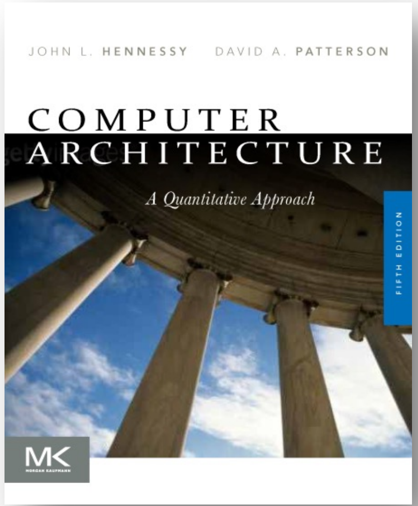


Chapter 6

Warehouse-Scale Computers to Exploit Request-Level and Data-Level Parallelism

Introduction

- Warehouse-scale computer (WSC)
  - Provides Internet services
    - Search, social networking, online maps, video sharing, online shopping, email, cloud computing, etc.
  - Differences with HPC “clusters”:
    - Clusters have higher performance processors and network
    - Clusters emphasize thread-level parallelism, WSCs emphasize request-level parallelism
  - Differences with datacenters:
    - Datacenters consolidate different machines and software into one location
    - Datacenters emphasize virtual machines and hardware heterogeneity in order to serve varied customers

Introduction

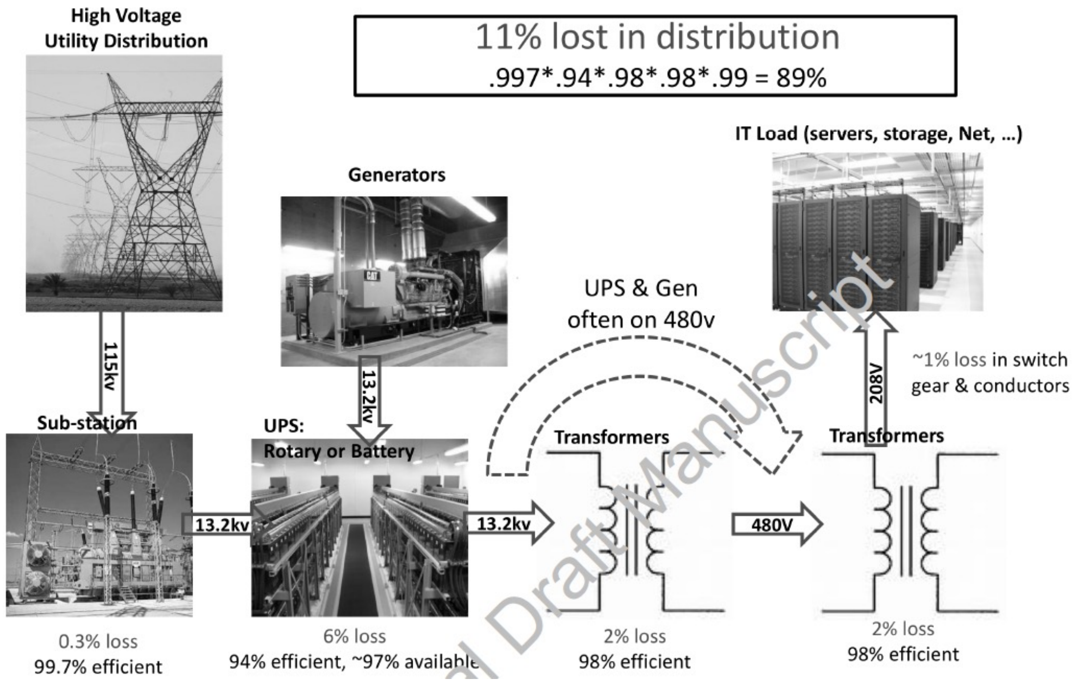
- Important design factors for WSC:
  - Cost-performance
    - Small savings add up
  - Energy efficiency
    - Affects power distribution and cooling
    - Work per joule
  - Dependability via redundancy
  - Network I/O
  - Interactive and batch processing workloads

