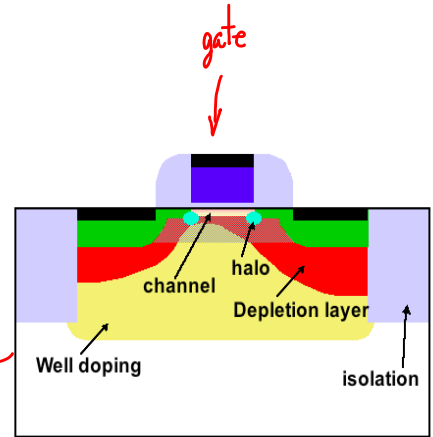
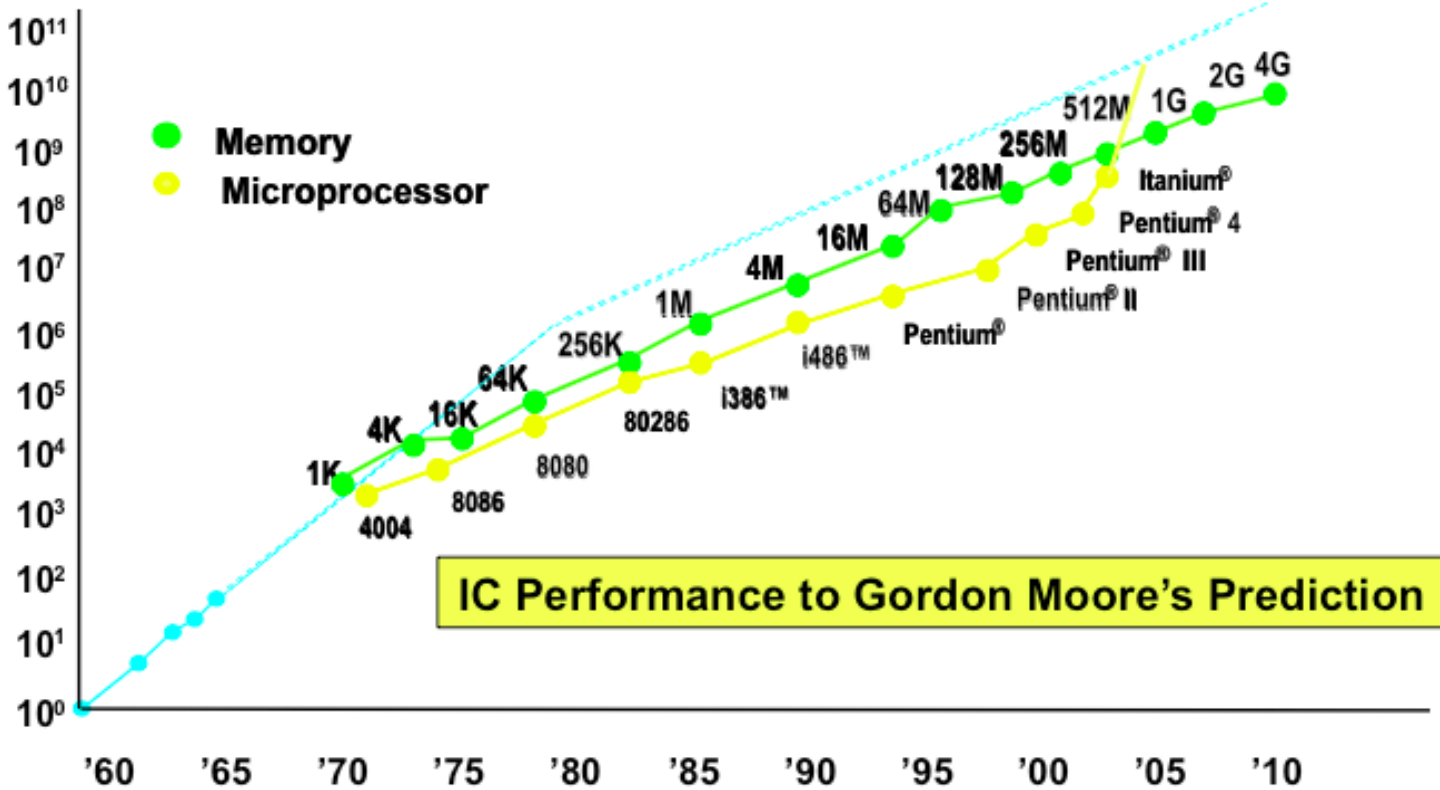


