CHAPTER 5

Server Grid

Server hardware forms a major component of the IT infrastructure in an enterprise. The server infrastructure currently deployed in enterprises typically consists of sets of servers, dedicated to serve the needs of a specific business unit or application. This does not allow flexibility in reprovisioning the hardware to serve more pressing demands on computing power elsewhere, leading to lack of responsiveness, reduced physical-asset efficiency, and huge expenditures on server hardware. In this chapter, we will describe how the Enterprise Grid Computing model for server infrastructure solves these problems by virtualizing servers, allowing server resources across the enterprise to be pooled and shared by multiple business applications. We will discuss how the service levels for multiple applications can be managed in a cost-effective manner by dynamically provisioning server resources to these applications based on their current needs. We'll provide an overview of Oracle technologies that enable enterprises to take advantage of a server grid.

Enterprise Server Infrastructure

Enterprise server infrastructure provides the processing power for running enterprise applications. Processing power in the raw form is delivered by servers that have one to many CPUs, volatile memory (also called *RAM*), and connections to other servers and storage. This server is made usable for the enterprise applications by deploying the operating system and application software that performs specific functions onto it, for example, a database or an application server. Figure 5-1 shows a typical server infrastructure within an enterprise.

Based on the high-level application functions served, server infrastructure may be classified into three tiers, the presentation tier, business application tier, and data tier. This classification is shown in Figure 5-1. In addition to these servers, there are the employee desktop systems, which are not considered "enterprise class servers" and will not be discussed in this book. However, in reality, many of the concepts and issues related to management and configuration of enterprise servers apply equally well to desktop systems.

Presentation Tier Server infrastructure at the presentation tier performs customer-facing functions such as running the company's web site and

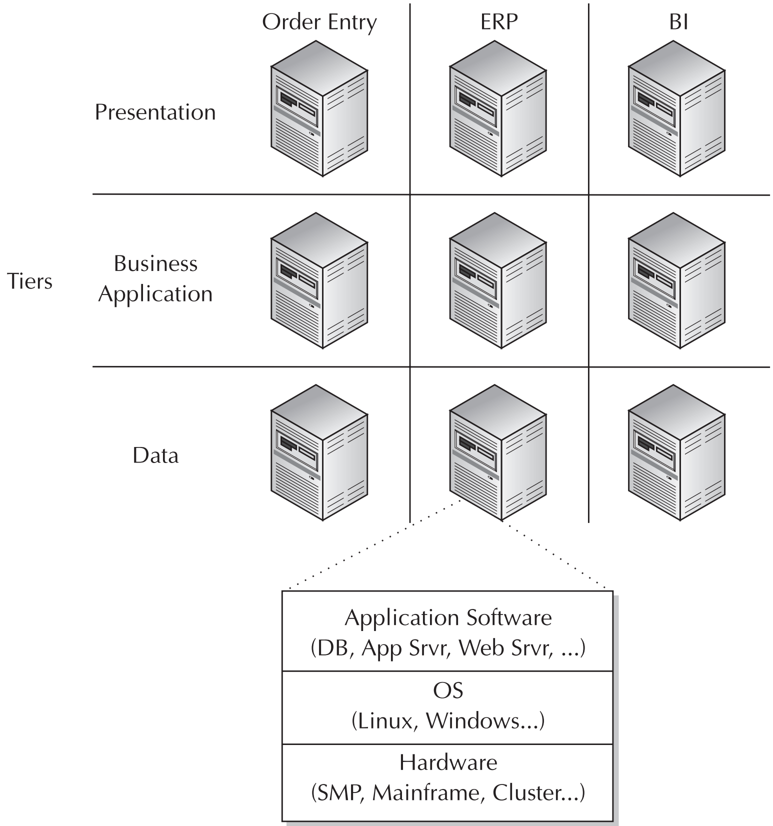

FIGURE 5-1. *Enterprise server infrastructure*

responding to end-user requests. The presentation tier deals with client interactions—displaying information and accepting and preprocessing requests from the client. This tier includes the web cache and web servers that typically provide the first point of contact for customers and employees interacting with enterprise applications.

Business Application Tier These servers run the enterprise application software that includes the actual business logic and process flows. Servers running products like application servers, application integration brokers, and even collaboration utilities like e-mail, web conferencing, etc., also fall into this tier. Often, the presentation tier and business logic are implemented in the same piece of software such as an application server.

Data Tier Servers at the data tier query and update the data that is used by the business application. This tier provides persistence for business data. Typically, this tier runs commercial database software and is also responsible for life cycle management of the business data and information.

Requirements from Server Infrastructure

As we can see, a typical enterprise requires a fairly elaborate server infrastructure. Server hardware and OS and application software, together with IT architecture and processes that comprise the server infrastructure, must be able to deliver requisite service levels to the specific business function it serves. Application service-level requirements are usually expressed in terms of performance, reliability, and security needs. For a business user, the server infrastructure is simply about having access to computing capacity as needed. For an administrator, the infrastructure should support ease of monitoring, management, and automation. For an IT executive, the server infrastructure must do all the above in a cost-effective manner.

Let's review these requirements in some detail.

Performance

The server infrastructure at each tier must satisfy a varying set of requirements for response times and throughput for the applications it serves. For instance, databases used for transaction processing must handle thousands of transactions per second, while those used for data warehousing need high data processing throughput. The presentation and middle tier must be able to handle thousands of user connections simultaneously. These requirements are typically met using a combination of techniques and the server infrastructure must be flexible to respond as these needs change.

Business Continuity

Different enterprise applications have differing degrees of availability requirements, which the server infrastructure should be able to handle. Mission-critical applications must be reliable and fault-tolerant across all tiers. These applications need to be up and running within a short amount of time, even after disasters. With 24×7 e-businesses, if a company's web site or the order-processing system is down for even a few minutes, it could lead to significant loss of revenue. With the shrinking window for planned downtime, it should be easily possible to repair, replace, or upgrade a server hardware or software component extremely fast with minimal downtime. IT management processes must be set up to support such stringent business continuity needs.

Security

Enterprise servers have varying security needs depending on their function. Critical servers, especially those facing the Internet, must be constantly protected from external intrusion such as hackers, viruses, etc. Servers running databases that hold critical business data must be secured to prevent unauthorized access. While security is primarily a concern with respect to the software running on the server and its associated data, the design of the server hardware infrastructure can either greatly simplify or hinder management of security for the application running on it.

Ease of Management

A typical large enterprise employs hundreds to thousands of servers, and hence managing servers is a major component of IT management tasks. Enterprises on an average spend one-to-four times the acquisition cost on management. In order to provide expedient response to a problem, it is imperative to be able to monitor, configure, and manage servers remotely. It should be possible to quickly identify a failed unit and replace it without impacting other running units. Utilities that automate mundane server management tasks can be extremely valuable in helping the administrative staff scale to support a large number of servers.

Provisioning

The computing needs of the enterprise continually evolve due to changing business demands, increased number of users, etc. The server infrastructure must be able to quickly adapt to these needs by provisioning new servers at various tiers. For example, it should be possible to allocate additional servers at the presentation or middle tier to handle increased connection load or to allocate additional servers to the data tier to handle year-end reporting. The time to provision additional hardware to existing applications must be short. Lead times to deploy a new application should be of the order of days and not months. Ideally, the provisioning process should be transparent and happen without the knowledge of the application user. If this is not possible, the downtime during provisioning must at least be known in advance so that application users can plan for any disruptions. This process shouldn't lead to unplanned downtime for enterprise applications. From the viewpoint of the administrator, the provisioning process should be simple, reducing the risk of human error, thereby making it more predictable and timely.

Usage Tracking and Capacity Planning

Enterprises would like to continuously track and measure the value that they are extracting from their server investments. Monitoring server utilization helps enterprises in tracking usage and aids in their future capacity planning. Growth in server utilization can be monitored and used to justify future hardware purchase in a proactive manner. Monitoring server utilization can also help in identifying places where server investments could be reduced.

Reducing Cost

These days, reducing costs is a mantra at every layer of the IT infrastructure and servers are no exception. Enterprises today are reluctant to purchase new servers for a short-term increase in computing power. There is a strong incentive to squeeze every last bit of value out of the investment in servers. While evaluating purchasing decisions for the new servers, enterprises would like to buy the most cost-effective servers that address their needs. Enterprises are also looking for ways to reduce overall cost of the server infrastructure, via simplified management and automation.

Unfortunately, these requirements are not very easy to achieve in currently deployed enterprise architectures. In the next section, we will highlight why this is the case.

Enterprise Server Infrastructure Today

As we have mentioned in the prior chapters, enterprise IT infrastructure today consists of islands of resources, dedicated to specific applications or business units. Server infrastructure suffers from some of the same problems as storage. Typically, each business application has separate server infrastructure for its own components, such as application servers and backend databases. It is not uncommon to find applications running on many different hardware and operating system platforms and versions. Part of the reason is that servers have been purchased from different vendors at different times and various business units have their own purchase and upgrade cycles. Each of these islands is managed individually, resulting in a large number of ad hoc and redundant management processes.

Problems with the Silo Architecture

The approach of deploying server resources, where each business application "owns" exclusive rights to specific server hardware, suffers from several problems, outlined next.

