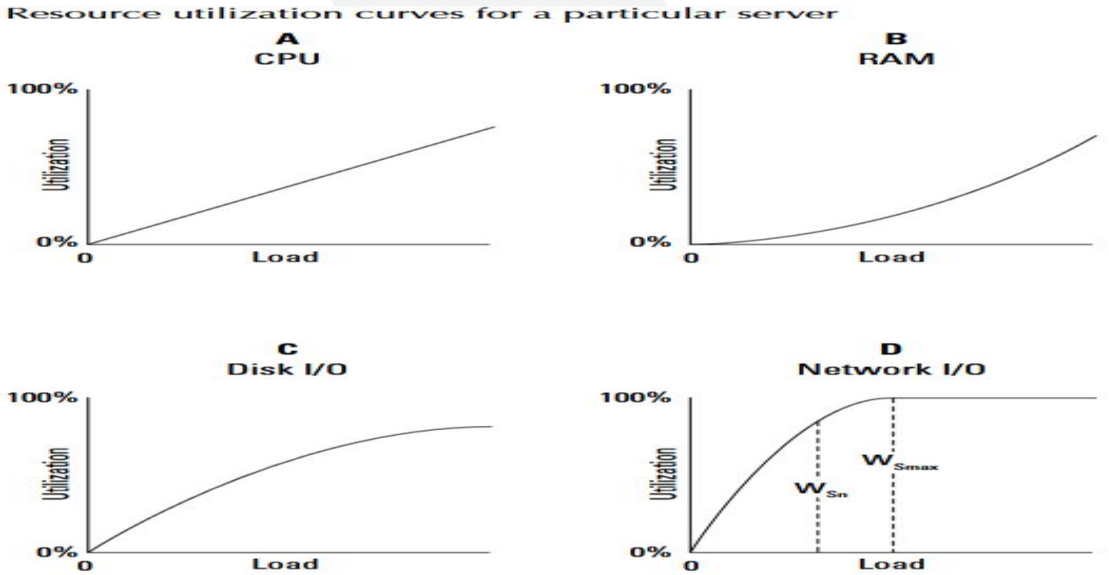
Resource Ceilings

- Whatever performance measurement tool you use, the goal is to create a set of resource utilization curves similar to the ones shown in Figure. The server at different load levels and measure utilization rates.
- The graphs in Figure 6.3 indicate that over a certain load for a particular server, the CPU (A), RAM (B), and Disk I/O (C) utilization rates rise but do not reach their resource ceiling.
- In this instance, the Network I/O (D) reaches its maximum 100-percent utilization at about 50 percent of the tested load, and this factor is the current system resource ceiling.
- Network I/O is often a bottleneck in Web servers, and this is why, architecturally, Web sites prefer to scale out using many low-powered servers instead of scaling up with fewer but more powerful servers.
- Adding more (multi-homing) or faster network connections can be another solution to improving Web servers' performance.
- Unless you alter the performance of the server profiled in Figure 6.3 to improve it, you are looking at a maximum value for that server's workload of Wsmax , as shown in the dashed line at the 50-percent load point in graph D.

Resource Ceilings

- The server is overloaded and the system begins to fail. Some amount of failure may be tolerable in the short term, provided that the system can recover and not too many Web hits are lost, but this is a situation that you really want to minimize.

- Consider WSn to be the server's red line, the point at which the system should be generating alerts or initiating scripts to increase capacity.



Resource Ceilings

- The parameter you are most interested in is likely to be the overall system capacity, which is the value of WT. WT is the sum over all the Web servers in your infrastructure: WT = Σ(WSnP W SnV).
- WSnP represents the workload of your physical server(s) and WSnV is the workload of the virtual servers (cloud-based server instances) of your infrastructure.
- The amount of overhead you allow yourself should be dictated by the amount of time you require to react to the challenge and to your tolerance for risk.
- A capacity planner would define a value WT such that there is sufficient overhead remaining in the system to react to demand that is defined by a number greater than WMAX by bringing more resources on-line.
- For storage resources that tend to change slowly, a planner might set the red line level to be 85 percent of consumption of storage; for a Web server, that utilization percentage may be different.
- This setting would give you a 15-percent safety factor.

