

## Data Centers

Kai Shen

4/22/2014

CSC 258/458 - Spring 2014

1

## Data Centers

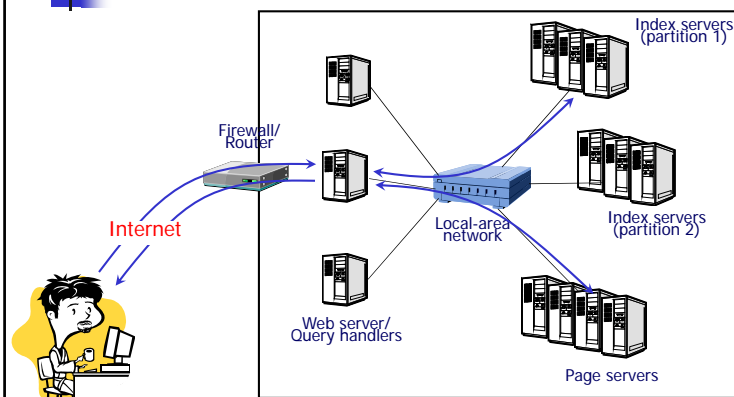
- Large distributed systems of computers and storage (size of warehouses)
  - Host interactive web services
  - And also batch processing / data mining
- Can be at very large scale:
  - Up to 10s/100s of thousands of machines (millions of processors)
  - megawatts or more in power usage
  - Serve millions of users (thousands at a time)
  - Host large amount of data: partitioning and replication

4/22/2014

CSC 258/458 - Spring 2014

2

## Data Center Hosting Web Search



4/22/2014

CSC 258/458 - Spring 2014

3

## Motivations for Extreme Scale

- Highly popular applications /services
- Big data
- Cost efficiency at scale (buying bulk is cheaper)
  - Resources to applications can be elastically managed
  - Repair and maintenance is centralized
  - Mistakes from individual incidents are learned and used to improve management at scale

4/22/2014

CSC 258/458 - Spring 2014

4

## Data Center Architecture

- Earlier days of parallel computing
  - Scientific computing is parallelizable, but complex synchronization glue the parallel processes
  - Distributed databases are also tightly coupled
  - Many points of failure in commodity machine clusters
  - **Supercomputers with custom network and node design**: close integration but expensive, hard to scale
- Late 90s, early 2000
  - Embarrassingly parallel web services: web search, e-commerce
  - **Commodity machines and networks, glued together by software**
  - Turn independent failures from a curse to a strength, migrate a server farm
  - Cheaper at scale since the software cost is amortized

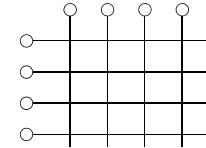
4/22/2014

CSC 258/458 - Spring 2014

5

## Network Fabric

- Network matters to distributed data processing
- A Gbps network switch allows simultaneous traffic at full link speed
  - Crossbar: Complexity grows up quickly with the number of ports



- Switch with a small number of ports is cheaper (~\$30 per port)
  - Cost goes up non-linearly (an order of magnitude higher per port cost)
- One design choice
  - Cheap, few-port switch to link machines on a rack
  - Expensive, many-port switch to link all the racks together

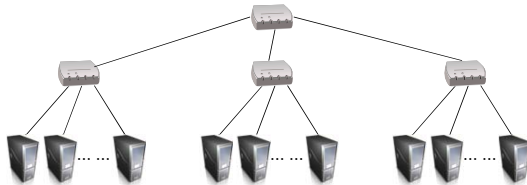
4/22/2014

CSC 258/458 - Spring 2014

6

## Network Switching at Data Centers

- Thousands of servers with hierarchical network switching



- Reduce congestions:
  - Network load balance across nodes
  - Place highly communicating server modules within same switch domain, or better yet, in same machines
  - Redundant backbone links → Equal-Cost Multi-Path (ECMP) routing

4/22/2014

CSC 258/458 - Spring 2014

7

## Storage Architecture

- Each machine has local storage, managed by distributed file system such as GFS (Colossus), HDFS
- In addition (alternatively), storage-level parallelism
  - specialized storage systems plugged into the network switch that manages replication, parallel I/O etc. (scalable RAID system)
- Contrast
  - Storage-level parallelism generally provides better I/O performance
  - File system parallelism integrates better with higher-level software (such as MapReduce), facilitates data locality through smart software management, and puts less stress on network

4/22/2014

CSC 258/458 - Spring 2014

8

## Memory/Storage Hierarchy

- CPU cache, memory, Flash SSD, and disk
  - Larger space but longer latency
- In a data center, we have CPU cache, local memory, local disk, rack memory, rack disk, data center memory, data center disk
  - Figures 1.3, 1.4 (page 10)

4/22/2014

CSC 258/458 - Spring 2014

9

## Data Center Workloads

- Unique, large jobs
  - E.g., compute page ranks from the 50B crawled web pages
- Many, small tasks
  - E.g., Amazon cloud computing, Google search service
- Parallelism
  - Data parallelism
  - User / request parallelism

4/22/2014

CSC 258/458 - Spring 2014

10

## Platform / Application Adaptation

- Platform: basic hardware and infrastructure system software
- Applications developed on top
- Platform / application adaptation
  - Platforms are somewhat homogeneous
  - Application upgrades are fast