Introduction

- Ample computational parallelism is not important
  - Most jobs are totally independent
  - “Request-level parallelism”
- Operational costs count
  - Power consumption is a primary, not secondary, constraint when designing system
- Scale and its opportunities and problems
  - Can afford to build customized systems since WSC require volume purchase
- Location counts
  - Real estate, power cost; Internet, end-user, and workforce availability
- Computing efficiently at low utilization
- Scale and the opportunities/problems associated with scale
  - Unique challenges: custom hardware, failures
  - Unique opportunities: bulk discounts

Efficiency and Cost of WSC

- **Location of WSC**
  - **Proximity to Internet backbones, electricity cost, property tax rates, low risk from earthquakes, floods, and hurricanes**
- **Power distribution**



Prgrm'g Models and Workloads

- Batch processing framework: MapReduce
  - **Map:** applies a programmer-supplied function to each logical input record
    - Runs on thousands of computers
    - Provides new set of key-value pairs as intermediate values
  - **Reduce:** collapses values using another programmer-supplied function

Prgrm'g Models and Workloads

- Example:
  - **map (String key, String value):**
    - **// key: document name**
    - **// value: document contents**
    - **for each word w in value**
      - **EmitIntermediate(w,"1"); // Produce list of all words**
  - **reduce (String key, Iterator values):**
    - **// key: a word**
    - **// value: a list of counts**
    - **int result = 0;**
    - **for each v in values:**
      - **result += ParseInt(v); // get integer from key-value pair**
    - **Emit(AsString(result));**

