IC-UNICAMP

MO401

IC/Unicamp

Prof Mario Côrtes

Capítulo 6

Request-Level and Data-Level Parallelism in Warehouse-Scale Computers

IC-UNICAMP

Tópicos

- Programming models and workload for Warehouse-Scale Computers
- Computer Architecture for Warehouse-Scale Computers
- Physical infrastructure and costs for Warehouse-Scale Computers
- Cloud computing: return of utility computing

IC-UNICAMP

Introduction

- Warehouse-scale computer (WSC)
  - Total cost (building, servers) $150M, 50k-100k servers
  - Provides Internet services
    - Search, social networking, online maps, video sharing, online shopping, email, cloud computing, etc.
  - 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
  - Differences with HPC “clusters”:
    - Clusters have higher performance processors and network
    - Clusters emphasize thread-level parallelism, WSCs emphasize request-level parallelism

IC-UNICAMP

Important design factors for WSC

- Requirements shared with servers
  - Cost-performance: work done / USD
    - Small savings add up → reducing 10% of capital cost → $15M
  - Energy efficiency: work / joule
    - Affects power distribution and cooling. Peak power affects cost.
  - Dependability via redundancy: > 99.99% → downtime/year = 1h
    - Beyond “four nines” → multiple WSC mask events that take out a WSC
  - Network I/O: with public and between multiple WSC
  - Interactive and batch processing workloads: search and Map-Reduce

IC-UNICAMP

Important design factors for WSC

- Requirements not shared with servers
  - Ample computational parallelism is not important
    - Most jobs are totally independent
    - DLP applied to storage; (in servers, to memory)
    - “Request-level parallelism”, SaaS, little need for communication/sync.
  - Operational costs count
    - Power consumption is a primary, not secondary, constraint when designing system. (em servidores, só preocupação do peak power não exceder specs)
    - Costs are amortized over 10+ years. Costs of energy, power, cooling > 30% total
  - Scale and its opportunities and problems
    - Opporunities: can afford to build customized systems since WSC require volume purchase (volume discounts)
    - Problems: flip side of 50000 servers is failure. Even with servers with MTTF = 25 years, a WSC could face 5 failures / day

IC-UNICAMP

Exmpl p 434: WSC availability

Approx. number events in 1st year	Cause	Consequence
1 or 2	Power utility failures	Lose power to whole WSC; doesn't bring down WSC if UPS and generators work (generators work about 99% of time).
4	Cluster upgrades	Planned outage to upgrade infrastructure, many times for evolving networking needs such as recabling, to switch firmware upgrades, and so on. There are about 9 planned cluster outages for every unplanned outage.
1000s	Hard-drive failures	2% to 10% annual disk failure rate [Pinheiro 2007]
	Slow disks	Still operate, but run 10x to 20x more slowly
	Bad memories	One uncorrectable DRAM error per year [Schroeder et al. 2009]
	Misconfigured machines	Configuration led to ~30% of service disruptions [Barroso and Hölzle 2009]
	Flaky machines	1% of servers reboot more than once a week [Barroso and Hölzle 2009]
5000	Individual server crashes	Machine reboot, usually takes about 5 minutes

Figure 6.1 List of outages and anomalies with the approximate frequencies of occurrences in the first year of a new cluster of 2400 servers. We label what Google calls a cluster an *array*; see Figure 6.5. (Based on Barroso [2010].)

IC-UNICAMP

Example Calculate the availability of a service running on the 2400 servers in Figure 6.1. Unlike a service in a real WSC, in this example the service cannot tolerate hardware or software failures. Assume that the time to reboot software is 5 minutes and the time to repair hardware is 1 hour.

Answer We can estimate service availability by calculating the time of outages due to failures of each component. We'll conservatively take the lowest number in each category in Figure 6.1 and split the 1000 outages evenly between four components. We ignore slow disks—the fifth component of the 1000 outages—since they hurt performance but not availability, and power utility failures, since the uninterruptible power supply (UPS) system hides 99% of them.

Exmpl p 434: WSC availability

$$\text{Hours Outage}_{\text{service}} = (4 + 250 + 250 + 250) \times 1 \text{ hour} + (250 + 5000) \times 5 \text{ minutes}$$

$$= 754 + 438 = 1192 \text{ hours}$$

Since there are 365×24 or 8760 hours in a year, availability is:

$$\text{Availability}_{\text{system}} = \frac{(8760 - 1192)}{8760} = \frac{7568}{8760} = 86\%$$

That is, without software redundancy to mask the many outages, a service on those 2400 servers would be down on average one day a week, or *zero* nines of availability!

IC-UNICAMP

Clusters and HPC vs WSC

- Computer clusters: forerunners of WSC
  - Independent computers, LAN, off-the-shelf switches
  - For workloads with low communication reqs, clusters are more cost-effective than Shared Memory Multiprocessors (forerunner of multicore)
  - Clusters became popular in late 90´s →100´s of servers → 10000´s of servers (WSC)
- HPC (High Performance Computing):
  - Cost and scale = similar to WSC
  - But: much faster processors and network. HPC applications are much more interdependent and have higher communication rate
  - Tend to use custom hw (power and cost of i7 > whole WSC server)
  - Long running jobs → servers fully occupied for weeks (WSC server utilization = 10% - 50%)

IC-UNICAMP

Datacenters vs WSC

- Datacenters
  - Collection of machines and 3rd party SW → run centralized for others
  - Main focus: consolidation of services in fewer isolated machines
    - Protection of sensitive info → virtualization increasingly important
  - HW and SW heterogeneity (WSC is homogeneous)
  - Largest cost is people to maintain it (WSC: server is top cost, people cost is irrelevant)
  - Scale not so large as WSC: no large scale cost benefits

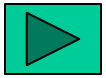
IC-UNICAMP

6.2 Prgrm'g Models and Workloads

- Most popular batch processing framework: MapReduce
  - Open source twin: Hadoop

	Aug-04	Mar-06	Sep-07	Sep-09
Number of MapReduce jobs	29,000	171,000	2,217,000	3,467,000
Average completion time (seconds)	634	874	395	475
Server years used	217	2002	11,081	25,562
Input data read (terabytes)	3288	52,254	403,152	544,130
Intermediate data (terabytes)	758	6743	34,774	90,120
Output data written (terabytes)	193	2970	14,018	57,520
Average number of servers per job	157	268	394	488

Figure 6.2 Annual MapReduce usage at Google over time. Over five years the number of MapReduce jobs increased by a factor of 100 and the average number of servers per job increased by a factor of 3. In the last two years the increases were factors of 1.6 and 1.2, respectively [Dean 2009]. Figure 6.16 on page 459 estimates that running the 2009 workload on Amazon's cloud computing service EC2 would cost $133M.



IC-UNICAMP

Prgrm'g Models and Workloads

- **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
- Example: calculation of # occurrences of every word in a large set of documents (here, assumes just one occurrence)
  - **map (String key, String value)**:
    - **// key: document name**
    - **// value: document contents**
    - **for each word w in value**
      - **EmitIntermediate(w,"1"); // produz lista de todas palavras /doc e contagem**
  - **reduce (String key, Iterator values):**
    - **// key: a word**
    - **// value: a list of counts**
    - **int result = 0;**
    - **for each v in values:**
      - **result += ParseInt(v); // soma contagem em todos os documentos**
    - **Emit(AsString(result));**

IC-UNICAMP

Prgrm'g Models and Workloads

- **MapReduce runtime environment schedules map and reduce task to WSC nodes**
  - **Towards the end of MapReduce, system starts backup executions on free nodes → take results from whichever finishes first**
- **Availability:**
  - **Use replicas of data across different servers**
  - **Use relaxed consistency:**
    - **No need for all replicas to always agree**
- **Workload demands**
  - **Often vary considerably**
    - **ex: Google, daily, holidays, weekends (fig 6.3)**