Underutilization

In the silo approach, servers have to be provisioned to meet the peak load of the application. Therefore, the servers are largely idle most of the time, when the peak load is not reached. For example, retailers in the United States make almost 50 percent of their sales during the month prior to Christmas. Fortunately, it is not common for all applications to peak at the same time. For example, in Figure 5-2, we show two applications: Order Processing and Business Intelligence (BI). In the months prior to Christmas, there may be heavy investment in BI to identify a suitable sales strategy. As Christmas approaches, the order-processing system experiences increased load.

The problem with isolated silos of server systems is that it is nearly impossible or very time-consuming to divert idle resources assigned from one application (say BI) to meet the immediate needs of another application (say Order Processing). As a result, headroom capacity has to be separately allocated to each silo and the overall server infrastructure is highly underutilized. Various surveys put the average utilization of servers in a typical enterprise to often much less than 20 percent.

Today's businesses must be very agile and hence the peak demand is a constantly-moving target. It is not possible to precisely predict the peak demand and when it may change. This leads enterprises to want to purchase much more capacity than they could possibly need, which creates a very costly combination of overprovisioning and underutilization.

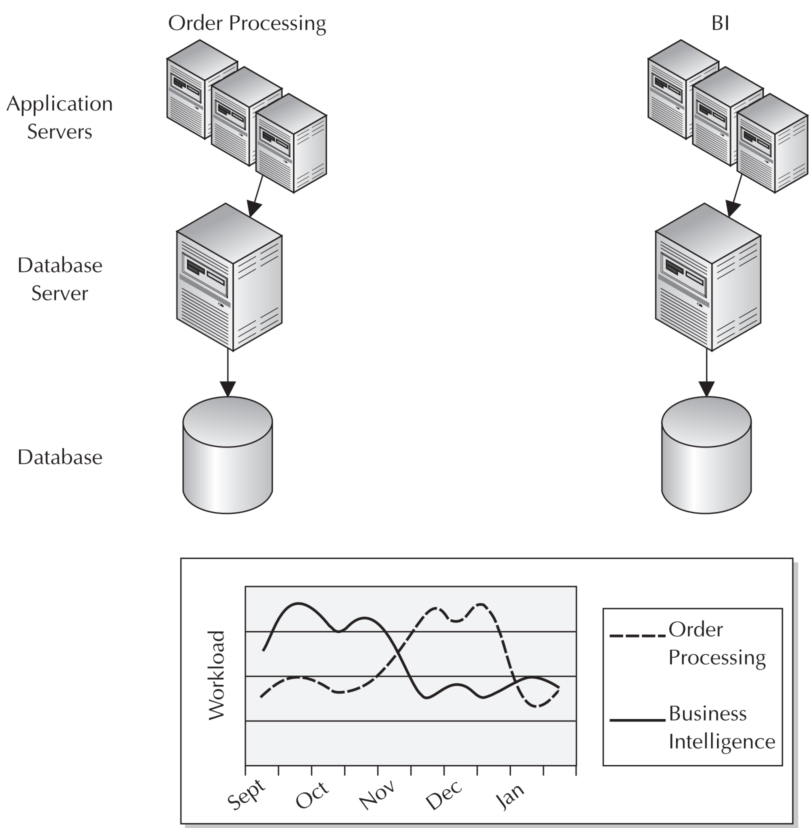

FIGURE 5-2. *Silos of server infrastructure—the underutilization problem*

Server Infrastructure Costs

Enterprise servers, especially those running databases, often use large symmetric multiprocessor (SMP) servers that provide high-performance computing capacity. However, they cannot be easily extended. It is possible to add more processors to an 8-way box, but along with it more memory must be purchased; in some cases a faster bus may be needed and so on. Thus, upgrading a server can be an extremely complex endeavor. Servers in these configurations are also very expensive. If a new server is purchased, much of it may lie idle for a long time until the excess capacity is needed by the application. This means that it is not cheap to incrementally increase the computing capacity assigned to an application—it requires a large investment. Enterprises may instead force their users to stick to the overloaded server until there is enough demand to justify a new server. It is a constant battle among utilization, complexity, and cost.

Server hardware cannot operate by itself—it needs software to function. Enterprises must include hardware- and software-licensing costs in their IT budgets for each business application. Software is usually licensed according to the number of processors in the system on which it is deployed, not according to how many are actually utilized. Thus, combined with the overprovisioning and underutilization problem, software-licensing costs can be a severe drag on the IT budget.

Data Center Costs

Data centers form a significant portion of the IT budget. Data center real estate is at a premium and servers occupy a significant portion of this space. So it is important to be able to pack as much computing power into as little space as possible. Power and cooling requirements for servers also add to the data center costs. Secondly, the servers must be connected to storage and the network—the resulting complexity in cabling can make data center management extremely complex, which adds to the administrative costs. Underutilized servers multiply all these associated costs; indeed, statistics show that the management, power, cooling, and real-estate costs for unused capacity typically far outweigh the acquisition and support costs of the server hardware and operating systems. The problem is further exacerbated when each business unit owns and operates its own separate data center, which is not uncommon in large enterprises.

Management Challenges

Over the years, managing the IT hardware infrastructure has grown incredibly complex. Data centers are a maze of cables and server boxes. IT staff has to constantly keep this infrastructure running to meet the availability and performance requirements of each business application. If a new server is needed, administrators have to install requisite software, allocate storage, and connect the cables to the network and storage. Tasks have to be undertaken by members of different teams, thus making provisioning time-consuming, expensive, and risk-prone. For example, the server administrator installs the OS, perhaps a different security administrator sets up the firewall configuration, and the network admin assigns IP addresses, DNS entries, etc. As the number of servers increases, manually provisioning servers quickly becomes infeasible.

Software patching is required very frequently, especially due to security vulnerabilities like viruses and buffer overflows. In an architecture where each application has its own servers, enterprises invariably end up with diverse combinations of server hardware and software configurations. It is very hard to test all these configurations when performing hardware or software maintenance. This means that more often than not planned downtime does not go as planned and is followed by extended unplanned outages.

It is not surprising that the life of an administrator is a struggle against daily fires. There is no room for proactive management, well-planned upgrade schedules, or preventative maintenance activities. With already shrinking IT budgets, the IT staff is stretched very thin.

In the next section, we will discuss how applying the Enterprise Grid Model to the server infrastructure better equips enterprises to tackle these issues. We will also discuss how the currently-deployed infrastructure may be evolved to a server grid using a combination of technology and management best practices.

Enterprise Grid Computing Model for Server Infrastructure

Advances in server hardware and management technology are making it possible for enterprises to move away from the silo approach to the Enterprise Grid Computing model for server infrastructure. Figure 5-3 illustrates a rationalized server architecture that alleviates the problems we had discussed with the silo architecture. We call such an infrastructure a "server grid." In this picture, all like server resources in the enterprise are consolidated into a small number of shared pools. All servers are accessible and managed through a common management interface. Management processes across heterogeneous components follow a common operational model; for example, managing servers from different vendors or patching different OS configurations is done in a standardized fashion.

In this architecture, the server resources are virtualized from the perspective of an individual application, or in other words, the application does not have dedicated server hardware. In fact, the application does not

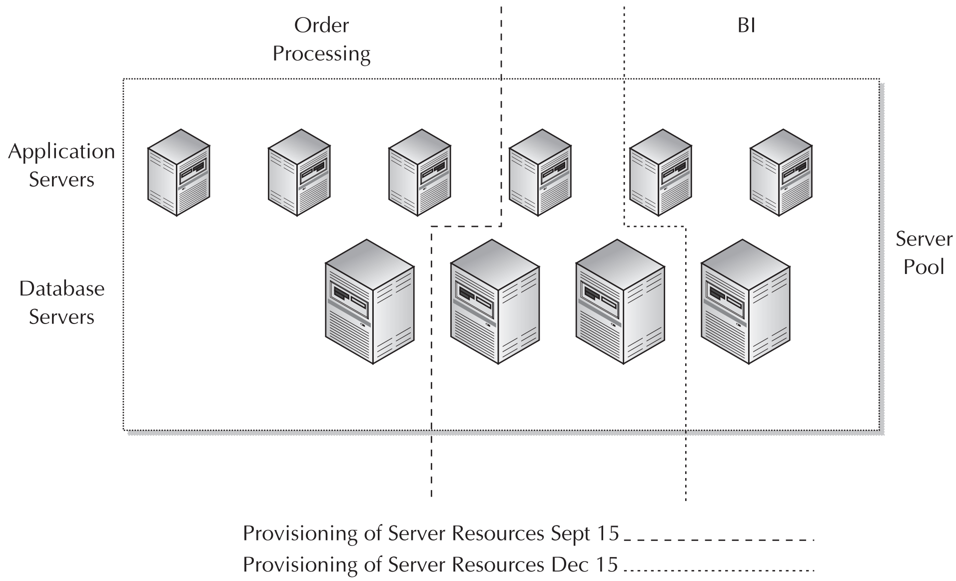

FIGURE 5-3. *Applying the Enterprise Grid Computing Model to server infrastructure*

care which physical server box it is connected to. Rather, the infrastructure ensures that the application is provisioned adequate server resources based on its need. Additional computing power is allocated to the application if needed, and when no longer required, the servers are taken away and redeployed elsewhere. For example, in Figure 5-3, more resources are given to BI in the months prior to Christmas and diverted to Order Processing as Christmas approaches. A key benefit of this architecture is that headroom capacity is shared among the client applications, and hence the total investment in headroom capacity can be lowered. As peaks of two applications do not generally coincide, each individual application using the server grid will tend to have more overall headroom for unexpected increased in load than it would have had in the silo model. You will rarely have a situation where some application has server resources lying idle while another application is suffering from lack of resources.

