaction.....	12
Identity Management.....	13
Systems Management	14
Conclusion	15
Getting Started	15
A Few Examples	16
The Future.....	17

Introduction

Opportunities for improving IT efficiency and performance through centralization of resources have increased dramatically in the last few years with the maturation of technologies such as SOA, virtualization, grid computing, and management automation. A natural outcome of this is what has become increasingly referred to as “cloud computing”, where a consumer of computational capabilities sets up or makes use of computing “in the cloud” (i.e. over a network) in a self-service manner, without direct involvement in how that computing is resourced. Initially referring to services provided by third-parties over the Web, cloud computing is now also evolving in a “private” variant whereby enterprises set up cloud-like, centralized shared infrastructure with automated capacity adjustment that internal departmental “customers” utilize in a self-service manner. Realization of efficiency, performance, and agility benefits is reinforcing this trend.

Oracle Fusion Middleware provides a comprehensive foundation for enterprise private clouds. With mature, reliable, high-performance clustering mechanisms as well as pervasive, unified security and management, Fusion Middleware delivers the most effective dynamic resourcing in the industry. Modularity, sharability, and composability enabled by Fusion Middleware SOA, BPM, and user interaction technologies complement the dynamic resourcing to support a powerful self-service platform of reusable components. These platform components are both easily managed by a central IT function and easily composed into applications by departments within the enterprise.

There remain technologies to develop and practices to refine, but enterprises can make significant progress in achieving the benefits of cloud computing with products available today. Start with Oracle Fusion Middleware.

"The maturation of application infrastructure virtualization technologies that enable dynamic, automated provisioning of enterprise application server-based applications on grid-like clusters of commodity servers will support cloud-like, cost-optimized, elastic scalability, thus enabling on-premises cloud-style architectures."

Gartner: *Key Issues for Platform Middleware, 2009*

Overview: Cloud Computing and the Enterprise

One can think of the history of IT as a pendulum that swings between the extremely centralized and the extremely distributed. Early mainframes were an example of centralized computing. The PCs and workstations of the 70s and 80s were a strong contrast, with computation occurring at the outermost nodes of the architecture. The ever improving connectivity and bandwidth driven by Internet technologies is bringing us back to greater centralization.

Both centralization and distribution have important merits for enterprise IT. Centralization of control enables consistency, economies of scale, and efficient rollout of innovations that are applicable across the enterprise. Distribution of control enables agility for departments, allowing flexibility to respond quickly to needs and imperatives specific to their roles within the organization. Optimizing the balance between centralization and distribution is an ongoing challenge for IT architecture, organization of the people who support that architecture, and organization of the people who use that architecture.

Client-server computing was the first big step toward more centralization from the world of distributed PCs and workstations. As the ubiquity and quality of Internet-based networking made client-server more viable and widespread, much of computing shifted to the server in a more centralized model.

The Siloed World: Laborious Set-up and Manual Expansion

Initially the shift to server-based computing was supported with a highly siloed architectural approach: each application got a dedicated "stack". Early on that stack was typically a single server, but as the stack subdivided into multiple layers and as those layers became standardized, applications became distributed across multiple servers. A single application still typically had exclusive ownership of the servers in these multi-tier architectures, so it was still very much a siloed architecture, often with very low hardware utilization.

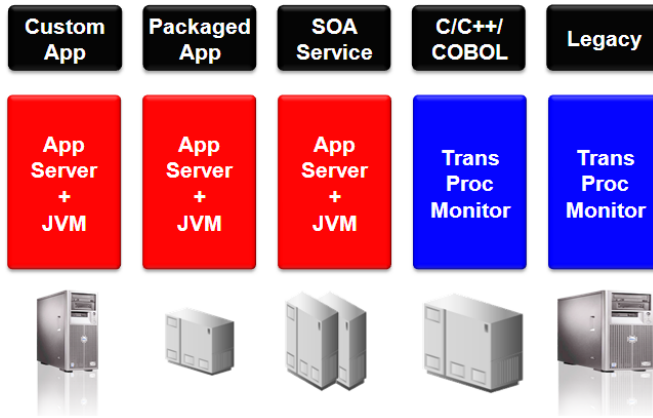


Figure 1. Siloed server-side architecture.

Breaking Down the Silos

Underutilization and the complexity of managing growing sprawl have motivated several trends during the last several years. Decomposition of applications into components (beyond macro-level tiers such as Web/UI tier, application/business logic tier, and database tier) was enabled by service-oriented architecture (SOA). Decoupling of applications from specific, dedicated hardware has been enabled by virtualization and grid techniques. Automation of hardware allocation across applications has been enabled by increasingly sophisticated metrics, management, and control.

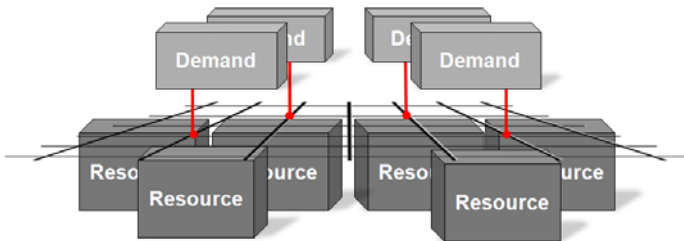


Figure 2. Decoupling of demands from specific resources enabled by grid computing.