Google: CPU utilization distribution

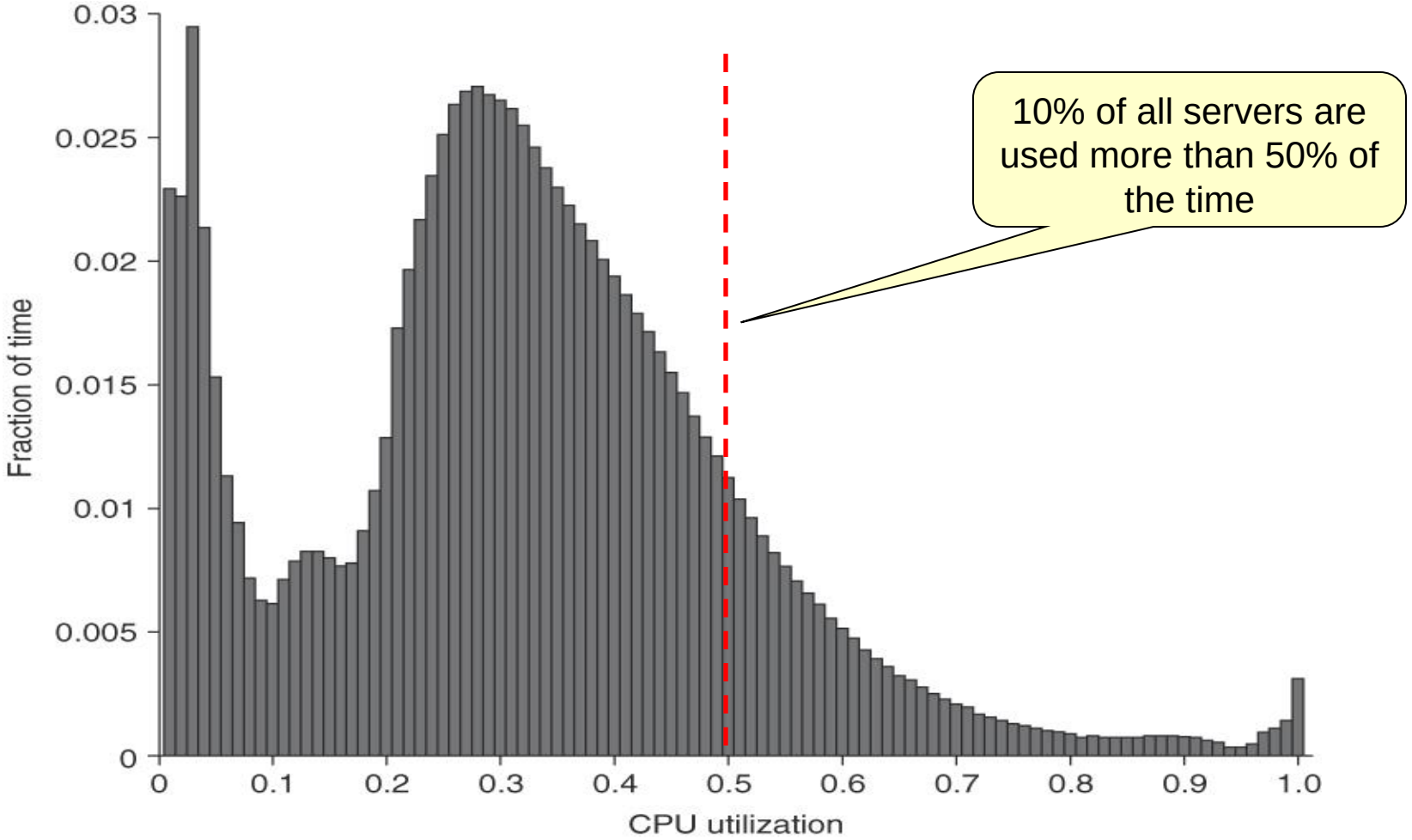


Figure 6.3 Average CPU utilization of more than 5000 servers during a 6-month period at Google. Servers are rarely completely idle or fully utilized, instead operating most of the time at between 10% and 50% of their maximum utilization. (From Figure 1 in Barroso and Hölzle [2007].) The column the third from the right in Figure 6.4 calculates percentages plus or minus 5% to come up with the weightings; thus, 1.2% for the 90% row means that 1.2% of servers were between 85% and 95% utilized.

IC-UNICAMP

Exmpl p 439: weighted performance

Example As a result of measurements like those in Figure 6.3, the SPECPower benchmark measures power and performance from 0% load to 100% in 10% increments (see Chapter 1). The overall single metric that summarizes this benchmark is the sum of all the performance measures (server-side Java operations per second) divided by the sum of all power measurements in watts. Thus, each level is equally likely. How would the numbers summary metric change if the levels were weighted by the utilization frequencies in Figure 6.3?

Answer Figure 6.4 shows the original weightings and the new weighting that match Figure 6.3. These weightings reduce the performance summary by 30% from 3210 ssj_ops/watt to 2454.

Load	Performance	Watts	SPEC weightings	Weighted performance	Weighted watts	Figure 6.3 weightings	Weighted performance	Weighted watts
100%	2,889,020	662	9.09%	262,638	60	0.80%	22,206	5
90%	2,611,130	617	9.09%	237,375	56	1.20%	31,756	8
80%	2,319,900	576	9.09%	210,900	52	1.50%	35,889	9
70%	2,031,260	533	9.09%	184,660	48	2.10%	42,491	11
60%	1,740,980	490	9.09%	158,271	45	5.10%	88,082	25
50%	1,448,810	451	9.09%	131,710	41	11.50%	166,335	52
40%	1,159,760	416	9.09%	105,433	38	19.10%	221,165	79
30%	869,077	382	9.09%	79,007	35	24.60%	213,929	94
20%	581,126	351	9.09%	52,830	32	15.30%	88,769	54
10%	290,762	308	9.09%	26,433	28	8.00%	23,198	25
0%	0	181	9.09%	0	16	10.90%	0	20
Total	15,941,825	4967		1,449,257	452		933,820	380
				ssj_ops/Watt	3210		ssj_ops/Watt	2454

Figure 6.4 SPECPower result from Figure 6.17 using the weightings from Figure 6.3 instead of even weightings.

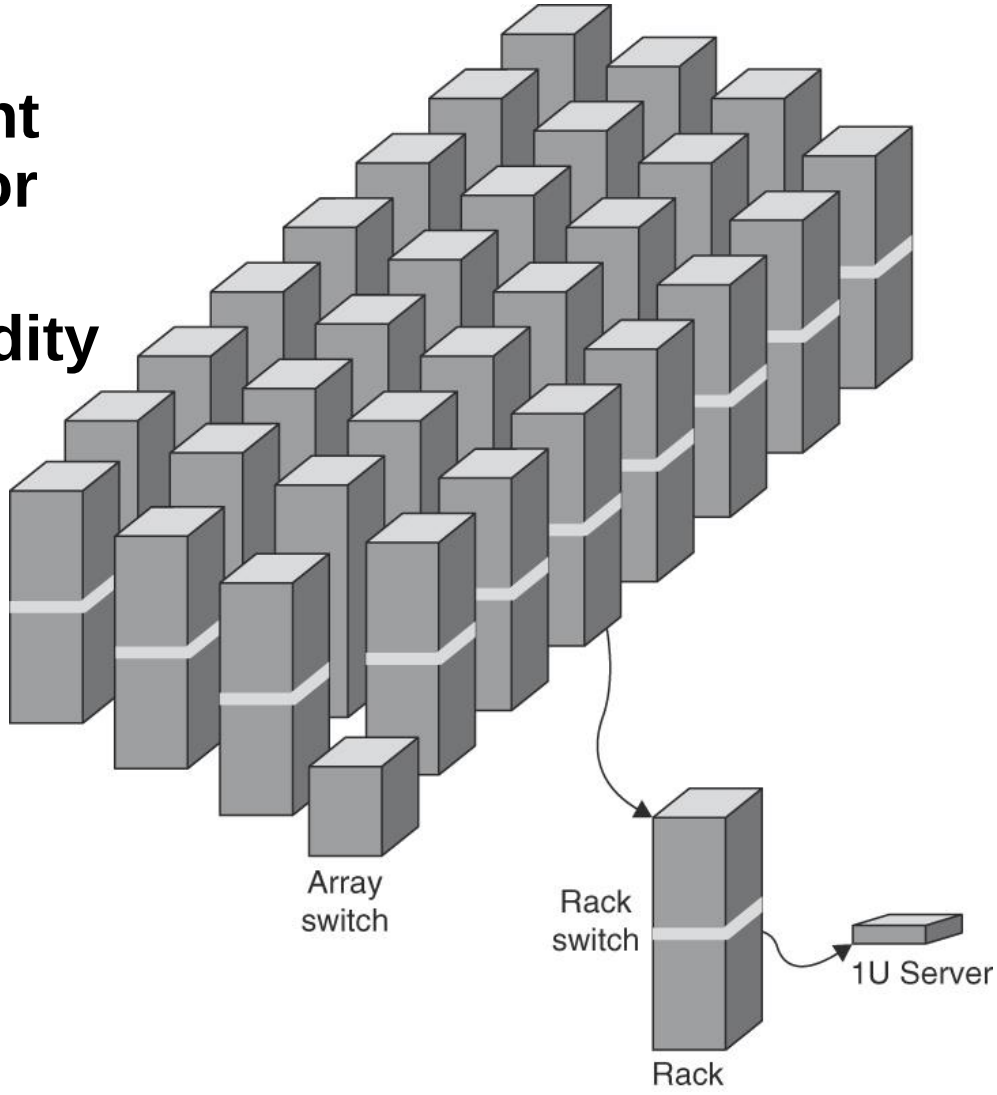
IC-UNICAMP

6.3 Computer Architecture of WSC

- **WSC often use a hierarchy of networks for interconnection**
- **Standard framework to hold servers: 19” rack**
  - **Servers measured in # rack units (U) they occupy in a rack. One U is 1.75” high**
  - **7-foot rack → 48 U (popular 48-port Ethernet switch); $30/port**
- **Switches offer 2-8 uplinks (higher hierarchy level)**
  - **BW leaving the rack is 6-24 x smaller (48/8 – 48/2) than BW within the rack (this ratio is called “Oversubscription”)**
- **Goal is to maximize locality of communication relative to the rack**
  - **Communication between different racks → penalty**

Fig 6.5: hierarchy of switches in a WSC

- **Ideally: network performance equivalent to a high-end switch for 50k servers**
- **Cost per port: commodity switch designed for 50 servers**



IC-UNICAMP

Storage

- **Natural design: fill the rack with servers + Ethernet switch; Storage??**
- **Storage options:**
  - **Use disks inside the servers, or**
  - **Network attached storage (remote servers) through Infiniband**
- **WSCs generally rely on local disks**
  - **Google File System (GFS) uses local disks and maintains at least three replicas → covers failures in local disk, power, racks and clusters**
- **Cluster (terminology)**
  - **Definition in sec 6.1: WSC = very large cluster**
  - **Barroso: next-sized grouping of computers, ~30 racks**
  - **In this chapter:**
    - **array: collection of racks**
    - **cluster: original meaning → anything from a collection of networked computers within a rack to an entire WSC**