Resource Ceilings

- There are more factors that you might want to take into account when considering an analysis of where to draw the red line.
- When you load test a system, you are applying a certain amount of overhead to the system from the load testing tool—a feature that is often called the "observer effect." Many load testers work by installing lightweight agents on the systems to be tested.
- Those agents themselves impact the performance you see; generally though, their impact is limited to a few percent. Additionally, in order to measure performance, you may be forced to turn on various performance counters.
- A performance counter is also an agent of sorts; the counter is running a routine that measures some aspect of performance such as queue length, file I/O rate, numbers of READs and WRITEs, and so forth.
- While these complications are second order effects, it's always good to keep this in mind and not over-interpret performance results.
- Before leaving the topic of resource ceilings, let's consider a slightly more complicated case that you might encounter in MySQL database systems.

Resource Ceilings

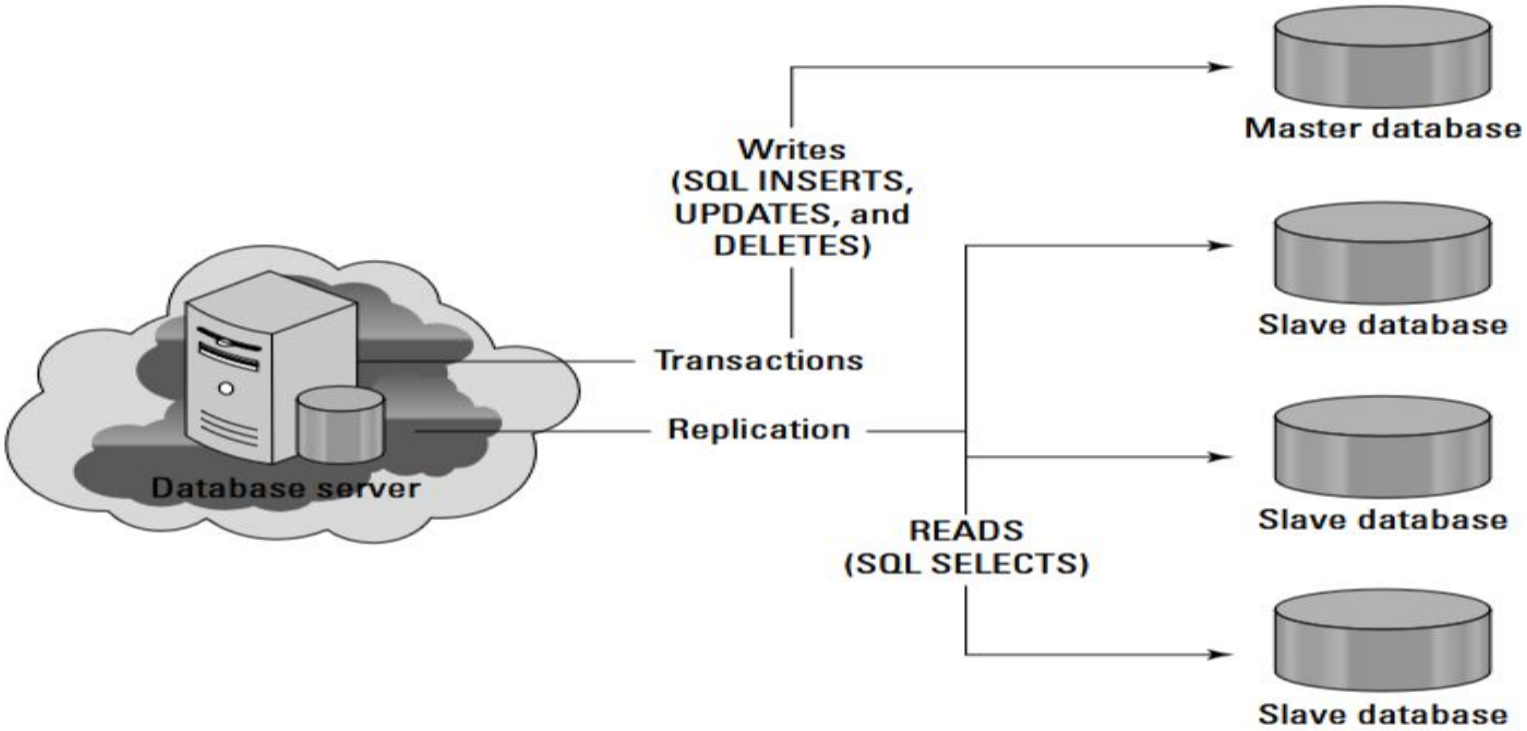
- Database servers are known to exhibit resource ceilings for either their file caches or their Disk I/O performance. To build high-performance applications, many developers replicate their master MySQL database and create a number of slave MySQL databases.
- All READ operations are performed on the slave MySQL databases, and all WRITE operations are performed on the master MySQL database.
- A master/slave database system has two competing processes and the same server that are sharing a common resource: READs and WRITEs to the master/slave databases, and replication traffic between the master database and all the slave databases.
- The ratio of these transactions determines the WSn for the system, and the WSn you derive is highly dependent upon the system architecture.
- These types of servers reach failure when the amount of WRITE traffic in the form of INSERT, UPDATE, and DELETE operations to the master database overtakes the ability of the system to replicate data to the slave databases that are servicing SELECT (query/READ) operations.