Cloud Computing

The decoupling of “server-side” software from hardware—and the flexibility, dynamics, and automation with which that software is run—has led naturally to a model referred to today as “cloud computing”. Most simplistically interpreted, the cloud metaphor is meant to convey that a user (whether a person or another system) of an asset (application, storage, etc.) “in the cloud” has no worry or need to worry where or how that asset is resourced. The cloud obscures the location of the asset; the only thing the user is concerned with is that the expected asset exists and it works. The essential two characteristics of a cloud environment are that

- the asset is set up, deployed, or engaged instantly or near-instantly over a network (usually the Internet/Web) without binding to specific physical compute resources, and
- the capacity of the application (supported number of concurrent users, transactions per unit time, amount of storage, etc.) is adjusted automatically as demand fluctuates so that manual sizing and provisioning are eliminated both up-front and over time

The notion of cloud initially evolved in the context of applications accessed over the Internet and Web. Initially referred to as “hosted applications” provided by “application service providers” (ASPs), this phenomenon later became known as “software as a service” (SaaS).

Types of Cloud Computing

As the cloud metaphor and terminology evolved in the last few years, the notion of SaaS begged the question of what else could be offered “as a service”. A number of different offshoots have appeared, but the ones that have gained the most traction divide the range of cloud types into three categories: the original SaaS (which refers mainly to complete applications), platform as a service, or “PaaS”, and infrastructure as a service, or “IaaS”. IaaS refers to very basic compute capability—machines with operating systems and storage. The most prominent current example of IaaS is Amazon Elastic Compute Cloud (“EC2”). PaaS is between IaaS and SaaS and refers to an environment where one builds and runs an application platform in the cloud using whatever pre-built components and interfaces are provided by that particular PaaS platform.

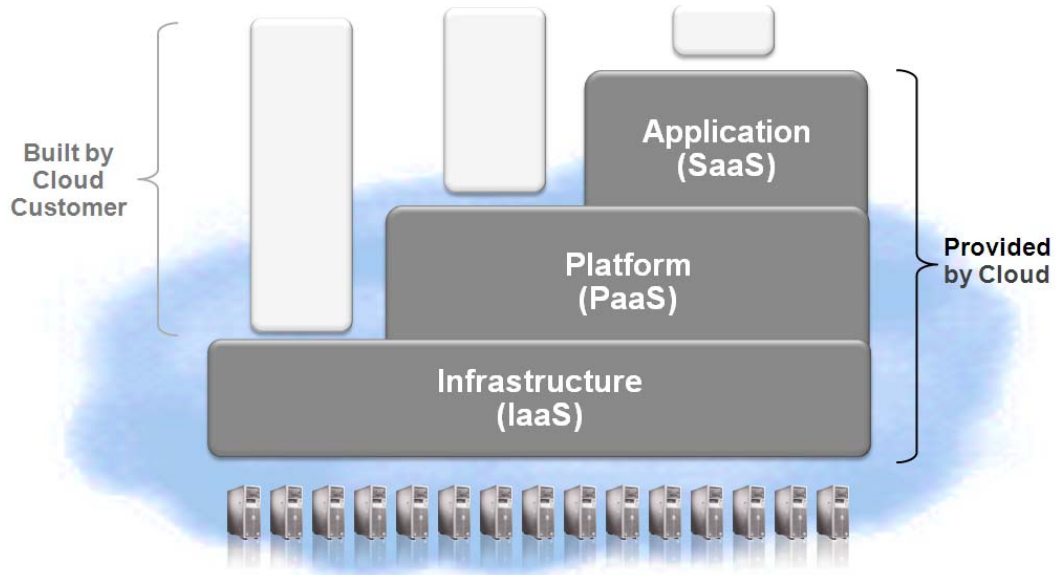


Figure 3. Different types of cloud computing .

Whether an enterprise chooses SaaS, PaaS, or IaaS for a particular solution depends on a number of factors, including the availability of a SaaS application to suit the needs at hand, the

enterprise’s level of expertise developing custom applications, and the amount of flexibility required. Essentially the spectrum between IaaS and SaaS represents a trade-off between effort and flexibility.

Public Cloud vs. Private Cloud

Besides the split into SaaS, PaaS, and IaaS, cloud computing has divided along another dimension. Though initially cloud referred to software accessed over the public Internet/Web, enterprises quickly realized that environments could be set up internally that have the essential cloud characteristics enumerated above: network-based self-service deployment and elastic capacity. Such “on-premise” or “internal” clouds have come to be referred to as “private cloud”.

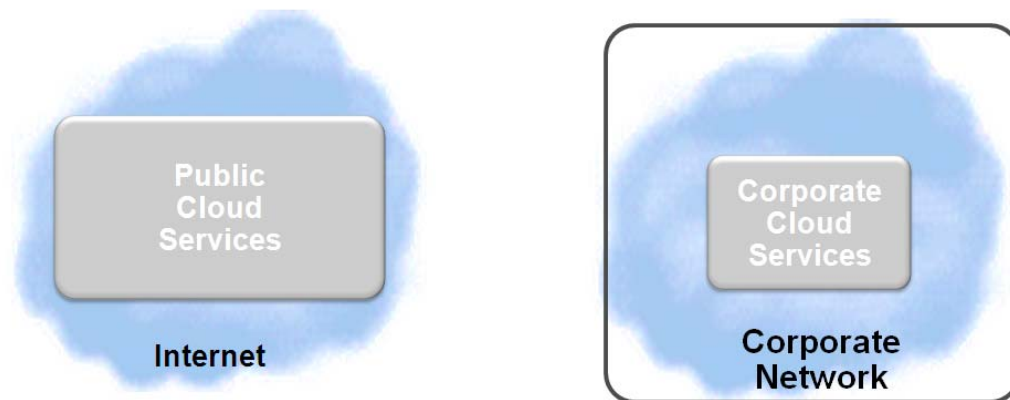


Figure 4. A public cloud offering is provided by a third party over the Internet; a private cloud is set up and managed by an enterprise’s own IT department and runs inside the corporate firewall.

