CS 537
Section 11
Large Scale Systems

Michael Swift

Recent Trends

- Computing is moving away from the desktop
  - To mobile device: smart phones
  - To data centers: cloud computing
- Why?
  - Cheap communication
    - Enables a smart phone to be useful
    - Enables low-latency communication with a data center
  - Cheap computation & storage
    - Can carry enough power with you to do interesting things
    - Can build a data center to do interesting things for many people

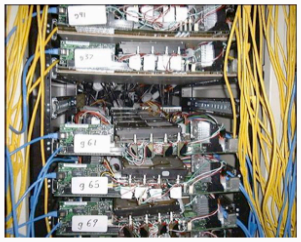
Google Design Philosophy

Truckloads of low-cost machines

- Workloads are large and easily parallelized
- Care about perf/$, not absolute machine perf
- Even reliable hardware fails at our scale

- Why?
  - At large scale (100,000+ machines), things will fail and software will handle it
  - Workload is independent requests; can spread across many independent machines

Effects of Google's HW philosophy



Google - 1999

- Software must tolerate failure
- Application's particular machine should not matter
- No special machines - just 2 or 3 flavors

Failure happens in the real world

Typical first year for a new cluster:

~0.5 overheating (power down most machines in <5 mins, ~1-2 days to recover)
~1 PDU failure (~500-1000 machines suddenly disappear, ~6 hours to come back)
~1 rack-move (plenty of warning, ~500-1000 machines powered down, ~6 hours)
~1 network rewiring (rolling ~5% of machines down over 2-day span)
~20 rack failures (40-80 machines instantly disappear, 1-6 hours to get back)
~5 racks go wonky (40-80 machines see 50% packet loss)
~8 network maintenances (4 might cause ~30-minute random connectivity losses)
~12 router reloads (takes out DNS and external vips for a couple minutes)
~3 router failures (have to immediately pull traffic for an hour)
~dozens of minor 30-second blips for dns
~1000 individual machine failures
~thousands of hard drive failures

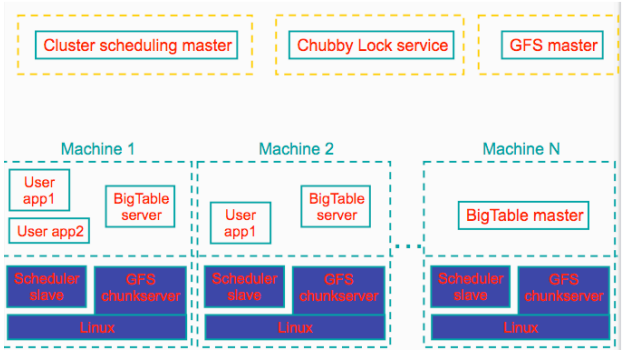
slow disks, bad memory, misconfigured machines, flaky machines, etc.

Google

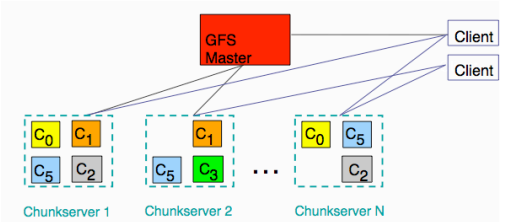
Google Software Design

- Linux kernel everywhere (an old version)
- Infrastructure services shared by all applications
  - Google File System (GFS) for sharing data
  - MapReduce programming model for accessing data
  - Chubby Lock Service for synchronizing access to data
  - BigTable for structured data, such as database tables
- Services hierarchically decomposed:
  - Small number of masters for complex synchronization
  - Workers for distributing load across many machines

Software Architecture



Example: GFS



- Master: Manages file metadata
- Chunkserver: Manages 64MB file chunks
- Clients talk to master to open and find files
- Clients talk directly to chunkservers for data

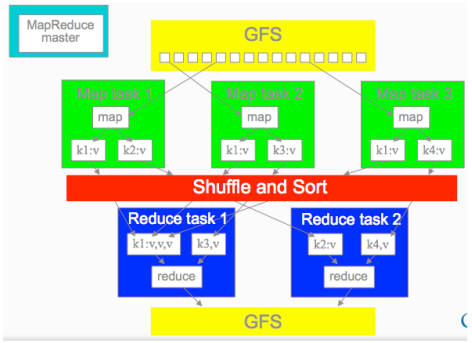
Example: MapReduce

- Google's batch processing tool of choice
- Users write two functions:
  - Map: Produces (key, value) pairs from input
  - Reduce: Merges (key, value) pairs from Map
- Library handles data transfer and failures
- Used everywhere: Earth, News, Analytics, Search Quality, Indexing, …

Example: Document Indexing

- Input: Set of documents D_1, …, D_N
- Map
  - Parse document D into terms T_1, …, T_N
  - Produces (key, value) pairs
    - (T_1, D), …, (T_N, D)
- Reduce
  - Receives list of (key, value) pairs for term T
    - (T, D_1), …, (T, D_N)
  - Emits single (key, value) pair
    - (T, (D_1, …, D_N))