Resource Ceilings

- The more slave databases there are, the more actively the master database is being modified and the lower the WS for the master database is.
- This system may support no more than 25-35 percent transactional workload as part of its overall capacity, depending upon the nature of the database application you are running.
- Increase the working capacity of a database server that has a Disk I/O resource ceiling by using more powerful disk arrays and improving the interconnection or network connection used to connect the server to its disks.
- Disk I/O is particularly sensitive to the number of spindles in use, so having more disks equals greater performance.
- Ability to alter the performance of disk assets in a virtual or cloud-based database server is generally limited.
- A master/slave MySQL replication architectural scheme is used for smaller database applications. As sites grow and the number of transactions increases, developers tend to deploy databases in a federated architecture.

Resource Ceilings

Resource contention in a database server



Resource Ceilings

- A single server has a single application with a single resource ceiling. In the second case, you have a single server that has a single application with two competing processes that establish a resource ceiling.
- What do you do in a situation where your server runs the full LAMP stack and you have two or more applications running processes each of which has its own resource ceiling.
- Isolate each application and process and evaluate their characteristics separately while trying to hold the other applications' resource usage at a constant level.
- Examining identical servers running the application individually or by creating a performance console that evaluates multiple factors all at the same time.
- Real-world data and performance is always preferred over simulations. As your infrastructure grows, it becomes more valuable to create a performance console that lets you evaluate your KPIs (Key Performance Indicators) graphically at an instant.

Resource Ceilings

- Many third-party tools offer performance consoles, and some tools allow you to capture their state so you can restore it at a later point.
- An example of a performance analysis tool that lets you save its state is the Microsoft Management Console (MMC).
- In the Amazon Web Service, the statistics monitoring tool is called Amazon CloudWatch, and you can create a performance monitoring console from the statistics it collects.

Server and Instance Types

- One goal of capacity planning is to make growth and shrinkage of capacity predictable. Greatly improve your chances by standardizing on a few hardware types and then well characterizing those platforms.
- In a cloud infrastructure such as Amazon Web Server, the use of different machine instance sizes is an attempt to create standard servers.
- Reducing the variability between servers makes it easier to troubleshoot problems and simpler to deploy and configure new systems.
- As much as possible, you also should assign servers standardized roles and populate those servers with identical services.
- A server with the same set of software, system configuration, and hardware should perform similarly if given the same role in an infrastructure.