Both public and private clouds share the essential benefits of

- Efficiency
- High availability
- Elastic capacity

These benefits derive from the virtualization and/or grid architecture used as the foundation for clouds—one gets efficiency from the high utilization of sharing physical servers, availability from clustering across multiple physical servers, and elastic capacity from the ability to dynamically resize clusters and/or migrate live cluster nodes to different physical servers. In addition to these benefits, public clouds offer

- Low up-front costs
- No hardware set up by the enterprise
- Minimal or no systems management by the enterprise

Potential downsides of using a public cloud offering include

- Greater total costs over time
- Difficult integration constrained by limitations of interfaces the offering provides
- No direct control over quality of service or security

Private cloud, on the other hand, does entail some up-front investment in hardware and set-up as well as ongoing administration on the part of the enterprise. However, the downsides of public cloud are then mitigated since the private cloud offers:

- Lower total cost over time
- Full access and flexibility to integrate
- Direct control over quality of service and security

PaaS Private Cloud – Natural for Enterprises

Because integration flexibility and control over quality of service and security are high priority for larger enterprises, and because such enterprises likely have the financial resources to optimize for costs over time rather than up-front costs, many enterprises will naturally gravitate towards the private variant in their adoption of cloud computing. An additional aspect strengthens this bias: adopting private cloud practices will be a small change for many enterprises. IT departments have in many cases already gone significantly down the path of consolidating infrastructure and setting up shared services, and enabling cloud's self service and automated dynamic capacity will often be a relatively small incremental step. This is in contrast to adoption of a public cloud offering, which will dramatically change how departmental users obtain application support.

Organizational Support for PaaS Private Cloud

In setting up a private cloud, the natural organizational structure comprises a central IT function that sets up and manages the cloud itself and various functional or product departments across the enterprise that are “customers” of the cloud.

For an private cloud, the most appropriate type is PaaS. If the central IT function were to set up an offering at the IaaS level, the departmental users would need too much IT expertise themselves in order to make use of the cloud, thus defeating the economies of centralizing the IT function in the first place. At the other end of the spectrum, an internal SaaS offering would not likely make sense in very many cases because departments wouldn't have the flexibility to create the specific functionality they require—there are very few applications that would fulfill a majority of functional needs across multiple departments. The platform level of PaaS is the right balance between flexibility and ease of use for the departmental cloud customers.

Depending on industry or domain characteristics as well as company-specific business strategies, different enterprises will have different balances between what is in a shared cloud platform and what individual departments will need to create. For example, in a consumer products company organized such that different departments represent different products, the cloud-based platform might have functionality such as a consumer-facing portal, a catalog, order processing, and customer service, and each product department would have only minimal customization beyond the basics provided by the platform. In another example, a telecommunications company’s private cloud platform might provide basic customer record functionality, with each department creating deeply specialized applications. In the former example, most of the IT expertise would reside in the central IT function with very little expertise required of the departments; in the latter example, each department likely has Java programmers. Each enterprise will have a unique set of capabilities provided in the central cloud platform; the goal will be to centralize as much as possible while allowing the departments the flexibility they need for their roles in making the overall business competitive.

Requirements and Reference Architecture for PaaS Private Cloud

The figure below shows the basic architecture of a platform-as-a-service cloud offering that a central IT function would set up within an enterprise.

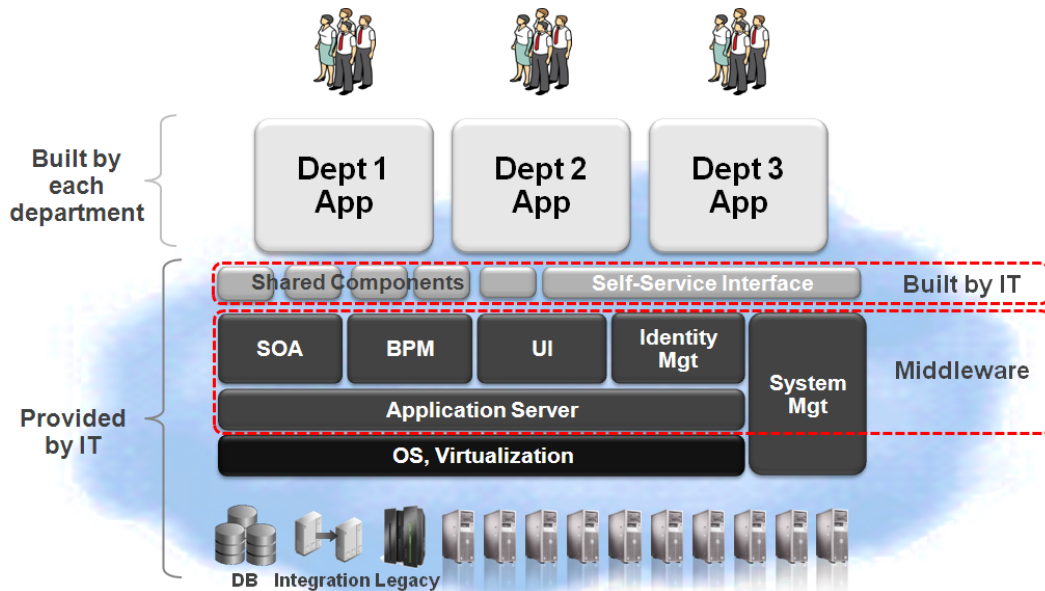


Figure 5. Basic platform-as-a-service cloud architecture.