IC-UNICAMP

Array Switch

- **Switch that connects an array of racks**
- **Much more expensive than a 48-port Ethernet switch**
- **Array switch should have 10 X the bisection bandwidth of rack switch → cost is 100x**
  - **bisection BW: dividir a rede em duas metades (pior caso) e medir BW entre elas (ex: 4x8 2D mesh)**
- **Cost of *n*-port switch grows as n^2**
- **Often utilize content addressable memory chips and FPGAs**
  - **packet inspection at high rates**

IC-UNICAMP

WSC Memory Hierarchy

- **Servers can access DRAM and disks on other servers using a NUMA-style interface**
  - **Each server: Memory =16 GB, 100ns access time, 20 GB/s; Disk = 2 TB, 10 ms access time, 200 MB/s. Comm = 1 Gbit/s Ethernet port.**
  - **Pair of racks: 1 rack switch, 80 2U servers; Overhead increases DRAM latency to 100 μs, disk latency to 11 ms. Total capacity: 1 TB of DRAM + 160 TB of disk. Comm = 100 MB/s**
  - **Array switch: 30 racks. Capacity = 30 TB of DRAM + 4.8 pB of disk. Overhead increases DRAM latency to 500 μs, disk latency to 12 ms. Comm = 10 MB/s**

	Local	Rack	Array
DRAM latency (microseconds)	0.1	100	300
Disk latency (microseconds)	10,000	11,000	12,000
DRAM bandwidth (MB/sec)	20,000	100	10
Disk bandwidth (MB/sec)	200	100	10
DRAM capacity (GB)	16	1,040	31,200
Disk capacity (GB)	2000	160,000	4,800,000

Fig 6.7: WSC memory hierarchy numbers

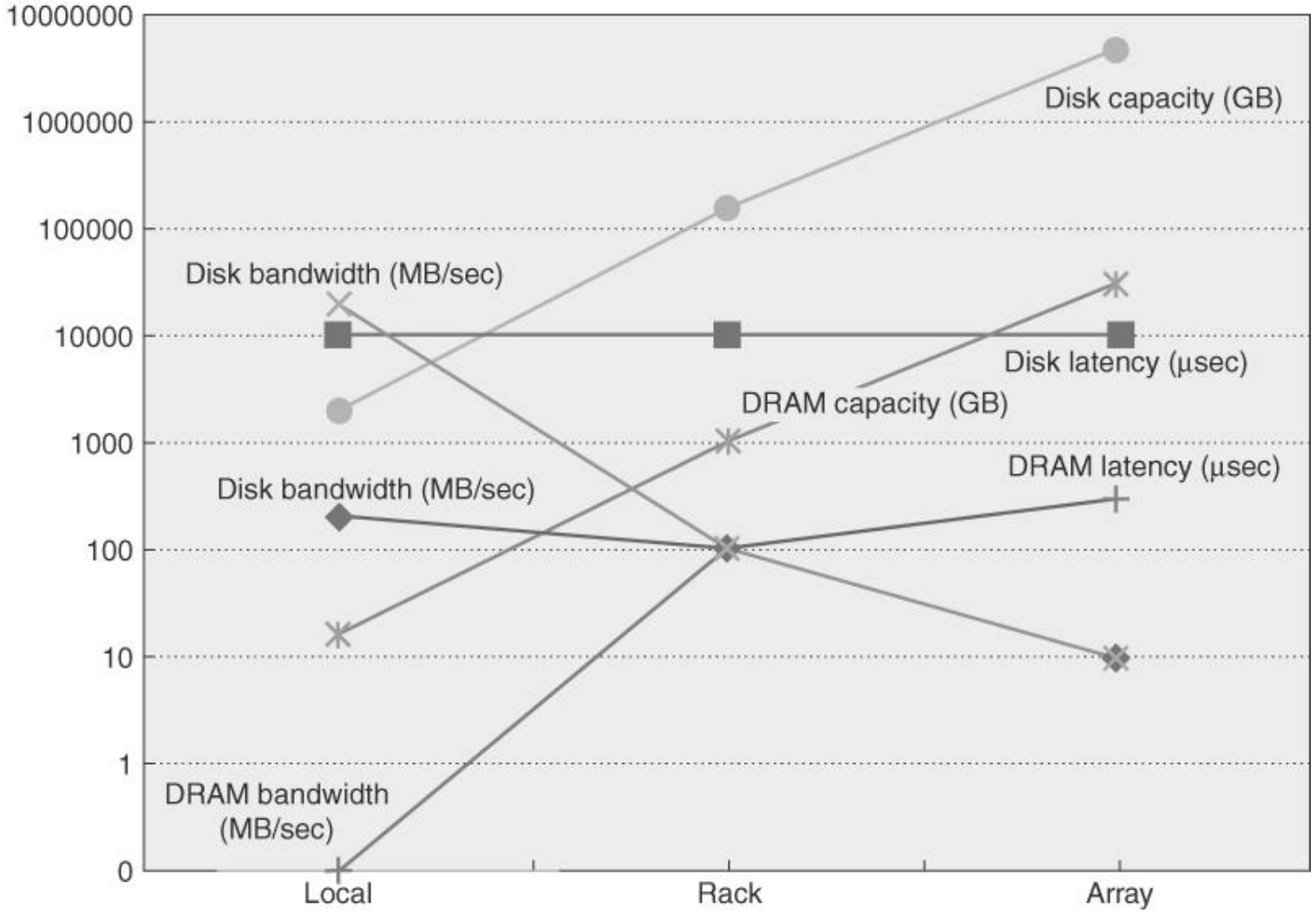


Fig 6.8: WSC hierarchy

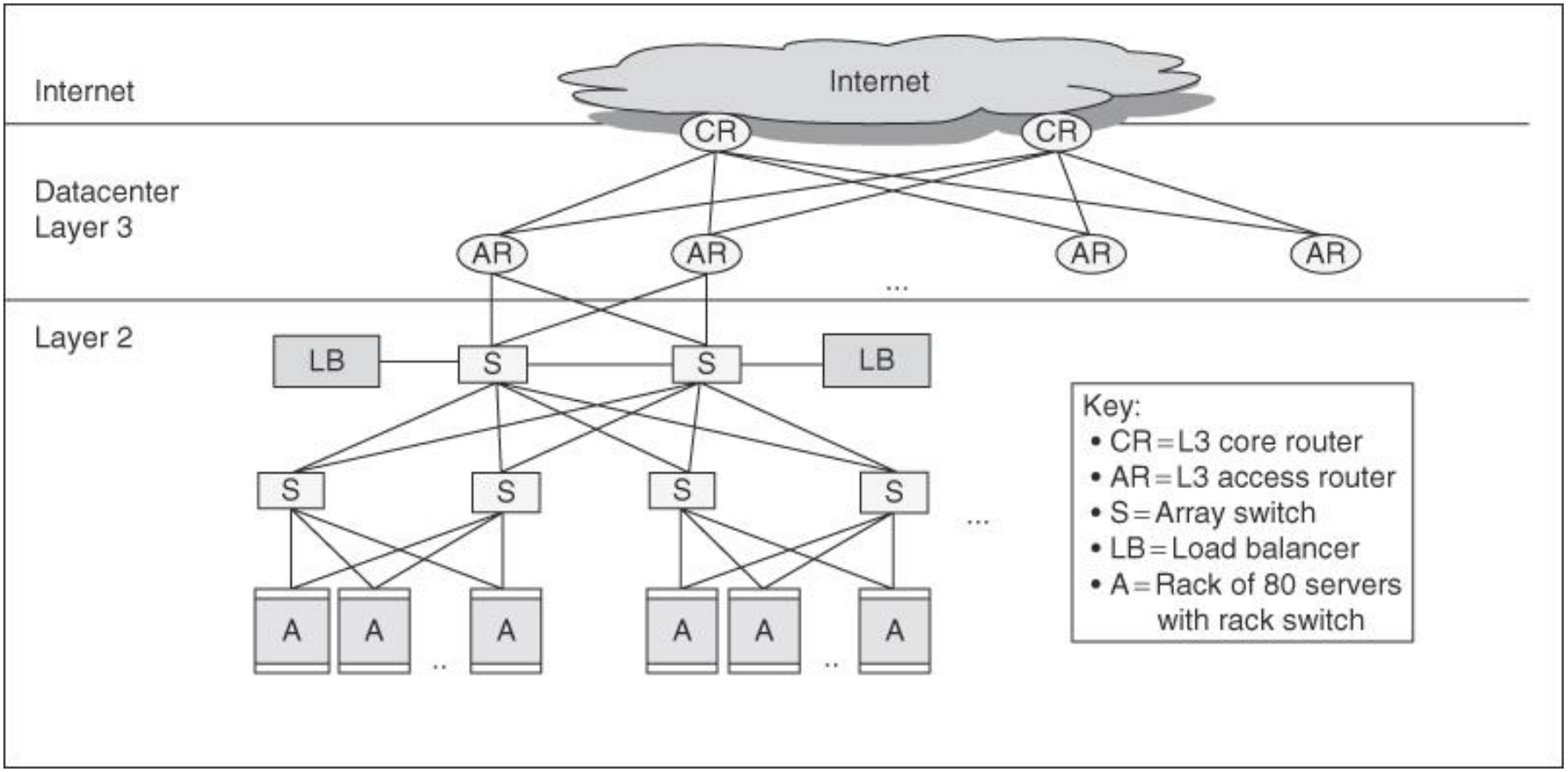


Figure 6.8 The Layer 3 network used to link arrays together and to the Internet [Greenberg et al. 2009]. Some WSCs use a separate *border router* to connect the Internet to the datacenter Layer 3 switches.