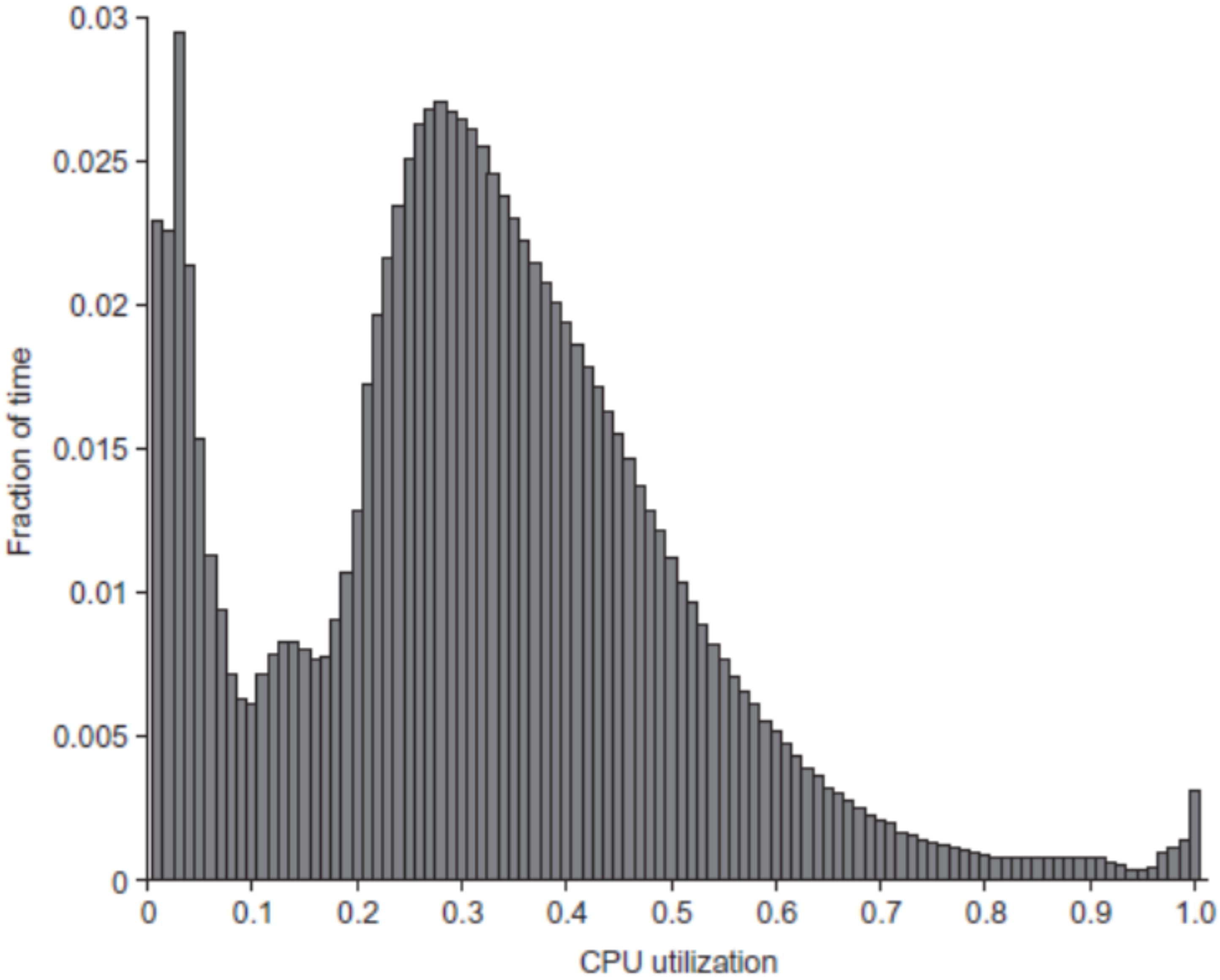
Prgrm'g Models and Workloads

- **Availability:**
  - **Use replicas of data across different servers**
  - **Use relaxed consistency:**
    - **No need for all replicas to always agree**
- **File systems: GFS and Colossus**
- **Databases: Dynamo and BigTable**

Prgrm'g Models and Workloads

- **MapReduce runtime environment schedules map and reduce task to WSC nodes**
  - **Workload demands often vary considerably**
  - **Scheduler assigns tasks based on completion of prior tasks**
  - **Tail latency/execution time variability: single slow task can hold up large MapReduce job**
  - **Runtime libraries replicate tasks near end of job**

Prgrm’g Models and Workloads



Computer Architecture of WSC

- WSC often use a hierarchy of networks for interconnection
- Each 19” rack holds 48 1U servers connected to a rack switch
- Rack switches are uplinked to switch higher in hierarchy
  - Uplink has 6-24X times lower bandwidthGoal is to maximize locality of communication relative to the rack

Storage

- Storage options:
  - Use disks inside the servers, or
  - Network attached storage through Infiniband

  - WSCs generally rely on local disks
  - Google File System (GFS) uses local disks and maintains at least three relicas