⇒ Focus on basic cost-effectiveness on the platform construction and let application software adapt to the platform

4/22/2014

CSC 258/458 - Spring 2014

11

## Infrastructure Software: Buy, Adapt, Build

- Infrastructure software: operating system, file systems, databases, parallel computation, replication, ...
- Choices:
  - Buy from vendors
  - Take open-source software and make adaptation
  - Build new software from scratch
- Google's argument: adapt or build, but don't buy
  - Need full control (performance debugging), flexibility, customization (nosql)
  - Few vendors (think of databases) are capable of building and testing their software at the datacenter scale

4/22/2014

CSC 258/458 - Spring 2014

12

## Cost-Efficient Analysis on Multiprocessor

- You have N parallel tasks
  - c is the computation time, d is the number of data accesses,  $l_{\text{local}}$  is the latency of local data access,  $l_{\text{remote}}$  is the latency of remote data access
- If each task runs on a separate machine
  - Assume data is evenly distributed, the chance of a data access to be local is  $1/N$
  - Then the execution time is:  $c + d \cdot (l_{\text{local}} \cdot 1/N + l_{\text{remote}} \cdot (1-1/N))$
- If you have a large multiprocessor (host all N tasks):
  - Then the execution time is:  $c + d \cdot l_{\text{local}}$
- In practice, your multiprocessor isn't large enough to host all (host  $n < N$  tasks)
  - Then the execution time is:  $c + d \cdot (l_{\text{local}} \cdot n/N + l_{\text{remote}} \cdot (1-n/N))$
  - Figure 3.2 (page 36)

4/22/2014

CSC 258/458 - Spring 2014

13

## Power

- Power consumption of a large data center is significant
- The power distribution system is also expensive
  - In building cost and power lost during distribution
- People care about the total energy usage and peak power
- What do people do?
  - Build your data center next to a hydro power plant
  - Energy conservation through server consolidation
    - Weigh spare capacity against load spikes and speed of server powerup
  - Lower power spikes through load shifting or throttling

4/22/2014

CSC 258/458 - Spring 2014

14

## Conventional vs. Wimpy CPUs

- Conventional CPU (regular CPUs on desktops / servers)
- Wimpy CPU (what you have in a smart phone)
- Why wimpy CPUs?
  - With good parallelism, a larger number of wimpy CPUs can do the job done by a smaller number of conventional CPUs
  - Cluster of wimpy CPUs may save power and save money
  - E.g., a 2-Watt wimpy processor is 5 times slower than a 50-Watt server processor  $\Rightarrow$  potentially 5 times higher energy efficiency
- But the assumption of "good parallelism"
  - Now you know that good parallelism isn't easy to get

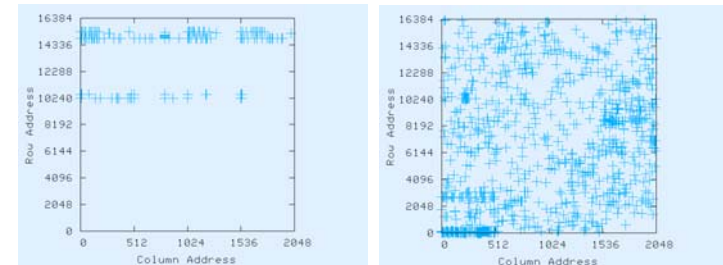
4/22/2014

CSC 258/458 - Spring 2014

15

## Reliability at Scale

- Chance of failures/anomalies to occur in a large system is high



- Permanent faults on two memory chips from thousands of memory chips we examined in a production system
- Transient faults were also found

4/22/2014

CSC 258/458 - Spring 2014

16

## Reliability at Scale

- Why does a machine fail?
  - Hardware problems, software problems
- Reliability rate for one machine is  $p$ , for  $n$  machines is  $p^n$
- In a very large system, you may not always be able to find out problem cause and fix the problem, so
  - Detect problem and tolerate / mitigate it
  - Redundancy / replication
    - Memory ECC
    - Machines
    - Data storage
      - Replication of mutable states introduces consistency problems
    - Networking
    - Power supply

4/22/2014

CSC 258/458 - Spring 2014

17

## Reliability at Scale

- Performance anomalies
  - A particular request / map task is very slow
  - Technique to tolerate slow tail: perform work redundantly if the current run is too slow
- Recovery-oriented computing
  - Components are inherently unreliable, design a system in which components can be independently recovered (rebooted) without significant effects on others
  - Make sure that restart is fast (multi-scaled restart)

4/22/2014

CSC 258/458 - Spring 2014

18

## Failure Correlation

- Correlated catastrophic failures
  - A bug-triggering input / request: Canaries
  - Load-triggered failures
  - Malicious activities

4/22/2014

CSC 258/458 - Spring 2014

19

## Management

- Easy to add/remove components
  - Container-based data center modules

4/22/2014

CSC 258/458 - Spring 2014

20



## Disclaimer

- Preparation of this class was helped by materials in the book “The Datacenter as a Computer” by Luiz A. Barroso, Jimmy Clidaras, and Urs Holzle.