IC-UNICAMP

Exmpl p445: WSC average memory latency

Example What is the average memory latency assuming that 90% of accesses are local to the server, 9% are outside the server but within the rack, and 1% are outside the rack but within the array?

Answer The average memory access time is

$$(90\% \times 0.1) + (9\% \times 100) + (1\% \times 300) = 0.09 + 9 + 3 = 12.09 \text{ microseconds}$$

or a factor of more than 120 slowdown versus 100% local accesses. Clearly, locality of access within a server is vital for WSC performance.

IC-UNICAMP

Exmpl p446: WSC data transfer time

Example How long does it take to transfer 1000 MB between disks within the server, between servers in the rack, and between servers in different racks in the array? How much faster is it to transfer 1000 MB between DRAM in the three cases?

Answer A 1000 MB transfer between disks takes:

Within server = 1000/200 = 5 seconds

Within rack = 1000/100 = 10 seconds

Within array = 1000/10 = 100 seconds

A memory-to-memory block transfer takes

Within server = 1000/20000 = 0.05 seconds

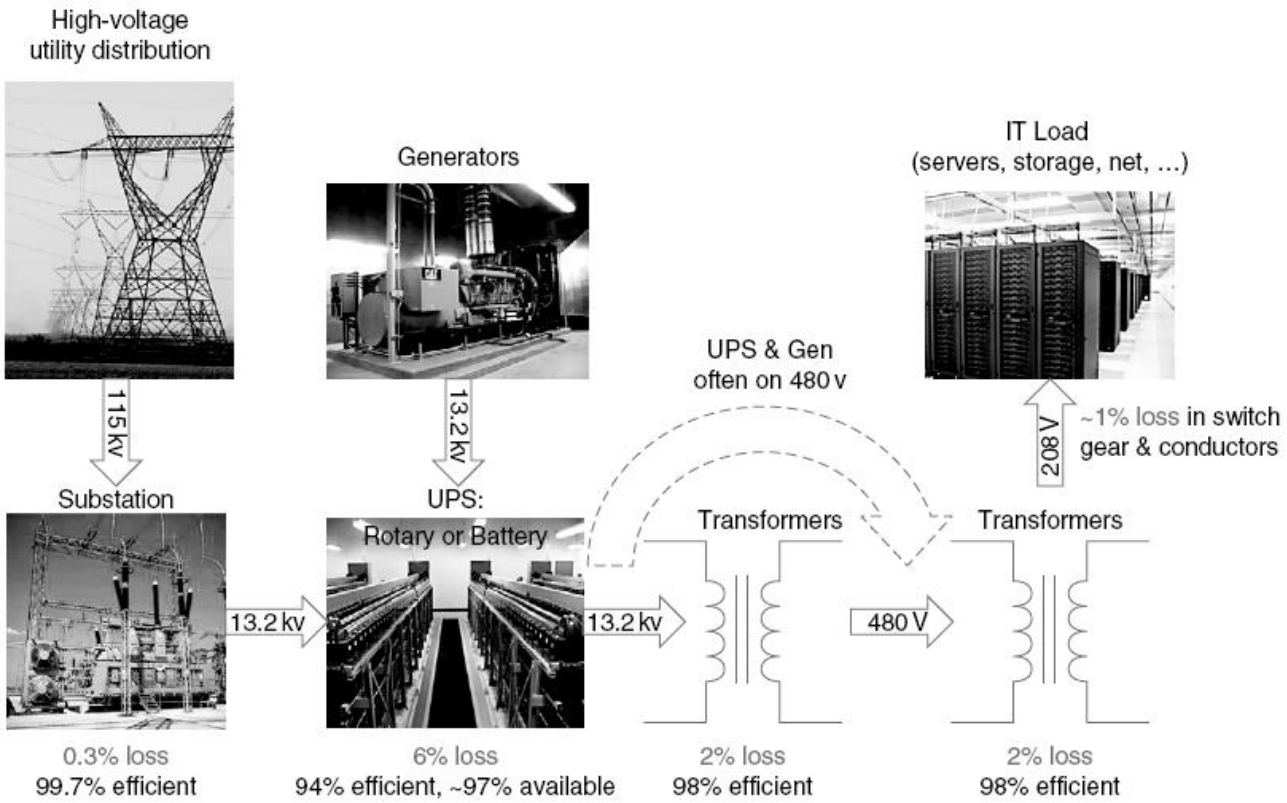
Within rack = 1000/100 = 10 seconds

Within array = 1000/10 = 100 seconds

Thus, for block transfers outside a single server, it doesn't even matter whether the data are in memory or on disk since the rack switch and array switch are the bottlenecks. These performance limits affect the design of WSC software and inspire the need for higher performance switches (see Section 6.6).

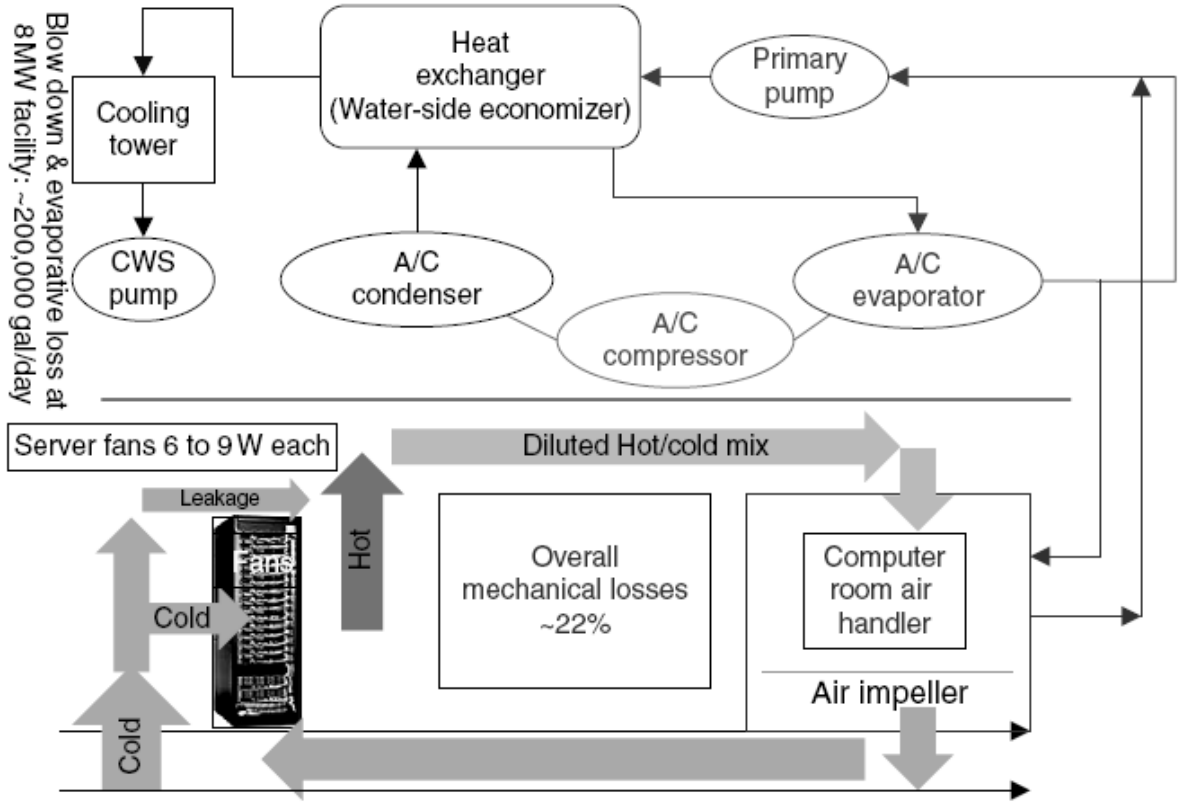
6.4 Infrastructure and Costs of WSC

- Location of WSC
  - Proximity to Internet backbones, electricity cost, property tax rates, low risk from earthquakes, floods, and hurricanes
- Power distribution: combined efficiency = 89%



Infrastructure and Costs of WSC

- Cooling
  - Air conditioning used to cool server room
  - 64 F – 71 F (18ºC – 22ºC)
    - Keep temperature higher (closer to 71 F)
  - Cooling towers can also be used: Minimum temperature is “wet bulb temperature”



IC-UNICAMP

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 → deschedule lower priority tasks in case workload shifts
- Power components:
  - processors 33%, DRAM 30%, disks 10%, networking 5%, others 22%

IC-UNICAMP

Measuring Efficiency of a WSC

- Power Utilization Effectiveness (PUE)
- Performance

Power utilization effectiveness

- Power Utilization Effectiveness (PUE)

