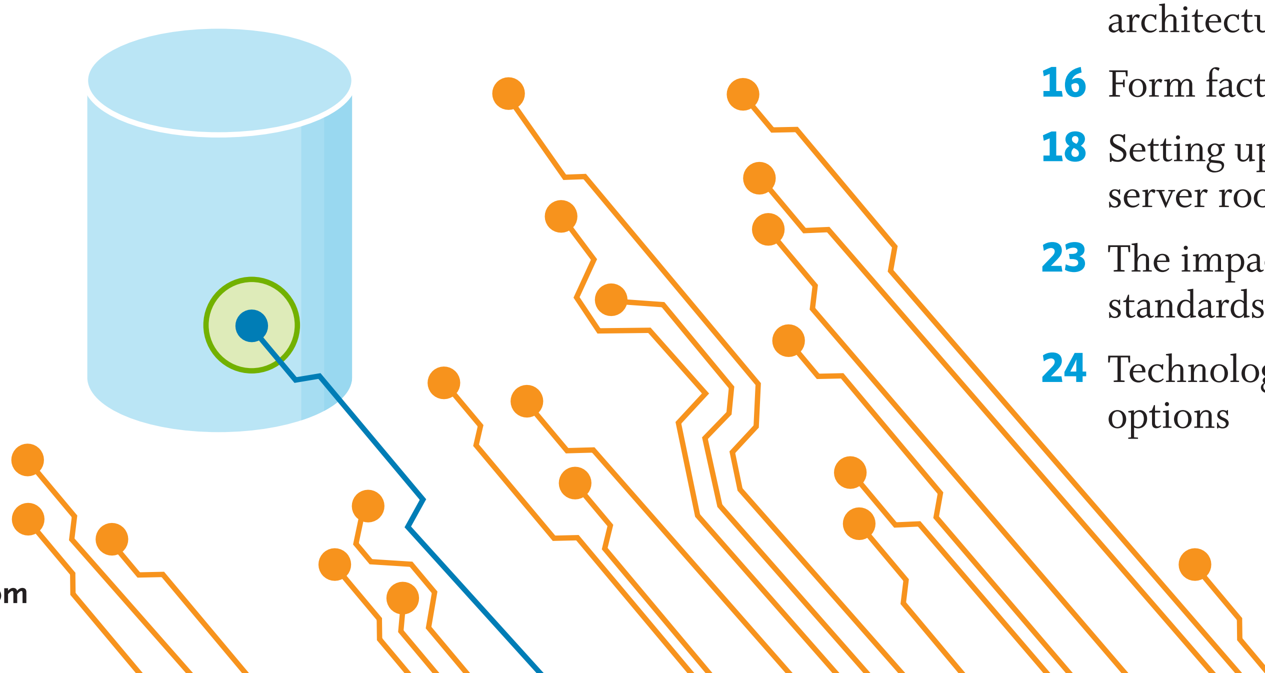
Choosing your server

Why selecting, installing and implementing your server is such a critical challenge



inside

SearchWinComputing.com

Introduction

SELECTING, INSTALLING AND implementing the servers in your organization is a critical challenge. Even after you've selected a server, careful planning for its installation and management is crucial. Indeed, the future of your organization may depend on it.

This chapter is an excerpt from the SearchWinComputing.com e-book, Servers and Storage. The goal of this chapter is to provide administrators and IT managers with information to guide their decision-making in the selection and installation of servers, systems and subsystems.

Servers are as diverse as the population of a small city. They vary in platform, shape, size and purpose.

This excerpt from the e-book examines server hardware, beginning with server architectures, and ways to build servers from these components. We'll focus on symmetrical multi-processor (SMP) systems, clustering and grid computing.

We'll next turn our attention to installation. We'll cover which server form factors best suit particular types of applications, and then we'll discuss what going into a successful server installation.

We'll also discuss scalability – the extent to which you can change the resources and performance of the system to match the growing needs of the company.

We'll look at some of the factors you'll want to consider in setting up your server room. We'll discuss why you want a dedicated server room, and what type of environmental factors you should consider.

Lastly we'll look at the technology lifecycle, and how that should enter into your purchasing decision for a server (or any other IT component).

ABOUT THE AUTHORS:

RENÉ J. CHEVANCE has written numerous papers and a book on server architectures, *Server Architectures – Multiprocessors, Clusters, Parallel Systems, Web Servers, Storage Solutions*. He has taught computer science for 20 years at the University of Paris. After 30 years with Groupe Bull, he left the company in 1999 to become an independent consultant in information system architecture and system integration, hardware and software technologies, system performance and high availability systems. He can be reached through his Web site www.chevance.com

PETE WILSON is chief scientist at Kiva Design, a small consulting firm and research company focusing on issues surrounding the move to multi-core computer platforms. Prior to Kiva, he spent seven years at Motorola/Freescale.

Choice of server depends on application

A SERVER IS SUCH a fundamental component of an information-processing system that its operational state directly affects a company's ability to run the business. Selecting, installing and implementing your server(s) are critical challenges.

In deciding which servers to invest in, first carefully consider your applications. Start by making a thorough list of the applications your company needs to run properly. Then talk to the likely users of those applications. The applications themselves can generally be put into two broad categories:

- **"SERVICES"** applications provide services such as e-mail server, Internet access, intranet facilities and (when needed) an extranet capability.
- **"DOMAIN-SPECIFIC"** applications handle business-oriented work. These applications will usually be free or commercial products that are externally acquired, such as SAP AG and Oracle applications, or else custom applications developed specifically for the company.