Server and Instance Types

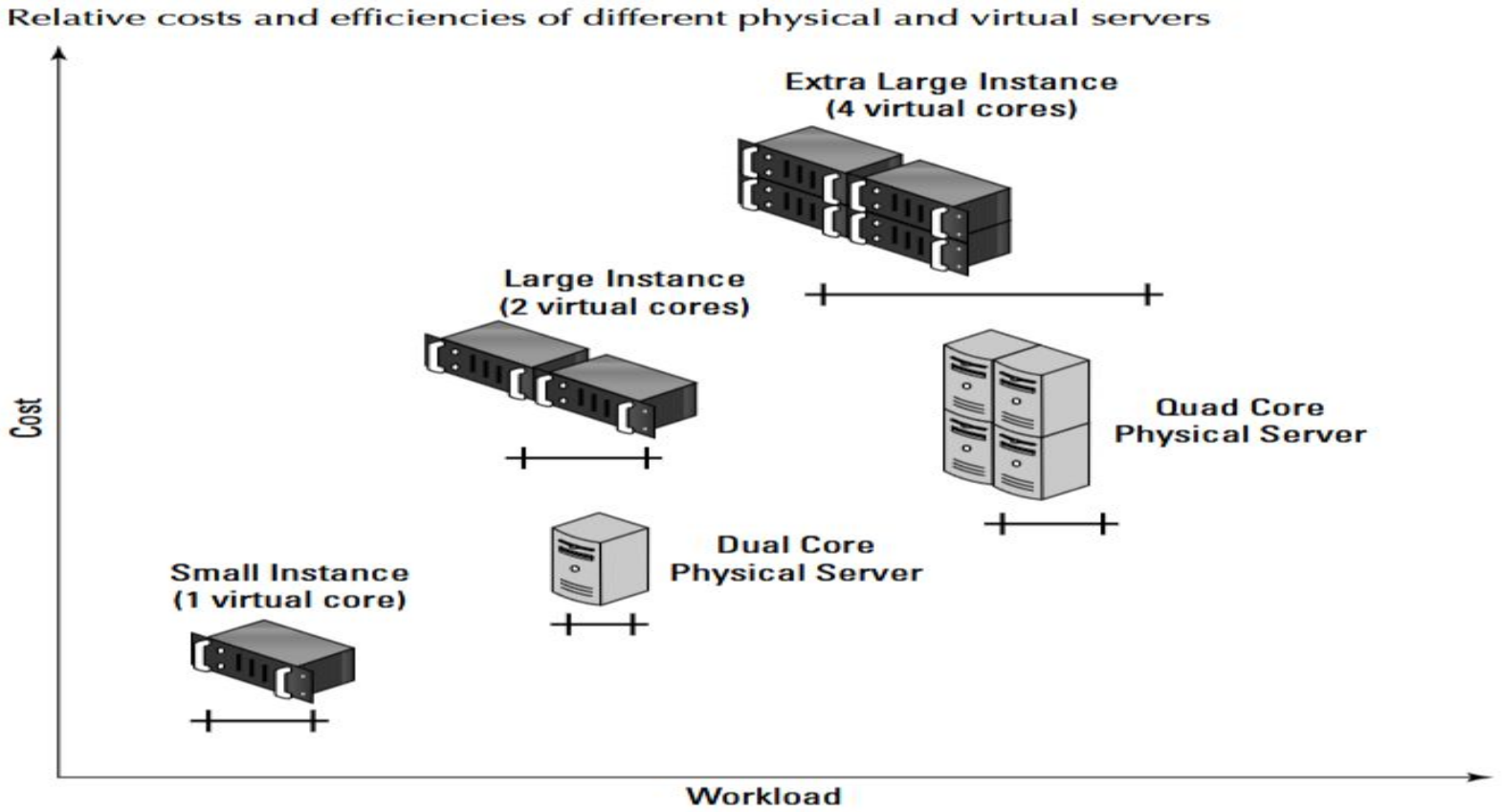
- There is more performance variability in cloud machine instances than you might expect. This variability may be due to your machine instances or storage buckets being moved from one system to another system.
- Therefore, you should be attentive to the increase in potential for virtual system variability and build additional safety factors into your infrastructure. System instances in a cloud can fail, and it is up to the client to recognize these failures and react accordingly.
- Virtual machines in a cloud are something of a black box, and you should treat them as such because you have no idea what the underlying physical reality of the resources you are using represents.
- Capacity planning seeks to compare the capability of one system against another system and to choose the solution that is not only the right size but provides the service with the best operational parameters at the lowest cost.
- In Figure a graph of different server types is shown, with some hypothetical physical servers plotted against Amazon Machine Instance types.
- This type of graph allows you to add the appropriate server type to your infrastructure while performing a cost analysis of the deployment.