  = Total facility power / IT equipment power

- PUE
  - always >1
  - ideal =1

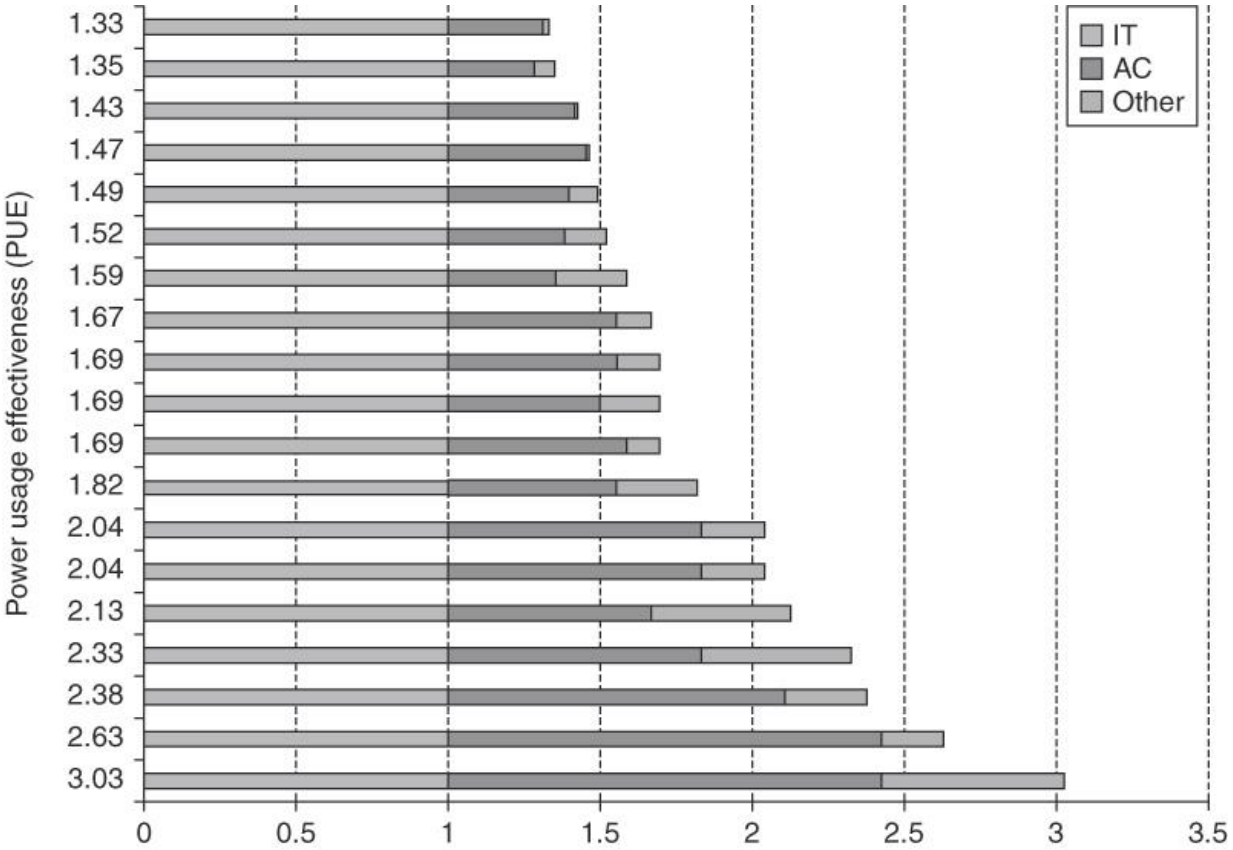


Figure 6.11 Power utilization efficiency of 19 datacenters in 2006 [Greenberg et al. 2006]. The power for air conditioning (AC) and other uses (such as power distribution) is normalized to the power for the IT equipment in calculating the PUE. Thus, power for IT equipment must be 1.0 and AC varies from about 0.30 to 1.40 times the power of the IT equipment. Power for "other" varies from about 0.05 to 0.60 of the IT equipment. Median = 1.69

IC-UNICAMP

Measuring Performance efficiency of a WSC

- Latency is important metric because it is seen by users
  - experimental data: cutting system response time in 30% → average interaction time reduced by 70% (people have less time to think with fast responses; people less likely to get distracted)
- Bing study: users will use search less as response time increases

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%

Figure 6.12 Negative impact of delays at Bing search server on user behavior Schurman and Brutlag [2009].

- Service Level Objectives (SLOs)/Service Level Agreements (SLAs)
  - E.g. 99% of requests be below 100 ms

IC-UNICAMP

Cost of a WSC

- Capital expenditures (CAPEX)
  - Cost to build a WSC
- Operational expenditures (OPEX)
  - Cost to operate a WSC

IC-UNICAMP

6.5 Cloud Computing (as utility)

- WSCs offer economies of scale that cannot be achieved with a datacenter:
  - 5.7 times reduction in storage costs: $4.6 / GB (WSC)
  - 7.1 times reduction in administrative costs: 1000 server / administrator
  - 7.3 times reduction in networking costs: $13 / (MB/s . month)
  - This has given rise to cloud services such as Amazon Web Services
    - “Utility Computing”
    - Based on using open source virtual machine and operating system software
- Scale: discount prices os servers and networking (Dell, IBM)
- PUE: 1.2 (WSC) vs 2.0 (Datacenters)
- Internet services: much more expensive for individual firms to create multiple, small datacenters around the world
- HW utilization: 10% (Datacenters) → 50% (WSC)

IC-UNICAMP

Case: AWS – Amazon Web Services

- 2006: Amazon started S3 (Amazon Simple Storage Service) and EC2 (Amazon Elastic Computer Cloud)
  - Virtual machines: x86 commodity computers + Linux + Xen virtual machine solved several problems:
    - protection of users from each other
    - software distribution: customers install an image, AWS automatically distribute it to all instances
    - ability to kill a virtual machine → resource usage control
    - multiple price points per virtual machine: different VM configurations (processors, disk, network….)
    - hiding (and using) older hardware, that could be unattractive to users if they know
    - flexibility in packing cores (more or less) per VM
  - Very low cost: in 2006, $0.10 / hour per instance !! (low end = 2 instances / core)

IC-UNICAMP

Case: AWS – Amazon Web Services (cont)

- Initial reliance on open source SW: → lower price
  - Recently, AWS offers instances with 3rd party SW, at higher $
- No (initial) guarantee of service. Initially, AWS offered only best effort (but cost so low, one could live with it).
  - Today, SLA of 99.95%.
  - Amazon S3 was designed for 99.999999999% durability. Chances of loosing an object → 1 in 100 billion
- No contract required

IC-UNICAM

Instance	Per hour	Ratio to small	Compute units	Virtual cores	Compute units/core	Memory (GB)	Disk (GB)	Address size
Micro	$0.020	0.5–2.0	0.5–2.0	1	0.5–2.0	0.6	EBS	32/64 bit
Standard Small	$0.085	1.0	1.0	1	1.00	1.7	160	32 bit
Standard Large	$0.340	4.0	4.0	2	2.00	7.5	850	64 bit
Standard Extra Large	$0.680	8.0	8.0	4	2.00	15.0	1690	64 bit
High-Memory Extra Large	$0.500	5.9	6.5	2	3.25	17.1	420	64 bit
High-Memory Double Extra Large	$1.000	11.8	13.0	4	3.25	34.2	850	64 bit
High-Memory Quadruple Extra Large	$2.000	23.5	26.0	8	3.25	68.4	1690	64 bit
High-CPU Medium	$0.170	2.0	5.0	2	2.50	1.7	350	32 bit
High-CPU Extra Large	$0.680	8.0	20.0	8	2.50	7.0	1690	64 bit
Cluster Quadruple Extra Large	$1.600	18.8	33.5	8	4.20	23.0	1690	64 bit

Figure 6.15 Price and characteristics of on-demand EC2 instances in the United States in the Virginia region in January 2011. Micro Instances are the newest and cheapest category, and they offer short bursts of up to 2.0 compute units for just $0.02 per hour. Customers report that Micro Instances average about 0.5 compute units. Cluster-Compute Instances in the last row, which AWS identifies as dedicated dual-socket Intel Xeon X5570 servers with four cores per socket running at 2.93 GHz, offer 10 Gigabit/sec networks. They are intended for HPC applications. AWS also offers Spot Instances at much less cost, where you set the price you are willing to pay and the number of instances you are willing to run, and then AWS will run them when the spot price drops below your level. They run until you stop them or the spot price exceeds your limit. One sample during the daytime in January 2011 found that the spot price was a factor of 2.3 to 3.1 lower, depending on the instance type. AWS also offers Reserved Instances for cases where customers know they will use most of the instance for a year. You pay a yearly fee per instance and then an hourly rate that is about 30% of column 1 to use it. If you used a Reserved Instance 100% for a whole year, the average cost per hour including amortization of the annual fee would be about 65% of the rate in the first column. The server equivalent to those in Figures 6.13 and 6.14 would be a Standard Extra Large or High-CPU Extra Large Instance, which we calculated to cost $0.11 per hour.

IC-UNICAMP

Exmpl p 458: cost of MapReduce jobs

Example Calculate the cost of running the average MapReduce jobs in Figure 6.2 on page 437 on EC2. Assume there are plenty of jobs, so there is no significant extra cost to round up so as to get an integer number of hours. Ignore the monthly storage costs, but include the cost of disk I/Os for AWS's Elastic Block Storage (EBS). Next calculate the cost per year to run all the MapReduce jobs.

Answer The first question is what is the right size instance to match the typical server at Google? Figure 6.21 on page 467 in Section 6.7 shows that in 2007 a typical Google server had four cores running at 2.2 GHz with 8 GB of memory. Since a single instance is one virtual core that is equivalent to a 1 to 1.2 GHz AMD Opteron, the closest match in Figure 6.15 is a High-CPU Extra Large with eight virtual cores and 7.0 GB of memory. For simplicity, we'll assume the average EBS storage access is 64 KB in order to calculate the number of I/Os.

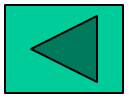
Figure 6.16 calculates the average and total cost per year of running the Google MapReduce workload on EC2. The average 2009 MapReduce job would cost a little under $40 on EC2, and the total workload for 2009 would cost $133M on AWS. Note that EBS accesses are about 1% of total costs for these jobs.