Because the operational success of the business depends on at least some of these applications being available 24/7, the next step is to grade the applications in terms of business criticality. Use this information to determine what you will need in terms of high-availability hardware.

Beyond the applications, you should also look at how you will update the server system, including the hardware, applications and the operating system.

Some updates can be done without interruption; others cannot. For example, replacing a hard disk should not require the server to be brought down. Identify the types of backups that you will need and how often they must be done.

Also, look at which upgrades or other maintenance will be needed for the connected workstations, such as operating system upgrades, new software releases and capacity upgrades.

Installation considerations

When you add a newly acquired server to an existing systems set-up, there are two key actions: installation and test. Yes, the manufacturer will have conducted basic tests of the hardware and software, but now you must exhaustively test the system running your organization's applications.

The goal of this second round of testing is to ensure that the new system produces the same results as the current systems when given the same input. The set of applications selected for these tests must exercise all aspects of the system—computation, file system, LAN and WAN—and should involve the use of either real or simulated workstations.

However, if your new system is intended to replace rather than supplement an existing system, you face additional work. How much work depends on whether the new system is completely compatible with the old one.

If the systems are compatible, the test process should pay special attention to storage. Here you have two options: Reuse the old storage on the new system (after a complete backup), or create a complete backup of the data on the current system and restore it to the new system.

Plan to keep the old system in good operational condition for a while after the migration of applications to the new server. This will allow you to resume operations on the older machine should the new system run into problems.

In cases where the new server is incompatible with the older machine, the same test and transfer process must be followed, but you must take exceptional care with your data.

For instance, there are two types of data representation (referring to the order of bytes): Little Endian and Big Endian. Little Endian represents the low-order byte and high-order byte of a number stored in memory at the lowest and highest address respectively; Big Endian represents the high-order byte of the number stored at the lowest address. Systems based on the Intel x86 architecture use Little Endian, while many RISC-based systems use Big Endian. Some programs are sensitive to data representation, making this important to test.

The initial testing of the new server also provides an excellent opportunity to check that the operating procedures, especially backup and restore, are in good shape.

Scalability considerations

WHEN CHOOSING A server you must determine what degree of scalability you need. In this context, scalability means the extent to which you can change the resources and performance of the system to match the growing needs of the company without having to resort to a complete system replacement. Such adaptability can take place across several factors, such as data handling capacity (processing power), main memory size and LAN or WAN network connections.

Server scale-up and scale-out are not mutually exclusive.

As we shall see when we look at server architectures more closely, data handling capacity can be scaled in two ways:

- **SCALE-UP**, or vertical growth, by adding processors to an SMP system
- **SCALE-OUT**, or horizontal growth, by adding extra individual servers (or nodes) in a cluster or grid system

Naturally, scale-up and scale-out are not mutually exclusive. In a cluster built from SMP nodes, you can increase the horsepower available to a node and still increase the number of nodes. You can also move from a single SMP system to a cluster by adding further SMP systems and the appropriate interconnect.

Understanding basic server architecture

In the computing world, *architecture* is a fuzzy term used in several contexts. It generally refers to structural principles. For example, the *instruction set architecture* of a computer defines how the computer interprets memory contents when executing them as a program. Any given system may have its architecture described at many different levels, including the Industry Standard Architecture (ISA), I/O, virtual memory and interconnects.

> Server architecture should include the ability to configure servers by connecting various mixes of subsystems.

A SERVER EXISTS to provide services to its clients. Workloads vary and an architecture designed to efficiently support one class of applications could be very different from that needed by another class.

For economic reasons, a manufacturer wants to minimize the number of system components it needs to create, maintain and support. The manufacturer builds many different configurations from a reasonably small number of components or subsystems, allowing itself the freedom to update one subsystem class without having to update all of them. A key capability for any useful server architecture is an appropriate degree of *composability* – the ability to structure or configure servers by simply connecting various mixes of subsystems.

For this to be possible, a server must be constructed from reasonably independent subsystems, which are then connected using a relatively small number of interconnects. Interconnects used to couple very high-performance subsystems (such as the processor to its cache or multiple processors to each other) have many more constraints and demands than those needed to hook up slower devices, such as disks.

Generally, the intimate high-performance interconnects are proprietary to the vendor and tuned for a specific purpose, while those further out toward the edge of the system are more likely to adhere to standards. Key interconnects and the principles behind them will be discussed next.

Given an appropriate collection of interconnects and subsystems, it becomes possible to build systems of many different shapes and sizes, varying the mix of processors, memory, storage and connectivity. Elsewhere we will discuss the major system arrangements, covering SMP systems, clusters and grids.

Buses used to couple high-performance subsystems have more constraints than those needed to hook up slower devices, such as disks.

Interconnects

Often informally referred to as buses, interconnects hold the job of moving data (or program instructions) from one place to another. Interconnects have three important characteristics:

BANDWIDTH: The amount of data they can move per second.

LATENCY: How long it takes them to deliver data once you request it.

CONNECTIVITY: Whether they connect a pair of subsystems or many subsystems.

In meeting the needs of each interconnect class, you must balance the factors of cost, physical size, power and bandwidth. As subsystems are implemented as single chunks of silicon, VLSI (Very Large Scale Integration, a process for creating integrated circuits) dictates the inter-chip interconnect traits.

VLSI and interconnects

A KEY BENEFIT of the physics behind scaling silicon technology to ever-smaller dimensions is that with the reduced size you generally obtain higher performance while lowering power demands. Smaller transistors take less energy to switch state.