Server and Instance Types

An Amazon Machine Instance (AMI) is described as follows:

- **Micro Instance**: 633 MB memory, 1 to 2 EC2 Compute Units (1 virtual core, using 2 Cus for short periodic bursts) with either a 32-bit or 64-bit platform
- **Small Instance (Default):** 1.7GB memory, 1 EC2 Compute Unit (1 virtual core with 1 EC2 Compute Unit), 160GB instance storage (150GB plus 10GB root partition), 32-bit platform, I/O Performance: Moderate, and API name: m1.small
- **High-Memory Quadruple Extra Large Instance:** 68.4GB of memory, 26 EC2 Compute Units (8 virtual cores with 3.25 EC2 Compute Units each), 1,690GB of instance storage, 64-bit platform, I/O Performance: High, and API name: m2.4xlarge
- **High-CPU Extra Large Instance**: 7GB of memory, 20 EC2 Compute Units (8 virtual cores with 2.5 EC2 Compute Units each), 1,690GB of instance storage, 64-bit platform,
- **I/O Performance:** High, API name: c1.xlarge

Server and Instance Types



Server and Instance Types

- While you may not know what exactly EC2 Compute Unit or I/O Performance High means, at least we can measure these performance claims and attach real numbers to them.
- Amazon says that an EC2 Compute Unit is the equivalent of a 1.0-1.2 GHz 2007 Opteron or 2007 Xeon processor, but adds that "Over time, we may add or substitute measures that go into the definition of an EC2 Compute Unit, if we find metrics that will give you a clearer picture of compute capacity.
- What they are saying essentially is that this was the standard processor their fleet used as its baseline, and that over time more powerful systems may be swapped in.
- Whatever the reality of this situation, from the standpoint of capacity planning, all you should care about is measuring the current performance of systems and right-sizing your provisioning to suit your needs.
- In cloud computing, you can often increase capacity on demand quickly and efficiently.
- Not all cloud computing infrastructures automatically manage your capacity for you as Amazon Web Service's Auto Scaling (http://aws.amazon.com/autoscaling/) feature does.

Server and Instance Types

- For example, Amazon Web Services' infrastructure runs not only clients' machine instances, but also Amazon.com's machine instances as well as Amazon.com's partners' instances.
- If you need more virtual servers on Black Monday to service Web sales, you may be out of luck unless you have an SLA that guarantees you the additional capacity.
- Finding an SLA that you can rely on is still something of a crap shoot. If you are in the position of having to maintain the integrity of your data for legal reasons, this imposes additional problems and constraints.
- Easy to get started and requires little up-front costs. However, cloud computing is not cheap. At the point where you have a large cloud-based infrastructure, you may find that it is more expensive to have a metered per-use cost than to own your own systems.
- The long-term efficiencies are realized in the reduced need for IT staff and management, the ability to react quickly to business opportunities, and the freeing of a business from managing networked computer systems to concentrate on their core businesses.
- The Total Cost of Ownership (TCO) of a cloud-based infrastructure might be beneficial, it might be difficult to convince the people paying the bills that this is really so.

Network Capacity

Any cloud-computing system resource is difficult to plan for, it is network capacity. There are three aspects to assessing network capacity:

- Network traffic to and from the network interface at the server, be it a physical or virtual interface or server
- Network traffic from the cloud to the network interface
- Network traffic from the cloud through your ISP to your local network interface (your computer)

- This makes analysis complicated. You can measure factor 1, the network I/O at the server's interface with system utilities, as you would any other server resource.
- For a cloud-based virtual computer, the network interface may be a highly variable resource as the cloud vendor moves virtual systems around on physical systems or reconfigures its network pathways on the fly to accommodate demand.
- To measure network traffic at a server's network interface, you need to employ what is commonly known as a network monitor, which is a form of packet analyzer.
- Microsoft includes a utility called the Microsoft Network Monitor as part of its server utilities,

Network Capacity

The site Sectools.org has a list of packet sniffers at http://sectools.org/ sniffers.html.

- Wireshark (http://www.wireshark.org/), formerly called Ethereal
- Kismet (http://www.kismetwireless.net/), a WiFi sniffer
- TCPdump (http://www.tcpdump.org/)
- Dsniff (http://www.monkey.org/~dugsong/dsniff/)
- Ntop (http://www.ntop.org/)
- EtherApe (http://etherape.sourceforge.net/)

Network Capacity

Factor 2 is the cloud's network performance, which is a measurement of WAN traffic. A WAN's capacity is a function of many factors:

- Overall system traffic (competing services)
- Routing and switching protocols
- Traffic types (transfer protocols)
- Network interconnect technologies (wiring)
- The amount of bandwidth that the cloud vendor purchased from an Internet backbone provider Again, factor 2 is highly variable and unlike factor 1, it isn't easy to measure in a reliable way.
- Tools are available that can monitor a cloud network's performance at geographical different points and over different third-party ISP connections.
- This is done by establishing measurement systems at various well-chosen network hubs.