Array Switch

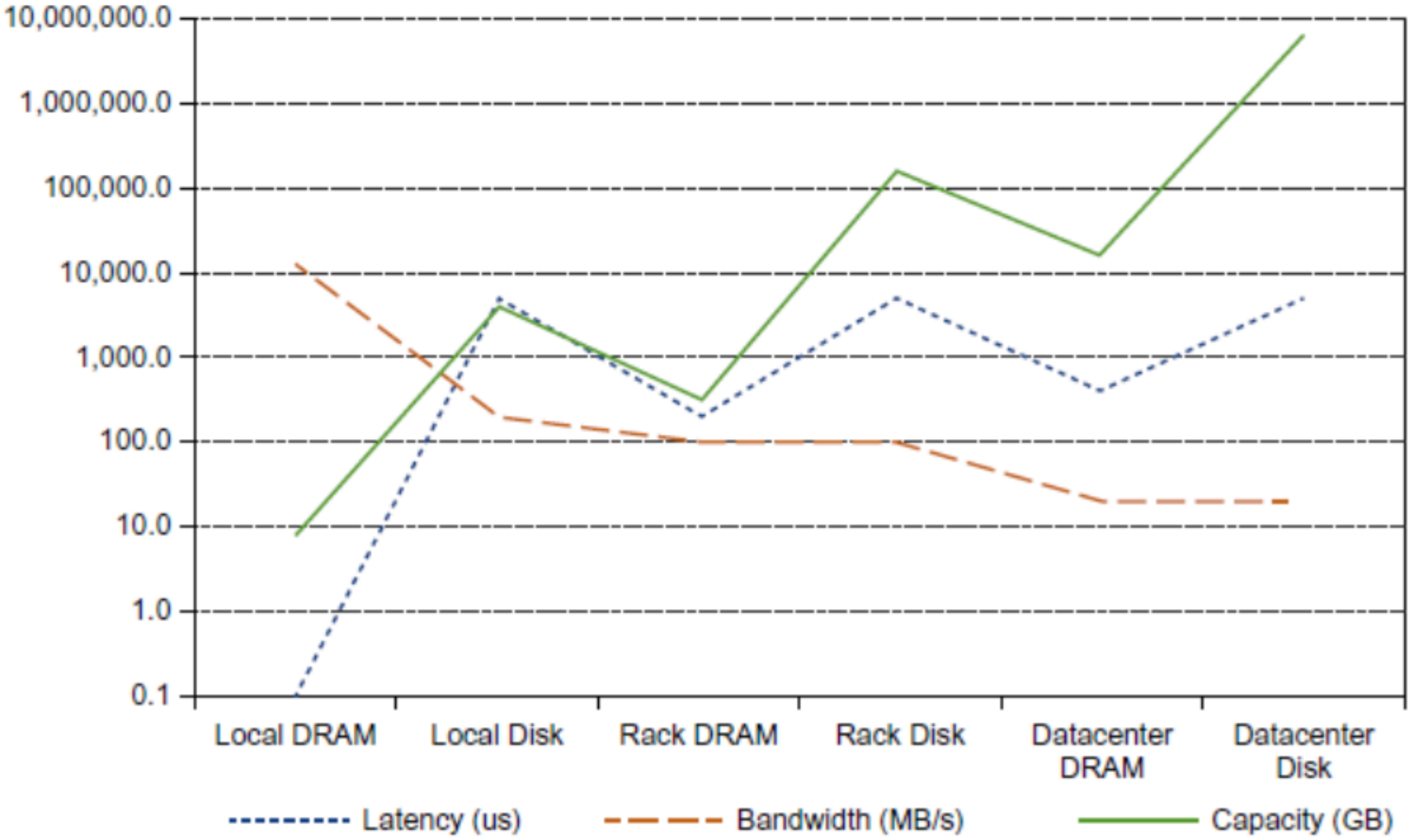
- **Switch that connects an array of racks**
  - **Array switch should have 10 X the bisection bandwidth of rack switch**
  - **Cost of *n*-port switch grows as n^2**
  - **Often utilize content addressible memory chips and FPGAs**