However, problems occur when you need to connect these ever-shrinking transistors to actual wires in order to transfer data between chips, which are some distance apart. The energy needed to drive signals out of a chip package, across a board and into a distant chip does not scale with transistor technology. Therefore, larger transistors must be used for the inter-chip connections than are used for internal logic.

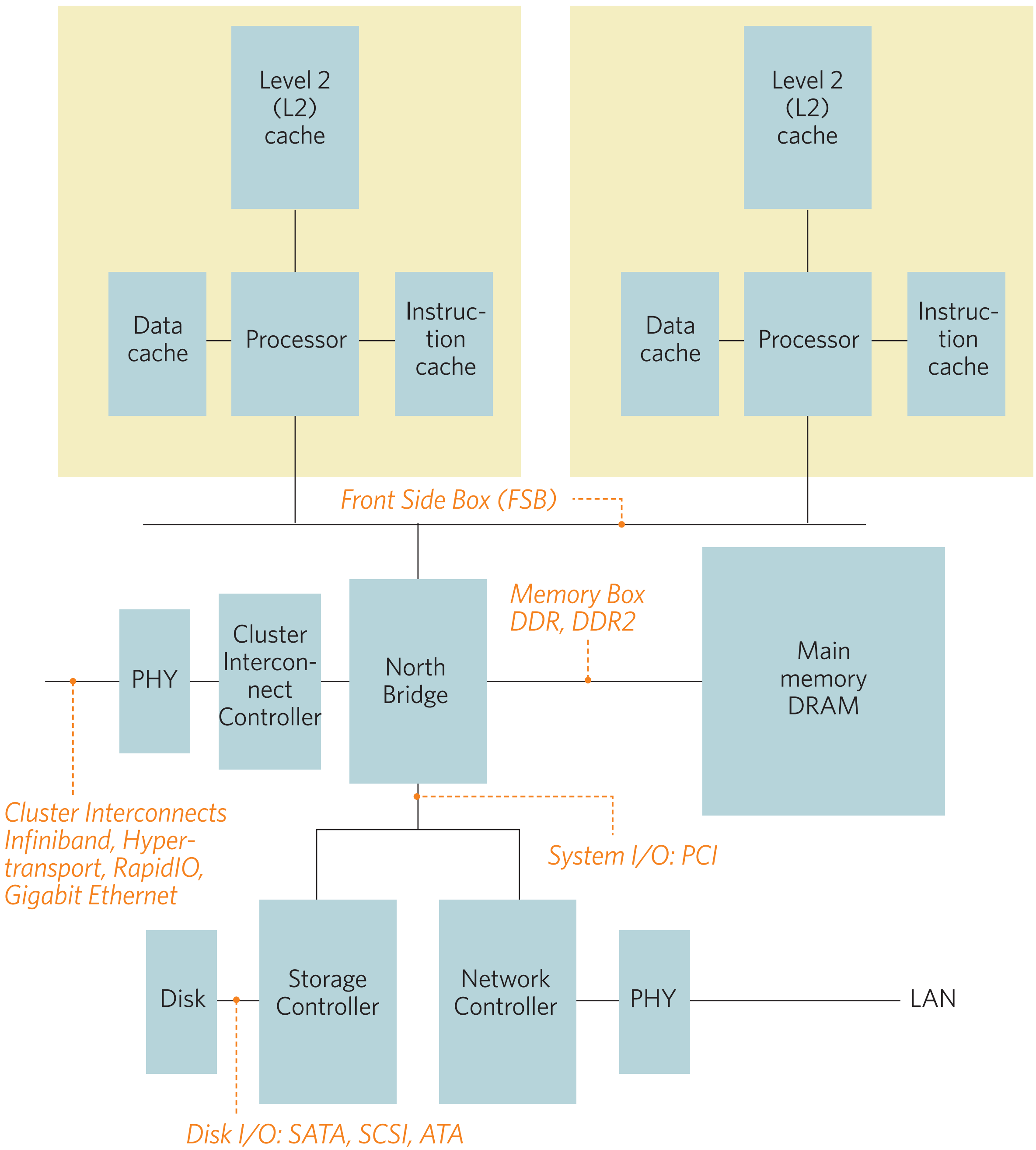
The system of the future is going to look more and more like a network.

Furthermore, the transistors needed to handle the logical activities in the interface represent a silicon area that shrinks with every generation of chip. However, the driver devices do not shrink, and thus occupy an increasing percentage of physical space and power.

Another challenge is keeping signals on many wires in sync with each other. When you are signaling at very high frequency, minor differences in wire length can produce significant differences in time of arrival. Connections that hook up multiple chips (for example, classical buses) require even bigger transistors.

The effect of all this is that systems are moving away from traditional buses and moving to interconnects that are unidirectional and point-to-point. What this means is that the system of the future is going to look more and more like a network, with familiar network components such as switches embedded inside the computer.

Modern server system interconnects



Modern server system interconnects are depicted in the diagram to the left. In a modern server, buyers can choose from a variety of interconnects as follows:

BACK-SIDE AND FRONT-SIDE BUSES: The front-side bus connects one processor to other processors and often a high-performance North Bridge component, which connects the processors to the rest of the system.

The back-side bus connects Level Two or Level Three cache to the processor. In some machines, the back-side bus is shared among a number of processor cores or else there is one bus per core. These buses are specific to the processor design.

Most processors integrate two levels of cache; some designs also integrate a sub-stantial Level Three cache or provide on-chip controllers to manage external SRAM (static RAM) as an L3.