In a server grid, management functions are also handled in a holistic fashion within the context of the entire enterprise and not on behalf of an individual application silo or a business unit. As a result, management responsibilities and priorities throughout the enterprise can be shared and logically centralized. For example, routine maintenance activities such as applying operating system patches may be performed collectively by the same group of administrators across all systems, rather than on each individual system. This makes server maintenance a more predictable activity. Management processes can be standardized and eventually automated to minimize the need for human involvement in mundane and repetitive activities. For instance, provisioning of servers to an application may be automated using predefined policies. This enables a small group of administrators to manage a very large collection of servers. Administrators can now focus on proactive management and use their time to better handle exceptional or challenging situations.

Sharing of servers translates to sharing of the hardware and some software-licensing costs. As the server resources are better utilized, there will be an overall reduction in expenditure on hardware and software.

In the next few sections, we will discuss various ways—both technology and management practices—that can help an enterprise move towards this rationalized picture. We start with technology and deployment architectures for a server grid.

Advances in Server Hardware

Advancements in server hardware, networking technology, and infrastructure software such as databases, file systems, and operating systems are making it feasible to implement a server grid today. In this section, we will discuss several technologies, including low-cost modular servers, server clusters, and virtualization at various layers, all of which facilitate improved utilization of server infrastructure within an enterprise. We will also discuss how these technologies apply to Oracle environments, in particular to Oracle Database 10*g* and Oracle Application Server 10*g*.

Note that the mention of new technology does not mean that enterprises that wish to deploy a server grid should simply replace all their existing hardware with new products. However, as we will see, standardizing on these technologies over time could lead to significant reduction in infrastructure costs.

Low-cost Modular Servers

Intel has driven the mass production of CPUs. This has resulted in commoditization of servers. It is now possible to deploy en masse low-cost modular servers comprised of one to four CPUs. Popular form factors for these modular servers are rack-optimized servers and blade servers. Let's visit these modular servers in a little more detail.

Rack-optimized Servers

Rack-mounted or rack-optimized servers have been around for many years. Each rack-optimized server is a complete computer in itself. Data center racks and servers come in multiples of a standard size, called *1U* (a rack unit equals 1.5" in height). A typical data center rack is 42U to 70U, and a typical rack-mountable (so-called *pizza box*) server slides horizontally into the rack taking up 1U or 2U of space. The acquisition cost of a system built using Intel-based rack-optimized servers can be drastically lower than that of an equivalent SMP system.

Blade Servers

Blade servers, often simply called *blades*, are a recent innovation in extremely compact server hardware and provide a way for a single data center to hold a very large number of servers. While blades are not as cheap as rack-optimized servers in terms of initial acquisition costs, they provide a much higher density of servers and offer a practical alternative to expanding or building a data center when real estate is at a premium.

Almost every major server vendor including IBM, HP, Fujitsu, Dell, etc., has a blade server offering. While the detailed specification of blades differs by vendor, a typical blade server is a self-contained computer with its own processors and memory, plus interfaces for networking and storage, all on a single motherboard. Figure 5-4 shows the architectural difference between a traditional rack-mounted design and blades. Several blades are inserted vertically into a single chassis (also called a *blade-frame*) and all or most connections are via the connector that links the blade to the chassis. All the blades in a chassis typically share power, cooling and integrated networking, and sometimes integrated storage networking. The benefit of this architecture is that there are no cables for each server, thereby simplifying the cabling requirements and reducing the overall administrative costs.

Bare-metal Provisioning

Bare-metal provisioning involves installing the operating system and other associated software onto a server that is assumed to not have any operating system or other software previously installed. Many vendors including IBM, HP, Sun, Fujitsu, Oracle, etc., have started offering bare-metal provisioning technologies for installing especially Linux OS on modular servers. For example, Oracle Grid Control offers bare-metal provisioning for installing Linux OS and Oracle software. Most blade vendors also offer blade management software that supports bare-metal provisioning on their blade offerings.

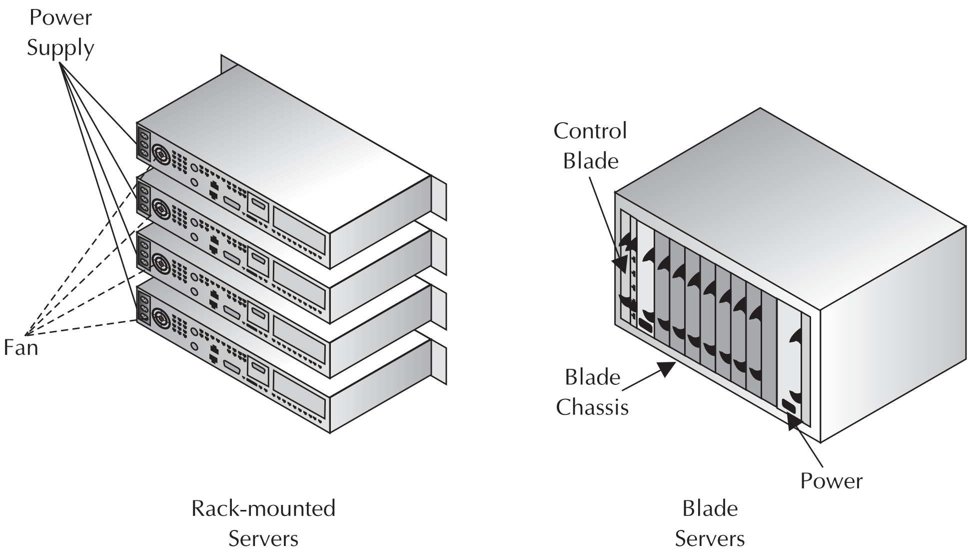

FIGURE 5-4. *Blade servers versus Rack-mounted servers*

Running Applications on Modular Servers

Since each modular server is an independent server, one rack or blade-frame can serve many enterprise applications. The same administrator can manage the server resources for several applications. Using automated bare-metal provisioning software, it is possible to dynamically and reliably re-assign a server from one application to another. Multitier enterprise applications typically have multiple components, one or more of which can be run on multiple servers. Some software such as application servers and databases like Oracle (with Real Application Clusters) can be run in a cluster configuration on multiple servers. We discuss server clustering in the next section.

Commodity Server Clusters

The deployment architecture for infrastructure software such as databases has traditionally consisted of a single large SMP server. When it is time to scale this architecture due to increased demands, two solutions present themselves. One approach is to replace the current large server (typically >= eight processors) with an even larger, more complex, and expensive server. An alternative approach is to "scale out" using a group of smaller and cheaper (one to four processors) servers—this ability for multiple servers to be used to perform the function of one larger server is known as *clustering*.

As we discussed in the previous section, over the past few years, server hardware is fast becoming a commodity. Enterprise class operating systems such as Linux and Solaris are now available for Intel/AMD-based hardware. Many vendors like Dell, HP, IBM, and Sun sell low-cost modular servers that come preconfigured with one of these operating systems. This has led to the emergence of clustering using low-cost modular servers as a viable alternative to large SMP systems to run infrastructure software. This is especially relevant in Oracle environments where both Oracle Application Server and Oracle Database are capable of being deployed on a server cluster configuration. In the following section, we will discuss this approach and its benefits in more detail.

Server Clustering in the Middle Tier

Clusters can generally be classified as either loosely coupled, where the multiple server nodes do not have much interaction with each other, or tightly coupled, where the multiple server nodes must work together and share data or state to handle the application workload.

A loosely coupled configuration is ideal for deployment in the middle tier to scale out web servers and application servers. In this architecture, often referred to as a *server farm*, each server runs an independent copy of the application software and a load-balancing mechanism is used to distribute workload across multiple servers. There are many benefits to this architecture over a single large SMP server. Depending on the application, this architecture by itself can improve performance through load balancing. It provides better reliability and availability. If one server goes down, the entire tier is not affected. The failed server can be replaced with another one, again while keeping the application online. Also, software upgrades such as OS patches can be applied a few servers at a time without bringing down the entire application.

The emergence of commodity modular servers makes this architecture cost-effective and extensible. This extensibility can help address the problem of underutilization of servers across an enterprise. In a cluster-based architecture, additional computing capacity can be deployed as and when needed by adding a new server to the cluster. Servers can also be re-allocated from one application to another. Thus, the headroom capacity can be shared by many applications and thereby overall utilization of the entire server infrastructure can be improved. Last but not least, the provisioning process can be automated.

Oracle 10*g* Application Server provides the necessary infrastructure to support application deployments on a farm of low-cost servers. We will discuss this in more detail in the section on dynamic resource management and server provisioning for application servers.