WSC Memory Hierarchy

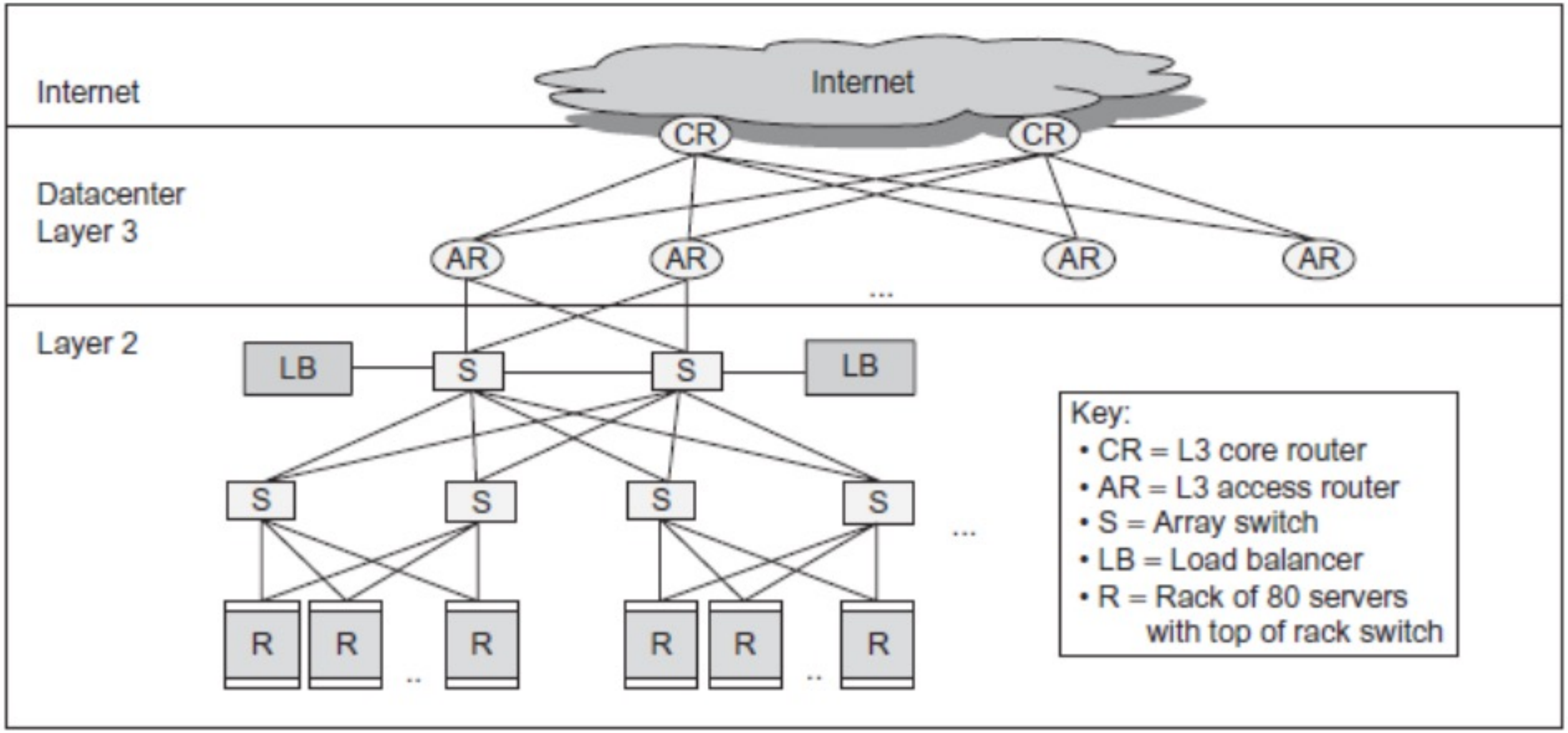
- Servers can access DRAM and disks on other servers using a NUMA-style interface

	Local	Rack	Array
DRAM latency (μs)	0.1	300	500
Flash latency (μs)	100	400	600
Disk latency (μs)	10,000	11,000	12,000
DRAM bandwidth (MB/s)	20,000	100	10
Flash bandwidth (MB/s)	1000	100	10
Disk bandwidth (MB/s)	200	100	10
DRAM capacity (GB)	16	1024	31,200
Flash capacity (GB)	128	20,000	600,000
Disk capacity (GB)	2000	160,000	4,800,000