Exmpl p 458: cost of MapReduce jobs (cont)

IC-UNICAMP

	Aug-04	Mar-06	Sep-07	Sep-09
Average completion time (hours)	0.15	0.21	0.10	0.11
Average number of servers per job	157	268	394	488
Cost per hour of EC2 High-CPU XL instance	$0.68	$0.68	$0.68	$0.68
Average EC2 cost per MapReduce job	$16.35	$38.47	$25.56	$38.07
Average number of EBS I/O requests (millions)	2.34	5.80	3.26	3.19
EBS cost per million I/O requests	$0.10	$0.10	$0.10	$0.10
Average EBS I/O cost per MapReduce job	$0.23	$0.58	$0.33	$0.32
Average total cost per MapReduce job	$16.58	$39.05	$25.89	$38.39
Annual number of MapReduce jobs	29,000	171,000	2,217,000	3,467,000
Total cost of MapReduce jobs on EC2/EBS	$480,910	$6,678,011	$57,394,985	$133,107,414

Figure 6.16 Estimated cost if you ran the Google MapReduce workload (Figure 6.2) using 2011 prices for AWS ECS and EBS (Figure 6.15). Since we are using 2011 prices, these estimates are less accurate for earlier years than for the more recent ones.



IC-UNIC

Example Given that the costs of MapReduce jobs are growing and already exceed $100M per year, imagine that your boss wants you to investigate ways to lower costs. Two potentially lower cost options are either AWS Reserved Instances or AWS Spot Instances. Which would you recommend?

Answer AWS Reserved Instances charge a fixed annual rate plus an hourly per-use rate. In 2011, the annual cost for the High-CPU Extra Large Instance is $1820 and the hourly rate is $0.24. Since we pay for the instances whether they are used or not, let's assume that the average utilization of Reserved Instances is 80%. Then the average price per hour becomes:

$$\frac{\frac{\text{Annual price}}{\text{Hours per year}} + \text{Hourly price}}{\text{Utilization}} = \frac{\frac{\$1820}{8760} + \$0.24}{80\%} = (0.21 + 0.24) \times 1.25 = \$0.56$$

Thus, the savings using Reserved Instances would be roughly 17% or $23M for the 2009 MapReduce workload.

Sampling a few days in January 2011, the hourly cost of a High-CPU Extra Large Spot Instance averages $0.235. Since that is the minimum price to bid to get one server, that cannot be the average cost since you usually want to run tasks to completion without being bumped. Let's assume you need to pay double the minimum price to run large MapReduce jobs to completion. The cost savings for Spot Instances for the 2009 workload would be roughly 31% or $41M.

Thus, you tentatively recommend Spot Instances to your boss since there is less of an up-front commitment and they may potentially save more money. However, you tell your boss you need to try to run MapReduce jobs on Spot Instances to see what you actually end up paying to ensure that jobs run to completion and that there really are hundreds of High-CPU Extra Large Instances available to run these jobs daily.

IC-UNICAMP

Examples of use (p 460)

- Farm Ville (Zynga): 1 million players 4 days after lauch, 10 million after 60 days, 60 millions after 270 days
  - deployed on AWS: seamless growth of number of users
- NetFlix video streaming: 2011, conventional datacenter → AWS
  - ability to switch video format of a film (cell phone → TV) → heavy conversion batch processing
  - today, Netflix is responsible for 30% of download traffic at peak evening hours

6.6 Crosscutting issues

- WSC Network as a bottleneck
  - 2nd level networking gear is significant fraction of WSC cost: 128-port 1 Gb datacenter switch (EX8216) = $716,000
  - Power hungry: EX8216 consumes 500-1000 x a server
  - Manually configured manufactured → fragile. But because of high price, difficult to afford redundancy → limited fault tolerance
- Using energy efficiently inside the server
  - PUE: WSC power efficiency. But, inside one server?
  - Power supply has low efficiency: lots of conversion, oversized, worst efficiency at (normal) 25% load
  - Climate Savers Computing Initiative: Bronze, Silver, Gold power supplies (fig 6.17)
  - Goal should be “energy proportionality” → energy should be proportional to work performed (fig 6.18, next slide)

Energy proportionality

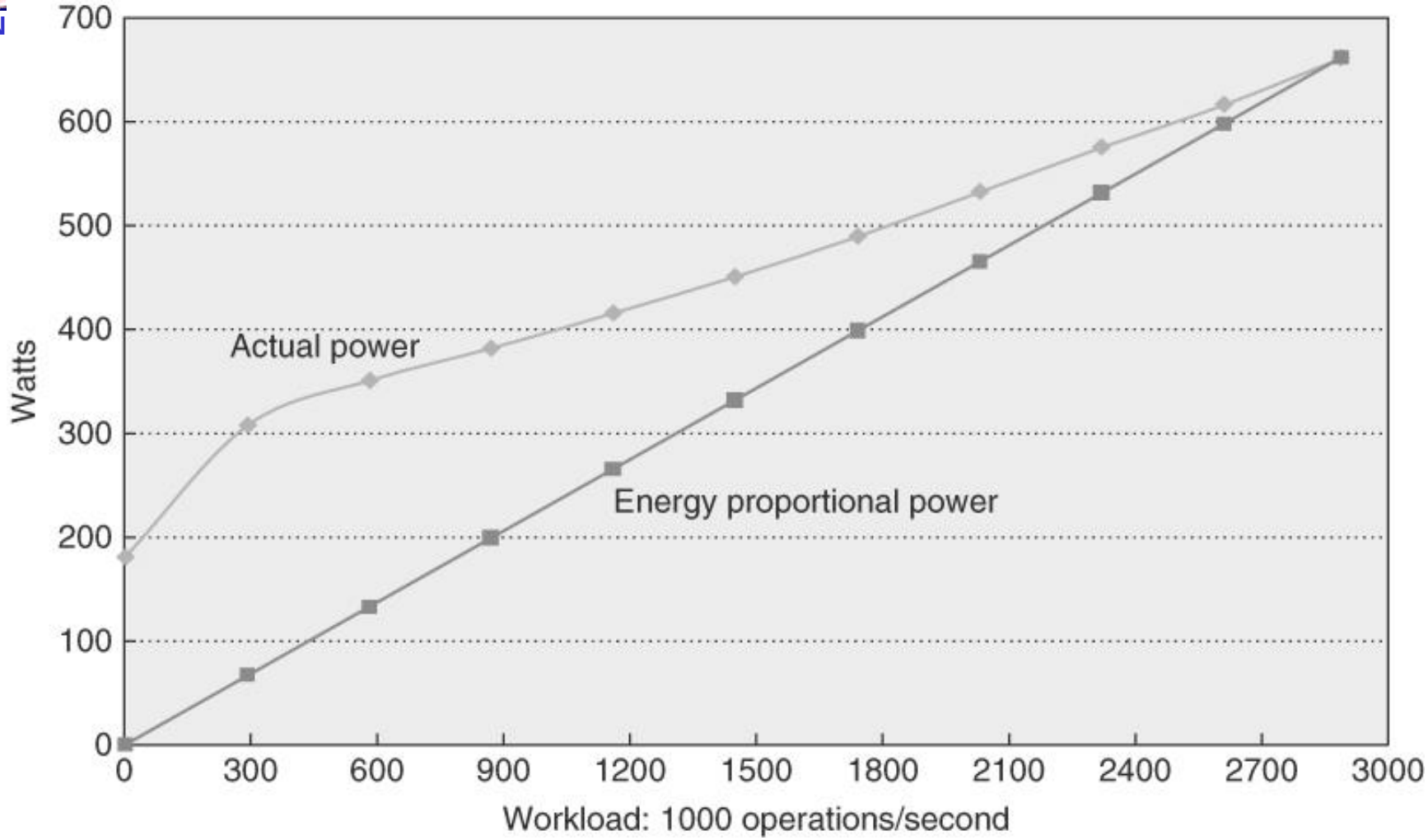


Figure 6.18 The best SPECpower results as of July 2010 versus the ideal energy proportional behavior. The system was the HP ProLiant SL2x170z G6, which uses a cluster of four dual-socket Intel Xeon L5640s with each socket having six cores running at 2.27 GHz. The system had 64 GB of DRAM and a tiny 60 GB SSD for secondary storage. (The fact that main memory is larger than disk capacity suggests that this system was tailored to this benchmark.) The software used was IBM Java Virtual Machine version 9 and Windows Server 2008, Enterprise Edition.

IC-UNICAMP

Exmpl p 463: energy proportionality

Example Using the data of the kind in Figure 6.18, what is the saving in power going from five servers at 10% utilization versus one server at 50% utilization?

Answer A single server at 10% load is 308 watts and at 50% load is 451 watts. The savings is then

$$5 \times 308/451 = (1540/451) \approx 3.4$$

or about a factor of 3.4. If we want to be good environmental stewards in our WSC, we must consolidate servers when utilizations drop, purchase servers that are more energy proportional, or find something else that is useful to run in periods of low activity.

