Oracle® Reference Architecture

Engineered Systems

Release 3.0

E24268-01

November 2011

ORA Engineered Systems, Release 3.0

E24268-01

Primary Author: Ellsworth Wente, Bob Hensle

Contributing Author: Jim Baty

Contents

List of Figures

Send Us Your Comments

ORA Engineered Systems, Release 3.0

E24268-01

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this document?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us at its_feedback_ww@oracle.com.

Preface

A new architectural component, the engineered system, has recently begun to infiltrate commercial data centers. An engineered system is a complete set of integrated hardware and software designed to reach a predetermined level of capability, capacity, and scale. This relatively new concept of an integrated hardware and software solution dedicated to providing a specific service, changes the way one must think about architecture in IT environments. The intent of this document is to delve into the reasons the industry is moving towards engineered systems, how they impact a customer's architecture, and how they impact both enterprise and cloud architectures.

Audience

This document is intended for Enterprise Architects and Solution Architects who are interested in engineered systems and how engineered systems can be incorporated into a enterprise architecture. Some level of understanding of computer systems design and concepts is expected since this document is not intended to provide a primer for computer systems.

How to Use This Document

This document is intended to be read from start to finish. However, each section is relatively self contained and could be read independently from the other sections. This document can be used to understand what engineered systems are, the benefits they provide, and their architectural impacts.

Document Structure

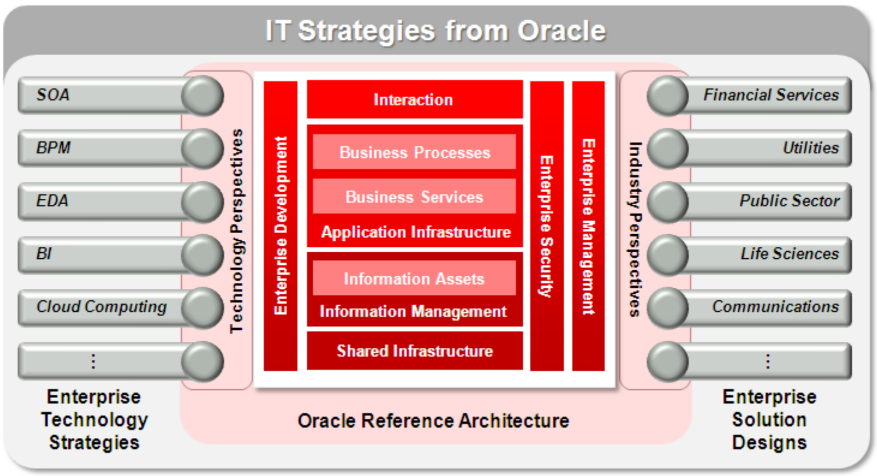
This document is organized into chapters that build upon each other to define engineered systems and discuss the architectural impacts and benefits incurred from adopting engineered systems. The chapters are organized as follows:

- Chapter 1 introduces the concept of engineered systems and puts them into historical and enterprise architecture context.
- Chapter 2 describes the architectural impacts and benefits of engineered systems.
- Chapter 3 describes the relationship between engineered systems and cloud computing.
- Chapter 4 illustrates some example deployment architectures for engineered systems.
- Chapter 5 is a brief summary of the document contents.

- Appendix A lists various documents that may be of interest to readers of this document.

Related Documents

IT Strategies from Oracle (ITSO) is a series of documentation and supporting material designed to enable organizations to develop an architecture-centric approach to enterprise-class IT initiatives. ITSO presents successful technology strategies and solution designs by defining universally adopted architecture concepts, principles, guidelines, standards, and patterns.



ITSO is made up of three primary elements:

- **Oracle Reference Architecture (ORA)** defines a detailed and consistent architecture for developing and integrating solutions based on Oracle technologies. The reference architecture offers architecture principles and guidance based on recommendations from technical experts across Oracle. It covers a broad spectrum of concerns pertaining to technology architecture, including middleware, database, hardware, processes, and services.
- **Enterprise Technology Strategies (ETS)** offer valuable guidance on the adoption of horizontal technologies for the enterprise. They explain how to successfully execute on a strategy by addressing concerns pertaining to architecture, technology, engineering, strategy, and governance. An organization can use this material to measure their maturity, develop their strategy, and achieve greater levels of adoption and success. In addition, each ETS extends the Oracle Reference Architecture by adding the unique capabilities and components provided by that particular technology. It offers a horizontal technology-based perspective of ORA.
- **Enterprise Solution Designs (ESD)** are industry specific solution perspectives based on ORA. They define the high level business processes and functions, and the software capabilities in an underlying technology infrastructure that are required to build enterprise-wide industry solutions. ESDs also map the relevant application and technology products against solutions to illustrate how capabilities in Oracle's complete integrated stack can best meet the business, technical, and quality of service requirements within a particular industry.

This document is one of the series of documents that comprise Oracle Reference Architecture. ***ORA Engineered Systems*** describes the benefits and architectural impacts of engineered systems.

Please consult the ITSO web site for a complete listing of ORA documents as well as other materials in the ITSO series.

Conventions

The following typeface conventions are used in this document:

Convention	Meaning
boldface text	Boldface type in text indicates a term defined in the glossary for this document, the ***ORA Master Glossary***, or in both locations.
italic text	Italics type in text indicates the name of a document or external reference.
underline text	Underline text indicates a hypertext link.

1 Introduction

IT departments are under constant pressure to reduce costs without sacrificing quality or functionality. One common approach to reducing costs is to standardize on a particular hardware and software stack. This reduces the myriad configurations possible for hardware, operating system, software version, patches, etc.; thus substantially reducing the complexity of support and maintenance for the IT staff. This standardization usually results in improved support and substantial cost reductions.

Historically, IT departments that wanted to standardize their hardware and software stack would spend time and effort defining, constructing, configuring, and testing a combination of hardware and software which, once proven successful, would then become the standard stack. Engineered systems remove this burden from the IT department.

Engineered systems are a hardware and software stack designed and built as a single product. This allows both the hardware and software to be configured and even modified to provide the very best compatibility; thereby providing improved reliability, availability, security, and performance. Since the engineered system is a product, individual IT departments are no longer building one-off standardized stacks. By simply adopting the appropriate engineered system, the IT department gets a standardized stack that is the same standardized stack as other IT departments that adopt the engineered system.

1.1 Definition of Engineered Systems

Many companies today have recognized the need to reduce the load on their in-house systems administration and development staff by moving towards proven vendor provided solutions. Engineered systems are on the end of a continuum of vendor provided solutions that an IT department can choose from when constructing a computing environment. The choices available to IT departments are:

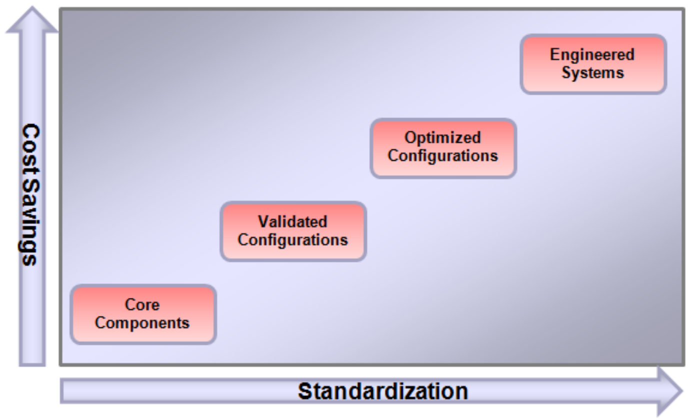
- **Core Components:** Servers, storage, networking, operating systems and virtualization, databases and middleware, and enterprise management tools. Core components are best for customers that have a unique IT requirement, need to integrate those core components with other technologies, and the customer has a high level of expertise in system design
- **Validated Configurations:** Self-service configurations with basic functionality testing and deployment guidance. Validated configurations are useful for expert IT departments and system integrators that need to do customization of systems but are looking for guidance that will shorten development, testing, and deployment.
- **Optimized Configurations:** Pre-defined configurations including hardware and software optimized and tested to work together with documented deployment best practices. Optimized configurations are for companies that want to

implement a system with greater flexibility of configuration and software/hardware combinations to adjust the system to fit within their existing IT environment.