conducting
interconnect layers



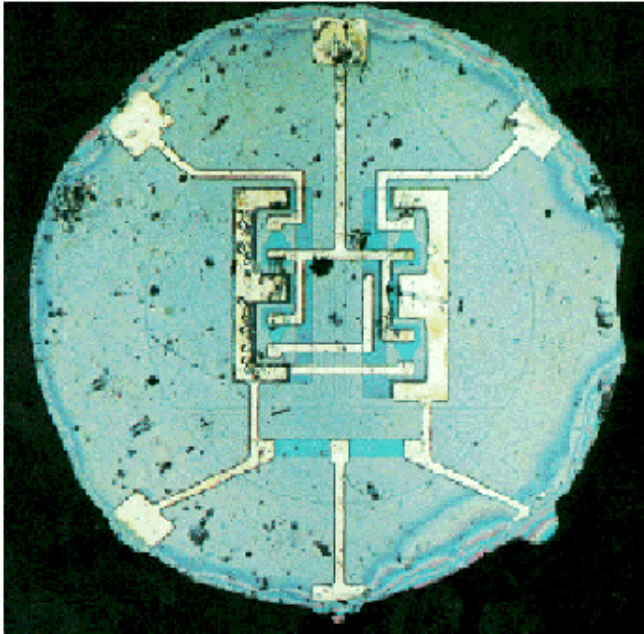
TRANSISTOR

Transistors Per Die

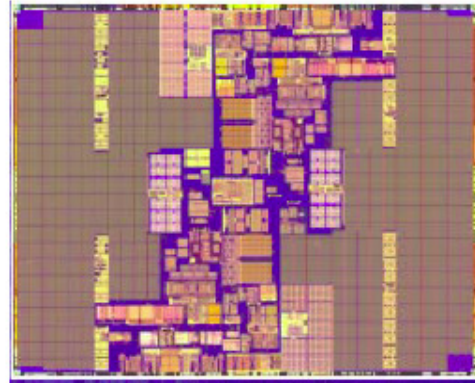


Source: Intel

First planar integrated circuit (1961)



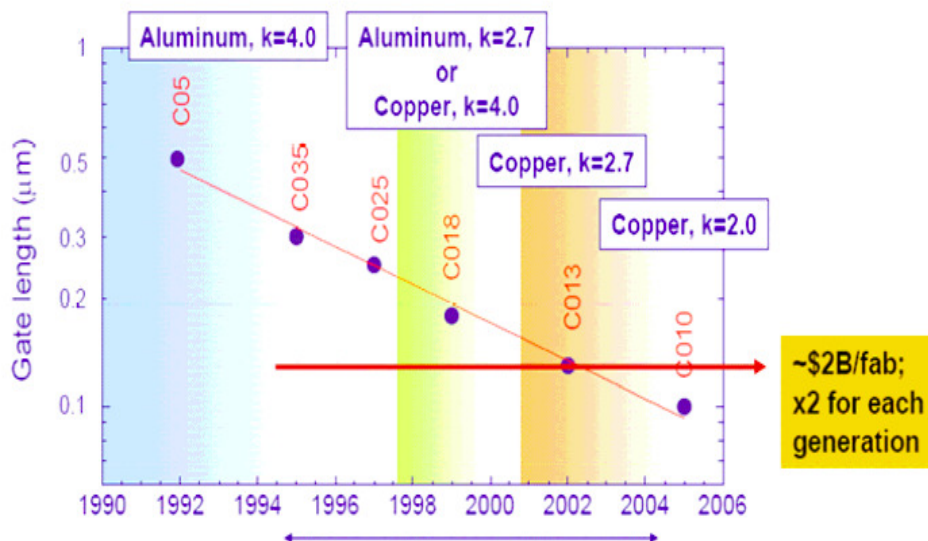
90 nm Intel's processor
Montecito (2004)
Itanium Processor Family



Transistors: 1.72 Billion
Frequency: >1.7GHz
Power: ~100W

Source: Intel Developer Forum,
September, 2004

Cost limit



Fundamental limits

From: **thermodynamics, quantum-mechanics, electromagnetics**

- Limit on energy transfer during a binary switching:
 $E(\min) = (\ln 2) kT (=kT \log_e N, N=2)$ (J. Neumann)
- Heisenberg's uncertainty principle:
 $\Delta E > h/\Delta t$ \rightarrow forbidden region for power-delay
- electromagnetics $\rightarrow \tau > L/c_0$ (limited time of electromagnetic wave travelling across interconnects)

Why $E(\min) = kT \times \ln 2$?