The physical infrastructure includes servers, legacy systems such as mainframes, integrations, and database resources. The lowest layer of software above this is at the operating system level and may or may not include virtualization technologies such as the Oracle Virtual Machine hypervisor. Above this resides middleware such as Oracle Fusion Middleware, including

application servers and technologies such as service-oriented architecture (SOA), business process management (BPM), user interface (UI) technologies, and identity management. Systems management such as Oracle Enterprise Manager spans the entire stack. Upon this foundation are the custom elements built by the central IT function, including shared components such as SOA services and BPM processes as well as the self-service interface that the enterprise's internal cloud customers interact with.

The Basic PaaS Private Cloud Life-Cycle

Getting underway with a PaaS private cloud involves four macro-level steps. First, the central IT function builds the platform, starting with out-of-the-box middleware and creating the enterprise-specific shared components and self-service interface.

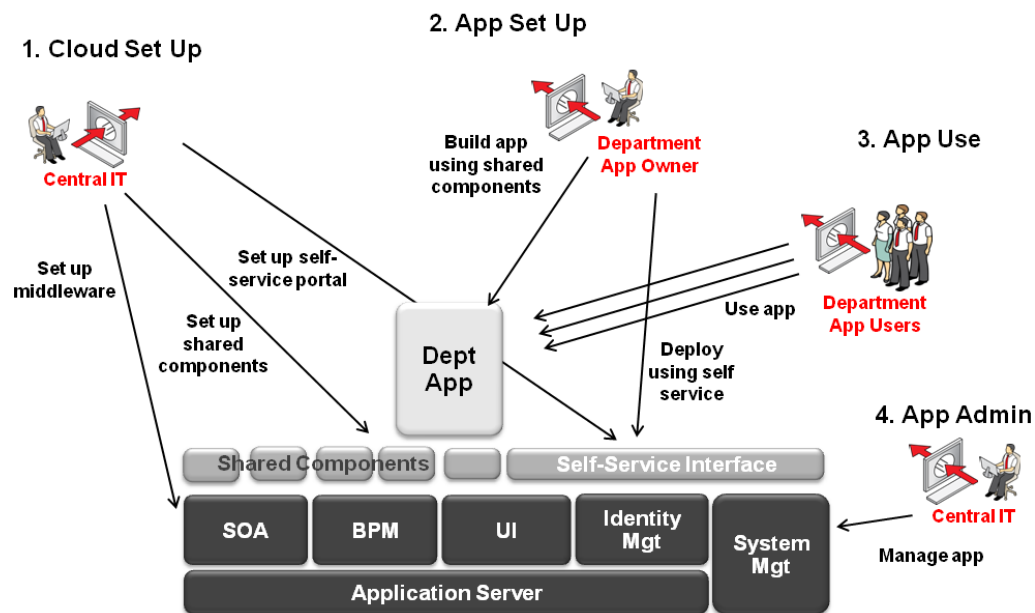


Figure 6. Basic PaaS private cloud life-cycle.

Once the basic platform is up and running, the application owners within the enterprise's departments can set up their respective applications. Again, depending on the nature of the domain and enterprise, this may involve fairly simple application composition using platform components, or it may involve a substantial amount of custom application development.

Once an application has been deployed on the platform, the third step is simply the use of the application. From the users' perspective, the application is no different from any other network/Web-based application they would use within the enterprise—there is nothing special about the fact that it's running on a cloud platform as far as they are concerned.

Finally, there is of course ongoing administration of the platform as well as the applications, which is carried out by the central IT function. Depending on the nature of the applications, the application owners may also carry out some amount of administration, such as adding and removing users or other high-level functions specific to the application.

Central IT is concerned with lower-level issues such as whether the application is resourced appropriately, is meeting its service-level agreements (SLAs), etc. One of the goals in setting up shared infrastructure in general and private cloud in particular is to exploit as many economies of scale and opportunities for efficiency as possible. Among these is the opportunity to automate dynamic resource allocation and optimization, enabling the elastic capacity that characterizes cloud. This also enables the continuously high responsiveness demanded by users irrespective of load and minimizes manual intervention.

PaaS Private Cloud with Oracle Fusion Middleware

In planning a platform-level private cloud, there are several important considerations to keep in mind. First is that much of the needed technology already exists. Oracle Fusion Middleware today provides an excellent foundation for private cloud with critical capabilities such as a rich portal technology for a self-service interface and support for elastic capacity. The second consideration is leveraging existing investments: you can get to cloud's architecture and benefits with minimal incremental effort and restructuring. Finally, cloud is not an all-or-none proposition. You can evolve from your current architecture to one that is more cloud-like with a phased approach, prioritizing and sequencing incremental changes in a way that reflects priorities specific to your business and in a way that minimizes disruption.