- **Engineered Systems:** Purpose-built hardware and software sold and supported as a single system (pre-defined, pre-engineered, pre-assembled, focused configurations). Engineered systems include a full range of specialized services to simplify deployment, maintenance, and support. Engineered systems are best for companies that want the highest system performance, rapid deployment, reliability, and security, with the lowest total cost of ownership.

The options available to IT departments can be visualized as a trade-off between standardization and cost as shown in Figure 1–1.

Figure 1–1 Trade-off between cost and standardization



Obviously, constructing a custom solution from core components provides the greatest flexibility to a company. However, this flexibility comes at substantial cost. While this cost may be justified for specialized needs, in general a standardized approach is superior from a cost/benefits perspective.

An important side effect of introducing engineered systems is the impact of increasing modularity and standardization on the IT environment. An increase in standardization ultimately makes it much simpler to upgrade or off load services in the future as business needs change.

1.2 Standardization and Enterprise Architecture Maturity

Enterprise Architecture maturity has a significant impact on the ability of IT to support business objectives and initiatives as illustrated in Figure 1–2.

Figure 1–2 Enterprise Architecture Maturity[1]

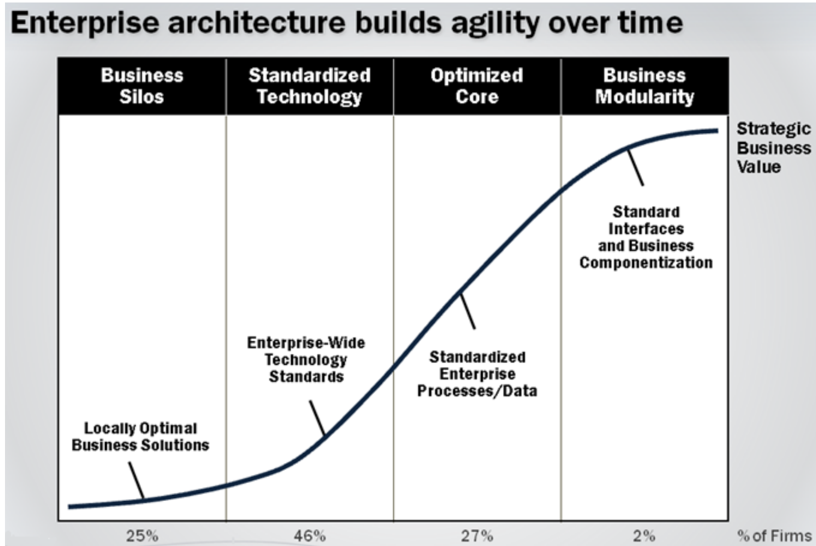


Figure 1–2 shows that the path to IT providing strategic business value includes standardization. Standardizing the technology is the first step toward reaching higher levels of strategic business value. As discussed above, technology standardization can be accomplished in-house by the IT department selecting specific products to be included in the enterprise architecture. Engineered systems increase the granularity of the overall architecture by providing a particular service at an optimized performance level. Since the engineered systems are specialized around services, they tend to increase granularity and modularity in the architecture by shifting those services to dedicated hardware which has been configured with all the necessary components to provide the service.

For example, if the IT department wants to provide application server functionality as a standardized service they must first decide on the standard platform that will provide that service. Historically this required the IT department to select of a variety of products including servers, operating system, virtualization software, application server software, and management software. With the advent of engineered systems, this entire selection process can be reduced to selecting an engineered system that provides application server functionality (e.g. Oracle Exalogic).

The addition of Engineered systems into the traditional IT infrastructure comprised of in-house custom architecture has the distinct advantage of moving an organization to a more standards-based infrastructure with greatly improved monitoring, performance, reliability, scalability, supportability and defined upgrade options. This leads to a significant reduction in engineering effort applied to designing and maintaining custom solutions and frees those resources to focus on business related development efforts.

1.3 Historical Perspective

Engineered systems can be viewed as a natural evolution of IT systems. Engineered systems are the convergence of dedicated resources and open standards. Although some consider engineered systems somewhat of a return to monolithic mainframes of the past, this is an inaccurate characterization. Rather than a return to monolithic,

[1] Source: ***Enterprise Architecture and Strategy: Creating a Foundation for Business Execution***, J. Ross, P. Weill, D. Robertson, HBS Press, 2006

general purpose systems, the Engineered system is optimized for performance around specific services. The introduction of these specialized, service-oriented systems ultimately increases the modularity and standardization in the overall architecture.

1.3.1 Monolithic, Multi-Use Servers

Historically, enterprise computing has been focused on large, monolithic, vertically-scaled servers (e.g. mainframes) in the company data centers which were frequently used to provide a “swiss army knife” mix of services. For example, a large server may have been purchased to provide a database instance. Over its lifetime, the expensive but under utilized machine may have had several other services added, such as providing **DNS** and **LDAP** for the company **WAN** environment. Another department may have needed a “small” departmental web server instance, which was also added to the server, and later grew to become a business critical web server.

Thus, what began as a dedicated database server, over time may have morphed into a complicated, multi-use enterprise system. Maintenance and upgrades to the system became much more complex due to the diversity of software hosted on the server. Any upgrades or maintenance could end up impacting multiple processes and multiple departments.

As IT departments move away from this monolithic server approach to a more highly distributed, shared-service architecture, engineered systems have an important role and substantial impact. The introduction of engineered systems helps to simplify the overall environment by modularizing service delivery and integrating the monitoring, support, security, and configuration around these specific modular services. This increase in service modularity is the key to improving performance in the IT environment. The optimization provided by specialized hardware and software tuning simply cannot be matched without the dedication of substantial engineering resources. Even if a custom solution meets the performance requirements, it is not guaranteed to scale or have a well defined upgrade path. Modular architecture which is standards based opens the IT environment to performance enhancement one service at a time while reducing the staffing load required to maintain non-standard custom solutions.

Take, for example, an engineered system dedicated to providing the database need in the case described above. That system would only provide the database, and it would be guaranteed to perform well, and provide a level of service which is well defined in advance. To move to such a system, the company must simply migrate its data into the new machine and by doing so receives the benefit of a completely supported, monitored, and serviceable solution which by nature moves their architecture into a more modular, service-oriented, industry standard-based architecture.

This standardization and shifting of focus to modularized services radically simplifies the purchasing decisions, patch administration, and future upgrade path. Though the enterprise system itself may be a locked device, the service is not. Since the company is now using a system which provides a database service, in the future, it becomes relatively simple to plug in a new and improved system which will provide that database service at a higher performance level.

1.3.2 Convergence of Open Systems with Standardization

In recent years, the industry has begun to embrace the concept of openness in system design in order to provide flexibility and compatibility to customers. In order for a system to coexist with systems provided by other vendors (both commercial and open source), adoption of industry standards for system features, elements, protocols and application programming interfaces (**API**s) has become a necessity.