Binary signal discrimination: the slope of the static transfer curve of a (CMOS) binary logic gate must be greater than unity in absolute value at the transition point where input and output voltage levels are equal \rightarrow **CMOS inverter**

$$V_{dd}(\min) = 2[kT/q] \left[1 + \frac{C_{fs}}{C_{ox} + C_d} \right] \ln \left(2 + \frac{C_d}{C_{ox}} \right)$$

$$V_{dd}(\min) \cong 2(\ln 2) \frac{kT}{q} = 1.38 \frac{kT}{q} = 0.036V @ T = 300K$$

Min signal energy stored on gate:

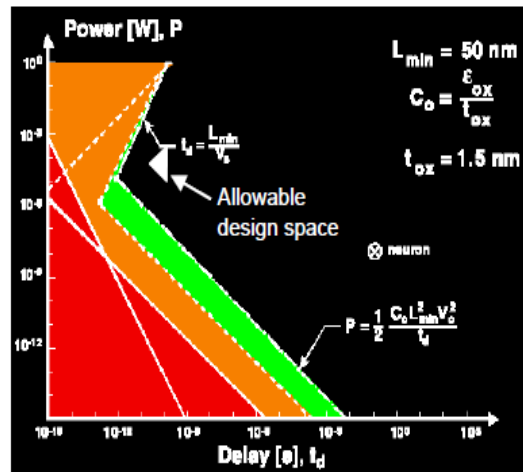
$$E_s(\min) = (1/2)Q_g V_{dd} = (1/2)q \times 2(\ln 2) \frac{kT}{q} = kT \times \ln 2 = 0.693kT$$

$$\text{with : } C_g = \frac{\epsilon_{ox} L_{\min}^2}{t_{ox}} \Rightarrow L_{\min} = \left[\frac{t_{ox}}{\epsilon_{ox}} \right] q^2 / [2(\ln 2)kT]^{1/2} = 9.3nm @ t_{ox} = 1nm$$

Source: J.D. Meindl, J. A. Davis, IEEE JSSC, Vol. 35, October 2000, pp. 1515-1516.

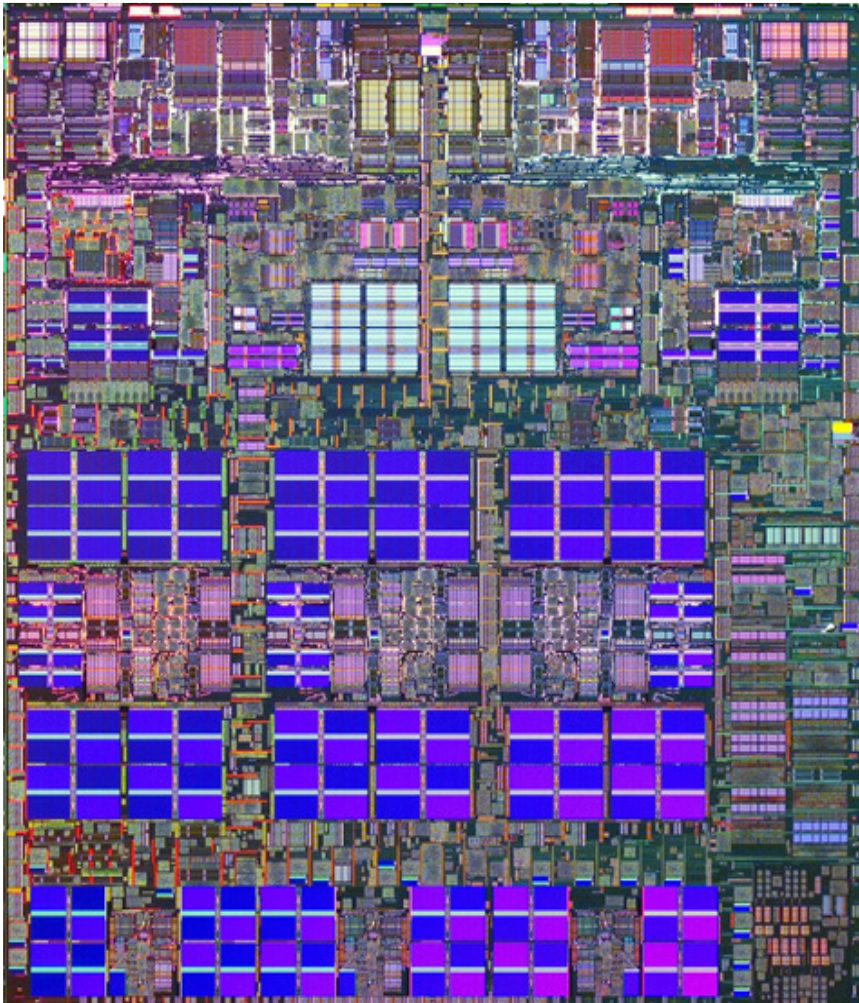
Fundamental limits

Average power transfer during a binary transition, P , versus transition time, t_d . The red, orange, and green zones are forbidden by fundamental, silicon material, and 50-nm channel length transistor device level limits, respectively.



