eCommerce – Chapter 03

Server Architectures

- **Two-tier**: Simple Web site consists only of a client and a server tier
- **Three-tier**: Extends the 2-tier approach to allow additional processing to occur before the web server responds to the web client's request (Often DBs)
- **N-TIER:** Used for tracking customer purchases, keep track of customer preferences, update in-stock inventory databases, etc.

Pre-Web Era
Application runs on a computer that performs the presentation & business logic.

Web Era
Application server performs only the business logic. In an Internet environment that hosts a variety of languages systems used to program databases queries and general business processing.

Blade Systems
- **Blade**: Spatially reduced server computer with a modular design developed to minimize the use of physical space and energy.
- **Blade enclosure**: Host multiple blade servers.

Centralized Server Architectures
- Consist of a few, very large and fast computers.
- Requires expensive computers
- More sensitive to technical problems
- Bad scalability

Distributed Server Architectures
- Uses a large number of less-powerful computers and divides the workload
- Many small less expensive computers
- Spreads the risk of failure
- Requires additional hubs and switches to interconnect the servers
- Use load-balancing systems to assign the workload efficiently

Capacity Planning and Scalability

Vertical Scaling / Scaling Up
- Adds resources to a system to make it more powerful
- Leads to single powerful supercomputer
- **Useful for:** processor-limited or memory limited applications

Horizontal Scaling / Scaling Out

- Adds capacity to system by adding more individual nodes
- Leads to server farm
- **Useful for:** distributed applications, deploy low cost commodity systems for high performance computers

Capacity Planning

- Seeks to match demand to available resources.
- Examines what systems are in place, measures their performance and determines patterns in usage that enables the planner to predict demand

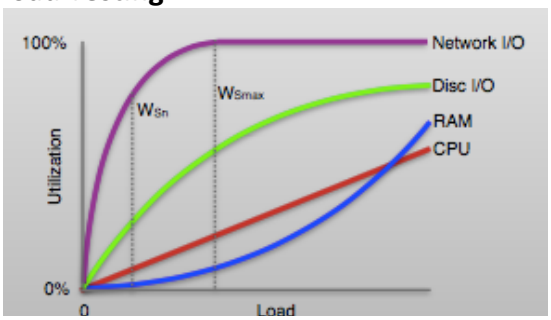
Procedure

1. Determine the characteristics of the present system
2. Measure the workload for the different resources in the system
3. Load the system until it is overloaded, determine when it breaks and specify what is required to maintain acceptable performance
4. Knowing when systems fail under load and what factor is responsible for the failure
5. Predict the future based on historical trends and other factors
6. Deploy or tear down resources to meet your predictions
7. Iterate Steps 1 to 5 repeatedly

System Metrics

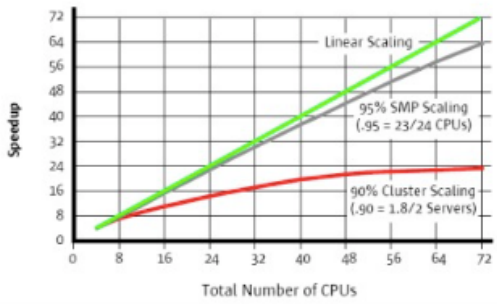
- Application-Level Statistics
  - Page view: *hits per second*
  - Transactions completed: *transactions/queries per second*
- System-Level Statistics
  - CPU, RAM, Disk, Network, Connectivity, …

Load Testing



- What is the maximum load that my current system can support?
- Which resource represents the bottleneck in the current system that limits the system's performance?

Speedup / Relative Performance



- Number of times an application runs faster on multiple processors or nodes vs on a single process
- **Scalability factor:** Percentage of the resource that is actually usable

If 5% of process power is lost every time a CPU is added: scalability factor is 0.95.

Load Balancing and Content Switching

Benefits

1. Improve reliability
2. Lower cost
3. Improve maintainability
4. Improve performance
5. Improve scalability and flexibility

DNS-based Load Balancing

- Allows single domain name to be associated with several IP addresses
- Request for resolving the URL is sent through the distributed DNS hierarchy
- Order of IP addresses is alternated for each time a new query is sent
- **Benefits:** Simple concept, as no additional hardware is required
- **Drawbacks:** May prevent load balancing

Layer-4-Switching

- Placed between the connection to the internet and the server farm
- Recognizes when a client is requesting a new session.
- Maintains a session server binding table that associates each active session with the selected server
- Recognizes when the session is terminated and removes the session-server binding from its binding table
- **Benefits**:
    - Good load balancing
    - No problems caused by the DNS data
    - Can be used in combination with sophisticated algorithms
    - Can consider failures of web servers
- **Drawbacks**: Does not consider what content is being requested

Content Switching

- Intelligently distributes traffic across delivery nodes, dynamically directing specific content requests to the best site and server at the moment.
- Based on content availability, application availability and server load.
- Types:
    - URL Switching
    - Cookie Switching
    - SSL Session-ID Switching

Load Balancing Algorithms

1. Static
    a. Random
    b. Round Robin
    c. Weighted Round Robin
2. Client aware
    a. Client Partition
3. Server aware
    a. Least Traffic First
    b. Least Weighted Load
    c. Fastest Response
    d. Least Load First

Cloud Computing

It is a style of computing where massively scalable IT-related capabilities are provided as service across the internet to multiple external customers.

Cloud Service Models

1. Infrastructure as a Service (Flexibility)
2. Platform as a Service
3. Software as a Service (Optimization)

Five Attributes

1. Multi-tenancy: *Multiple users use the same resources*
2. Massive scalability
3. Elasticity: *Users can increase and decrease their computing resources as needed*
4. Pay as you go: *Pay only for what you use*
5. Self-provisioning of resources

Cloud Deployment Models

- Public Clouds:
    - Managed by a third-party vendor for one or more data centers
    - Service is offered to multiple customers over a common infrastructure
- Private Clouds:
    - Only for a single organization
    - Internal IT or third party with contractual SLAs
- Hybrid Clouds:
    - Consist of multiple internal and/or external providers

Supply-Side Economics of Scale

- Cost of Power
    - Inexpensive energy cost for the cloud providers

- Infrastructure Labour Costs
    - An administrator can server more servers
- Security and reliability
    - Large providers have better security expertise
- Buying power
    - Large operators get discounts for hardware purchases

Demand-Side Economics of Scale
In non-virtualized data centers, the utilization of servers has traditionally been extremely low. Virtualization enables multiple applications to run a physical server within their optimized OS instance.

- Random Variability
- Time-of-Day Patterns
    - Daily cycles are responsible for demand peaks and low utilization
- Industry-specific Variability
- Multi-resource Variability
    - Different resources are important for different tasks.
    - For Search CPU is more important than Disk I/O
    - But for E-Mail is the other way around
- Uncertain growth patterns

Virtualization
- Abstraction of computer resources from applications and end users consuming the service.
- Type:
    - Server virtualization
    - Storage virtualization
    - Network virtualization: *VLAN, VPN*

Hypervisor / Virtual Machine Monitor
Virtualization technique which allows multiple operating systems to run concurrently on a host computer.

- Type-1-hypervisor:
    - Runs directly on the host's hardware to control it and to monitor the guest OSs
    - Can archive higher virtualization efficiency
- Type-2-hypervisor:
    - Runs with a conventional operating system environment
    - Manly used where support for a broad range of I/O devices is important