The concept and adoption of open systems greatly reduces the quantity and impacts of dependencies in an architecture, and by doing so, increases architectural modularity. Standards driven architectures no longer compete by introducing proprietary capabilities into an environment, but rather, the differentiators become the more meaningful elements of a system such as performance, reliability, footprint, power consumption, capacity, etc. Therefore, the move to standardize architectural components within an environment allows the company to focus on the actual needs of the business, rather than focusing on how to make a system from vendor A work well with a system from vendor B.

Open systems and industry standards has the added benefit of removing the previous convention of locking a company into a proprietary solution with a fixed upgrade path. The company may still choose to integrate proprietary components into their architecture for performance advantage, but they are no longer required to stay with that system should a better alternative or newer release become available.

Engineered systems can be viewed as the evolution of IT systems to incorporate open standards and modular service delivery. This results in fewer choices and less flexibility in the usage of IT infrastructure, but also delivers standard functionality at reduced cost and at an increased level of service.

2

Impact of Engineered Systems on Architecture and Organization

Engineered systems provide a new approach to delivering IT capabilities to the business. As with any new products and approaches, there are principles and impacts that a company should take into consideration. This chapter identifies and discusses both the architectural principles and the architectural and business impacts upon an IT organization when adding engineered systems.

2.1 Engineered Systems Architectural Principles

As an IT organization moves to a more standardized implementation and begins to introduce engineered systems into their architecture, a common theme of key architectural principles becomes apparent. These affect the IT architecture in different ways, but they will apply to future IT decisions and should be considered during the design phase.

- **Increased Modularity:** Architectural decisions are broken down into major components which reinforces the concept of architectural layers/tiers/zones. Adding engineered systems to an IT environment increases the modularity of that environment. Moving to a more modular architecture greatly simplifies the replacement of those modules as new performance requirements arise, or as upgrades become available.
- **Increased Granularity:** By moving to an engineered systems approach for specific capabilities within an architecture, the granularity of the architecture is naturally increased. Instead of selecting, deploying, and managing complex systems providing multiple services, engineered systems allow the IT department to work with systems focused upon providing individual services.
- **Specialization:** Engineered systems are optimized to provide one (or a few) specific services well. The systems are designed around that service with the intention of meeting specific performance requirements. Since unforeseen software and hardware changes are not permitted, the specialization leads to improved architectural awareness of the system's impact, and enables one to architect for specific requirements.
- **Standardization:** The push to provide industry standard external interfaces into engineered systems greatly improves the flexibility in the overall architecture of the IT environment. Standardization allows the architect to focus primarily on the key metrics of performance, capacity, scale, etc., without concern for the internals of how that service is actually being provided, or what architecture is being used internally to provide the service. In the past when a company built their own custom solutions, they often created roadblocks to growth and scale as an

unforeseen side effect. Many of these roadblocks occurred due to lack of standardization, and there would always be some concern as to whether the custom system actually met industry standard requirements.

- **Abstraction:** Interaction with engineered systems is limited to a defined set of services and high-level interfaces. Typically the end users of an engineered system only interact with the service the system provides and the administrative interface/controls that are available on the system. This closed system model greatly simplifies the use of the system and prevents the introduction of foreign software which would impact performance and create unwanted dependencies. Introducing an engineered system increases the level of abstraction within a company's architecture. The architecture, for example, would now contain a box whose entire reason for existence is to provide a database service (e.g. Oracle Exadata), or to provide middleware as a service (e.g. Oracle Exalogic).

2.2 Engineered Systems Architectural Impacts

The following are examples of impacts to the overall IT architecture that may be experienced following the introduction of engineered systems. In total these impacts really illustrate the advantages of moving to engineered systems and they generally lead to a reduction in custom engineering workload.

- **Increased Integration:** Engineered systems are pre-packaged, tested, supported, qualified for software, sized by service measure. The company must ready their existing architecture for the introduction of a standards based system which is providing a specific service.
- **Streamlining:** Engineered systems reduce decisions, actions, and preparations required to acquire and deploy, and greatly simplifies repeat purchases. The option to select a system to perform a task at a known price and performance level greatly simplifies the purchasing and deployment at the data center. There is no longer a need for the company to design a custom system from hardware components, assemble a software stack out of a set of widely variable tools, and hope that the result with actually meets their needs with minimal waste of resources. The engineered system greatly reduces time to deploy simply because it removes the need to spend months building a custom computer to provide the service.
- **Capacity Planning:** Engineered systems are sized for predetermined capacity and performance requirements. Capacity planning for engineered systems is dramatically simplified since they ship in pre-defined (and tested) sizes (e.g. small, medium, large) with well understood performance characteristics. Through sizing, the customer may maximize utilization while minimizing cost since it is the nature of the engineered system to be able to easily scale to meet future demand.
- **Combined Software/Hardware Decision:** Engineered systems combine the software and hardware purchasing decision governance process. Traditionally, organizations often separate their hardware and software purchasing decisions. The system administrators often specify and purchase hardware, while leaving the software purchase decisions to the client departments. The challenge in engineered systems is that the purchase of software and hardware is made at the same time. It becomes necessary to coordinate the purchase among teams early in the purchasing process.

2.3 Flexibility Levels of Engineered Systems and Appliances

Each engineered system typically has a pre-defined level of configuration flexibility. One example of an engineered system may be tightly controlled and allow only a few configuration options. Another engineered system may allow greater variability in configuration by, for example, allowing variability in which software components can be hosted by the system.

Having an engineered system with limited configuration flexibility may, at first, seem limiting; however, limited flexibility is actually a feature that has a positive impact upon an IT architecture. A truly self-contained, service-oriented engineered system tends to have its flexibility tightly controlled since any configuration options available to the customer are only those that are relevant to the service being provided, or how the engineered system interacts within the IT architecture. Thus, less flexibility in customization is a requirement of an engineered system in order to meet industry standards, but engineered systems have varying degrees of flexibility dependent upon the service roles they are designed to deliver.

For example, in order to achieve maximum performance, the Oracle Exadata product is a relatively closed engineered system which has well defined interfaces for customer access. On the Exadata, configuration is limited to the basic attributes necessary for monitoring and integration into the environment. The internal database engine is optimized for the included hardware and the internal system is well defined, controlled, and is generally not accessible to the end user. The end user interacts with the Exadata primarily through the database interface from the network and is not concerned with the internals of how the data flows inside the machine or how the Exadata chooses to distribute data on its internal storage.

Other engineered system instances may be more open to custom configuration e.g. a system optimized for hosting virtual machine images. The hardware and storage are optimized for providing virtual machines; however, the contents of the images are completely open to configuration by the customer. Thus, the virtual machine system is inherently much more flexible in its potential uses, since it may provide myriad services on multiple hosted operating systems.

The family of engineered systems includes systems referred to as appliances. Typically, appliances may be even more focused around a particular workload or function. The appliance is even more tightly controlled and locked down than the typical large Exadata style engineered system. Appliances are in essence engineered systems consisting of completely integrated and configured hardware and software solutions designed to perform a specific function in a predefined way. Thus, the flexibility level of an appliance is even more tightly controlled, and customer interaction with the internals of the system is not typically permitted other than the very basic site specific configuration information required to integrate the system into the infrastructure.

When implementing an architecture which includes engineered systems, the degree to which any engineered system or appliance is accepted into an architecture will depend upon a number of factors. There is invariably a trade-off between flexibility and performance optimization. The desired level of performance must be balanced with the need for custom configuration and integration with third party applications. Typically, more customization away from the standard optimized stack may lead to some degradation of performance and supportability. Therefore, it is important to be mindful of the architectural impact engineered systems may have in terms of overall flexibility.