MEMORY INTERFACE: Standard interfaces slowly change as new generations of dynamic RAM (DRAM) are introduced. Some systems may favor RAMBus variants, while most will choose a more mainstream technology, such as" Double Data Rate" DRAMs (DDR, DDR2, etc.). In the diagram on p. 9, the memory is shown connected to the North Bridge, which integrates the memory controller. However, newer designs are beginning to take advantage of VLSI to integrate the memory controller into the processor. This can substantially reduce latency to the memory, thereby increasing performance.

SYSTEM I/O: Most servers deploy some variant of PCI, generally a 64-bit-wide parallel version like PCI-X. However, we are now seeing a serial form such as PCI Express being used increasingly.

STORAGE: SERIAL ATA (SATA) or its eSATA variant is now often being used to replace the once preferred SCSI and commodity ATA interfaces. Similarly, the Fibre Channel (FC) interconnect is being replaced by SATA, although that interconnect does not have the same reach as FC, which still has its place in large configurations.

A storage controller will be connected to the system using the system I/O bus (usually PCI). However, some storage vendors offer storage area networks (SANs), a form of specialized LAN tuned for storage, and network-attached storage (NAS) to connect the system to a storage subsystem using Gigabit Ethernet.

LAN: An Ethernet variant, wired using Gigabit Ethernet

CLUSTER INTERCONNECTS: When systems are connected to a cluster, a specialized interconnect is typically employed, although Gigabit Ethernet can be used. When it comes to connecting computer systems in separate cabinets, Infiniband is slowly making inroads. To connect boards within a single cabinet, versions of Hypertransport and RapidIO can be used, though the latter is rarely seen outside networking equipment.

Interconnects that have become practically extinct include VME (generally used in industrial systems), MCA and EISA (used in PCs), NuBus (used in early modular Apple Macintosh computers) and the VESA Local Bus.

Types of server architectures

IT MAKES ECONOMIC sense for manufacturers to offer composable systems, which can be configured to meet a broad spectrum of needs simply by an IT person plugging together the right subsystems. A key consideration when selecting a server is the amount of computing or processing power needed. Because there are strict limits on computer performance available from a single processor, an Important factor in composability is the number of processors deployed in a system.

Ways in which multiple processors can be deployed within a system include:

- **SYMMETRIC MULTIPROCESSING (SMP):** This method arranges processors so each sees all system memory and all I/O, allowing programs to run on any processor (or processor core) and access all system resources.

- **CLUSTERING:** This technique builds the system by connecting separate, independent computer subsystems, each having its own processor, memory, storage and I/O.

- **GRID COMPUTING:** This clustering variant employs the clustering approach using real, separate computers as its building blocks, and a LAN or WAN as the system.

Each of these three approaches has its own strengths and weaknesses; each is appropriate for different classes of applications.

Note: Not all servers are Intel x86-based. As you will see on p. 18, processor architectures like Sun SPARC, the IBM POWER processor and Intel Itanium own a minority share of the market. Processor architectures vary, but the underlying physics remains the same for all machines. Processor architecture should enter into your server decision but should not be the deciding factor per se. A better approach is to consider the applications needed to run the business and choose the appropriate system based on factors such as price, performance and lifecycle cost.

SMP (Symmetric Multi-Processing)

In an SMP system there are multiple processors, each with the same access to system resources (the 'System interconnects' diagram shows an SMP system). Since modern systems require caches to deliver meaningful performance, this means that an SMP must be *cache-coherent*.

Cache-coherence ensures that whenever a processor accesses data, it sees the same value as any other processor would, regardless of whether the data is cached in one processor or maintained in memory. The mechanisms used to ensure this uniformity rely on a notification system that informs "interested caches" when a cached data value in several caches has been changed. SMPs use a shared bus, making it simple to maintain cache coherence since every processor sees every bus transaction.

A shared bus does not scale well. Large-scale systems must use some other form of interconnect.

However, a shared bus does not scale well. Large-scale systems must use some other form of interconnect, and so they generally appear as a collection of independent SMP systems (each with its own processors, cache and portion of main memory) connected through a cache-coherent interconnect.

In such a system, the time to access data increases when a processor in one SMP system needs to access data in the memory connected directly to another SMP system, rather than in local memory. Such a system is considered a cache-coherent, non-uniform memory access (CCNUMA) system, and the ratio between the access time to local memory and the access time to remote memory is called the non-uniform memory access (NUMA) factor.

If the software is not aware of the nature of the system when it comes to non-uniform memory access, and the NUMA factor goes above four, performance can be severely impacted.

If software is not aware of a system's NUMA nature, and the factor goes above four, bad things can happen to performance. CCNUMA systems require software variants, particularly in the memory management system. That software

The key advantage of using MP is that it reflects a view of how computers work.

determines where to allocate memory requests by scheduling processes so as to minimize overhead.

As an example, suppose a process running on processor A is suspended and resumes on a distant processor. The cost of moving data across the system and accessing data from a worse place can easily negate the performance gains from having more processors.

The key advantage of SMP is that it reflects a view of how computers work, one that is promulgated by most mainstream operating systems and assumed by mainstream sequential programming languages. To oversimplify, it promotes a programming model in which cooperating processes share memory. This allows low-cost interprocess communication; large chunks of data can be moved by passing pointers rather than copying the data itself.

But SMP's strength is also its weakness. SMP systems are fragile. An error in the cache of one processor, if not detected and fixed immediately, can pollute data for the whole system, causing an irrecoverable error.

A small SMP offers a simple means of getting greater server performance than a uniprocessor, and so, 2- to 4-processor systems are remarkably cost effective. Furthermore, generic mainstream software written with some awareness of multithreading should run faster on a small SMP.