In the following sections we highlight how each of several key Oracle Fusion Middleware technologies specifically helps enable private cloud, using the reference architecture we described above as a framework to structure the discussion.

Application Grid

Of the two characteristics of cloud computing identified earlier, elastic capacity and self-service deployment, elastic capacity is perhaps of more immediate priority for many enterprise IT managers. Having an infrastructure automatically adapt to the evolving needs of various departments and their applications means the IT shop can do more with less: it can provide better performance against application SLAs across the enterprise without having each application individually provisioned for its own worst-case load.

This is the fundamental goal of the application grid architecture. An application grid is a dynamically adjustable grid of “resources”—resources as seen and understood by applications. For enterprise Java applications, the entity that embodies these resources is the application

server. Thus it is at the application server level of the stack that application grid manifests abstractions of resources and carries out the dynamic allocation of those resources

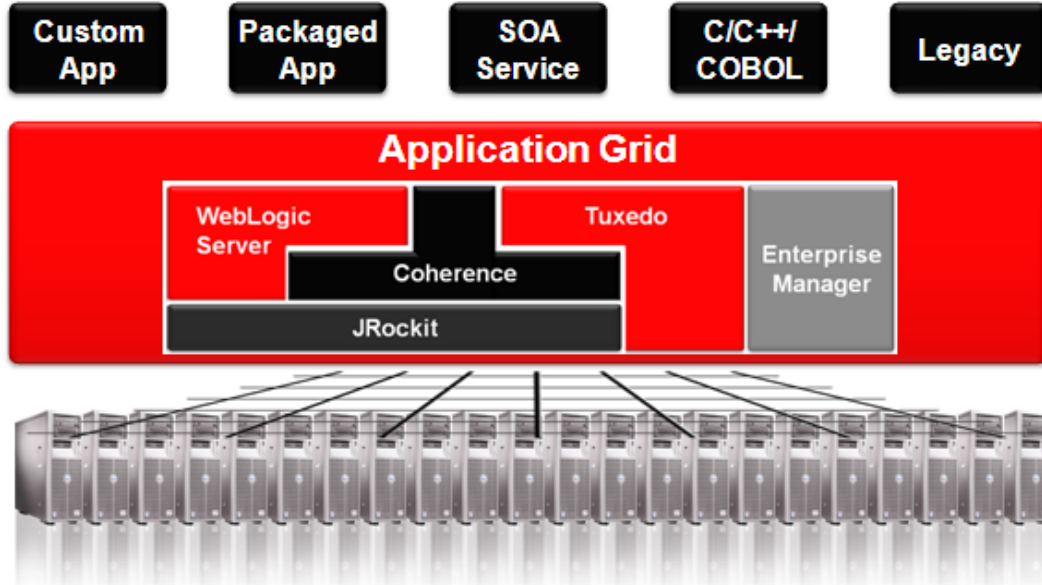


Figure 7. The application grid architecture decouples resources at the application server level, enabling sharing and dynamic adjustment.

Oracle WebLogic Server is the core application grid technology within Oracle Fusion Middleware. In addition to its general market leadership among Java EE application servers, WebLogic Server is particularly well-suited to application grid. Its clustering capabilities, with support for automated load balancing and failover as well as dynamic addition and removal of nodes, serves as the fundamental mechanism for adjusting capacity. WebLogic Server clustering’s support for both console-based and script-based automation of capacity adjustment gives flexibility for a wide range of needs while maximizing ease of use. WebLogic Server clustering management easily plugs into the Oracle Enterprise Manager framework, enabling unified management of the entire private cloud infrastructure from a single console.

WebLogic Server is complemented and enhanced by Oracle Coherence as a basis for application grid. In many environments the resource that requires the most frequent and fine-grained adjustment for optimum performance and efficiency is memory. Coherence enables an application running on one or several machines to use the memory of many machines as if it were “local”. An “in-memory data grid” based on Coherence allows an application to cache enormous amounts of data—potentially entire databases—in memory, accessing that data at memory speeds rather than disk access speeds. This achieves the obvious benefit of performance improvement due to the memory access speed, but has additional performance benefits based on the ability to parallelize computation across the data grid as well reliability benefits deriving from the way objects can be replicated in the grid, eliminating single points of failure. Coherence automatically “repartitions” or redistributes data objects optimally across the data grid as nodes

are added to or removed from the grid, again supporting the elastic capacity requirement of application grid and cloud. Management of this mechanism also plugs cleanly into Oracle Enterprise Manager.

WebLogic Server, Coherence, and Enterprise Manager are packaged with additional application grid elements such as the JRockit Java virtual machine (JVM) in an offering called WebLogic Suite. With the automatable, dynamic clustering mechanisms of WebLogic Server and Coherence managed by Enterprise Manager, consolidation of applications onto a centralized, shared application grid based on WebLogic Suite is often an IT shop's first step down the path to private cloud.

SOA and BPM

With a basic mechanism for elastic capacity in place such as application grid, the next consideration in setting up a private cloud that is not just infrastructure but a true *platform* will likely be the creation of shared components. Service-oriented architecture (SOA), with its approach of modularizing applications into reusable components accessible through standardized interfaces using XML, SOAP, and the various WS-* specifications, is the obvious starting point.