If the system is intended to provide for wide variability in the services hosted, there is little, if any, opportunity to optimize the system. Conversely, if the system is designed to provide a very specific service then the system can be engineered to deliver that specific service in the most optimum way and a vendor optimized engineered system may be the best fit.

2.4 Value Proposition for the Adoption of Engineered Systems

There are many distinct value propositions associated with the introduction of engineered systems into an IT architecture. The benefits of introducing engineered systems to replace dated methods of custom configuration on monolithic server architectures are:

- **Reference configuration work has already been done:** The customer does not have to spend months designing a new machine to provide a specific service in their environment. The customer no longer needs to dedicate resources to develop solutions.
- **Reduced time to deploy:** Engineered systems are well defined and documented with predefined system configuration options. The system can be installed, configured, and placed in service within days as compared to the traditional multi-month process most IT departments require when deploying a new server.
- **Known performance, capacity, and scalability:** Engineered systems are designed to run specific software well, i.e. the system is sized to provide the appropriate capacity and configured to alleviate bottlenecks. Additionally, engineered systems are designed to scale out to meet future needs.
- **Simplified ordering process:** Software and hardware is included in a single purchase, with the selection simplified to concepts such as small, medium, or large.
- **Unified vendor provided support:** The engineered system is treated as a single product with an associated support contract. The internal components are self monitored, often times including an optional "phone home" capability. Additionally, fail-in-place and hot swap service may be provided. These features are included to help provide a specific level of service uptime guarantee.
- **Future upgrade path well defined:** By moving to engineered systems, an IT architecture becomes standardized, thus simplifying the process of upgrading or scaling out an individual service within the environment as needed.
- **Ease of use:** The entire set of engineered systems from Oracle may be managed and monitored from a single point using Oracle Enterprise Manager.
- **Oracle options:** Oracle offers complete solutions from applications to tape backups, completely integrated and configured to work with the engineered systems offerings. This greatly reduces the architectural load required to provide the production computing environment. Instead of spending time and resources designing the production computing environment using commodity components, the IT department can focus their efforts on delivering business solutions.

2.5 Impacts upon the IT Organization

Aside from the more obvious technical aspects of engineered systems, organizations must also consider the potential for shifts in roles and responsibilities of their staff. Typically the motivation to add these complete solutions into an existing environment is driven by a specific set of business needs. These needs often drive the direction of the organization while at the same time, tend to morph existing staff roles to better meet the needs of the business. In essence, this is the natural evolution of the overall

IT organization, and though it isn't specific to engineered systems, it is a consideration that roles may be refactored at an accelerated pace.

These role shifts occur as a side effect of shifting the overall architecture of the IT environment to a higher level of abstraction. The addition of completely integrated solutions into a legacy environment creates a situation where the IT staff is no longer required to focus their attention on the small details of how to design, purchase, integrate, and maintain custom systems. Rather, the IT staff can now interact with their IT architecture at a higher level, e.g. they purchase a database machine to provide the database, a middleware machine to host the applications, a cloud infrastructure machine to host virtual system images.

Through shifting to a higher level of abstraction, scaling and management issues are reduced in exchange for a slight reduction in flexibility in the ability to customize system internals. This trade-off is not necessarily a disadvantage, since the move to higher abstraction implies an increase in standardization. Standardization is the key to reducing complexity and reducing roadblocks to future scaling.

2.5.1 Impact to the Purchasing Process

Since engineered systems are complete software and hardware solutions, they have an impact during purchasing phase, since the organization now must make a combined hardware and software decision. In legacy environments, hardware purchases have been typically made by the system administrators, while software purchases were often made by teams responsible for individual pieces of software. Therefore, when selecting an engineered system, an organization may experience changes in purchasing decision governance.

Upon selecting an engineered system, the application owner is most likely purchasing the system comprised of both hardware and software. Thus, in order to successfully incorporate the new system into the data center, the organizational workflow may be altered to accommodate the new system style. Approval processes are typically changed: primarily who makes the decisions, when they are made, and what steps are taken to complete the process. In general the entire purchasing process and architectural decisions become lighter touch and iterative. The decisions happen earlier in the process than organizations typically have experienced in the past. Since the systems are complete, there are fewer up-front choices to be made, but those choices tend to drive future architecture decisions. This is not necessarily a disadvantage, since the early decision ultimately raises the level of standardization in the overall architecture.

Prior to purchasing, the IT teams must collaborate to ensure that the environmental, infrastructure, management, and security requirements are well understood. The network administration team must be aware of the needs of the new system, and must allocate the necessary resources for integration into the existing network infrastructure. The systems administration team similarly must be engaged to assist with integration into legacy monitoring and backup environments, as well as for initial system configuration and migration. The electricians and HVAC engineers must be consulted to ensure that there will be enough power and cooling capacity to meet the requirements of the new machine, and any future scaling. And finally, it may be necessary for an organization to determine how the new system will fit into their internal cost structure given the shift in roles during production.

2.5.2 Impact to System Administrator Roles

Another consideration is the change in the way the systems are managed and provisioned. In the past, the systems administrators would manage resources, provision systems, and possibly build custom images for specific applications. Often the steps to provision new applications into legacy systems required a fully engaged systems administrator to handle all the necessary system configuration, interaction at the **CLI** level, and develop custom monitoring scripts and backup procedures. The engineered systems include self-service facilities to enable these activities designed with future scaling in mind. The role of the system administrators becomes lighter in the day-to-day operation of the machines, which ultimately enables a single administrator to manage many more systems. One of the primary problems of out of control growth in the IT environment is the manageability of large complex systems, so the move to more standardized service-oriented engineered systems directly addresses this problem.

Upon initial installation of an engineered system, the systems administration team load is dramatically reduced to basic configuration work since the system is already qualified and ready to run from the factory. The systems administration staff may now find more time to work on projects they have had to set aside in the past, since they are not required to purchase hardware and software, integrate a new server, qualify and certify the machine, and move production efforts to the new system. Since the process is so greatly simplified, the addition of future machines into the environment to address growth and scaling needs is much more streamlined.

2.5.3 Impact to Storage Administrator Roles

In legacy environments, storage administrators would be responsible for carving up and distributing disk resources, and developing backup and data migration plans. The engineered systems are highly automated and often do not allow for direct access to the internal storage structure. The traditional storage administrator role shifts from a more hands on internal custom partitioning approach to a higher level policy management role through the self-service administrative interface. Storage administrators may be involved during the initial integration into the existing backup infrastructure, but the daily management load specific to the internals of the system is greatly reduced. The storage administration team is freed to focus their efforts on more business critical projects such as maintaining infrastructure resources (e.g. archival storage, data backup, and replication).

One potential caveat regarding the internal storage is that it may be difficult for an organization to use traditional certification methods to qualify the internal storage for production. However, given that the storage is delivered as a vendor qualified, certified, and supported internal component of the system, it should be possible to alter the internal organizational policies to validate the machine.

2.5.4 Impact to Network Administrator Roles

In the past, network administrators may have built custom private **VLAN**s for individual applications, monitored traffic within the system, and spent hours tuning internal network implementation for performance and reliability. Through the self-service interface, much of these needs are automated and handled internally to the engineered system. Internal network structure has been predetermined to meet the performance requirements of the machine, and internal VLANs are automatically provisioned based upon application provisioning requests of the application owners. The network administrator role shifts more to high level tasks such as external network performance, and managing **LAN** infrastructure components such as load balancers, monitoring systems, traffic shaping, and firewalls.