Very large SMP systems comprised of 32 to 128 processors are probably best suited to scientific computing in which very large data structures are manipulated by parallelizable code. Commercial applications, including transaction processing, Web serving, Google databases and decision support with few updates, have workloads that emphasize multiple independent transactions or interactions and have less need for efficient cache coherence. Transaction processing in particular is not well served by a fragile system.

SMP systems are offered by all server vendors, including Dell, IBM, Sun and Hewlett-Packard, and even some of the PC makers, including Apple.

Clusters

A CLUSTER IS a collection of computer systems interconnected in a way so that a program running on one machine can access the resources of another machine. . .but only indirectly. For instance, a processor is unable to perform a load or store to remote memory. Instead, it must ask software running on a remote processor to perform the access and forward the data.

This sounds remarkably inefficient, but such an isolationist approach provides a major benefit: robustness. In a cluster, if a memory system or cache failure occurs on one node, it is quite difficult for the polluted data to instantly diffuse through the rest of the system.

The apparent inefficiencies matter more in some applications than others. Even for scientific computing, when large shared data areas are manipulated, it is possible for appropriate workloads to allocate the resources in such a way that the interprocessor communications costs are a small part of the overall computational burden.

However, a small cluster rarely makes sense, simply because the programming model of choice (for most software) is one that best fits an SMP view of the world. Writing software that works on a cluster requires not writing it as you would for an SMP. Shared data is basically not available; explicit message-passing is required. This means there's not a great deal of mainstream software for cluster architectures -- just those applications that scale well on clusters, such as databases and transaction-processing systems.

In the real world, there is no battle between SMP and cluster. Effective clusters are constructed by clustering small- to medium-scale SMPs. You should choose a cluster solution when your application requires robust scaling of data processing capabilities, memory size or data storage. However, a cluster can only be deployed when the application is already available for a cluster architecture.

As with SMPs, all the server vendors offer clusters: IBM, Sun, Hewlett-Packard, Dell, the other PC players and Apple.

> **Writing** *software that works on a cluster requires you to not write it as you would for SMP.*

Grid computing

Grid computing is a fashionable term referring to the ability, through software, to leverage many independent and possibly heterogeneous computer systems, which happen to be connected through a network. Its name suggests the concept of an electricity grid, a vanilla means of delivering enough electricity anywhere.

Grid computing is simply the result of being able to distribute software tasks to a large population of network-connected processors. Real-world grid computing leverages software that virtualizes available resources, with the goal of presenting a view of having a large enough, rich enough computing resource.

To the extent that virtualization works for tasks to be executed on a grid, the grid offers some advantages. A cluster has similar properties (and its nodes may be virtualized in the same manner), but the nodes of a cluster generally must be close together. While it's certainly possible to have I/O connections across a WAN-scale distance (perhaps using iSCSI), there are no such scalable, low-latency, WAN-friendly processor interconnects. A grid has no such restriction. Distributed computing systems relax the need for physical proximity that historically requires strong homogeneity. Therefore, grid computing allows organizations to deploy heterogeneous distributed systems.

Grid computing can be an excellent choice for workloads that naturally break into convenient parcels of computation on well-defined, relatively small amounts of data. In such cases, data and programs can be distributed to a remote computer, the task run and the results collected. With proper program sizing, execution time, data sizing and bandwidth, you can obtain fairly high efficiencies.

Candidates for grid computing include anonymous computation processes, such as the search for extra-terrestrial intelligence through digital signal processing analysis of radio signals, as well as simulation and what-if analyses for financial and aerospace domains. Virtualization also allows the potentially vast amounts of storage in a grid system to be geographically distributed and replicated transparently for increased robustness.

Grid computing is based on specialized software rather than hardware systems designed specifically for grid computing. Any machine can be used. All the major server vendors all offer grid support.

Server packaging

Servers come in a number of physical form factors or packaging, including:

TOWER: The simplest servers, such as high-end PCs, are provided in tower cases and share similar restrictions in their extensibility.

RACK-MOUNT: Larger servers often take rack-mount form. This is a standard, 19-inch wide enclosure that is some multiple of 1.75 inches tall and designed to be mounted into a larger enclosure termed a rack, which allows a site to create an appropriate mix of server configurations. A rack generally integrates some number of services such as power, storage and network connections.

BLADE: As the technologies used to fabricate components in building server computers has evolved, those components have become more integrated and substantially smaller. This allowed a server to be built on a single board so a rack-mount-sized enclosure could hold many such boards, each with its own processing, memory, network and (minimal) storage. This led to a new form factor for servers: blades. The developing trend is that blades will encroach and take over rack-mount form factors.

Here is a brief summary of these physical form factors and where to use them:

- **The TOWER form factor is used for low-capacity servers where scalability beyond two towers is not needed.**
- **The RACK-MOUNT form factor is used for servers of substantial capability and multiple nodes, with each node itself being highly capable and configurable.**
- **The BLADE form factor is best suited for servers containing large numbers of nodes of limited capability, each of which typically builds on PC technology with its low cost and small size.**

The following table further describes and compares these three packaging options.

	TOWER	RACK	BLADE
WHAT IT IS	A collection of server components (processors, memory, I/O controllers and sometimes peripherals) integrated in a compact floor-standing unit.	A rack 19 inches wide, into which either servers or storage can be inserted in any mix. The installable units come in heights that are multiples of 1.75 inches (called 1U). Maximum height for a rack is 42U.	An enclosure equipped to allow a fairly large number of PCI-format cards to be installed within. Each card is itself a computer system providing processors, memory, network connections and storage connections. A typical configuration is up to four processors, memory, and one or two hard disks for local storage.
WHAT TO USE IT FOR	Use a tower system when low cost is a priority, fairly minimal	Use a rackmount system when each node has to be of fairly large capacity (in terms of number of processors, memory size, storage capacity, etc.), when the overall server configuration should encompass an average to a large number of nodes, and when scalability is important. Physical footprint is moderate.	Use a blade system when each node has to be of reasonably small capacity (in terms of number of processors, memory size, storage capacity, etc.), when the overall server configuration should encompass an average to a large number of nodes, and when scalability is important. Physical footprint is fairly small.