Network Capacity

- A company that makes WAN network monitoring software, has set up a series of these points of presence at various Internet hubs and uses its networking monitoring software called PathView Cloud to collect data in a display that it calls the Cloud Performance Scorecard (http://www.apparentnetworks.com/CPC/scorecard.aspx).
- Figure shows this Web page populated with statistics for some of the cloud vendors that Apparent Networks monitors.
- Use PathView Cloud as a hosted service to evaluate your own cloud application's network performance at these various points of presence and to create your own scorecard of a cloud network.
- Current pricing for the service is $5 per network path per month. The company also sells a small appliance that you can insert at locations of your choice and with which you can perform your own network monitoring.
- The last factor, factor 3, is the connection from the backbone through your ISP to your local system, a.k.a.
- Internet connection is more like an intelligently managed thin straw that you are desperately trying to suck information out of. So Factor 3 is measurable, even if the result of the measurement isn't very encouraging, especially to your wallet.

Network Capacity

- Internet connectivity over the last mile (to the home) is the Achilles heel of cloud computing. The scarcity of high-speed broadband connections (particularly in the United States) and high pricing are major impediments to the growth of cloud computing.
- Many organizations and communities will wait on the sidelines before embracing cloud computing until faster broadband becomes available.
- Indeed, this may be the final barrier to cloud computing's dominance. That's one of the reasons that large cloud providers like Google are interested in building their own infrastructure, in promoting high-speed broadband connectivity, and in demonstrating the potential of high-speed WANs.
- Google is running a demonstration project called Google Fibre for Communities (http://www.google.com/appserve/fiberrfi/public/overview) that will deliver 1 gigabit-per-second fiber to the home
- The few lucky municipalities chosen (with 50,000 to 500,000 residents) for the demonstration project will get advanced broadband applications, many of which will be cloud-based.

Network Capacity

Apparent Networks' Cloud Performance Center provides data on WAN throughput and uptime at Internet network hubs.

Scaling

- In capacity planning, after you have made the decision that you need more resources, you are faced with the fundamental choice of scaling your systems.
- You can either scale vertically (scale up) or scale horizontally (scale out), and each method is broadly suitable for different types of applications.
- To scale vertically, you add resources to a system to make it more powerful. For example, during scaling up, you might replace a node in a cloud-based system that has a dual-processor machine instance equivalence with a quad-processor machine instance equivalence.
- You also can scale up when you add more memory, more network throughput, and other resources to a single node. Scaling out indefinitely eventually leads you to an architecture with a single powerful supercomputer.
- Vertical scaling allows you to use a virtual system to run more virtual machines (operating system instance), run more daemons on the same machine instance, or take advantage of more RAM (memory) and faster compute times.
- Applications that benefit from being scaled up vertically include those applications that are processor-limited such as rendering or memory-limited such as certain database operations—queries against an in-memory index.

Scaling

- Horizontal scaling or scale out adds capacity to a system by adding more individual nodes.
- In a system where you have a dual-processor machine instance, you would scale out by adding more dual-processor machines instances or some other type of commodity system.
- Scaling out indefinitely leads you to an architecture with a large number of servers (a server farm), which is the model that many cloud and grid computer networks use.
- Horizontal scaling allows you to run distributed applications more efficiently and is effective in using hardware more efficiently because it is both easier to pool resources and to partition them.
- Although your intuition might lead you to believe otherwise, the world's most powerful computers are currently built using clusters of computers aggregated using high speed interconnect technologies such as InfiniBand or Myrinet.
- Scale out is most effective when you have an I/O resource ceiling and you can eliminate the communications bottleneck by adding more channels.Web server connections are a classic example of this situation.

Scaling

- These broad generalizations between scale up and scale out are useful from a conceptual stand point, but the reality is that there are always tradeoffs between choosing one method for scaling your cloud computing system versus the other.
- The pricing model that cloud computing vendors now offer their clients isn't fully mature at the moment, and you may find yourself paying much more for a high-memory extra-large machine instance than you might pay for the equivalent amount of processing power purchased with smaller system equivalents.
- This has always been true when you purchase physical servers, and it is still true when purchasing virtual servers. Cost is one factor to pay particular attention to, but there are other tradeoffs as well.
- Scale out increases the number of systems you must manage, increases the amount of communication between systems that is going on, and introduces additional latency to your system.