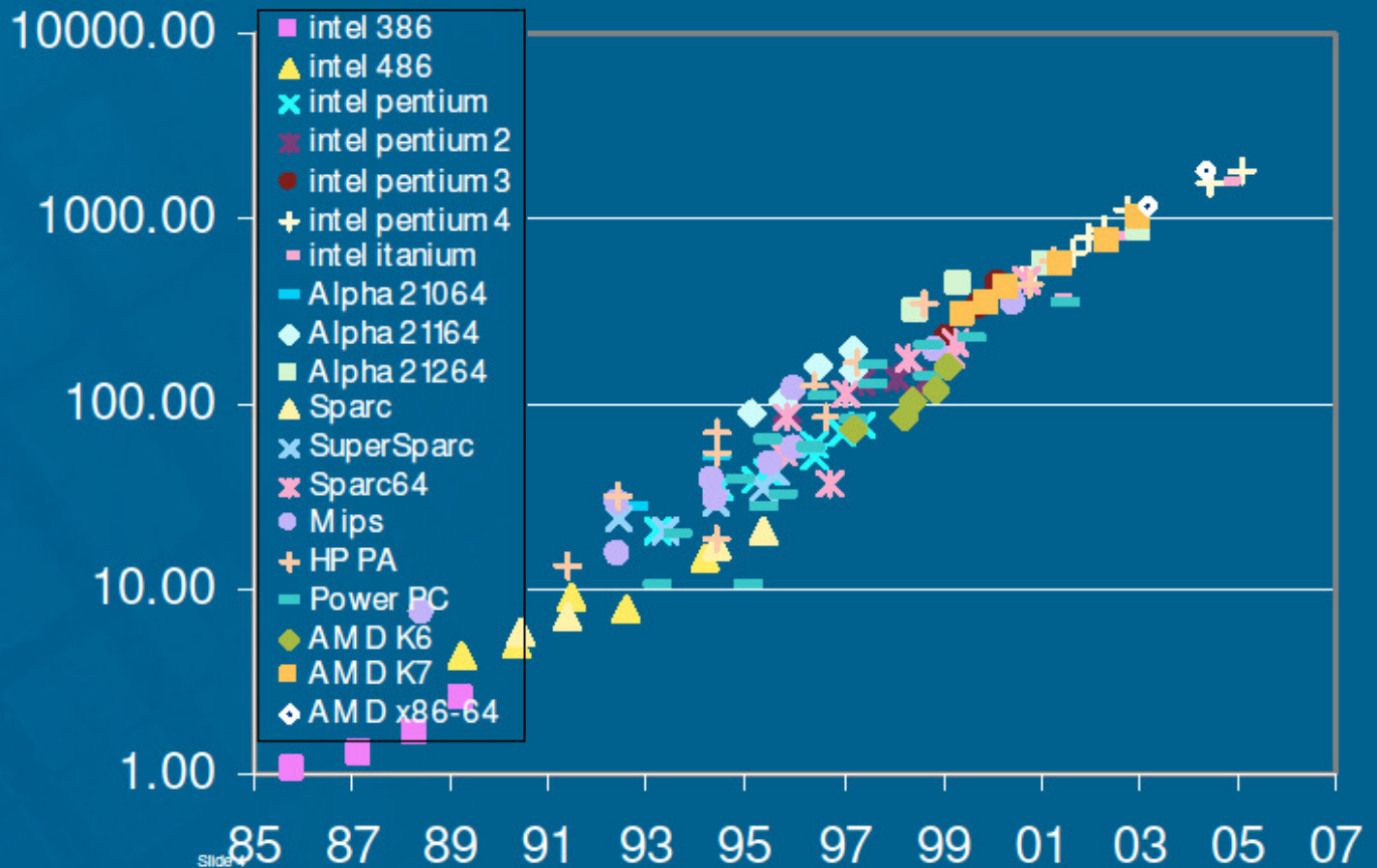
Source:

J. D. Meindl, Q. Chen, J. A. Davis, Science, Vol. 293, pp. 2044-2049, September 2001

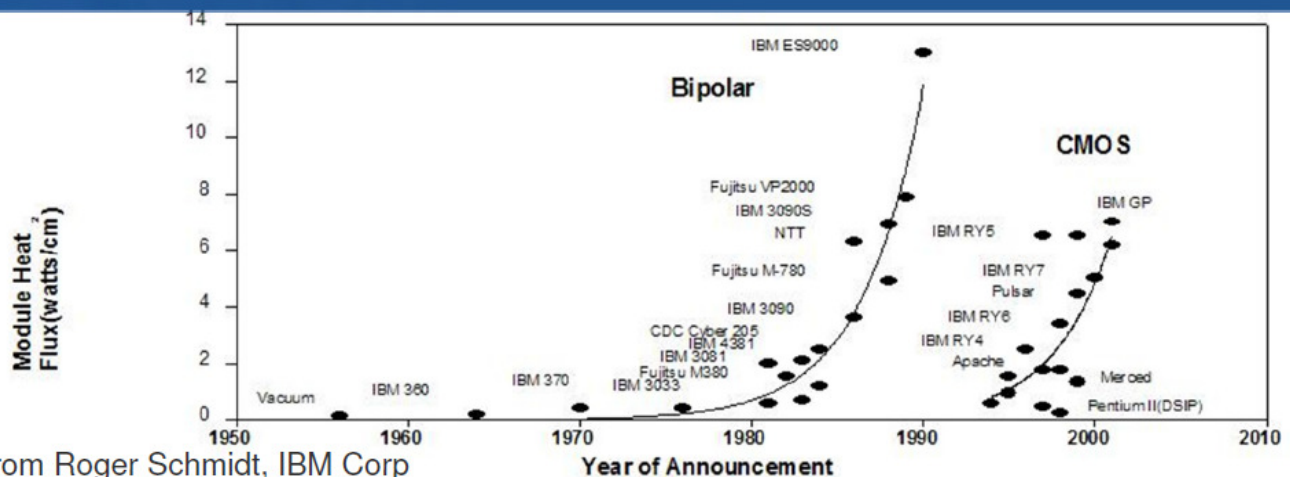


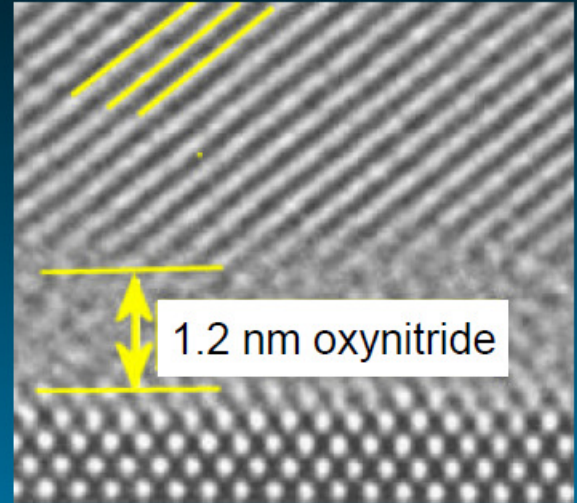
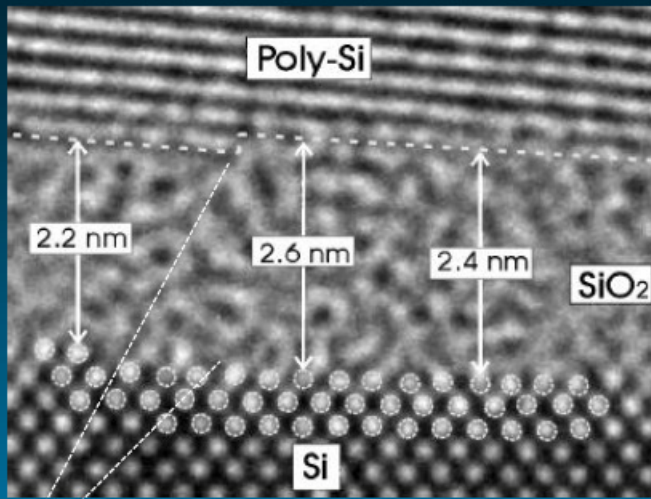
ibm power 5