2.5.5 Impact to Database Administrator Roles

Traditionally, database administrators have been responsible for working closely with the systems administration staff to deploy new database instances on custom hardware. Of course, there are many dependencies that must be overcome prior to going into production, from the initial purchase decision, waiting for integration, purchasing the software licenses, qualifying the new machine, etc. Generally, the time from purchase decision to production of a new database can be on the order of months. Ultimately the database administrator's role has been to provide a database service to the application owner clientele.

When moving to an engineered system such as the Exadata, the entire process becomes simplified and accelerated. The addition of an Exadata implies that the database administrators are now selecting an integrated hardware, software, and storage solution to provide the service. Initially the database administrators must handle the migration of data into the new system, and potentially they must address any issues that arise from database version mismatch between their legacy systems and the new Exadata system. Once the integration and migration is complete, the database administrators may focus their effort on business specific development rather than database software maintenance and tuning they have handled in the past on their legacy systems.

2.5.6 Impact Variance from Engineered System Types

The introduction of an Oracle Exalogic system into an existing environment has a slightly different impact, illustrating the variations among engineered systems. Traditionally, the system administration team would have been responsible for specifying and integrating custom systems to host corporate business applications. Much of their time was spent selecting the appropriate hardware from various vendors, purchasing licenses, integrating the software stack, qualifying the software, developing a security plan, all while being mindful of the future upgrade path. This is a daunting task and ultimately, systems are only typically designed to the capability level of the staff. Thus, the primary advantage of selecting an engineered system is that all of the work has been completed by the vendor in advance.

Engineered systems ship from the factory ready to go, and often may be floored and live within a single day. In the case where all software on the system is built entirely on Oracle products, the only delay is the time required to migrate to the new system. The impact on roles in the case of Exalogic may be rather minimal, in the sense that it is mostly freeing time from configuration, maintenance, and certification, while also enabling vendor support for the entire stack. The performance advantage tends to free up development staff to focus more on development rather than supporting legacy systems on highly varied custom platforms.

Following initial system installation, there may be a shift of responsibility to the application owners away from the IT staff for application specific activities. Since the software stack internal to an engineered system is typically pre-configured and vendor supported, responsibility for application development and deployment falls mainly upon the application owner. The need for system administrator coordination in deployment effort is greatly reduced, since the application team will generally have the necessary access to the management software for application provisioning and control. This shift in control to a self service framework dramatically reduces time to deployment and increases flexibility allowing the application developers a more rapid develop, test, and deployment cycle.

3 Engineered Systems in the Cloud

As companies move to cloud computing environments, the introduction of engineered systems into cloud architecture has an important role. The adoption of engineered systems shifts the architectural strategy away from a monolithic, shared-server architecture to a service-oriented, highly standardized, highly modularized architecture; thereby facilitating the transition to a dynamic cloud environment. Services provided by engineered systems may be considered as infrastructure or platform services provided within a dynamic cloud implementation.

For example, a database service which is utilized by a dynamically provisioned cloud cluster could provide the data storage backend to a large customer facing **OLTP** cloud. A middleware system (e.g. Exalogic) may be used to provide a cloud implementation in software which dynamically scales out to meet interactive web server requirements, while interacting with the database service for data persistence or analysis. The built in data migration capabilities in the Exadata may be used to distribute data to multiple database service instances around the world to provide automated redundancy and high availability through a well defined supported solution.

By enforcing standardization and service abstraction, the introduction of engineered systems simplifies horizontal scaling in a dynamically provisioned cloud environment for automated, demand-based load scaling. As cloud presence expands, the database service behind the cloud may expand to meet the cloud needs. However, since the database service is modular, that expansion can occur at a different rate to optimize for data capacity need and cost, independent from cloud processing need.

Another example of service independent scaling would be the introduction of an engineered system dedicated to providing an identity management solution. A cloud deployment may consist of thousands customer facing interactive web server nodes which need common identity management. A single identity system may potentially be able to provide service for that cloud, and more systems may be added as needed to address future cloud expansion when a capacity metric has been surpassed. Therefore, through the introduction of engineered systems, it is now possible to scale out individual services as needed and optimize them based upon individual customer application needs, ultimately optimizing the architecture around cost and performance on a per-service basis.

The primary influence of engineered systems in a cloud environment is to enable service abstraction with enhanced performance of the service while accepting the reduction in flexibility provided by completely custom implementations. This flexibility trade-off may be considered beneficial in the sense that the increase in performance and granularity within a given architecture outweighs the reduction in flexibility. At the same time, the IT department gains through reduction of support load on their staff, improved availability to the cloud client systems, and overall reduction of total cost of ownership.

Engineered systems may be considered as **PaaS** (Platform as a Service) solutions since they generally provide well defined service interaction at the API level. It is important to understand that PaaS is not required to exist on top of an **IaaS** (Infrastructure as a Service) implementation, but may be implemented entirely within a software solution as it is in the case of an Exadata. The hardware providing the PaaS and the software internal to the system has been optimized to provide the service at the highest possible performance multiple. Therefore, the advantage of integrating an engineered system into a cloud environment is the performance and reliability provided by those systems is understood, and may scale to provide the services to a dynamically scaled IaaS or PaaS cloud deployment. As the cloud service demand changes and the monitoring system indicates that the engineered system is approaching maximum capability, it is possible to horizontally scale the engineered system in concert with the cloud to meet the increased demand.

3.1 Oracle Engineered Systems

Oracle has embarked on a path to provide a variety of engineered systems and appliances to simplify the deployment of complex enterprise systems. Currently there are several examples of engineered systems that are available as products from Oracle: Exadata, Exalogic, Exalytics, Oracle Big Data Appliance, SPARC Supercluster, and the Oracle VM Optimized Solution for Enterprise Cloud Infrastructure. There are many other engineered systems that are in various stages of development, some nearing production while others are still in the inception phase.

3.1.1 Exadata

The Oracle Exadata database system is a mature example of an engineered system. The Exadata is designed to provide a turn-key database as a service solution while providing a substantial performance advantage over a traditional database on a dedicated server implementation. The Exadata hardware and software have been designed and tuned for performance, and since both aspects of the system are engineered together, it has been possible to greatly enhance the overall performance. A high-speed, low-latency **InfiniBand** interconnect is used internally to dramatically improve the network performance within the system. In addition, the software was refactored to add query off load directly to the storage hardware; thus moving the database query operations closer to disk and reducing internal network traffic. The Exadata internal storage hardware takes advantage of solid state disks for caching recently accessed data providing another performance boost.

These performance enhancing capabilities are not typically implemented in a customer designed system, nor would they be possible without standardization. Finally, the Exadata internal InfiniBand fabric has been designed with horizontal scaling in mind, making it possible to attach up to eight Exadata systems on a common fabric to increase capability and capacity without performance degradation.

The Exadata machine is positioned to replace existing database implementations as a universal system to handle both OLTP and data warehousing requirements. Unique internal automatic tuning mechanisms enable this capability, making the Exadata a highly capable upgrade for existing installations.

An Exadata implementation may be integrated into a cloud environment typically via a common 10 Gbps **Ethernet** LAN fabric to the cloud hosting servers in order to provide database as a service to cloud client customers. A common identity management scheme in a cloud environment may be used in conjunction with Exadata customer defined schemas to provide individual database as a service instances to each cloud customer. In the case of a corporate private cloud, an Exadata could be used to collect all OLTP data and customer information, while concurrently performing internal business intelligence processes on the same data.

Since cloud implementations are typically highly distributed, a company may wish to have multiple database instances in multiple locations around the world to implement high availability through redundancy. Oracle provides tools for data migration to remote sites, where two distant Exadata systems may be kept synchronized as live backup machines which may also be used to take on additional transactional load as cloud utilization expands. In addition, Oracle offers tape backup solutions to automatically migrate data to tape for archival purposes. All of these capabilities are known, qualified, and supported because Exadata is an engineered system; thus the time and effort to implement these capabilities is greatly reduced.