Oracle SOA Suite provides a comprehensive yet easy-to-use basis for creating the reusable components at the heart of your PaaS private cloud. Rich drag-and-drop SOA component features in JDeveloper and the SCA designer enable rapid creation of components and subsequent composition of those components into applications. Oracle Service Bus provides a simple way to make components available to department application creators using your PaaS cloud. End-to-end instance tracking and Oracle Business Activity Monitoring provide a range of metrics visualizations supporting both the central IT function charged with keeping the PaaS up and running and the departmental application owners concerned with business-level performance indicators.

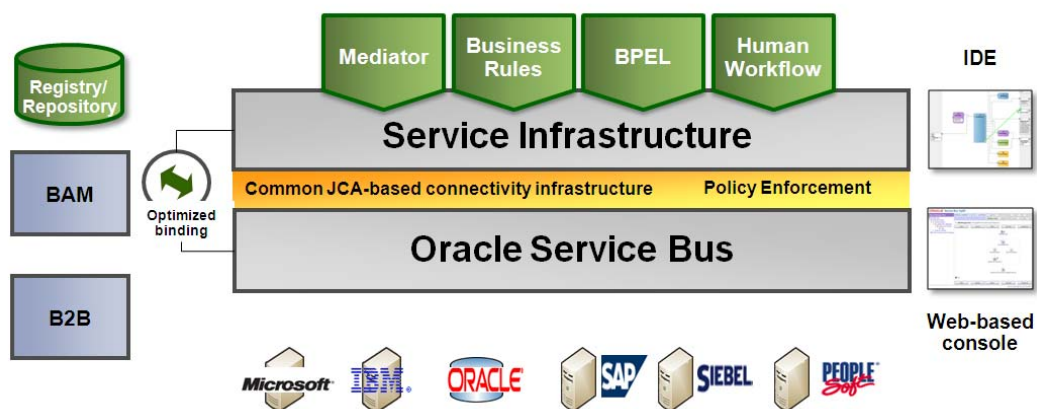


Figure 8. Oracle Service Bus and Registry/Repository allows PaaS platform components to be easily discovered and composed into departmental applications.

In addition to SOA components, many enterprises will want to include business process components managed within a unified BPM framework as part of their PaaS. Oracle BPEL Process Manager provides the federation capability to create BPEL process components out of new as well as legacy assets while also supporting the flexibility to enable multiple departments to incorporate PaaS-based BPEL components into their respective workflows.

User Interaction

Like SOA and BPM components, user interface (UI) components are great candidates for inclusion in an enterprise's PaaS. A centrally-managed library of UI components can give department application owners a great head start in composing their solutions and also gives the central IT function a desirable level of control over consistency across the enterprise's UIs. At the same time, a robust, standards-based UI framework can give the departments the flexibility they need to accommodate their specific functionality, customization, and personalization needs for applications and portal solutions.

UI technologies play an additional role in a PaaS environment as the basis of the self-service interface for the cloud. In many cases this will be a fairly extensive portal that must work closely with an identity management system to authenticate users, filter their access based on roles, and present the platform's shared components for application development and composition.

Oracle Web Center Suite provides a number of portal and user interaction capabilities that are ideal for creating reusable UI components as part of a PaaS. Themes and skins provide powerful facilities for tailoring the look and feel of applications in a tiered way—for an entire Web interface or for portions of a Web interface associated with a department. This enables consistency in look and feel while consolidating deployment. The Advanced Personalization Framework provides the ability to further tailor the usage of the UI and the information delivered to the UI based on users' activities. Powerful mashup integration capabilities enable business users to further personalize the information they want to see while maintaining enterprise information security. Common enterprise metadata services provide a revolutionary way to store and manage all look-and-feel changes, personalizations, and mashups via uniform metadata that enables in-place customizations at runtime and insulates the UI from changes to the base application.

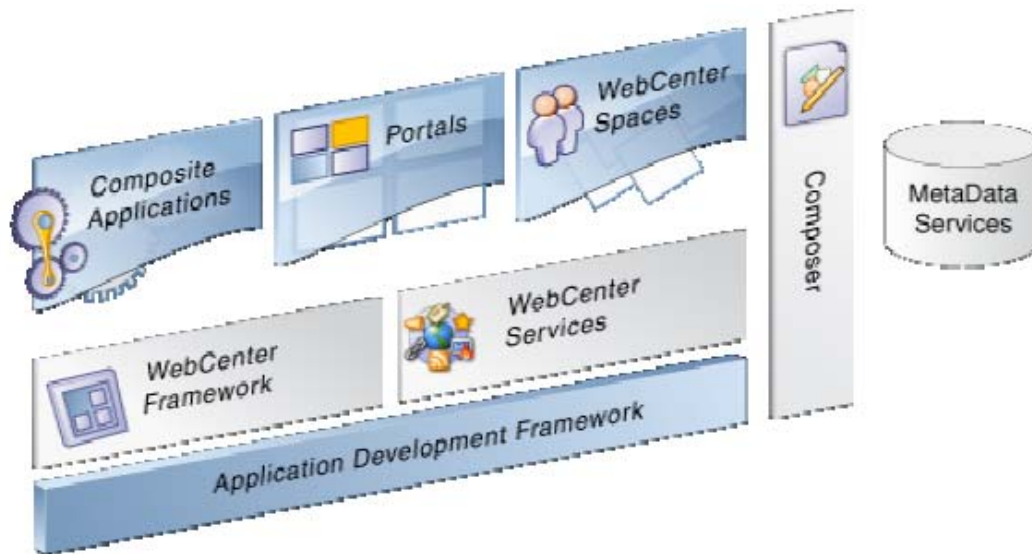


Figure 9. Oracle Web Center Suite supports a PaaS private cloud environment with a framework and services for composite applications and portals as well as a composer that extends Oracle JDeveloper. It also shares metadata services with the rest of Oracle Fusion Middleware.