CMOS Computer Performance



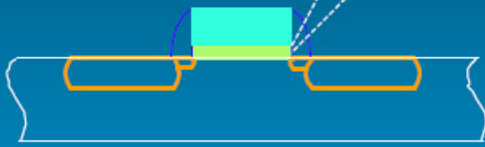
Power / cm²





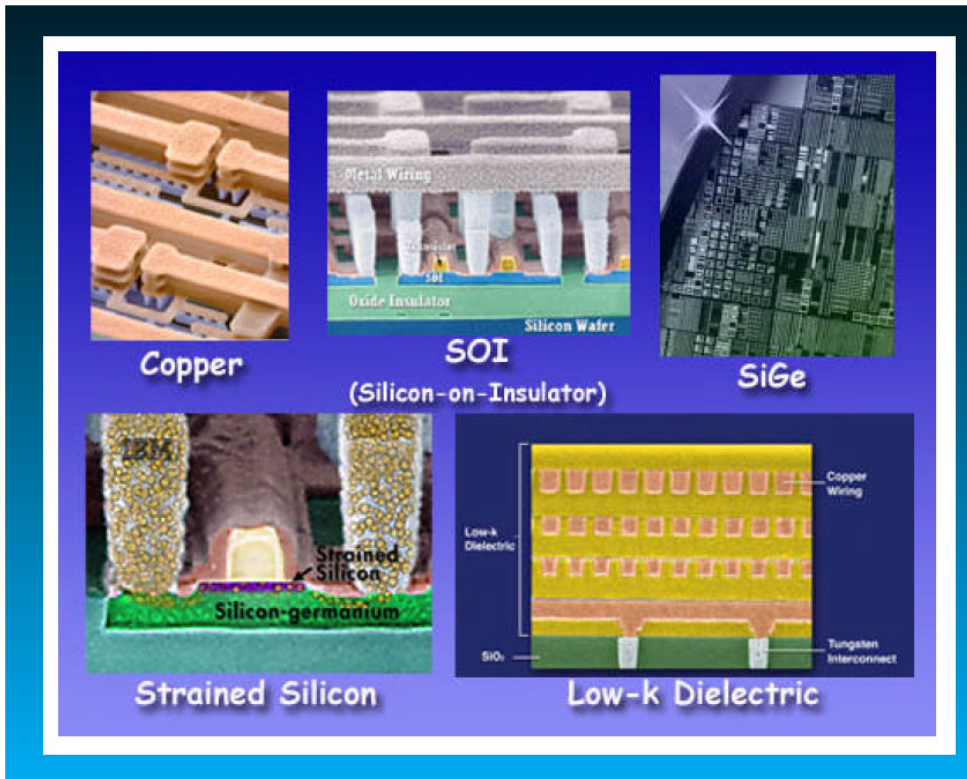
present

future



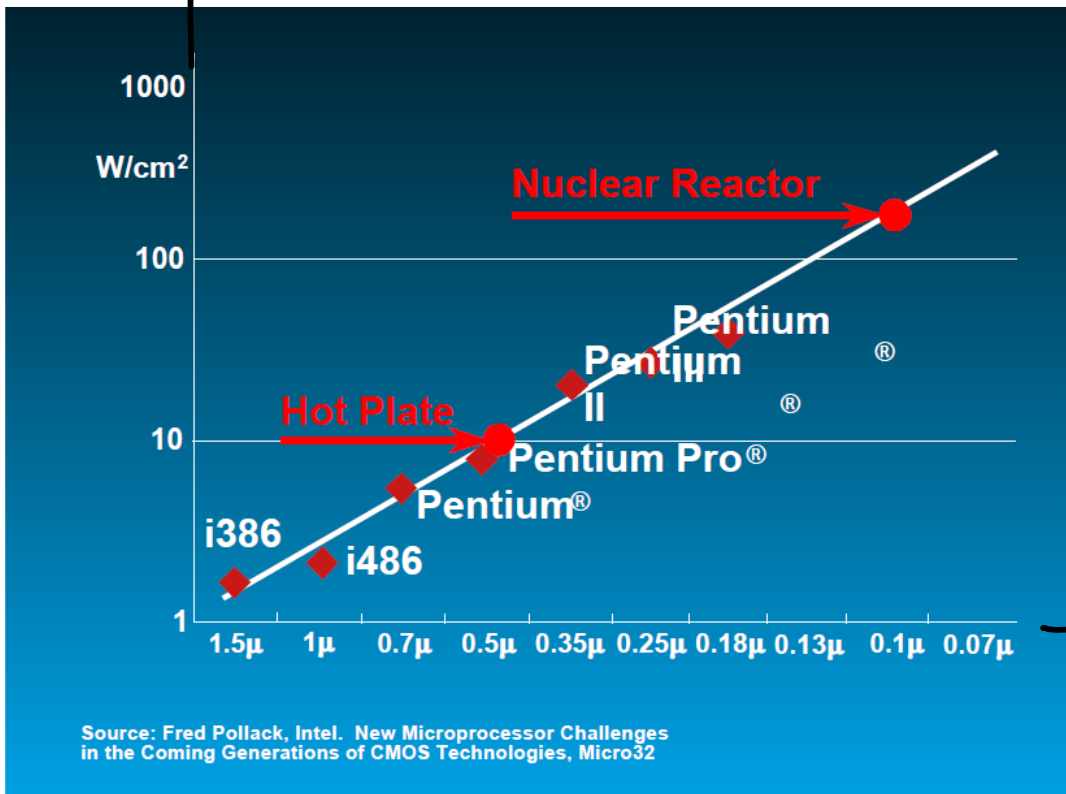
silicon bulk field effect transistor (FET)

