Lecture Notes 2

Todays Topics

- Cluster Architecture
- Scalability, Availability, Reliability, and Performance
- Replication and Partitioning
- CAP Theorem

Google Search Example

- Crawling and Indexing
- 10 Billion Web pages
- Average size of webpage = 20kb
- 10 Billion & 20KB = 200TB
- Time to read = 50 days
- To do something useful with the data?
- CPU -> Memory -> Disk

A Typical Cluster

Distributed Computing Challenges

Node Failures

- A single server stay up for 3 years (1000 days)
- 1000 servers in a cluster? **1 failure per day**
- 1 Million servers in a cluster? **1000 failures per day**
- **Availability and Consistency** - How to store data persistently and keep it available if nodes can fail
- **Reliability and Fault Tolerance** - Deal with node failures during a long-running computation
- **Scalable and Performance** - Network bottleneck and load balancing
- Network bandwidth = 1Gbps
- Moving 10TB takes approximately 1 day

- Balancing load across nodes?

Big Data Systems

Address the challenges of distributed computing

- Store data redundantly on multiple nodes (persistence and availability)
- Move computation close to data to minimize data movement
- Simple programming model to hide complexity of distributed computing

Scalable Data Processing

- Scalability - handle a growing amount of work
- Availability - The proportion of time a system is in a fuctioning condition
- Fault Tolerance - ability of a system to behave in a well-defined scenario where faults occur
- Performance - throughput, latency, (response time)

Scalability

- Ability of a system, network, or process to handle a growing amount of work in a capable manner
- Size Scalability
- Adding more nodes should make system linearly faster
- Growing the dataset should not increase latency

Availabilty

- The proportion of time a system in a functioning condition or node failures do not prevent from continuing tooperate
- Systems build with redundancy tolerant of partial failures and thus more available being fault tolerant

Fault-tolerance

- Ability of a system to behave in a well-defined manner once faults occur
- Define what faults you expect -> Design a system that is tolerant to them

Fault Tolerance and Performance

- Distributed Systems are constrained by
  - Increase in the number of nodes -> Failures
  - Distance between nodes -> Performance

Performance

- The amount of useful work accomplished by a system compared to the time and resources used
  - Throguhput
  - Response Time
  - Utilization of computing resources

Yahoo Experiement - Performance (Latency)

- High latency is bad for business

Paritioning (Sharding)

- Distribution of data between multiple nodes is very important
- Split data over multiple nodes -> Parallel Processing
- Partitioning improves performance by limiting the amount of data to be examined and by locating related data in same partition
- Partitioning improces availability by allowing partitions to fial independently

Replication

- Make copies of same data on multiple machines
- Replication - copying or reproducing something - is the primary way in which we can fight latency
- Replicaiton improves performance by making additional computing power and bandwidth applicable to new copy of data
- Replication improves availability by creating additional copies of the data, increasing the number of nodes that need to fail before availability is sacrificed **avoids single point of failure**

Consistency

- All nodes see same data at same time
- Replication -> Consistency Problems
- Consistency - there are multiple types
  - String consistency

- Eventual Consistency