WSC Memory Hierarchy

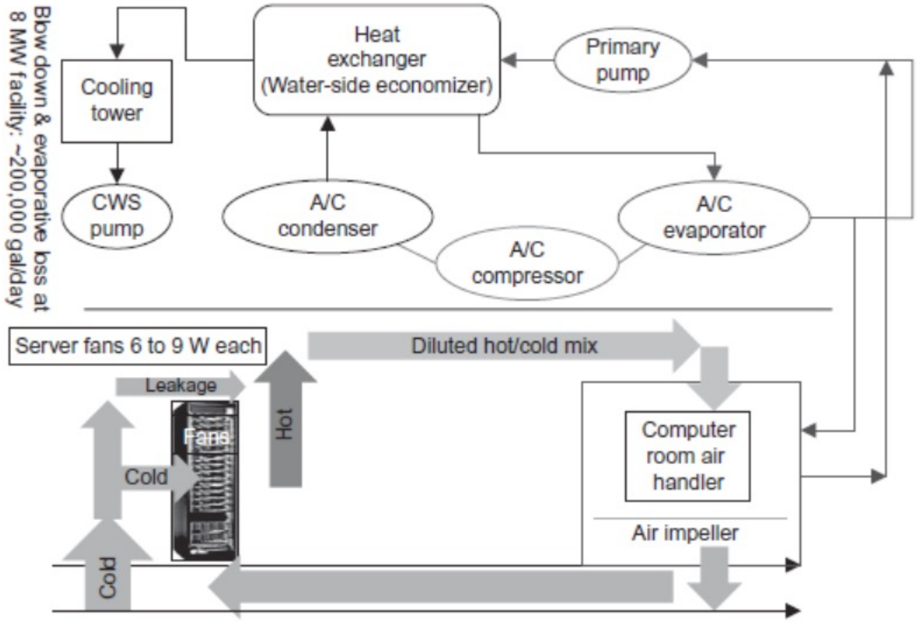


WSC Memory Hierarchy



Infrastructure and Costs of WSC

- Cooling
  - Air conditioning used to cool server room
  - 64 F – 71 F
    - Keep temperature higher (closer to 71 F)
  - Cooling towers can also be used
    - Minimum temperature is "wet bulb temperature"



Infrastructure and Costs of WSC

- Cooling system also uses water (evaporation and spills)
  - E.g. 70,000 to 200,000 gallons per day for an 8 MW facility
- Power cost breakdown:
  - Chillers: 30-50% of the power used by the IT equipment
  - Air conditioning: 10-20% of the IT power, mostly due to fans
- How man servers can a WSC support?
  - Each server:
    - "Nameplate power rating" gives maximum power consumption
    - To get actual, measure power under actual workloads
  - Oversubscribe cumulative server power by 40%, but monitor power closely

Infrastructure and Costs of WSC

- Determining the maximum server capacity
  - Nameplate power rating: maximum power that a server can draw
  - Better approach: measure under various workloads
  - Oversubscribe by 40%

- Typical power usage by component:
  - Processors: 42%
  - DRAM: 12%
  - Disks: 14%
  - Networking: 5%
  - Cooling: 15%
  - Power overhead: 8%
  - Miscellaneous: 4%

Measuring Efficiency of a WSC

- Power Utilization Effectiveness (PEU)
  - = Total facility power / IT equipment power
  - Median PUE on 2006 study was 1.69

- Performance
  - Latency is important metric because it is seen by users
  - Bing study: users will use search less as response time increases
  - Service Level Objectives (SLOs)/Service Level Agreements (SLAs)
    - E.g. 99% of requests be below 100 ms

Measuring Efficiency of a WSC

Server delay (ms)	Increased time to next click (ms)	Queries/ user	Any clicks/ user	User satisfaction	Revenue/ user
50	–	–	–	–	–
200	500	–	−0.3%	−0.4%	–
500	1200	–	−1.0%	−0.9%	−1.2%
1000	1900	−0.7%	−1.9%	−1.6%	−2.8%
2000	3100	−1.8%	−4.4%	−3.8%	−4.3%

Cost of a WSC

- **Capital expenditures (CAPEX)**
  - **Cost to build a WSC**
  - **$9 to 13/watt**