THE SERVER ROOM: Location, space and setup

We will now consider the physical aspects of server installation. These include the reasons for having a dedicated server room; what a server room should contain; environmental considerations; planning a server room; an installation checklist; and what makes an installation efficient.

Even a small server installation can benefit from a dedicated room. Small servers can impose an intolerable noise and thermal burden in an ordinary office environment, while serious security concerns may arise from an unprotected server that's easily accessible by passersby.

A large server configuration will quickly exceed both the tolerable noise level in an office environment and the regulatory limits allowed for noise (limits vary by country). The substantial thermal burden imposed by such a server configuration can be easily handled by a dedicated room with specialized air conditioning. A dedicated room also makes access control straightforward. By requiring electronic badges, you can restrict access and keep an accurate log of who has entered and who has left.

A dedicated server room generally will contain not only the server itself, but also disks, backup devices, interfacing cables, spare disks, blades, peripheral cards, fans and other equipment. Although server administration can be performed remotely, it makes sense to have a local console in the server room, so administrators can perform maintenance and administration on-site.

Environmental considerations

THE SERVER ROOM must be large enough not just to hold the server equipment; it should also allow you to rearrange equipment as needs change, including adjustments in air conditioning and electrical power. The room should also be able to accommodate replacement equipment such as extra logic cards, fans, disks and unused backup tape cartridges.

Underestimating these requirements could force you to move a server installation to a new room -- a serious disruption in service. That move also generates direct costs of acquiring, constructing and provisioning the new server room, as well as indirect costs that arise when inadequate computer or storage restrict the company's ability to work efficiently and respond effectively to competitors.

Note: Backup storage devices should not be stored in the server room. A fire could damage both the hardware and the backups, rendering the whole backup exercise useless. Backups should be stored offsite in fire-resistant spaces.

Make sure that the power supporting the server room is sufficient not just for today's usage but for server growth down the road. It is penny-wise and pound-foolish to scrimp on properly outfitting the server room. Allow for sufficient lighting and multiple electrical outlets; doing this right makes maintenance and enhancements simpler.

It may be necessary to equip the server room with dedicated air conditioning to keep equipment from overheating.

It is penny-wise and pound-foolish to scrimp when it comes to outfitting the server room in terms of lighting and electricity.

In addition, server equipment is sensitive to air quality. You should ensure building cleanliness, which may involve installing air filters.

Server room planning considerations

When a server is intended to support geographically-dispersed users, you will face the question of where to put the system. Consider the following points:

- **SPACE:** You need enough room to house the equipment and allow for future additions of hardware.
- **NOISE:** Equipment may be too loud to place near a work environment.
- **MECHANICAL VIBRATION:** Some candidate sites, such as those in or near large industrial factories, can suffer vibrations beyond what the system is designed to handle. If you believe this is a risk, measure vibration amplitudes and spectra, and plan accordingly.
- **ENVIRONMENTAL RISKS:** A server room should not be sited in a place prone to natural disasters, such as floods, earthquakes, volcanic eruptions or lightning. Such phenomena can interrupt the supply of electrical power to the site. In addition to identifying natural hazards, pay attention to the risks of riots, strikes and even terrorist attacks.
- **LOCAL SUPPORT AVAILABILITY:** You should be able to get support easily from both the equipment suppliers and internal resources.
- **COMMUNICATIONS AVAILABILITY:** You must be able to expand your network capabilities to ensure the right levels of performance (bandwidth, latency and uptime) and reasonable costs (connection costs, communications tariff structure).
- **QUALITY ELECTRICAL SUPPLY:** Consider the power supply's failure rate, the frequency of micro-outages, and the frequency of over-voltage or under-voltage.
- **SPARE EQUIPMENT:** You have to decide if storage of spare equipment should be onsite or nearby, so it is readily available in case of managed failures (i.e., data replication and backup servers).

Naturally, real-world decisions require some compromise, so it is good practice to revisit the data and your decisions on a fairly regular basis. The environment may undergo changes that call into question earlier decisions, resulting in a need to analyze the potential benefits, risks and costs of moving to a different site.

First steps in server room planning

Use this checklist when choosing and equipping a server room:

1. List factors to consider when choosing a site for the server room.
2. Choose the site.
3. Define the equipment to be installed (including power, air conditioning and air filtration) and how you will secure access to the room, such as electronic badges with a logging system.
4. Determine how you will secure access to the room, such as electronic badges with a logging system.
5. Create an implementation plan, approve budgets, choose suppliers and sign contractual agreements.
6. Ensure that work is completed on schedule and to appropriate standards.
7. Create an installation plan for the server room equipment. Have equipment delivered and installed.
8. Qualify the room and equipment before starting operation.
9. Set up a formal review process to periodically check on-site operation.

We cannot overstate the importance of this review process. It is rare indeed for any server installation — particularly if it's the first for a company — to proceed without problems. The review process allows you to analyze the system on a daily basis and evaluate proposed improvements and fixes.