- Oxide thickness is approaching a few atomic layers



faster w/o scaling

more W/cm^2

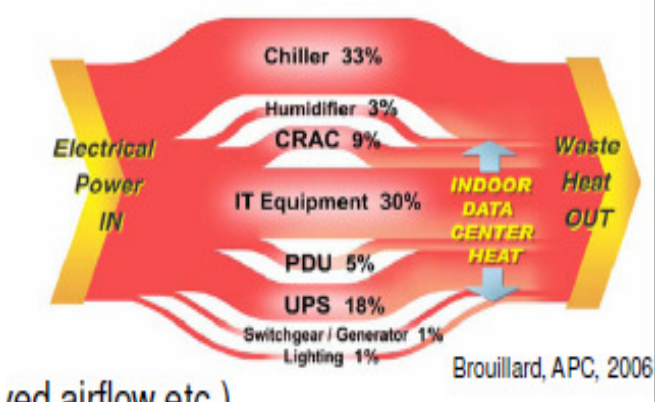
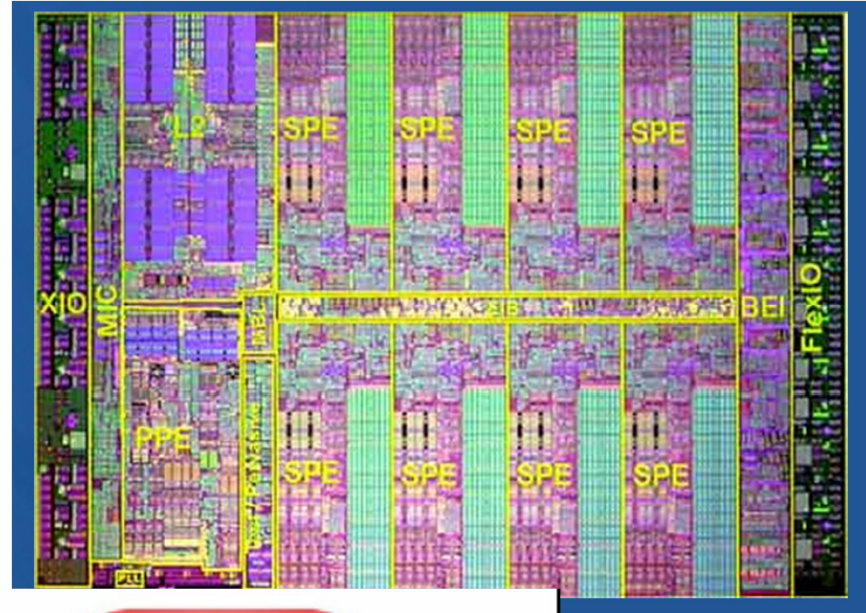