Database Clustering

Server clustering technology for the Oracle database has been available for many years. However, a major bottleneck used to be the "interconnect," which is the communications interface to coordinate the processing and share data among multiple nodes of a cluster. For clustering to be effective, a fast, reliable, and low-latency interconnect is required. Over the past decade, however, interconnect technology has improved in terms of latency and bandwidth, and many protocols have been incorporated into IEEE standards, leading to commoditization. Switches based on Gigabit Ethernet have been around for many years now. Certain vendors have even started offering 10 Gigabit Ethernet switches.

These high-speed interconnects combined with low-cost server blades make clustering a viable, cost-effective option to scale Oracle databases. All the benefits of clustering that we described in the preceding sections—improved performance, availability, and serviceability at a lower cost—also apply to the database tier. We discuss Real Application Clusters, Oracle's clustering technology for the database, in the next section.

Oracle Database Real Application Clusters Oracle Real Application Clusters (RAC) enables the Oracle database to run on a cluster of servers. Depending on the application needs, the cluster nodes may range from low-cost modular servers to large servers. RAC automatically load-balances database connections across the RAC nodes. Furthermore, a single database query may also be able to run in parallel while making use of multiple RAC nodes.

Typically, clusters require special software that is responsible for monitoring the health of the cluster and recovering from failures. Clusterware is platform specific and is usually provided by the cluster hardware vendor, operating system and server vendors, or platform utility vendors. Examples include SunCluster, Veritas Cluster Server and HP. Oracle provides its own portable framework, known as Oracle Clusterware, for all operating systems, which can eliminate the need to purchase, install, and configure third-party clusterware in many situations.

RAC provides support for dynamic provisioning, which we discuss in the section on server provisioning for Oracle databases.

To Cluster or Not?

Of course, there is no silver bullet here! The emergence of clusters does not mean that enterprises should simply replace all their large SMP machines. As noted earlier, a clustering architecture can indeed be incrementally and cheaply scaled up by simply adding a new server into the cluster. However, while it used to be true that to scale up a large SMP configuration would mean a large upfront investment in a bigger box, these days most vendors provide Capacity-On-Demand (discussed in detail in a later section), which allows enterprises to only pay for the portion of the large server that is in use.

An application running on a single server (i.e., a nonclustering architecture) is limited by the processing capacity of that server. You can combine many more processors in a cluster than you could in a single server, and so clusters can potentially provide much higher throughput for some applications. On the other hand, a large SMP configuration may outperform clusters of smaller ones in terms of latency for certain classes of applications or certain data sets.

The message to take home from this discussion is that advances in technology have created new cost-effective choices for scaling infrastructure software. Enterprises should determine the most cost-effective architecture to use by carefully evaluating the requirements of their particular application.

In the next section, we discuss the concepts and technology for server virtualization, which we believe is a key to evolving the server infrastructure into a server grid.

Virtualization

We noted earlier that servers in typical enterprises are deployed in monolithic configurations. Recall from Figure 5-1 that each server has several layers—hardware, an operating system, and possibly other software installed on it. Virtualization of a server means that the implementation of each layer is separated from the interfaces used to access it by the upper layer. This is typically done using an extra layer of hardware, software, or firmware.

The Value of Virtualization

Virtualization hides the implementation of one layer from another and as long as the interface between them remains the same, you can change the implementation of each layer without having to modify those that depend on it. This provides flexibility and the opportunity to implement additional functionality such as resource management or dynamic provisioning within the virtualization layer. For example, virtualization of an operating system using Solaris 10 Containers allows multiple applications to run on the same OS, isolated from each other in terms of resource usage. Virtualization of infrastructure software such as Oracle Database 10*g* RAC means that the application now no longer knows which database instance is being used to answer its queries. So, if the application load on the database increases, more servers can be provisioned to the database without the knowledge of the application that depends on it.

Virtualization adds flexibility to the server infrastructure and enables resources (at lower layers) to be shared (by the upper layers), thereby improving utilization. For example, virtualization of a physical server using hardware partitioning or virtual machine monitors like VMWare allows multiple operating systems to be installed on the same physical hardware.

The consolidation of similar resources enabled by virtualization can also simplify management. With virtualization, we divorce a resource at one layer from its consumer at the higher layer. This allows us to create a level of abstraction that can now be used as a common interface for management. A simplified example is that if all applications are run on virtual machines, then we can manage provisioning of server resources solely in terms of virtual machines and their configuration parameters without having to separately deal with underlying components like hardware, OS, etc. This enables centralized management of the entire server infrastructure, which further enables better tracking and possibly automation of various management functions. We will discuss centralized management and automation in a later section.

Virtualization Overview

Figure 5-5 provides an overview of virtualization mechanisms at various layers within a server. The following sections will discuss each of these technologies and their benefits in the context of an enterprise grid.

Hardware Virtualization

Virtualization of server hardware hides the exact physical hardware from the operating system running on it. There are two technologies that provide hardware virtualization—hardware partitioning and virtual machine monitors (VMMs). These technologies allow single server hardware to be efficiently shared among multiple applications, thus utilizing these hardware resources more efficiently.

Hardware Partitioning

Hardware partitioning technology partitions the bus or the backplane of an SMP, resulting in sets of components that behave as separate computers that are more or less electrically isolated from one another. Each hardware partition can behave like a stand-alone server with its own CPUs, operating system, memory, network connectivity, etc. If one of the partitions crashes,

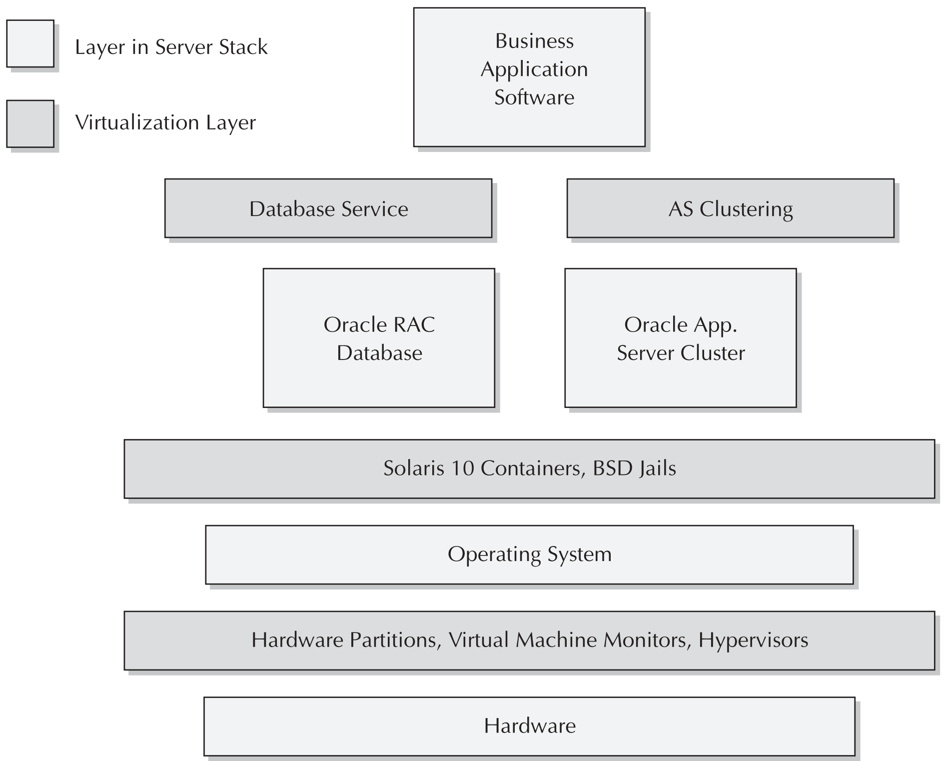

FIGURE 5-5. *Server virtualization technologies*

it has no impact on applications running on another partition. This type of hardware partitioning requires the support of the hardware and the operating system. This technique can help to improve the utilization of the server by enabling multiple business applications to run on one SMP, completely isolated from one another.

Hardware partitioning has been available on IBM mainframes for a long time, and more recently is available on Unix platforms, e.g., Sun's Dynamic System Domains for Sun Fire servers. The limitation of hardware partitioning is that the number of partitions that can be created is closely tied to the size of the hardware; e.g., it is not possible to create hardware partitions on a uniprocessor.

Capacity-On-Demand Hardware partitioning has resulted in the emergence of a new "pay-per-use" SMP pricing model, known as Capacity-On-Demand (COD). As discussed before, enterprises often overprovision servers to account for unexpected changes in the load. With SMP architectures, due to the large acquisition cost, this unused capacity within enterprises can be a huge drain on enterprises' IT budget. With COD, an enterprise pays only for the number of processors of the SMP that are in use at a particular time. The server is initially partitioned to use only a fixed number of CPUs, for example, 16 CPUs out of the available 32. The enterprise pays only for the 16 CPUs. If more CPU resources are needed over time, they could be "unlocked" and paid for at that time. Thus, the acquisition cost of the SMP can be amortized over a longer period of time. COD provides enterprises with an alternative to clustering to run Oracle databases, and also to run other business applications where clustering is not an option. IBM's Capacity Upgrade on Demand (CUoD) and Sun's COD are some examples of this type of a pricing model.