Identity Management

A high-priority concern for many enterprises in creating a private cloud is security. Particularly for firms in domains with a high level of regulation and/or sensitive customer data, cloud's self service can be a significant challenge. Balancing rich mechanisms for identity and access management with convenience features such as single sign-on is a must for cloud environments.

Implementing PaaS with a high degree of self service in a security-critical environment requires an approach where security pervades the entire architecture rather than being bolted on as an afterthought. An important strength of Oracle Fusion Middleware is that, in addition to each of the products having best-of-breed security in their respective categories, their security mechanisms are well integrated, enabling ease of deployment, ease of change, and high reliability.

Oracle Identity and Access Management Suite provides an ideal facility for managing access and security in a PaaS environment. Within the suite, Oracle Access Manager supports corporate directories and single sign-on. Oracle Entitlements Server provides centralized access control policies for a highly decentralized PaaS environment. Oracle Identity Manager is a best-in-class user provisioning and administration solution that automates the process of adding, updating, and deleting user accounts from applications and directories. It improves regulatory compliance by providing granular reports that attest to who has access to what. Oracle Identity Federation provides a self-contained and flexible multi-protocol federation server that can be rapidly deployed with your existing identity and access management systems. With its support for leading standards-based protocols, it ensures the interoperability to securely share identities across

vendors, customers, and business partners without the increased costs of managing, maintaining, and administering additional identities and credentials.

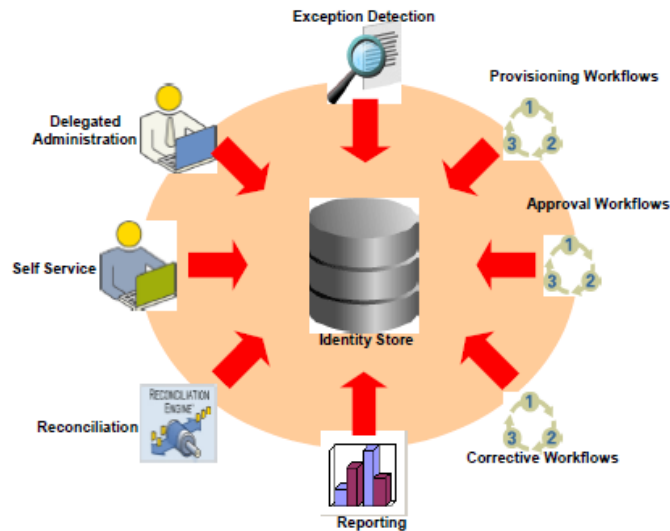


Figure 10. Comprehensive and well-integrated security and identity management with Oracle Identity and Access Management Suite.

Systems Management

The final area we highlight here is systems management. Like security, systems management is a characteristic that depends partly on functionality manifested in a particular software utility—in this case Oracle Enterprise Manager—and partly on capabilities infused throughout the other technologies in the platform.

Systems management is another strength of Oracle Fusion Middleware. Because of the consistent way in which WebLogic Suite, SOA Suite, WebCenter Suite, and Identity and Access Management Suite use metadata and plug into Enterprise Manager, a PaaS based on these technologies can be cleanly and effectively managed from Enterprise Manager’s Grid Control console.

Enterprise Manager provides a highly insightful set of visualizations that enable system administrators to monitor performance, diagnose problems, and make adjustments. In addition, Enterprise Manager operates as a powerful automation control mechanism, sensing when inputs cross certain user-specified thresholds and automatically taking appropriate actions, such as adding capacity to applications that see responsiveness compromised by load spikes.

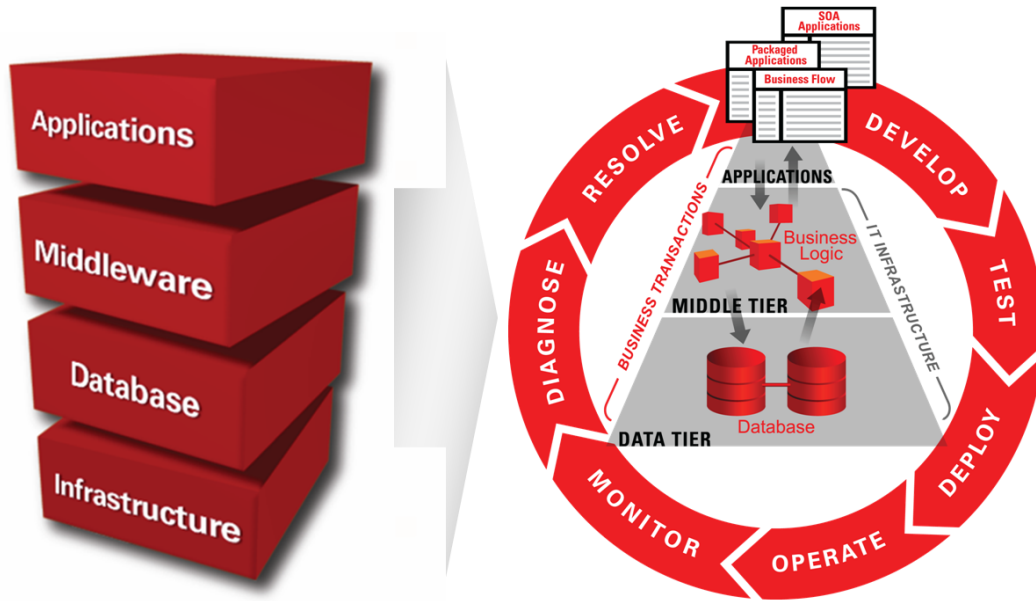


Figure 11. Oracle Enterprise Manager provides comprehensive management across the entire stack of technologies, including middleware.