Don't let the phrase "formal review" strike fear into your heart. This is not to propose a complex bureaucratic process with a specialized audit team, but a simple process tuned to the size and capabilities of the system so you can measure and review certain operational characteristics. You can even provide a suggestion book, either an actual book or electronic mailbox, where users and operations staff can record their experiences (and, of course, make suggestions). This provides an excellent source of feedback and can lead to useful improvements in service and operational practices.

In the end, it is possible that the audit resulting from the formal review process may lead you to question your original decisions. The question then becomes whether to fix the problems or move to a new site.

Once the first server is installed in a company, many users suddenly realize that they will need new services and capabilities.

Planning for growth

There are two cases to consider in thinking about how you will handle future system growth: You are working on installing your first system or you are enhancing an existing one.

If you are managing a first-time installation, there is no historical data available to guide estimates of future needs. Furthermore, it is often the case that installing the first server in a company or department leads to sudden (or not-so-sudden) realizations on the part of the users that they will need new services and capabilities.

That recognition in turn leads to unforeseen demands for increased server capability in data handling (computer) capacity and data storage. Under these circumstances, it's not at all unusual to see demand for processing capability and storage double in the first 18 months.

When the new server complements or replaces existing systems, the situation is simpler because you are working with a history of requirement needs, configuration enhancements and enhancement effectiveness. This makes it fairly straightforward to estimate the path of future requirements, at least for one to two years out.

In planning for the future, don't depend purely on technological advances. While it is certainly true that silicon and hard-disk technologies improve substantially year after year, it would be unwise to rely on these advances to provide for future needs. Better to look upon these technological leaps as providing some combination of budget relief and modest performance and capacity headroom.

Technology options

HARDWARE TECHNOLOGY ADVANCES in the IT universe have grown at an exponential rate over the past few decades (although in the silicon domain, this rate is slowing noticeably). This has a substantial effect on standards.

There are two types of standards: de jure and de facto. **DE JURE STANDARDS**, agreed upon and published by appropriate international standards' bodies, may be the rule in some domains. But data processing is dominated by **DE FACTO STANDARDS**, which are driven by market realities.

De facto standards are shaped by the introduction of successful products that met a market need and became accepted by buyers. Factors involved in such acceptance include cost, as well as the influence of the products' creators.

An example of the dynamics of real-world standards is the near-universal adoption of TCP/IP, despite a set of proposals created by industry experts and promulgated by the International Standards Organization (ISO). Of course, some international standards simply cast into de jure form some widely accepted de facto standard. This makes the standard none the worse.

The presence of both de facto and de jure standards makes decisions all the more difficult, leaving it up to managers to choose among options offered by competing companies, none of which have a dominant position in the marketplace.

Well-established standards offer stability, but emerging standards represent a business and technology risk. To help contain this risk, companies should consider establishing a regular market and technology review. Of course, doing this internally requires company resources. Doing it indirectly by relying on external analysts can be cheaper, but it presents another risk because such analysts can be influenced by fads and hype.

The phenomenon of technology evolution and marketplace hype has been captured in a semi-formal manner by Gartner, whose analysis we summarize here.

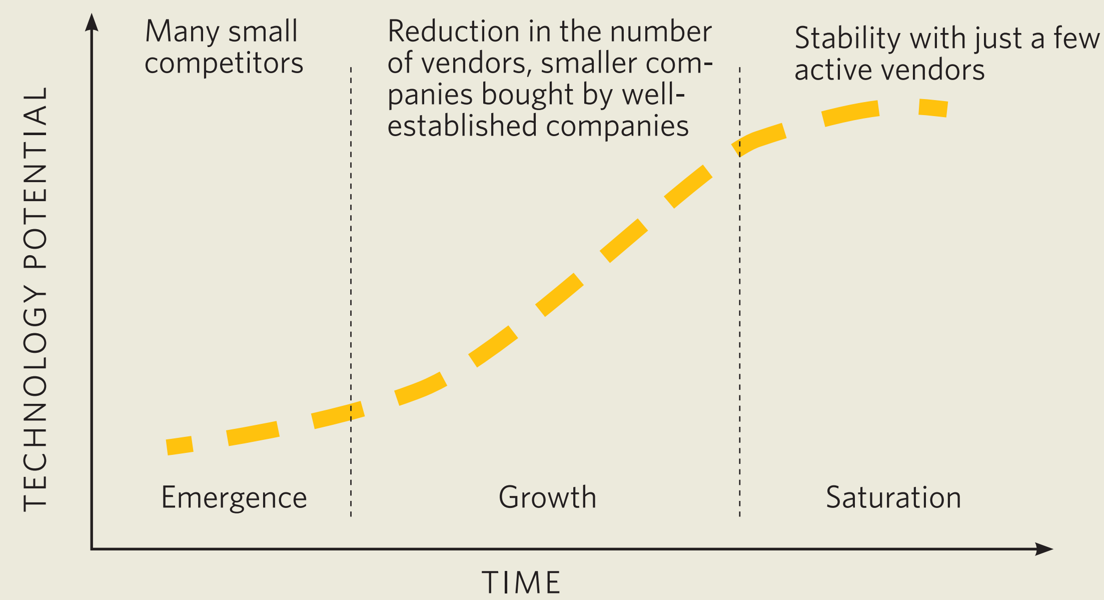
Technologies have a lifecycle. The first phase is *emergence*, which is characterized by the

> Well-established standards offer stability. Emerging standards represent both a business and a technology risk.

existence of many players, a number of which are typically startups. If there is some market acceptance, the technology can attain a degree of maturity, resulting in a concentration of companies offering the technology. Disappearing companies may go out of business or be absorbed by more successful players, defining the *growth* phase.