3.1.2 Exalogic

Another example of an engineered system is the Oracle Exalogic integrated application server machine. The Exalogic provides a hardware platform engineered to work with Oracle middleware software to provide a qualified, supported elastic cloud implementation. Each Exalogic system is comprised of hot-swappable compute nodes, a high performance disk subsystem, and a high bandwidth InfiniBand fabric in a standard 19" rack. The InfiniBand fabric includes the switches necessary for the internal nodes and includes free ports for attaching to additional Exalogic or Exadata racks. In addition to the InfiniBand, multiple 10 Gigabit Ethernet ports are provided for integration into an existing 10 Gbps Ethernet LAN.

Since the Exalogic system supports a variety of applications, the Exalogic system is more flexible than the locked down Exadata system from a custom configuration perspective. Additionally, it has the distinct advantage of utilizing unique Oracle software performance enhancements which take advantage of the internal InfiniBand fabric for low latency, as well as built in redundancy and scalability.

The Exadata provides a complete infrastructure for building and operating Oracle Fusion Middleware and Fusion Applications in a fully supported configuration. The overall system has been designed for a minimum of twice the performance multiple in comparison to a fully custom implementation built with off the shelf components. The hardware and software have been optimized to work together and ship from the vendor as a complete, fully supported solution. This provides a basis for a cloud application server with unmatched performance, scalability, and reliability.

3.1.3 Exalytics

A recent addition to the suite of Oracle engineered systems is the Oracle Exalytics business intelligence system. The Exalytics system is a closed appliance consisting of hardware and in-memory analytics software optimized to work together. The system has been designed to integrate with existing Oracle and non-Oracle data sources and provide a platform for incredibly fast business intelligence, modeling, forecasting, and planning applications. The key differentiator is that the system is closed and optimized to work out of the box with known performance characteristics, yet it is built upon industry standard interfaces and applications. Internal configuration is controlled for supportability, while maintaining enough flexibility for ease of integration into existing data infrastructures.

3.1.4 Oracle Big Data Appliance

The Oracle Big Data Appliance is an engineered system designed to help businesses integrate and maximize the value of large unstructured data sets. The Oracle Big Data Appliance provides a capability for handling data that is not readily accessible in enterprise data warehouses such as Weblogs, social media feeds, smart meter data, sensors, and other devices which generate massive volumes of data (commonly defined as 'Big Data').

The system is built upon a customized stack of open source software including Apache Hadoop, Oracle NoSQL Database, Oracle Data Integrator Application Adapter for Hadoop, Oracle Loader for Hadoop, and an open distribution of R. The appliance is designed to integrate with Oracle Exadata and Exalytics over a high speed InfiniBand connection. The combination creates a mechanism for analyzing unstructured data and migrating the data into a structured Oracle data warehouse providing an enterprise ready deeply-integrated environment for advanced analytics.

3.1.5 SPARC SuperCluster

The Oracle SPARC SuperCluster system is another recent addition to the Oracle engineered systems family. The SPARC SuperCluster is designed to be a highly flexible engineered system optimized for general purpose computing. The primary design goal of this system is to provide a scalable building block for creating an upgrade path for legacy Solaris system migration. Since the system is designed for general purpose computing, the system is much more open to end-user customization than the previous examples of engineered systems. The primary differentiators of this machine in comparison to traditional *à la carte* systems are the pre-integrated management infrastructure, redundant power supplies, integrated storage, redundant dual rail InfiniBand interconnect, and qualified performance capabilities for Solaris and SPARC virtualization environments.

3.1.6 Oracle Optimized Solution for Enterprise Cloud Infrastructure

In the x86 product line, another highly flexible example of an engineered system is the Oracle VM Optimized Solution for Enterprise Cloud Infrastructure (hereafter referred to as the Oracle VM Blade Cluster). This engineered system is completely flexible and open to client configuration while providing a solid infrastructure as a service platform with understood performance characteristics. The Oracle VM Blade Cluster is a single rack, integrated Sun Blade 6000 system qualified to run Oracle VM images within a unified management and monitoring framework. The Oracle VM Blade Cluster includes sample OS image templates for Oracle Enterprise Linux and Oracle Solaris optimized for Oracle VM software.

In a cloud environment the Oracle VM Blade Cluster includes hot-swappable Sun blades, a 10 Gbps Ethernet switch fabric, and a Sun **ZFS** storage appliance. Unlike the more narrowly focused systems previously described, the Oracle VM Blade Cluster provides a platform fully customizable for cloud compute hosting providing both Infrastructure-as-a-Service (IaaS) as well as Server-as-a-Service. Oracle VM images may be migrated to remote sites and cloud compute implementations and brought online in a completely controlled and monitored environment.

The system ships with a pre-installed set of OS options, including Oracle VM server, Oracle Linux, and Oracle Solaris. The systems may be customized as needed on demand and are hosted by the application aware Oracle VM software stack. The introduction of this system into a legacy environment has the unique impact of increasing standardization and modularity into the overall architecture. Moving to an Oracle VM solution enables the option of image migration, high availability, and automated workload demand based loading as is the ultimate goal in a cloud IaaS solution. All of these capabilities are accomplished under complete control on a single console via Oracle Enterprise Manager. Architecturally, moving to a standardized system such as this will simplify horizontal scaling in the cloud while working within known performance and scaling capabilities, thus greatly simplifying deployment and reducing deployment time.

As part of a high availability solution, custom configured Oracle VM images may be shipped to remote data centers and automatically enabled to meet load-based dynamic demand or to ensure remote site hot backup capabilities. The Oracle VM Blade Cluster is attached via 10 Gbps Ethernet fabric to the LAN fabric and may be used in combination with both Exalogic and Exadata engineered systems to provide a complete cloud infrastructure.

3.2 Centralized Management and Cloud Enablement

An important component of an engineered system is the included integrated management and monitoring software. Oracle Enterprise Manager 12c provides this management and monitoring capability for all Oracle engineered systems. As cloud systems scale horizontally, management complexity increases. Maintaining a high level of centralized control is of paramount importance when providing a high availability elastic cloud solution. Oracle Enterprise Manager 12c offers a single, integrated console for testing, deploying, operating, monitoring, diagnosing, and troubleshooting, implementation of an architecture.

The introduction of a centralized monitoring system has the overall impact of increasing standardization in the control infrastructure in an IT environment. Having a centralized, cloud capable management system both expands flexibility and provides a growth path for future cloud adoption. The increased standardization greatly simplifies the future integration of additional engineered systems expansion into a cloud solution to address future needs.

Oracle Enterprise Manager 12c fundamentally transforms the control architecture by following an Application to Disk paradigm, which:

- Monitors all components through a single pane of glass.
- Provides deep, out-the-of-box capabilities that have proven to reduce problem resolution time by more than 50%.

Enterprise Manager 12c also allows administrators to set up and enable their cloud services, so that end-users can provision resources on demand without being gated by IT for every request. Enterprise Manager comes with a service catalog that includes IaaS, DBaaS, and PaaS service variants. Using the solution, a cloud provider can create one or more of these services and also optionally charge back the tenants.