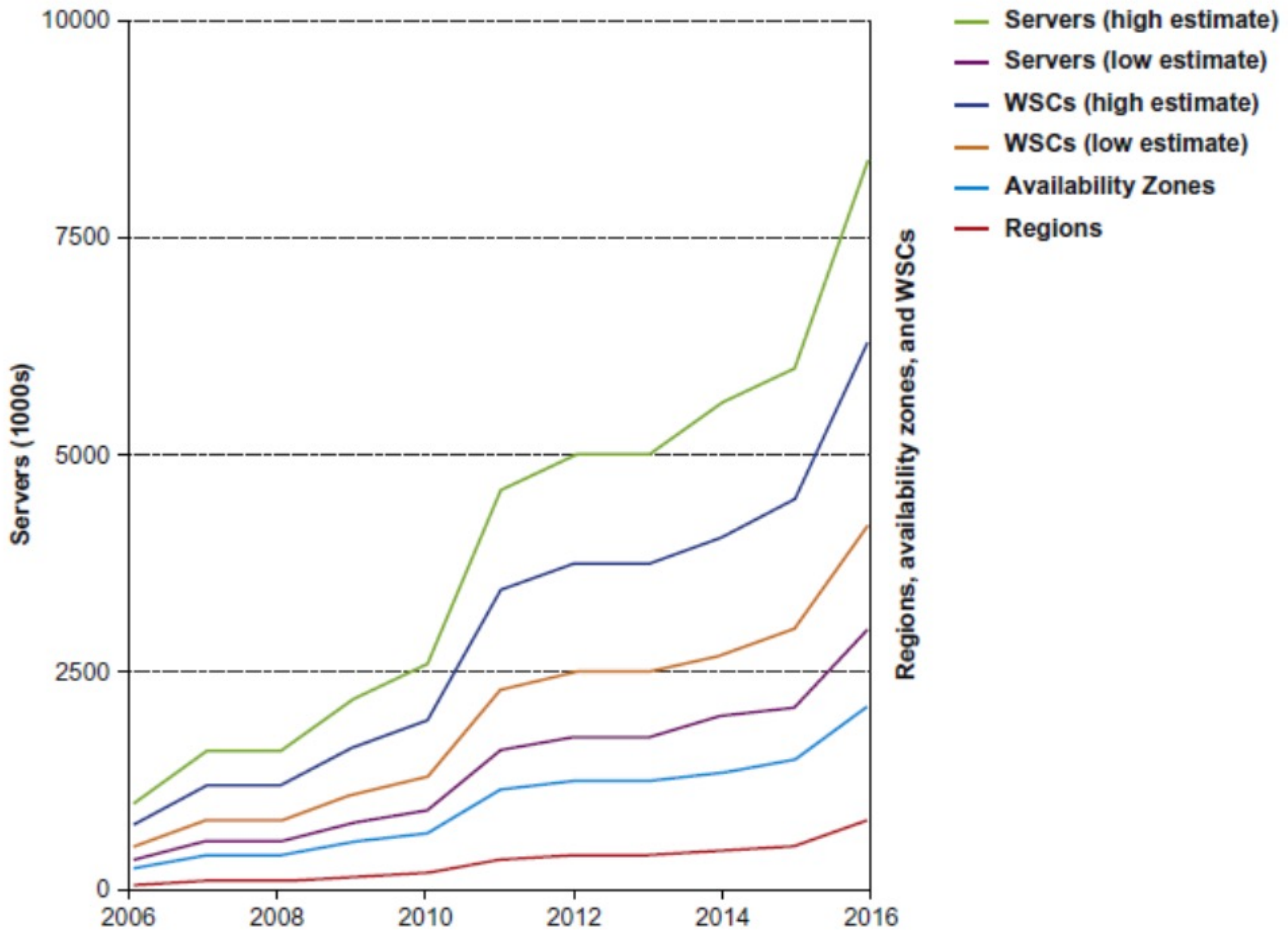
- **Operational expenditures (OPEX)**
  - **Cost to operate a WSC**

Cloud Computing

- Amazon Web Services
  - Virtual Machines: Linux/Xen
  - Low cost
  - Open source software
  - Initially no guarantee of service
  - No contract

Cloud Computing

- **Cloud Computing Growth**



Fallacies and Pitfalls

- Cloud computing providers are losing money
  - AWS has a margin of 25%, Amazon retail 3%
- Focusing on average performance instead of 99th percentile performance
- Using too wimpy a processor when trying to improve WSC cost-performance
- Inconsistent Measure of PUE by different companies
- Capital costs of the WSC facility are higher than for the servers that it houses

Fallacies and Pitfalls

- Trying to save power with inactive low power modes versus active low power modes
- Given improvements in DRAM dependability and the fault tolerance of WSC systems software, there is no need to spend extra for ECC memory in a WSC
- Coping effectively with microsecond (e.g. Flash and 100 GbE) delays as opposed to nansecond or millisecond delays
- Turning off hardware during periods of low activity improves the cost-performance of a WSC