In the final phase, *saturation*, the technology has become mainstream, technological advances have slowed dramatically and the marketplace is dominated by a small number of companies that compete on economies of scale rather than proprietary technological edge. This evolution is illustrated in the above diagram.

Technology Lifecycle



Understanding what phase a technology is in can help you decide which technologies to embrace and when.

As an IT manager or administrator, you have to make decisions about which technologies to embrace and when. To do so, you need to understand which phase the technology is in. During the emergence phase, risks are high; it is generally best to avoid selecting such technologies at all. During the growth phase, there are fewer vendor choices but each is of lower risk than during emergence. By using business knowledge to choose one supplier, you can gain, at reasonable risk, advantages in areas such as cost or performance.

But there is another dimension to new technologies: how the technology is *perceived* rather than what it is *capable* of.

Gartner captures the dynamics of technology perception with its *Hype Cycle.*

The moment a technology begins to show some potential, it becomes an area of interest to the marketplace, analysts and investors. The varying interests of these parties, coupled with the desire to make substantial money, drives a phased sequence of perceptions for a technology.

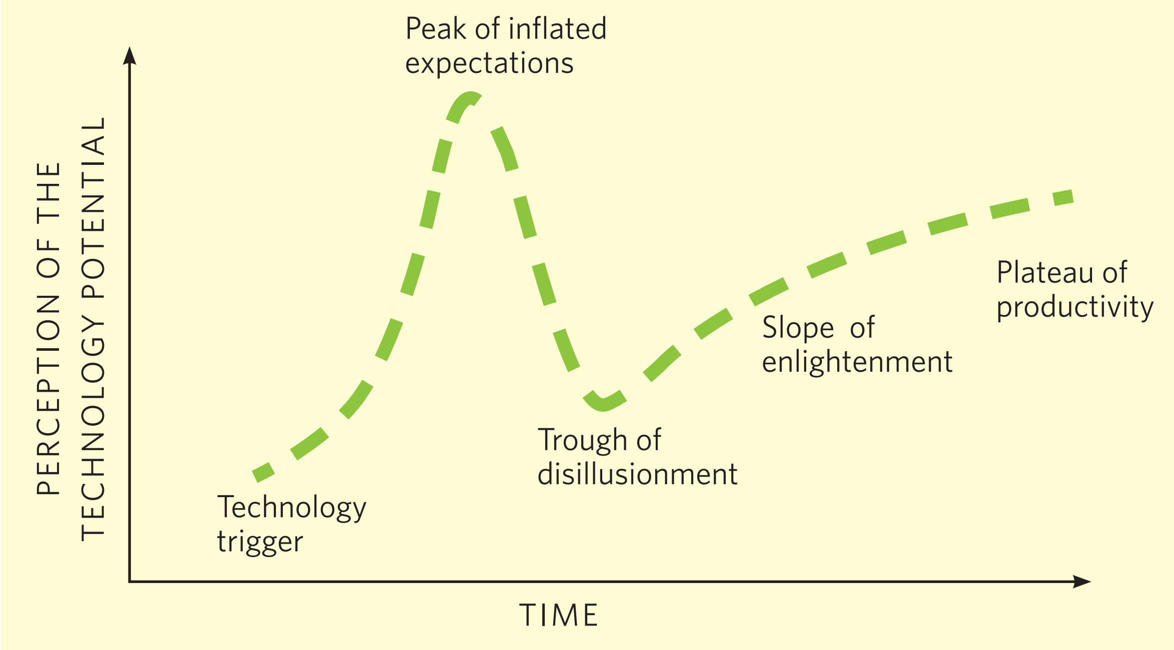
The first phase is the ability to demonstrate some technological capabilities; Gartner calls this the *Technology Trigger*. Demonstrations are often no more than laboratory experiments, with limits in how widely they can be applied and how long the machinery remains operational.

Observers tend to take the minimal results and extrapolate enthusiastically (and in public) to future *possibilities*, driving the perception of the technology into the second phase: *The Peak of Inflated Expectations*. This leads to widespread passion and enormous investments. As the nascent technology is used by early adopters enthused by the expectations, shortfalls are noted; these become bruited about, and the technology perception slumps into the *Trough of Disillusionment*.

If the technology has merit, it will persevere and slowly rise up through the traditional technology cycle with its perception progressing through the *Slope of Enlightenment* as it becomes accepted and finally reaches the *Plateau of Productivity,* where it is considered a commodity and accepted everywhere.

Looking at the technology lifecycle shows why it is so important for decision-makers to separate hype from reality.

Hype Cycle



Conclusions

In this chapter we discussed some of the basic aspects of computer architecture, looked at the major classes of servers (SMPs, clusters and grids) and then explored the issues involved in building a server room, choosing what general class of server to install and properly installing that server. Since the choice of a server (and system components) is so technology-driven, we examined the phases of a technology's evolution, saw how the perception of a technology itself has its own lifecycle, and explained how to leverage these lifecycle phases to determine what to buy and when.

FURTHER READING

There is surprisingly little on the practical aspects of computing in the literature on the subject. *Server Architectures – Multiprocessors, Clusters, Parallel Systems, Web Servers, Storage Solutions* by Rene J Chevance, Elsevier Digital Press 2005, provides a perspective on technologies, architecture and large systems missing in most academic works and the PC-oriented mainstream publications. In addition, material from this book has been summarized as a set of slides (in .pdf form) at http://www.chevance.com/slides.htm

For those who wish to delve deeper into computer architecture, we recommend *Computer Architecture: A Quantitative Approach* by John L Hennessy and David A Patterson, Morgan Kaufmann 2003.