4

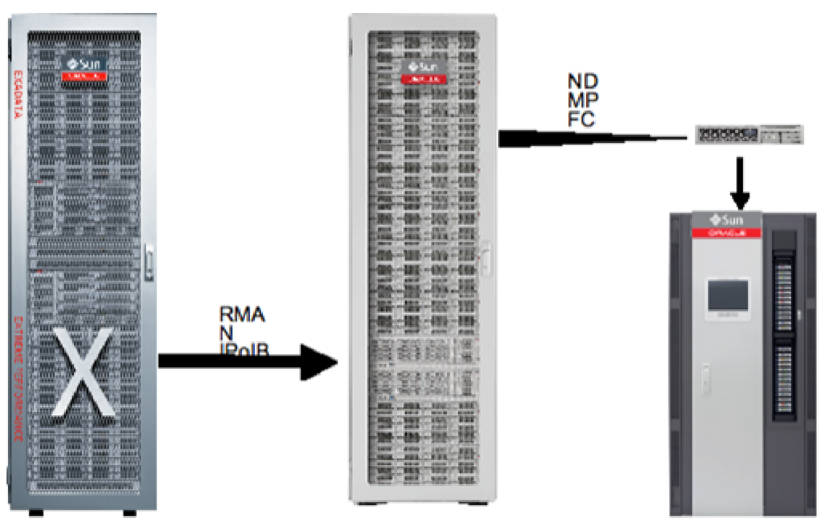
Deployment View

Engineered systems can be deployed to a data center in multiple different ways depending on the services desired, the existing environment, the expected load, etc. This chapter provides a few deployment examples to illustrate common ways in which the engineered systems described in the previous chapter might be deployed to data centers.

4.1 Example Deployment of Exadata with ZFS Storage

Figure 4–1 illustrates a simple Exadata system deployment where additional storage is necessary for a large data warehouse implementation. It is possible to attach additional Oracle ZFS storage appliances directly to the Exadata internal InfiniBand fabric to expand capacity while maintaining low latency data access capabilities.   For archival backup and Oracle tape drive and control head are shown attached via fiber channel for high performance backup and recovery.

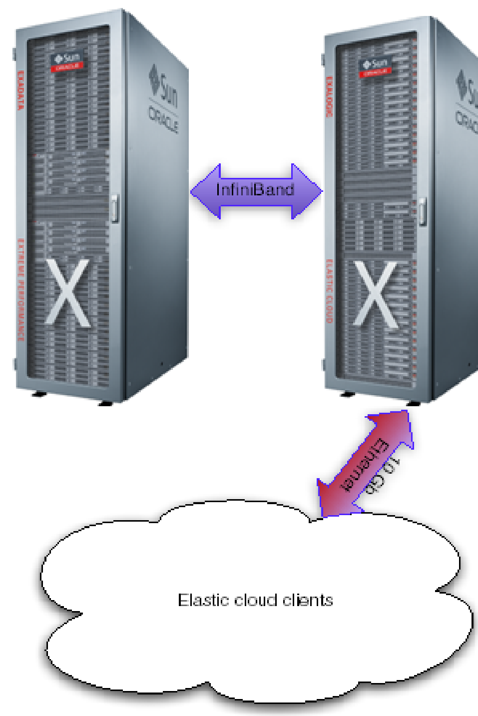
Figure 4–1   Exadata with ZFS Storage



4.2 Example Deployment of Exadata with Exalogic

Figure 4–2 is a simple architecture illustrating how the Oracle Exalogic integrated application server system may be directly attached to one or multiple Exadata systems via the internal InfiniBand fabric.   This architecture is a key element of a scalable system where multiple Exadata, Exalogic, ZFS storage appliances, and tape backup subsystems may be combined to meet customer needs based upon application workload and database capacity requirements. In this simple example, an elastic cloud application server may be implemented on the Exalogic system while OLTP and data warehouse capabilities are provided to the cloud implementation by the Exadata system.

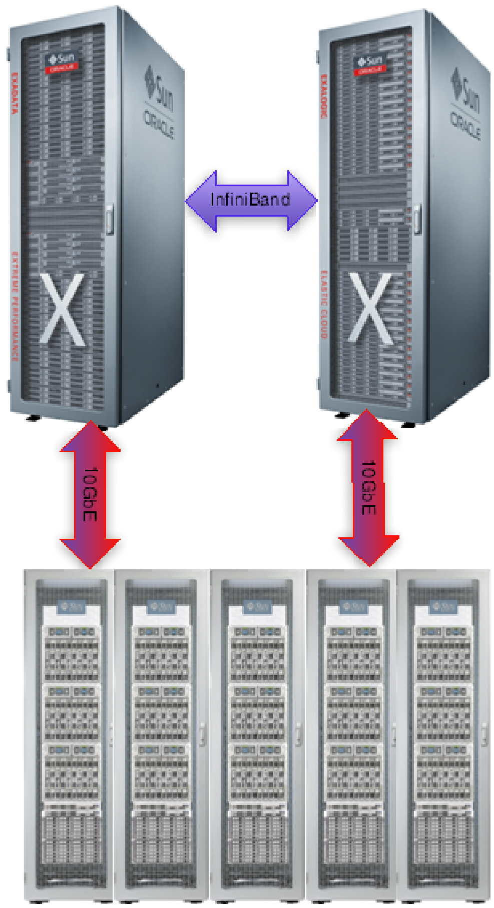
Figure 4–2 Exadata with Exalogic



4.3 Example Exadata, Exalogic, and Oracle VM Blade Cluster

In the case of a more compute intensive cloud implementation with a requirement for more IaaS flexibility, the prior example may be expanded upon through the addition of the Oracle VM Blade Cluster. This configuration allows for a completely supported database and middleware backend with a known performance level to provide service resources to a large IaaS cloud deployment. The Oracle VM Blade Cluster provides a highly flexible and scalable front-end virtualization platform which may be used as a dynamic customer facing cloud front-end, and or a consolidation platform for legacy applications and systems.

Figure 4–3 Exadata, Exalogic, and Oracle VM Blade Cluster

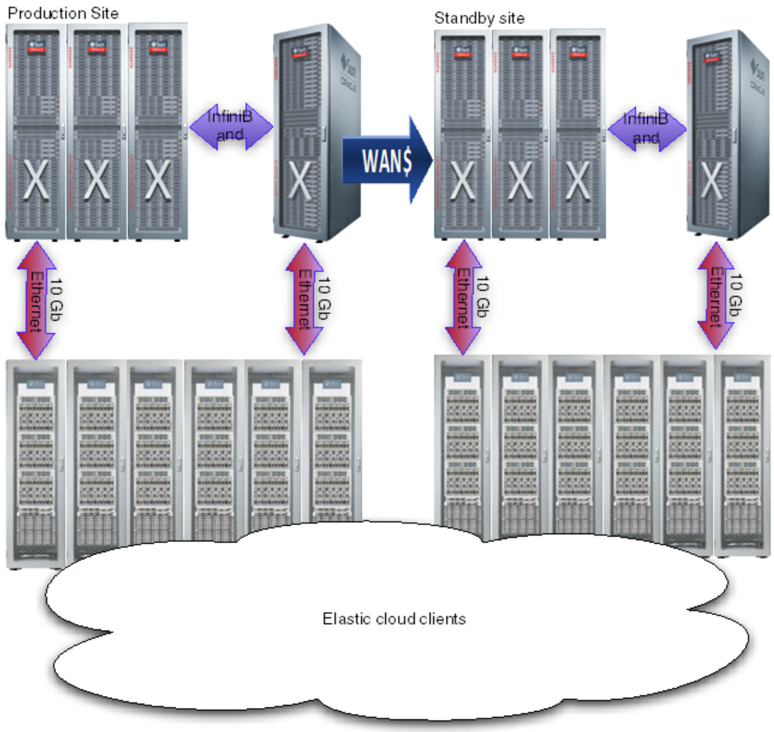


Oracle Optimized Solution for Enterprise Cloud Infrastructure

4.4 Example WAN Deployment

This deployment pattern shown in Figure 4–3 may be easily replicated in multiple data centers or within a single data center to provide a hot standby backup system, or additional capacity for cloud scale-out. Oracle Active Data Guard and **RMAN** tools may be used to synchronize Exadata database instances between two distant sites over the WAN, while allowing for image migration on the Oracle VM Blade Clusters. While the databases are set up in a master-slave configuration, the standby site database is available for local, high performance, read-only queries, while write transactions may be directed over the WAN to the master site. This architecture, shown in Figure 4–4, demonstrates the benefits provided by increasing service modularity. The elastic cloud IaaS front end that is deployed on the Oracle VM Blade Cluster only requires a database service; while behind the scenes, the database is actually being replicated to a remote site for increased availability and performance.