Soft Partitioning

An alternative technique to hardware partitioning is soft partitioning, which creates a virtualization layer using software or firmware, abstracting the physical hardware from the operating system. The virtualization layer is usually implemented by software called a virtual machine monitor (VMM), which allows users to create multiple virtual machines, each of which can be used to run a different operating system. A popular example of a VMM is VMWare's ESX Server, which allows any x86-based server to be divided into many virtual machines, each running on up to two processors. Another is IBM's LPARs, which implement a VMM for mainframes and its mid-range Unix server using a combination of firmware and software.

Why Use VMMs? One benefit of running operating systems within virtual machines is that the server hardware is isolated from the OS. A misbehaving OS (such as a Windows Blue Screen of Death) will potentially only bring down the virtual machine and not the physical server, while other OS instances running in separate virtual machines proceed unhindered. Note, however, that if the VMM itself fails, it will bring down all virtual machines and the OS instances running within them.

Some VMMs provide the capability to migrate an application running on one physical server to another, without interruption to the application. An example of this technology is VMWare's VMotion. This provides a way to perform a server upgrade or maintenance while keeping critical applications online. It also simplifies maintenance of applications that require different patch levels.

Physical server resources such as memory can be apportioned differently across multiple virtual machines depending on the application needs.

Para-virtualization One downside of traditional software VMMs, such as VMWare, is that they have some CPU overhead. It is not practical to have too many virtual machines running simultaneously on a single physical machine serving compute-intensive workloads. An alternative technique called *para-virtualization* can eliminate some of this overhead. This technique creates a thin layer that allows running and switching between multiple operating systems; however, much of the heavy-duty work (such as most device accesses, memory management, etc.) is executed directly by the host OS rather than going through the VMM. One disadvantage is that para-virtualization today typically requires the OS to be modified and hence recompiled. XenSource Inc. provides an open source para-virtualizing software known as the Xen hypervisor. Processor vendors such as Intel and AMD and software vendors such as RedHat, Novell, Sun, and HP have endorsed the Xen technology.

VMMs and Para-virtualization Going Forward There is a lot of interest and activity in the virtualization space, and the technology is constantly evolving. Processor vendors such as Intel and AMD are now embedding hooks for virtual machine monitors directly into their processors, for example, Intel's Vanderpool and AMD's Pacifica offerings. Going forward, this will enable virtualization software to run without modifying the OS and with much less CPU overhead. Both Xen and VMWare are expected to provide offerings that take advantage of these processor hooks in the very near future. VMWare now also provides a free entry-level version of its ESX server.

Suffice it to say that software-based virtualization is still an emerging area and time will tell which specific approach or product will dominate. However, it appears that the technique will play an important role in enterprise server grids in future. A passing note: at the time of this writing, Oracle supports running its software including Oracle Database, Application Server, and Grid Control on virtual machine monitors, notably VMWare for running test and development systems. However, Oracle does not yet support VMWare for running its software on production systems—the reader is advised to check Oracle's web site to see if their desired configuration is supported.

Virtual SMP

Another technology worth noting is one provided by Virtual Iron Inc. that allows a single copy of an OS to be run across a cluster of smaller low-cost x86 systems, making it appear like a single 4-way or 16-way SMP computer system. For applications, which cannot run on a cluster and require SMP architecture, Virtual Iron's technology can dramatically reduce the cost compared to a single large SMP, albeit at a lower performance. This technology is complementary to the single-server partitioning techniques discussed in the previous sections.

Virtualization is central to the Enterprise Grid Model. In the previous section, we discussed server hardware virtualization, which implies disassociating the physical hardware from an application. Enterprises can use a combination of all these technologies to create a virtualized pool of server hardware. Continuing on the concept, the next section discusses virtualization of infrastructure software like OS, databases, and application servers, from the applications using them.

Virtualization of Infrastructure Software

Virtualization of infrastructure software decouples the service it provides to an application from the lower layers of the server. In this section, we will discuss virtualization of the OS, and virtualization of Oracle database and application servers.

Operating System Virtualization

Operating system virtualization isolates multiple applications running on a single OS instance, so that each runs in its own sandbox. Examples of OS virtualization technology include BSD Jails and Solaris 10 Containers that divide the resources of an operating instance into multiple independent partitions. Each partition provides a complete operating environment with the simplicity of a Unix "root" model. These technologies also provide dynamic resource management across these partitions. Solaris 10 Containers also ensure that the violations such as stack overflow and buffer overflow (which are often exploited by viruses and other forms of security intrusions) are contained and cannot cause widespread damage. While operating systems such as HP-UX and AIX do not provide virtualized environments as Solaris 10 Containers do, these do provide dynamic workload and resource balancing of operating system resources across multiple applications running on the server. These workload management (often abbreviated as WLM) capabilities can be used to dynamically allocate computing resources (CPU, memory, disk I/O) on the server to multiple Oracle databases running on the same server, while ensuring that critical applications are not impacted by less important jobs in the system during peak demands.

Database Virtualization Using Service Names

Oracle 10*g* introduced the concept of *service names* for identifying connections from a specific application to a database. Service names (also called *services*) enable virtualization of an Oracle Database server from the applications using it. Instead of connecting to a physical host or a network address, the application now simply connects to a service name. Services allow segregation of various workloads such as batch from OLTP, planned reporting from ad hoc reporting, etc., and allow database resources to be appropriately allocated for these workloads.

When a service is defined, you can specify the database instances that support that service, known as *preferred instances*. Oracle Net Services will direct the request to the appropriate physical location hosting the database service. Oracle offers runtime-connection load balancing of the service requests among the multiple instances that serve that service. If there are multiple instances for a service, application connections can get routed to the least-loaded instance for the service, based on usage statistics. Services can be defined and managed entirely from Grid Control.

Services improve the availability of the database to the application. When defining a service, you can specify instances that should be used for a service, in case the preferred instance is down. This means that if the server is offline for maintenance, the database service may now be run on an instance on a different server; however, the application connection information does not need to change. Also, applications using Oracle ODBC/JDBC, OCI, and ODP.NET clients can take advantage of Failover Notification (FaN) events to automatically failover to an available instance for the service, in case of unexpected outages. FaN also provides automatic reconnection of the failed connections.

With services it is possible to consolidate multiple databases into a single database while ensuring the various applications get the required database computing resources. Various applications that formerly connected to different databases now connect to the appropriate service. A benefit of such virtualized databases is that you can monitor performance data by service, via Oracle 10*g* Automatic Workload Repository, which means it is now possible to track the use of the common database resources by various applications. Furthermore, service names can be used with the Database Resource Manager, described in a later section, to set up policies for resource usage for various applications. So, in effect, you have all the control of separate databases without the maintenance and cost overhead of managing them separately.

Oracle 10*g* Application Server Virtualization Using AS Clusters

Oracle 10*g* Application Server has a notion of an Application Server Farm, which is basically a management entity that allows managing many AS instances as a unit. Each AS Farm is associated with a common infrastructure repository that stores configuration and other metadata about the AS components. With an AS Farm, you can deploy a collection of Oracle AS instances as an AS Cluster—a collection of Oracle 10g AS instances, all of which have identical configurations and application deployments. Figure 5-6 shows an example of an AS Farm with two AS Clusters, one configured to run the Oracle Web Servers and another cluster defined for Oracle Containers for Java (OC4J) instances. The AS Cluster can in turn be physically deployed on a group of low-cost modular servers.

All instances within an AS Cluster can be configured to behave as if they were one logical AS instance, thereby virtualizing the specific application server instance from an application. There is built-in load balancing that routes HTTP and OC4J requests across the instances within a cluster. For example, if there is an OC4J AS Cluster, then the J2EE applications deployed on this cluster may in fact physically run on many different AS instances. Oracle AS will transparently route the application calls to one of the instances based on the load and application state, and handle the management of the Enterprise Java Beans running within the instances. An AS Cluster also provides virtualization from the point of view of failover—if any of the instances within an AS Cluster failed, another instance would be able to handle the application workload. Just as with databases, it is beneficial for many applications to share the same application server cluster.