Such automation is critical both for the elastic capacity and self-service provisioning aspects of private cloud. Another direction many enterprises will want to take with private cloud is departmental “chargeback”—an economic regime where departments are charged by the central IT function based on their usage. Enterprise Manager collects and logs the kinds of information—items such as times and numbers of users logged in to particular applications, amounts of data transferred, etc.—that an IT department would use as the basis for chargeback. With a fairly straightforward processing of the log files and generation of notifications, an effective internal billing system can be easily created.

Conclusion

Getting Started

Platform-as-a-service private cloud is not an all-or-none, black-and-white, rip-and-replace phenomenon. There are various levels or degrees of “cloudness”, such as self-service enablement, elastic capacity, etc., that you will pass through over time, and how far you go will depend on what is appropriate for your unique enterprise and domain.

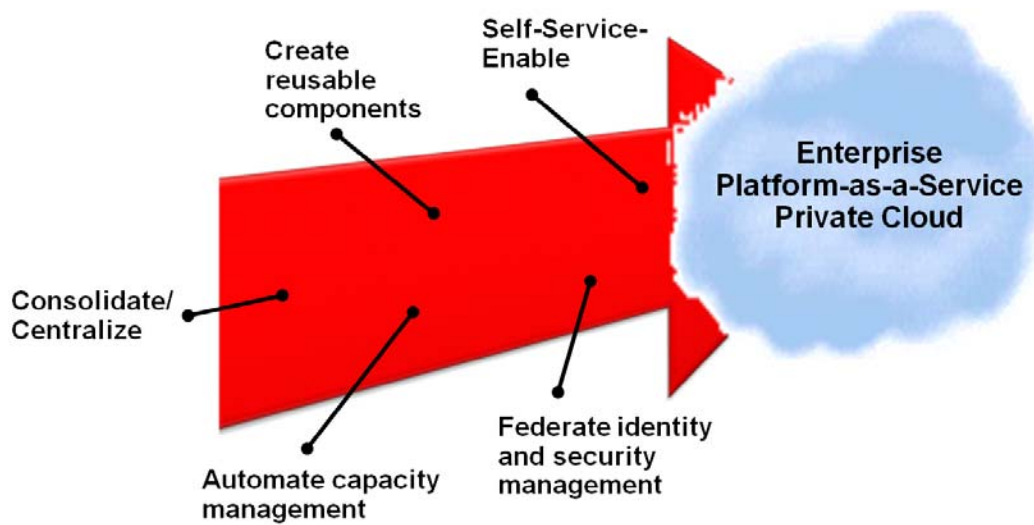


Figure 12. There are multiple entry points for starting on a path to PaaS private cloud.

There are also a number of different ways to get started. Probably the most common starting point is to achieve some level of centralization, consolidation, and standardization of at least some of your infrastructure. This has both technical and organizational implications as discussed earlier.

Once you have done that, or if you have already reached some level of centralization, the other entry points are not necessarily interdependent, and where you start will depend on your priorities. You may want to start on the elastic capacity side and increase automation there, such as by plugging in Enterprise Manager into WebLogic Server's or Coherence's cluster scaling mechanisms. You may want to start by creating reusable components such as SOA services or portlets. You may be in a domain whose highest priority is security, or you may have enough components and automation that your next step really is to put some self-service in front of the platform so as to reduce the burden on central IT, gaining agility for the departments and efficiency for the enterprise overall.

A Few Examples

Many Oracle Fusion Middleware customers have already gone far in setting up private cloud environments. A large hardware manufacturer has consolidated application server infrastructure for several hundred applications onto 2,000 instances of WebLogic Server, resulting in a reduction of administrative staff from 50 to 5 and a 4x reduction in deployment costs.

A top European telecommunications firm has similarly consolidated infrastructure at the level of application servers, explicitly invoking “middleware as a service” as their mantra. They are creating sharable components using Oracle Service Bus and are automating the provisioning of a standard build environment to enable a greater degree of self service for departmental customers.

A top global investment bank, with a high priority on security, has created a centralized Java platform based on WebLogic Server and other Oracle Fusion Middleware, with a focus on federated security. They have achieved a 7:1 consolidation ratio, reduced operational costs by 10-20%, and cut the average turnaround time for changes from months to days.

The Future

As evidenced by the practices and examples outlined in this paper, there is much that can be done today to realize the cloud promises of elastic capacity and self service. Much work remains to be done, both by vendors and enterprises. Oracle continues to invest heavily in capabilities throughout Fusion Middleware—in areas ranging from further automation of capacity adjustment to more out-of-the-box self service enablement to easier application composition—that will accelerate and simplify adoption of cloud computing. We look forward to working with you as the exciting phenomenon of cloud computing evolves.



Oracle Fusion Middleware and Enterprise
Private Cloud
October 2009
Author: Mike Piech

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



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

Copyright © 2009, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

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