Execution

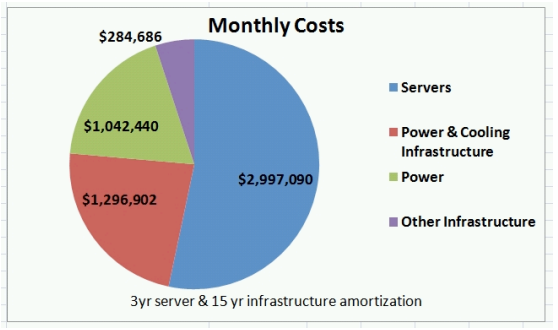


Hardware Design: Data Centers

Data Centers

- Buildings full of machines (thousands of identical machines)
- Machines stored in racks, that provide power, cooling (if water based), network
- State of the art trend: build data centers from shipping containers
- Key Concern: Power efficiency
  - Power usage is substantial part of cost
  - Cooling is a big part of power: must cool off every watt spent computing
  - Often located near cheap power (hydroelectric) or cheap cooling (cold weather)

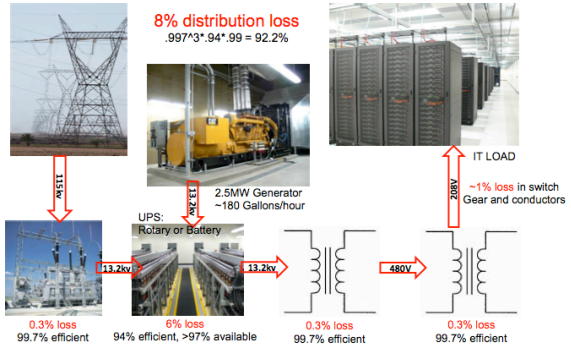
Costs in a Data Center



Where Does Power Go?

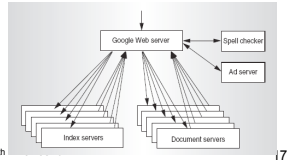
- **Assuming a pretty good data center with PUE ~1.7**
  - Each watt to server loses ~0.7W to power distribution losses & cooling
- **Power losses are easier to track than cooling:**
  - Power transmission & switching losses: 8%
    - Detailed power distribution losses on next slide
  - Cooling losses remainder:100-(59+8) => 33%
- **Data center power consumption:**
  - IT load (servers): 1/1.7=> 59%
  - Distribution Losses: 8%
  - Mechanical load(cooling): 33%

Power Distribution



Power

- **Power is a very big issue**
  - Google servers 400 watts/ft^2
  - High end servers 700 watts/ft^2
  - Typical commercial data center – 70-150 watts/ft^2
    ⇒ special cooling or additional space, anyway
    using high-end servers would make matters worse

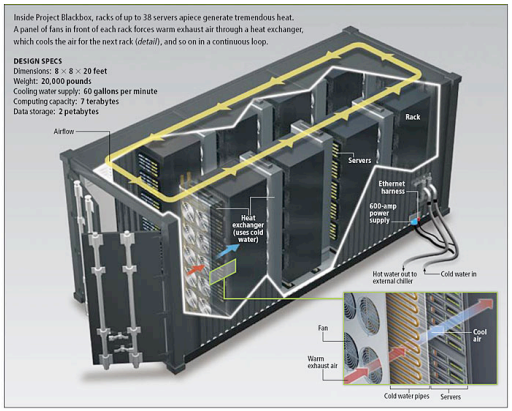


Shipping Container as Data Center Module

- Data Center Module
  - Contains network gear, compute, storage, & cooling
  - Just plug in power, network, & chilled water
- Increased cooling efficiency
  - Variable water & air flow
  - Better air flow management (higher delta-T)
  - 80% air handling power reductions (Rackable Systems)
- Bring your own data center shell
  - Just central networking, power, cooling, security & admin center
  - Can be stacked 3 to 5 high
  - Less regulatory issues (e.g. no building permit)
  - Avoids (for now) building floor space taxes
- Move resources closer to customer (CDN mini-centers)
- **Distributed, incremental fast built mini-centers**

Inside Google's Data centers

Inside a Container



Manufacturing & H/W Admin. Savings

- Factory racking, stacking & packing much more effic
  - Robotics and/or inexpensive labor
- Avoid layers of packaging
  - Systems->packing box->pallet->container
  - Materials cost and wastage and labor at customer site
- Data Center power & cooling expensive consulting contracts
  - Data centers are still custom crafted rather than prefab units
  - Move skill set to module manufacturer who designs power & cooling once
  - Installation design to meet module power, network, & cooling specs
- More space efficient
  - Power densities in excess of 1250 W/sq ft
  - Rooftop or parking lot installation acceptable (with secur
  - Stack 3 to 5 high
- Service-Free
  - H/W admin contracts can exceed 25% of systems cost
  - Sufficient redundancy that it just degrades over time
    - At end of service, return for remanufacture & recycli
  - 20% to 50% of systems outages caused by Admin error (A. Brown & D. Patterson)

1/21/2007 21