IC-UNICAMP

6.7 Putting all together: Google WSC

- Data from 2005, updated on 2007
- Container based WSC (Google and Microsoft): modular
  - external connections: networking, power, chilled water
- Google WSC: 45 containers in a 7000m^2 warehouse (15 stacks of 2 containers + 15)
  - location: unknown
- Power 10 MW, with PUE = 1.23
  - 0.23 PUE overhead: 85% (cooling) + 15% (power losses)
  - 250 KW / container

Google container

- 1160 servers
- 45 containers → 52,200 servers
- Servers stacked 20 high = 2 rows of 29 racks
- Rach switch: 48-port, 1 Gb/s

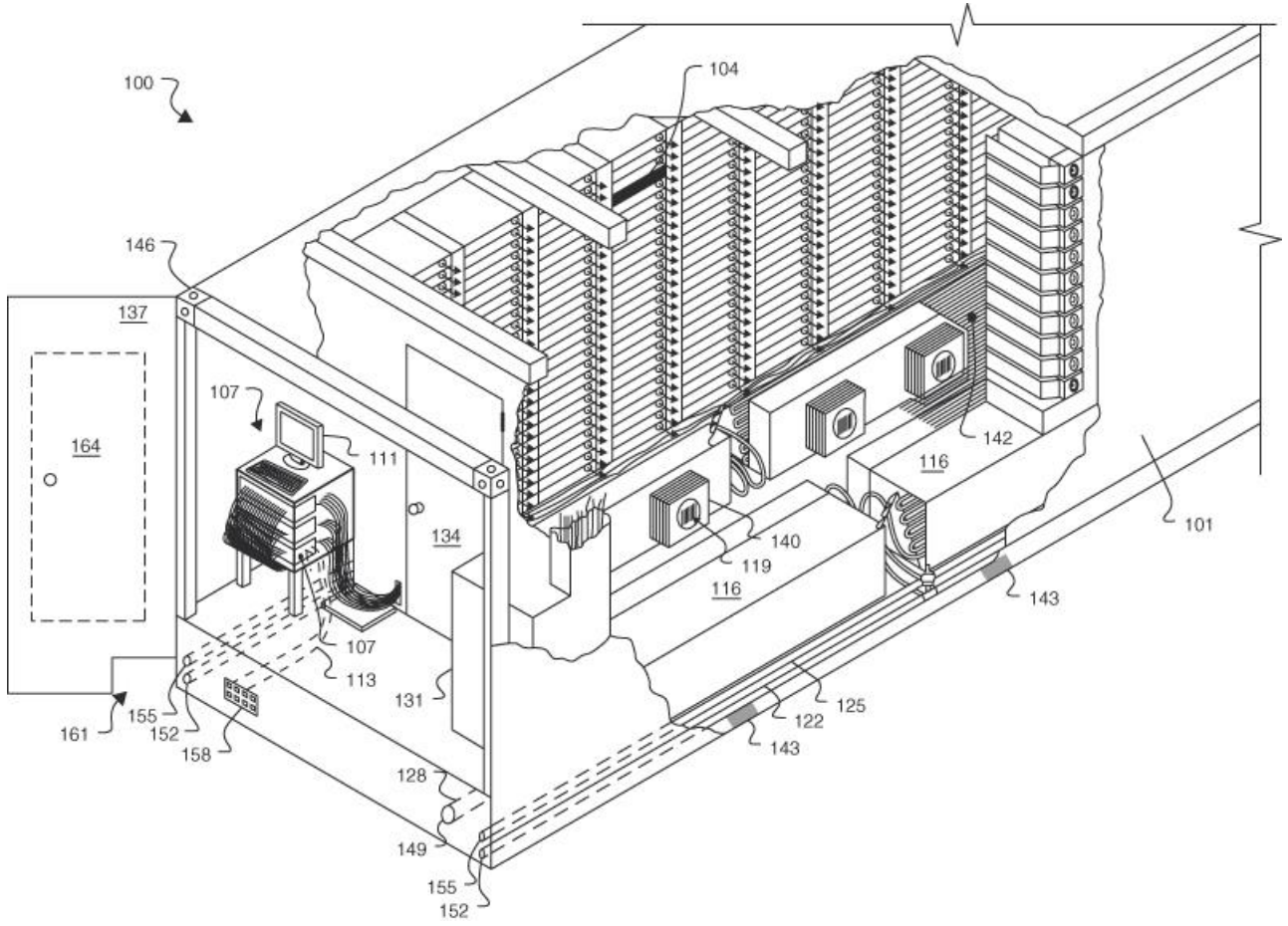


Figure 6.19 Google customizes a standard 1AAA container: 40 x 8 x 9.5 feet (12.2 x 2.4 x 2.9 meters). The servers are stacked up to 20 high in racks that form two long rows of 29 racks each, with one row on each side of the container. The cool aisle goes down the middle of the container, with the hot air return being on the outside. The hanging rack structure makes it easier to repair the cooling system without removing the servers. To allow people inside the container to repair components, it contains safety systems for fire detection and mist-based suppression, emergency egress and lighting, and emergency power shut-off. Containers also have many sensors: temperature, airflow pressure, air leak detection, and motion-sensing lighting. A video tour of the datacenter can be found at *http://www.google.com/corporate/green/datacenters/summit.html*. Microsoft, Yahoo!, and many others are now building modular datacenters based upon these ideas but they have stopped using ISO standard containers since the size is inconvenient.

Cooling and arflow

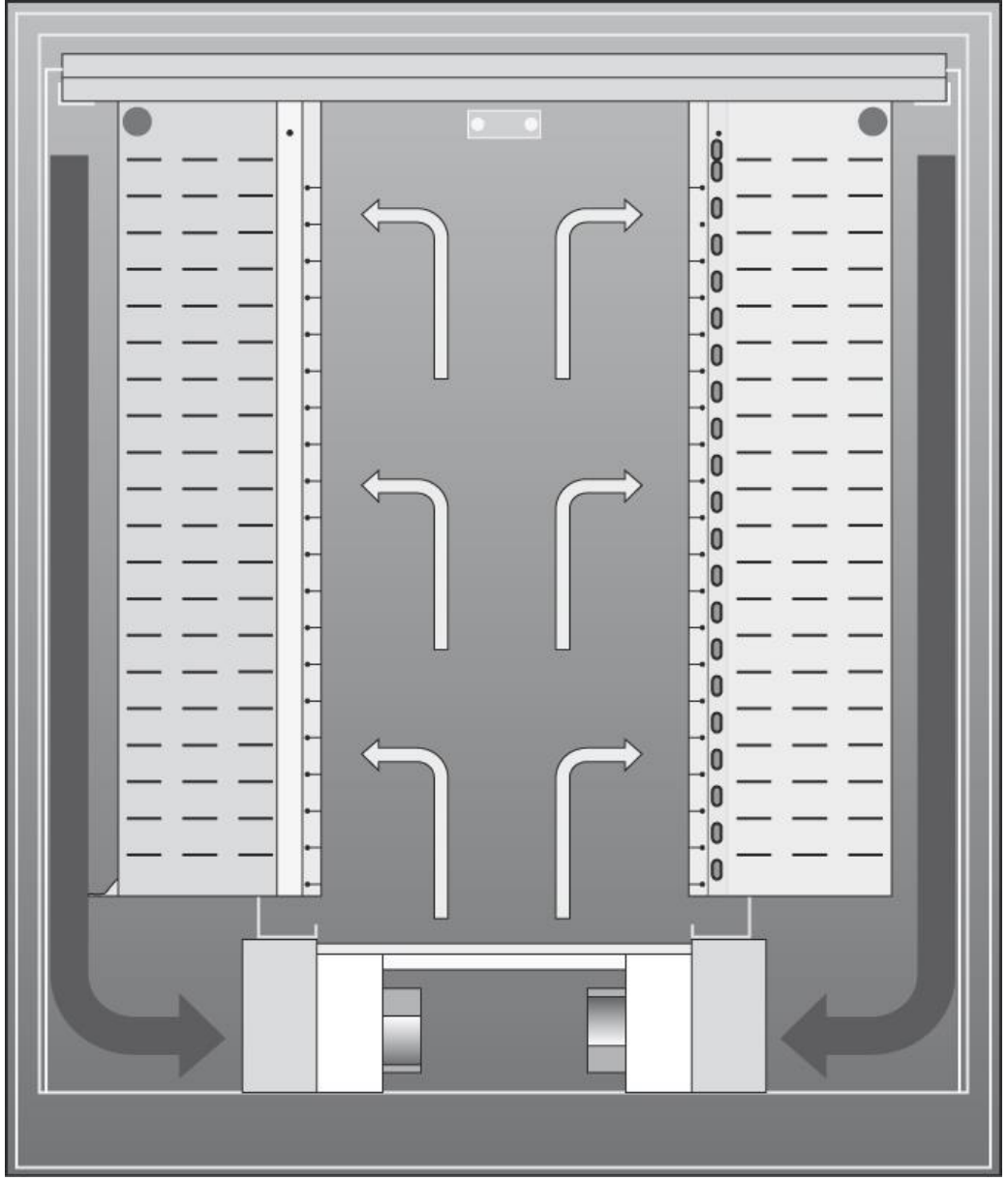


Figure 6.20 Airflow within the container shown in Figure 6.19. This cross-section diagram shows two racks on each side of the container. Cold air blows into the aisle in the middle of the container and is then sucked into the servers. Warm air returns at the edges of the container. This design isolates cold and warm airflows.