Figure 4–4 WAN Deployment



5 Summary

The constant push to provide enterprise compute resources at ever lower costs without sacrificing quality is driving IT departments to find new, better ways to deliver standardized services. Traditionally IT departments have been required to install and configure software that is hosted on hardware that was itself installed and configured separately. Engineered systems relieve this burden from the IT department by providing a pre-defined combination of hardware and software.

While removing the burden of installation and configuration provides significant cost savings in and of itself, this is only one of the many advantages delivered by engineered systems. An even larger advantage is the optimization that pre-defined hardware and software enables. Since the hardware and software are engineered together as a complete system, there are multiple opportunities to improve the overall system performance. The unified and integrated monitoring and management of the engineered system also provides cost savings through simplification of the overall environment.

Thus, engineered systems allow data center services to be delivered more efficiently via modular, dedicated systems. This greatly simplifies the entire purchase, deploy, configure, monitor, and manage lifecycle of the provided services.

The whole impetus for cloud computing is to provide specific services (whether IaaS, PaaS, or **SaaS**) as efficiently as possible; therefore, the increased efficiency delivered by engineered systems make them ideal candidates for the “building blocks” necessary to create a cloud computing environment.

In addition to creating a cloud computing environment, these "building blocks" may be used to comfortably address a variety of enterprise architecture requirements to provide a path for enterprise maturation alongside the move to cloud adoption. Each building block may be selected as needed to meet specific maturation needs, and thus ultimately they provide an increase in overall architectural flexibility. For example, an architect may choose to integrate an Exadata and SPARC SuperCluster or Oracle VM Blade Cluster early on to begin the process of consolidation into a scalable architecture, and then expand the system later to meet future demand. Shifting the overall environment into a more standardized architecture opens possibilities for future shift into cloud adoption and/or continuation down the traditional enterprise computing maturation continuum while providing for future scale.

A

Further Reading

The ***IT Strategies From Oracle*** series contains a number of documents that offer insight and guidance on many aspects of technology. In particular, the following documents from the ORA series may be of interest:

ORA Cloud Foundation - This document defines the core concepts and terms for cloud computing, provides a conceptual architecture depicting the key capabilities required, and identifies the standards and architectural principles for successful cloud implementation.

ORA Cloud Infrastructure - This document connects the conceptual architecture with a logical architectural view and includes the functional components necessary. Topics include: logical architecture, deployment considerations, and Oracle product mapping.

In addition, the following materials and sources of information relevant to Oracle engineered systems may be useful:

Information on Oracle Exadata

A Technical Overview of the Oracle Exadata Database Machine and Exadata Storage Server - http://www.oracle.com/technetwork/database/exadata/exadata-technical-whitepaper-134575.pdf

Exadata Hybrid Columnar Compression - http://www.oracle.com/technetwork/middleware/bi-foundation/ehcc-twp-131254.pdf

Oracle Solaris on the Oracle Exadata Database - http://www.oracle.com/technetwork/server-storage/solaris11/documentation/exadatawp-394593.pdf

Deploying Oracle Maximum Availability Architecture with Exadata Database Machine - http://www.oracle.com/technetwork/database/features/availability/exadata-maa-131903.pdf

Information on Oracle Exalogic

Oracle Exalogic Elastic Cloud: A Brief Introduction - http://www.oracle.com/us/products/middleware/exalogic-wp-173449.pdf

Oracle Solaris on the Oracle Exalogic Elastic Cloud - http://www.oracle.com/technetwork/server-storage/solaris11/documentation/wp-solaris-on-exalogic-cloud-345932.pdf?ssSourceSiteId=ocomen

Consolidating Oracle Applications on Exalogic - http://www.oracle.com/us/products/middleware/app-consolidation-exalogic-395610.pdf

Information on Oracle Cluster Optimized Solution for Enterprise Cloud Infrastructure

Whitepaper -
http://www.oracle.com/us/solutions/opt-solutions-cloud-technical-wp-405951.pdf

Solution brief -
http://www.oracle.com/us/solutions/oracle-opt-sols-cloud-inf-sol-brief-406108.pdf

Glossary

The following terms and abbreviations are included here for easy reference. Please see the ***ORA Master Glossary*** for other terms used in the various ORA documents.

Application Programming Interface (API)

A set of commands, functions, and protocols which programmers can use when building software for a specific operating system.

Command Line Interface (CLI)

The traditional text based (non graphical) user interface used to interact with a system.

DataBase as a Service (DBaaS)

A concept in cloud computing which delivers database functionality to an end user abstracted to a service model, where the end user is not necessarily aware of how the service is provided or the physical location of the database within a virtual environment.

Domain Name System (DNS)

A service which maps numeric Internet addresses to human readable names.

Ethernet

A standards based physical network connection typically used in local area network environments for intra-computer communication.

Infrastructure as a Service (IaaS)

A concept in cloud computing which delivers computer infrastructure - typically a platform virtualization environment - as a service, along with raw (block) storage and networking.

InfiniBand

A switched fabric communications link used in high-performance computing and enterprise data centers. Its features include high throughput, low latency, quality of service and failover, and it is designed to be scalable.

Local Area Network (LAN)

A computer network which is limited to a small area such as within a system, physical location, or data center.

Lightweight Directory Access Protocol (LDAP)

An application protocol for accessing and maintaining distributed directory information services over an Internet protocol network. An LDAP directory service typically provides information such as a corporate email directory, along with information such as user names, physical addresses, organizational information and telephone numbers.

On-Line Transaction Processing (OLTP)

A class of systems that facilitate and manage transaction-oriented applications typically for data entry and retrieval transaction processing.

Oracle Virtual Machine (Oracle VM)

An Oracle product which provides server virtualization capabilities for supporting both Oracle and non-Oracle applications.

Platform as a Service (PaaS)

The delivery of a computing platform and solution stack as a service. PaaS offerings facilitate deployment of applications without the cost and complexity of buying and managing the underlying hardware and software and provisioning hosting capabilities, providing all of the facilities required to support the complete life cycle of building and delivering web applications and services entirely available from the Internet.

Oracle Recovery Manager (RMAN)

An Oracle provided utility for backing-up, restoring and recovering Oracle Databases.

Software as a Service (SaaS)

A cloud computing concept of "on-demand software" in which software and its data are hosted centrally, typically over the Internet from a cloud provider, which is generally accessed by users using a thin client and/or client software such as a web browser.

Virual Local Area Network (VLAN)

A computer network which is limited to a collection of nodes based on something other than physical location.

Wide Area Network (WAN)

A computer network which typically spans more than one physical location and may cross Internet connections.

Zettabyte File System (ZFS)

A combined file system and logical volume manager designed by Sun Microsystems which by design can store up to 256 quadrillion zettabytes (270 bytes) of data. The ZFS name is a trademark of Oracle.