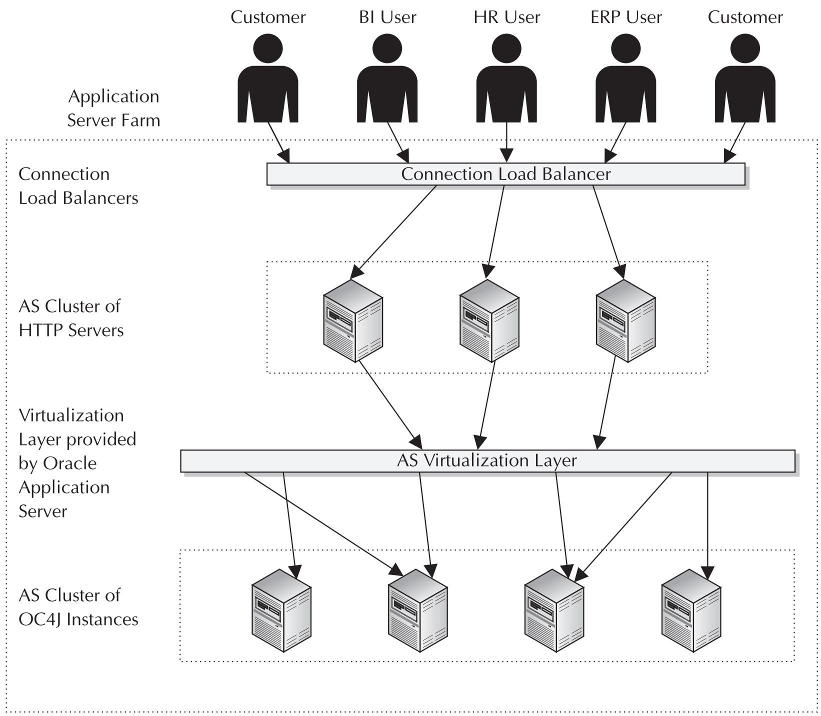

FIGURE 5-6. *Oracle Application Server virtualization*

We have seen how virtualization enables aggregation of resource at various layers in the server stack. Virtualization alleviates the utilization problem that plagues enterprises today. With a shared server infrastructure, resource allocation does not have to be static, but resources can be provisioned to applications as and when needed. The next section discusses dynamic server resource management and provisioning, particularly for Oracle software.

Dynamic Server Provisioning and Resource Management

Perhaps the largest consumers of server resources in an enterprise are databases that provide persistence and access to data, and application servers that run the business logic within enterprise applications. Recall that one of the characteristics of enterprise applications is that their load changes over time. In an architecture where each business unit has its own silo of resources, IT departments have an uphill battle to ensure that adequate resources are made available to the database and application servers so that business users are able to do their work. On the other hand, if the server infrastructure is virtualized and pooled, using the technologies discussed in the previous section, new mechanisms for automatic and dynamic resource provisioning now become possible. Oracle 10*g* includes some very powerful features that enable automated, policy-based provisioning of server resources for the database and the application server. The following sections will discuss these features.

Resource Provisioning for the Oracle Database

Resource provisioning for the Oracle Database can be thought of along two dimensions—one is to provision additional server resources to the database and the other is to allocate existing database server resources across multiple applications using the database, based on application needs or priorities. These two dimensions are equally important to address in a grid environment, and Oracle provides a broad set of tools for both purposes. Oracle 10g has three mechanisms for flexible resource management without interruption to the application—addition or removal of nodes in a RAC configuration, dynamic resizing of the database server, and the database resource manager itself. Let's discuss each of these in turn.

Dynamic Server Provisioning for Oracle Database RAC

Oracle Database RAC includes several features for dynamic provisioning of server resources without interruption to the applications accessing the database. Note that these features are generally unaware of the physical hardware—for example, a cluster can be made of physical servers (such as blades) or virtual ones (such as those created using hardware or software partitions).

It is possible to set up a single server cluster that runs many Oracle RAC databases. As the workload of these databases changes, it is possible to start up or shut down database instances to change the server resource allocation to these databases. As an example, consider a cluster with five nodes.

A through E that support two RAC databases, “ERP” and “CRM”. During normal times, RAC database “ERP” runs on nodes A through C and “CRM” runs on nodes D and E. As the load on database “CRM” increases, an additional “CRM” instance can be started on node C while the “ERP” instance on node C can be shut down, thus changing the server resource allocation among the two databases. Oracle Clusterware supports Oracle databases of multiple versions to be run on it. This further simplifies sharing of cluster among multiple databases as well as facilitates change management via rolling upgrades.

It is also possible to add a new node to the cluster running a RAC database. To do so, a new node needs to be added at the clusterware layer, and database software needs to be installed on the new node and the new node added to the RAC database. All of this can be performed through the Oracle Grid Control management tool. For the RAC database on Linux, even the OS can be installed on the newly added node. A convenient method for adding more nodes to a RAC cluster is to clone an existing node. In Chapter 8 we will discuss in more detail these software-provisioning aspects of Grid Control, in particular software cloning. Note that all of these steps can be done while keeping the applications on the database online.

Dynamic Resizing of the Database Server

One of the benefits of server virtualization is that the hardware or software partitions can be enlarged (or shrunk) dynamically. Oracle can take advantage of such changes in the underlying server resource allocation by dynamically resizing the database server. The database can dynamically increase its shared-memory size and add server processes to take advantage of the additional resources. On some operating systems like Solaris, this can be done automatically but on others it may need to be initiated manually. Note, however, that there is no interruption to the database service. This ability to dynamically change resource allocations clearly illustrates the power of server virtualization. The database server can also be downsized if the resources are no longer necessary.

The Database Resource Manager

Resource management tools provided by virtualization offerings and operating systems can be used only for coarse-grained resource management for the database. For example, they may be used to limit what portion of a large server is used by an Oracle database (say, to reduce licensing costs). However, these resource managers are not aware of the internal state and cannot control resource distribution across services within a database. In this case, the Oracle Resource Manager can be used to provide fine-grained control over the use of the computing resources of the underlying server by the various applications or application services using the database.

Database administrators can group user sessions with similar resource requirements into resource consumer groups. The administrator can set up limits on resources used by any group by defining a Resource Plan. Figure 5-7 shows an example of two resource plans that allocate the CPU resources of a single SMP server across many applications accessing the database.

Different resource plans can be set up to be applied at different times of day. For example, in Figure 5-7 a data warehousing–oriented plan is in effect at night, which allows a higher percentage of CPU for batch load

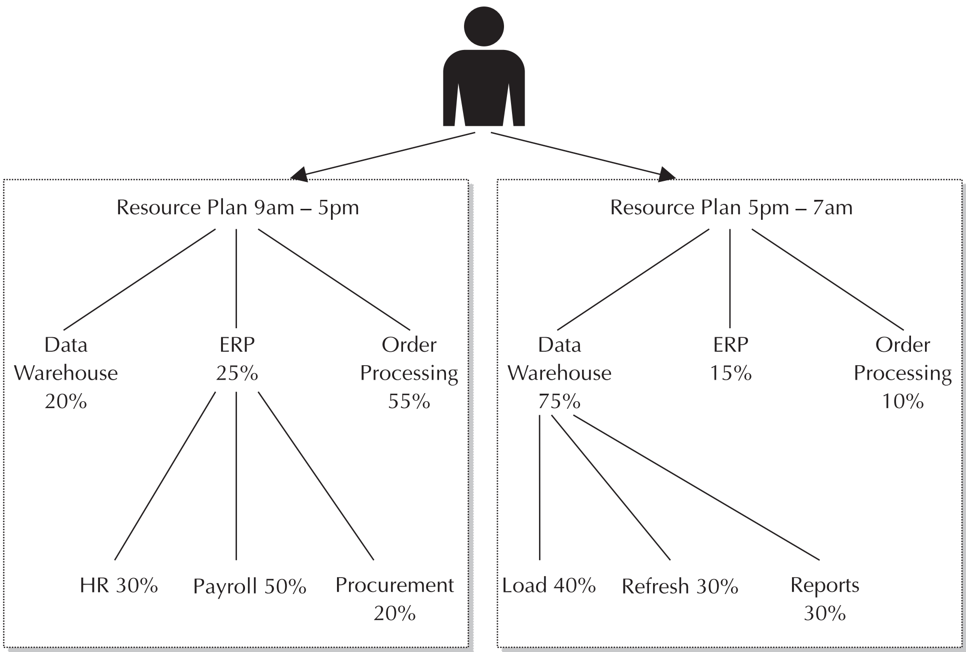

FIGURE 5-7. *DBA sets up resource plans and policies*

applications. During the day, a different plan is in place to give priority to order-processing systems, to ensure responsiveness to interactive or transactional users of the system. Resource plans can use various methods of resource allocation. For example, the allocation method may specify the percentage of CPU allocated to different resource consumer groups, or allocation may be done by limiting the number of active sessions allowed for each group or by limiting the degree of parallelism allowed to queries issued by a group.

Resource consumer groups and resource plans can be created and managed entirely from Oracle Grid Control. Furthermore, the administrator can define a mapping that is used to associate a user session with an appropriate resource consumer group, using session attributes such as the login name and the database service name. The resource consumer group can also be changed interactively; for example, if a session executes for too long, it can be switched over to a group with lower resource allowances.

Resource Provisioning for Oracle Application Server

Earlier we discussed how Oracle AS 10*g* can be deployed in an AS Cluster configuration. An AS Cluster supports dynamic server provisioning and resource management.

Dynamic Provisioning of AS Instances in an AS Cluster

Within an AS Cluster, if additional resources are required for the specific type of instance, a new instance can be added to the cluster (using the common configuration used within the cluster). For example, if the incoming connections to the enterprise web site spikes suddenly, a new Web Server instance can be added to the Web Server cluster. It is possible to start up and shut down an AS instance to change allocations of the AS server resources across different AS Clusters. To add a new server from scratch, all the software required to provision an actual server machine for the AS Cluster can be installed via Oracle Grid Control (including bare-metal provisioning for Linux OS), as will be discussed in detail in Chapter 7.