Server for Google WSC

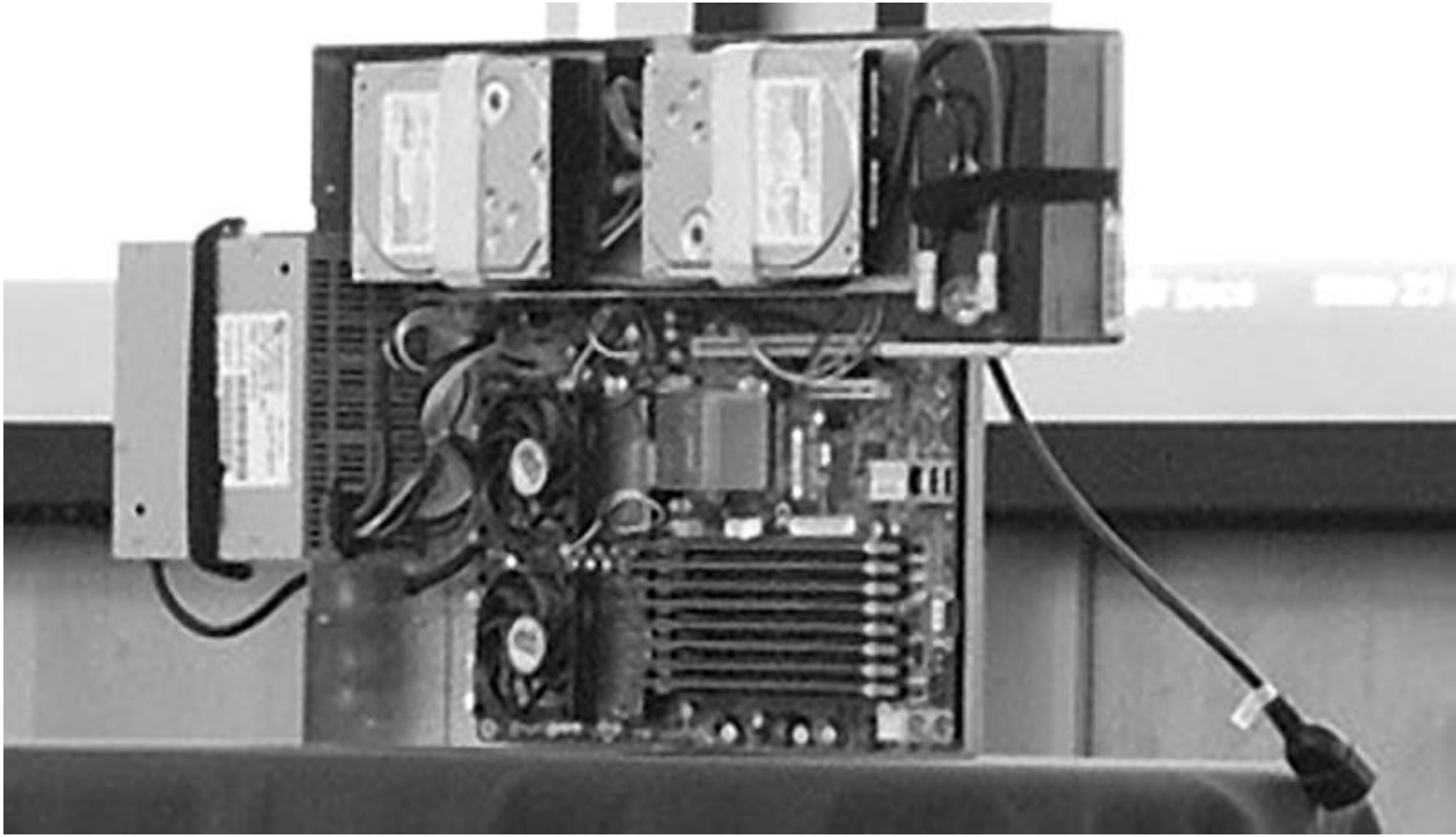


Figure 6.21 The power supply is on the left and the two disks are on the top. The two fans below the left disk cover the two sockets of the AMD Barcelona microprocessor, each with two cores, running at 2.2 GHz. The eight DIMMs in the lower right each hold 1 GB, giving a total of 8 GB. There is no extra sheet metal, as the servers are plugged into the battery and a separate plenum is in the rack for each server to help control the airflow. In part because of the height of the batteries, 20 servers fit in a rack.

IC-UNICAMP

Server for Google WSC

- Two sockets, each with a dual-core AMD Opteron processor running a 2.2 GHz
- Eight DIMMS: 8GB of DDR2 DRAM, downclocked to 533 MHz from standard 666 MHZ (low impact on speed but high impact on power)
- Baseline node: diskfull, or
  - second tray with 10 SATA disks
  - storage node takes up two slots in the rack → 40,000 servers rather than 52,200

PUE of 10 Google WSCs

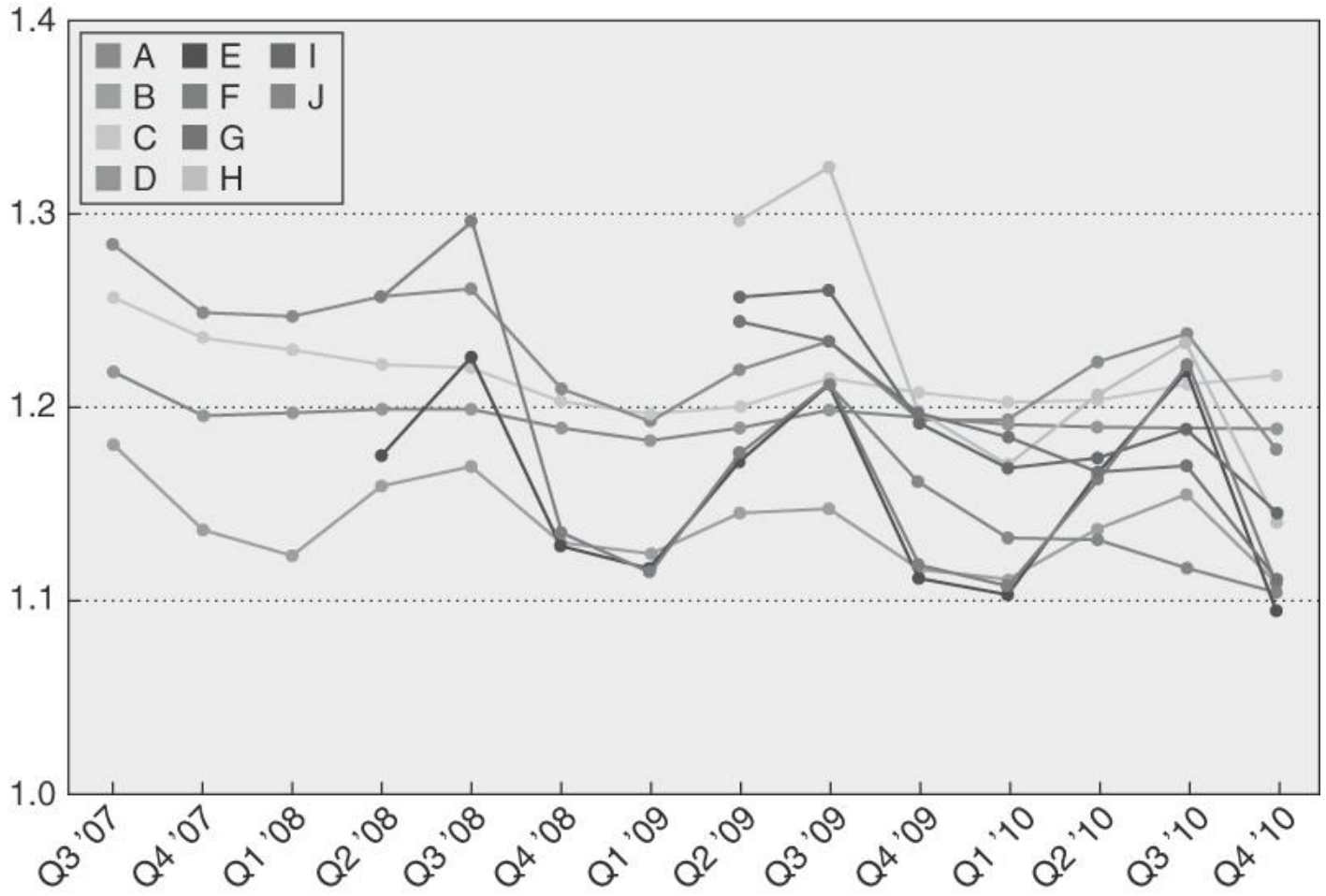


Figure 6.22 Google A is the WSC described in this section. It is the highest line in Q3 '07 and Q2 '10. (From *www.google.com/corporate/green/datacenters/measuring.htm*.) Facebook recently announced a new datacenter that should deliver an impressive PUE of 1.07 (see http://opencompute.org/). The Prineville Oregon Facility has no air conditioning and no chilled water. It relies strictly on outside air, which is brought in one side of the building, filtered, cooled via misters, pumped across the IT equipment, and then sent out the building by exhaust fans. In addition, the servers use a custom power supply that allows the power distribution system to skip one of the voltage conversion steps in Figure 6.9.

IC-UNICAMP

Networking in a Google WSC

- The 40,000 servers are divided in three arrays, called clusters (Google terminology)
- 48-port rack switch: 40 ports to other servers, 8 ports for uplinks to the array switches
- Array switches support up to 480 1 Gb/s links + few 10 Gb/s ports
- There is 20 times the network bw inside the switch as there was exiting the switch
  - Applications with significant traffic demands beyond a rack → poor network performance

IC-UNICAMP

Google WSC: conclusion / innovations

- Inexpensive shells (containers): hot and cold air are separated, less severe worst-case hot spots → cold air at higher temperatures
- Shrinked air circulation loops → lower energy to move air
- Servers operate at higher temperatures
  - evaporative cooling solutions (cheaper) are possible
- Deploy WSCs in temperate climate → lower cooling costs
- Extensive monitoring → lower operating costs
- Motherboards that need only 12 V DC → UPS function supplied by standard batteries (no battery room)
- Careful design of server board (under clocking without performance impact) → improved energy efficiency
  - no impact on PUE but WSC overall energy consumption reduction