Oracle AS Clusters can also change their membership via dynamically discovering new instances. The Oracle Notification Service (ONS) provides communications between instances within an AS Cluster, such as addition of new AS instances, failures, metrics, etc. Using ONS, when new OC4J instances get added to a cluster, the routing logic automatically discovers the new instance and includes it in servicing future application requests. ONS can also be used to dynamically bind new applications to an OC4J instance.

Dynamic Monitoring Service (DMS) and Resource Management Directives (RMDs)

The DMS of Oracle AS provides support for monitoring the AS instances within a cluster. Administrators can set up Resource Management Directives (RMDs), which specify an action to be triggered when a certain condition is met or a threshold exceeded. The RMDs are integrated into the Policy Manager in Grid Control, discussed in more detail in Chapter 9. The actions are implemented using OPMN, which is the Process Management component of Grid Control. For example, an RMD may be this: if the CPU usage of an OC4J instance goes over 85 percent, a new server node should be provisioned and an OC4J instance started on it. The RMD can also include exceptions that allow dealing with situations when an action fails to complete.

We have seen the impact of new hardware technology and the value of virtualization and dynamic provisioning on the server infrastructure. In the next section we will discuss technologies and processes that can simplify the management of the server infrastructure.

Server Grid Management

Centralized management of the IT infrastructure is another core theme in the Enterprise Grid Computing model. Recall that virtualization at any layer makes it possible to model the various properties of that layer using a consistent interface, which in turn enables logically centralized management of that layer. As management is done at higher and higher levels of abstraction, fewer moving parts to administrators are exposed. Enterprises can do more with less staff due to economy of scale. Management processes become less ad hoc and more standardized and reliable. Repetitive processes, once standardized, can be automated.

So how can an enterprise get from its current silo model to this rationalized model for server management? Generally, this will involve a combination of people, process, and technology. *Process* involves pooling of server resources and standardizing operational procedures for server management, if possible using an integrated management tool. By *technology*, we mean making effective use of provisioning and automation features of modern server hardware and software. Finally, *people* refers to the administrators whose role needs to evolve to be more proactive than reactive. In the following sections, we will explore all three aspects in detail. We start with a discussion on consolidation or pooling of server resources.

Pooling of Server Resources

Effective centralized management of server infrastructure in the grid model first and foremost requires that the enterprise's server assets are no longer "owned" by business units or applications. Instead, they are managed as a collective, in terms of a small number of shared resource pools on behalf of the business units. Server assets within a pool would typically have similar capabilities or operational features such as performance or availability. For instance, there may be one pool of modular servers used for application servers and web servers, and one pool of SMP machines for low-latency applications. This logically consolidates spare capacity throughout the enterprise, thereby providing administrators with the flexibility to deploy resources to applications as and when needed. It enables deployment of enterprise-wide best practices for security, patching upgrades and other maintenance activities. It provides a holistic viewpoint for matching resource requests to available resources, based on priorities and costs.

Exploiting Server Hardware and Software Features

Many of the technologies discussed in this chapter are indeed amenable to centralized management. Most modular servers provide remote management, rebooting, and power-down capabilities. Not requiring the administrator to physically go to the server to perform some activity can be a huge time-saver and reduces risk of human error. Bare-metal provisioning simplifies provisioning servers with the appropriate software stack, without human intervention. Server hardware often includes features that enable maintenance activities to be done in a localized fashion; for example, server blades allow hot-swapping without bringing down other servers in the chassis. Some software, when deployed on a server farm or cluster configuration, allows rolling upgrades—while one or more servers in the cluster are being patched or upgraded, the entire application does not need to be shut down. Oracle 10*g* provides rolling upgrades for the Application Server suite and also rolling-patch apply capability for certain patches for the RAC database.

Effective use of the automation features of server technology can thus greatly simplify management. Another avenue towards simplifying management is the use of an integrated management tool, discussed next.

Using an Integrated Management Tool

Emerging server management standards, which we will discuss in a subsequent section, are making it increasingly possible to manage a wide variety of server hardware using a common tool. The management tool can provide value-add in terms of asset tracking, change management, server-usage tracking, capacity planning, etc. Some examples of server management frameworks are IBM Tivoli, HP Openview, and CA Unicenter. Oracle Grid Control also provides centralized management capabilities for the Oracle environment, including servers.

Managing Heterogeneous Servers

Ideally, enterprises should standardize their server hardware to a common hardware and OS platform. However, this is easier said than done. When the many server islands are consolidated, enterprises will typically end up with a disparate array of server assets. Also for business reasons, enterprises may not want to lock-in with a single vendor, or some software vendors may certify their software only for a specific server hardware or operating system stack. As a result, the management software can be crucial when it comes to dealing with heterogeneity, in terms of both different classes of servers (for example, SMP, blades, rack-mounted servers, and clusters) and also servers from different vendors.

Asset Tracking and Usage Reporting

As servers get provisioned flexibly from a common pool across the enterprise, server-asset tracking becomes important to appropriately charge business units for the amount of computing power used. Historical usage reports can be used for capacity planning to determine how much of the server resources are idle and if more are required. As the number of servers and software in

an enterprise grows, it is difficult to keep track of the software installed on each server, the configuration and patch levels, the security patch status, etc. Hence, configuration and change tracking is an essential piece of any server management software. Oracle Grid Control
can be used to keep track of the inventory of servers used to run Oracle software, along with their configurations, software versions, and patch levels.

Monitoring, Failure Detection, and Failure Handling

Effective server management requires monitoring of server operations and rapid detection of failures. Failures may be a software system failure or a hardware failure such as a network or storage component failure. The root cause of a problem may not be in the same subsystem as where the failure or the symptom was observed. An integrated management tool can provide visibility into server operations and pinpoint the exact cause of the failure. This is extremely important in a dynamic grid environment where the same component may be serving multiple applications.

Policy-based management can be used to set up both alerts for unexpected events as well as a procedure for automatic resolution. For instance, if there is an unexplained crash, the server could be set up for automated reboot. In the case of a network component failure, a failover to a redundant network path could be triggered. With Oracle Grid Control, administrators can set alerts to notify them in case of a performance problem in the Oracle database. It is also important that failure incidents are recorded and tracked so that steps can be taken to reduce the chance of them repeating. Management software that tracks failures and provides reports can help an IT manager determine how well the server grid is functioning.

In Chapters 8 and 9, we will discuss the features of Oracle Grid Control, which can serve as a centralized management tool for Oracle software. A pooled and centrally-managed server infrastructure can also utilize system administrator capabilities more effectively, as we will see in the next section.

Role of Administrators in a Server Grid

One of the problems faced by enterprises today is that administrators are so bogged down by the scale of day-to-day management activities that even preventable disasters cannot be handled effectively. With a shared server infrastructure, administrators can perform management at a higher level rather than deal with many individual components. Routine operations such as provisioning and patching can be automated so that only exceptions require human intervention. This changes an administrator's job to be more proactive than reactive.

A server grid allows administrators to really scale and take on the broader job of data center–wide system management. In place of managing each system individually, the skills of the administrator can be utilized to define best-practice system configurations across the entire data center. Administrators can define policies to ensure system security that can be deployed enterprise-wide. Administrators can define scripts for repetitive jobs such as backups, etc., once, and then deploy these across the various systems. Thus, with server grid, an administrator becomes more versatile and his or her skills are leveraged to play a bigger role in IT data center management.

We have seen how a combination of people, process, and current technology can be deployed to rationalize the enterprise server infrastructure into a small number of centrally managed pools. This leads us to our next section, where we show how management of the server infrastructure can now be viewed holistically in terms of management of computing services and their service levels throughout the enterprise.

Service-level Management

Consolidating and centralizing management should be seen as the first step to move an enterprise away from the problems of today's IT architectures. The ultimate goal is to enable easier and more adaptable management of computing resources in order to meet the business demands in the most cost-effective manner. In Chapter 3, we discussed the general notion of service-level management using SLAs and SLOs. In this section, we discuss service-level management from the point of view of the server infrastructure.

Understanding the Enterprise's Server Needs

Server infrastructure has to be aligned to address the needs of various business applications. The first step in this process is to define the SLAs and the corresponding SLOs for the enterprise applications that server infrastructure strives to meet. Some examples of SLOs for the server infrastructure include the following:

- **Throughput for the Application** How many application requests need to be served? It is also important to know how these requirements change over time.
- **Response Time for the Application** For example, a web site application may require that any customer query return data within two seconds.
- **Recovery Time Objective** This ensures that in case of a failure, operation would be restored within the specified time limit.
- **Total Downtime per Year** The business may specify a maximum acceptable planned downtime per year.

Provisioning Resources

Application requirements, defined in terms of SLOs, can guide the decisions regarding the specific configuration of server hardware, OS, and software required to meet the SLOs. Typically, this is a very complex and time-consuming process. However, the Enterprise Grid Computing model allows enterprises to standardize this process by creating a small number of standard configurations that can be utilized to deliver the required service. Furthermore, the inherent flexibility of grid architecture means that it is less critical to get the initial provisioning perfectly right the first time because it is possible to adjust it later without much difficulty.

Monitoring and Enforcing the SLA

The service-level delivery is not a one-time process and clearly does not end with initial provisioning. Key performance metrics relevant for the SLOs must be constantly measured and monitored to ensure they meet acceptable levels. Repeated failures to deliver SLOs may lead to violation of the SLA and impose penalties on the IT department. This is where the benefits of the Enterprise Grid Computing Model become apparent. The monitoring and reporting features of an integrated management framework can be used to proactively determine that a metric is approaching unacceptable levels. For instance, Oracle Grid Control, with its policy-based monitoring and alert infrastructure, can play a big role in monitoring service levels for Oracle database and application server software. Rather than relying on irate users to call the IT department, policies can be set up to change resource allocation automatically when load increases. For example, the DRM feature of Oracle AS can be used to automatically provision a new server running an additional OC4J instance with the spike in the workload. Once the load is back to normal, the server is reclaimed automatically. Again, shared resources from a spare server pool can provide capacity to handle such unexpected spikes.

Creating a Feedback Loop

Service-level management of the server infrastructure provides the added benefit of efficiently tracking server resource utilization. The actual data on service levels and their successful or failed delivery over a given year can be analyzed for capacity-planning analysis for the subsequent year. For example, if it turns out that despite available spare capacity, the SLO came close to being violated this year, it may be that the overall requirements of the application have increased and more servers need to be purchased. On the other hand, it may also indicate a configuration problem with a certain class of servers. As historical data about various incidents is recorded and analyzed, it may expose trends that indicate inefficiencies in the IT process, which can then be streamlined. This makes the planning process more efficient with use of real data.

Server Management Standards

We mentioned earlier than it is not practical or realistic to assume a homogeneous server environment throughout the enterprise. It is also unlikely that a single vendor will be able to supply a complete end-to-end management solution. Therefore, in order to achieve centralized management and better automation, it is necessary that products from multiple software and hardware vendors be managed using a consistent management interface. Multiple management tools must be interoperable to create an effective integrated management framework. Such interoperability is best achieved through the standardization of the interfaces among various components, in terms of both the data models implied in the interactions and the protocols and message/data formats. We will now discuss the standardization activities that are underway today in the context of server management.

In Chapter 2, we introduced standards bodies, Distributed Management Task Force (DMTF) and OASIS, which are organizations that work on standards for enterprise applications and management. In this section, we provide a brief overview of server management standards currently being developed by DMTF and OASIS.

CIM

The Common Information Model (CIM) is a mature standard that continues to be developed by DMTF. CIM allows exchange of management information in a platform-independent and technology-neutral manner. It enables end-to-end multivendor interoperability in server management systems, thereby simplifying their integration and reducing costs. We will discuss CIM in greater detail in Chapter 9.

ASF

Alert Standard Framework (ASF) provides a standard framework for the management of servers and networking components without the presence of an operating system. It provides remote-control and alerting interfaces for networked elements. This also helps in bare-metal provisioning of server resources.

SMASH

SMASH (Systems Management Architecture for Server Hardware) is a suite of specifications for architectural semantics, industry-standard protocols, and profiles to unify the management of the data center. The SMASH Command Line Protocol (CLP) specification enables simple and intuitive management of heterogeneous servers in the data center, independent of machine state, operating system state, server system topology, or access method. It facilitates local and remote management of server hardware in both Out-of-Service and Out-of-Band management environments. SMASH includes the SMASH Managed Element Addressing Specification, SMASH CLP-to-CIM Mapping Specification, SMASH CLP Discovery Specification, and SMASH Profiles, as well as a SMASH white paper.

DCML

Data center markup language (DCML) is a proposed OASIS standard for describing the data center environments, dependencies among various data center components, and construction and management of these environments. DCML provides a structured data model to describe data center environments. DCML can be used to describe, replicate, and automate various data center processes. The primary mission of DCML is to drive automation for data center environments in order to realize enterprise grid computing. [DCML-OASIS]

Future Directions

We have seen how server technology has been evolving to support the Grid Computing paradigm. We are far from complete automation of server infrastructure management. Tools available today and in the near future will begin to address many of the problems encountered by enterprises today. In this section, we provide a roadmap for things to come, which should help further solidify the Grid Model.

New Licensing Models

As the Grid Computing Model becomes more prevalent, the current licensing model, where license costs are effectively paid for on an estimated peak-load basis, needs to be revised. Newer licensing models, which allow more flexible use and deployment of resources, are needed. The ideal situation would be one where enterprises are billed for the software and hardware they use. We already saw an example of the pay-per-use model in our discussion on Capacity-On-Demand for SMPs. One of the key challenges for such licensing models is the lack of accurate metering technologies to measure and track usage.

Server Interoperability

There is a need to standardize the management technology for the server infrastructure. It should be possible to manage and provision all the servers from multiple vendors within a data center from a single central console.

This management technology needs to take into account various workload management and virtualization technologies on those servers for optimizing resource allocations for various enterprise needs. Bare-metal technology needs to be standardized to span various OS environments. With this, enterprises would no longer be locked into a proprietary technology or a single-vendor solution.

Multicore Processors

Processor technology has so far successfully followed Moore's law and has consistently increased processor performance. Low-cost modular dual-core servers from Intel and AMD are available for mass consumption today, and quad-core processors from Intel and AMD are on the horizon. Systems based on multicore processors essentially create lower-cost SMP systems, for example, SPARC T1–based systems have eight cores and four threads per core, so they behave like a 32-way SMP. In addition, these modular multicore systems can be used in clustered configurations, essentially providing even more computing capability at low cost. Multicore processor technologies with virtualization and clustering technologies provide enterprises with lots of choices to lower the costs of server hardware, while utilizing their server infrastructure more efficiently.

Novel software licensing models for systems using multicore processors are emerging. Most major vendors, including Oracle, have already established new rules to govern pricing of software running on these multicore processors.

Unified Networking for Servers and Storage

As enterprises consolidate their hardware resources into a single data center, the natural next step would be to unify the computing and storage networking fabric. The emergence of technologies like Infiniband and 10 Gigabit Ethernet is already starting to make this happen.

Summary

The enterprise architecture with islands of computing resources can no longer sustain the dynamic nature of today's businesses. With the Enterprise Grid Computing Model, enterprises can begin to address their problems by consolidating all their computing resources to a small number of shared pools. Standardized low-cost modular servers with clustering architectures provide a cost-effective alternative to expensive large SMP systems. Virtualization technologies allow better utilization of existing server infrastructure. Oracle Database and Application Server include many features that support the grid model via clustering, virtualization, and dynamic server provisioning. Oracle Grid Control provides end-to-end monitoring, maintenance, and provisioning of systems used to run Oracle software.

Continuing up the IT stack, in the next chapter we will look at the enterprise application layer. We will see how Service-oriented Architectures provide a paradigm for applying the principles of enterprise grid computing to enterprise applications in order to deliver flexible and dynamic business processes.

References

[EGC] Strong, Paul. Enterprise Grid Computing, ACM, Queue, Enterprise Distributed Computing, Vol. 3 No. 6. July/August 2005.

[N1-Grid] Carolan, Jason, et al. Building N1(TM) Grid Solutions: Preparing, Architecting, and Implementing Service-Centric Data Centers.

[IT Spending Gartner] Gomolski, B. Gartner 2004 IT Spending and Staffing Survey Results. October 2004.

[Predicts 2004] Bittman, T. Predicts 2004: Server Virtualization Evolves Rapidly. Gartner Research Note. November 2003. SPA-21-5502.

[The DCJ] The Data Center Journal http://www.datacenterjournal.com/

[Intel Virtualization] Uhlig, Rich, et al. Intel Virtualization Technology, IEEE Computer, Volume 38, Issue 5. May 2005. ftp://download.intel.com/technology/computing/vptech/vt-ieee-computer-final.pdf

[XEN] XenSource Enterprise-Grade Open Source Virtualization http://www.xensource.com/

[VMWare] Virtualization Overview
http://www.vmware.com/pdf/virtualization.pdf

[IDC 10*g* Grid] Kusznetsky, Dan; Olofson, Carl. Oracle 10*g*: Putting Grids to Work. IDC Paper. April 2004.
http://www.oracle.com/technology/tech/grid/collateral/idc_oracle10g.pdf

[Oracle DB Grid] Oracle Database 10*g*: Database for the Grid (an Oracle White Paper). January 2005.
http://www.oracle.com/technology/tech/grid/collateral/10gDBGrid.pdf

[Oracle App Server] Oracle Application Server 10*g* Release 2 and 3, New Features Overview (an Oracle White Paper). October 2005.
http://www.oracle.com/technology/products/ias/pdf/1012_nf_paper.pdf

[Oracle-EM] Provisioning and Patching Your Oracle Environment with Oracle Enterprise Manager, 10*g*
http://www.oracle.com/technology/products/oem/pdf/prov_patch.pdf

[DCML-OASIS] OASIS web page on DCML
http://www.oasis-open.org/committees/dcml-network/faq.php

[SMASH] DTMF Systems Management Architecture for Server Hardware (SMASH) initiative http://www.dmtf.org/standards/smash/

[ASF] DTMF Alert Standard Format http://www.dmtf.org/standards/asf/