smaller line size

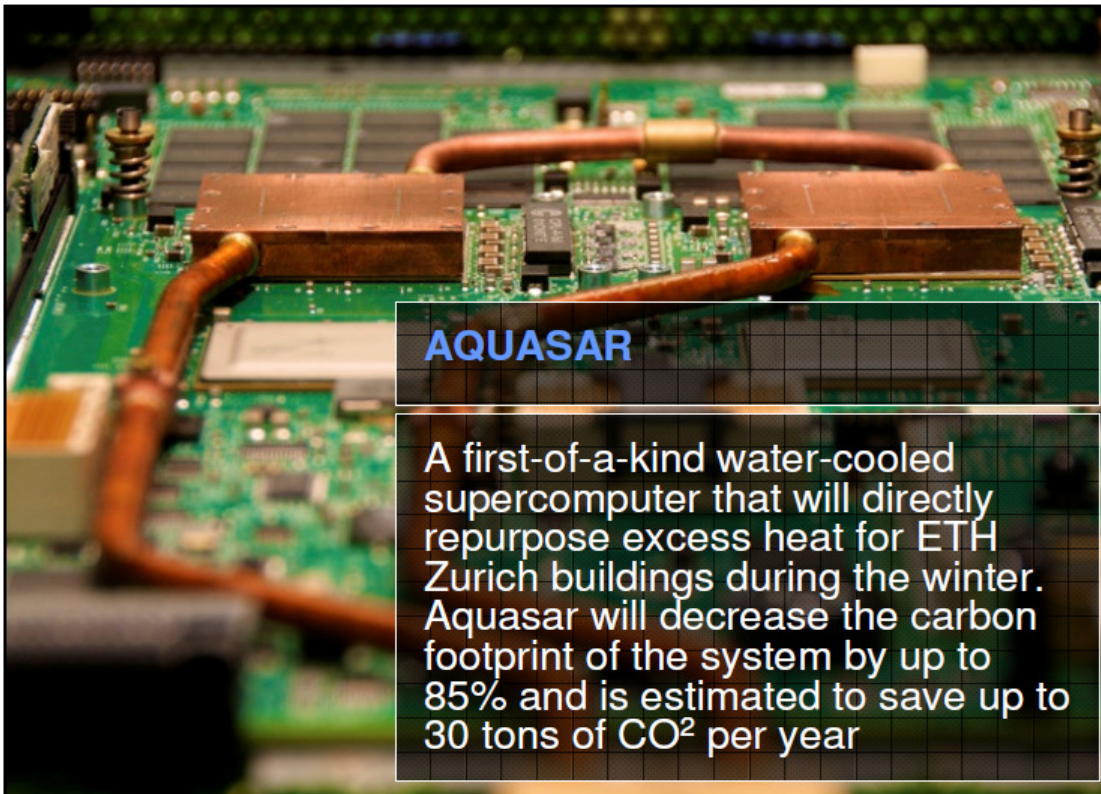
Heat Density

IBM Cell

Hetero- Many- core



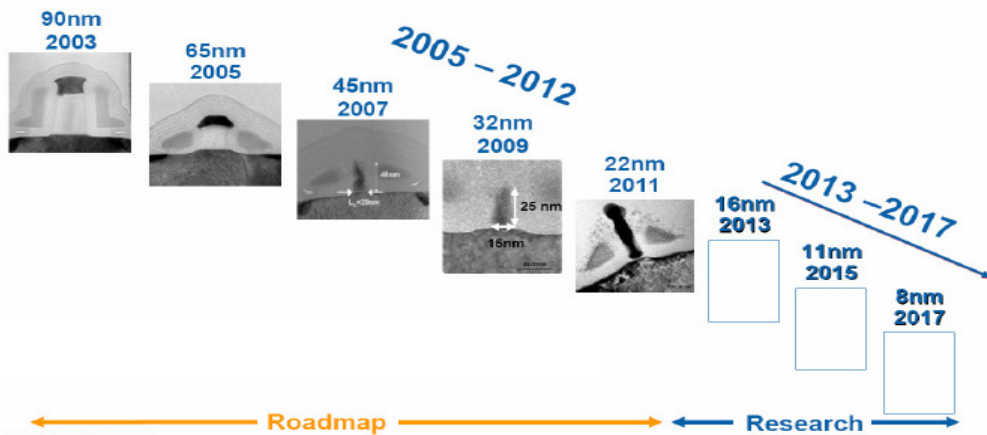
red airflow etc)



AQUASAR

A first-of-a-kind water-cooled supercomputer that will directly repurpose excess heat for ETH Zurich buildings during the winter. Aquasar will decrease the carbon footprint of the system by up to 85% and is estimated to save up to 30 tons of CO² per year

Moore's Law: based on CMOS scaling



Source: Intel, march 2008

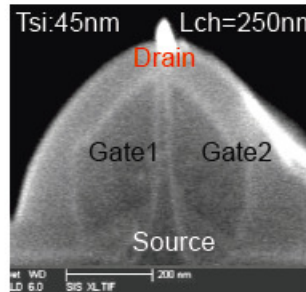
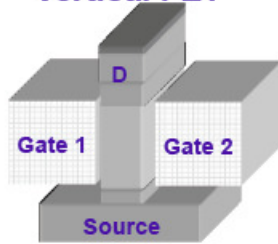
19

Ringvorlesung ETH Springsemester 2010 – Ronald Lütjen, IBM research – Zurich, "IT energy challenges – a new approach"

© 2010 IBM Corporation

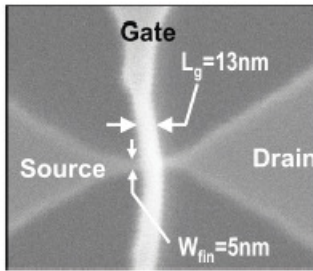
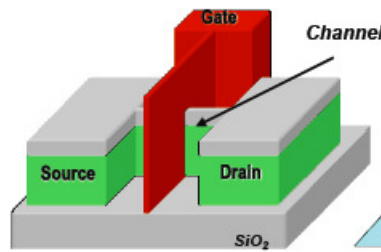
Non Planar MOSFETs

Vertical FET



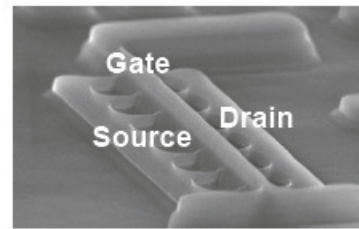
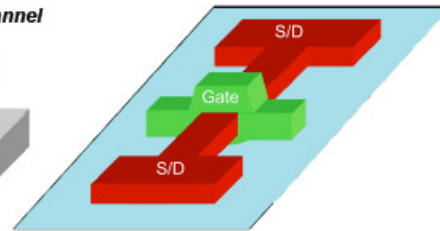
Stanford

Double Gate FinFET



UC Berkeley

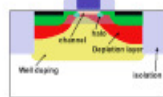
Tri Gate FET



Intel

Summary: Technology Progression

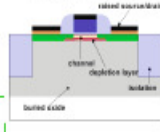
Bulk CMOS



Cu interconnect

Low-k ILD

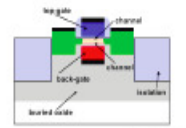
FD SOI CMOS



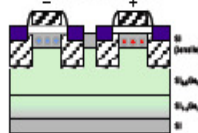
Metal gate

High k gate dielectric

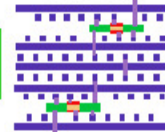
Double-Gate CMOS



Strained Si Ge channel

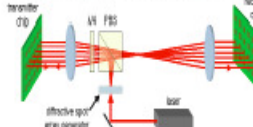


3D, heterogeneous integration



Wafer bonding
Crystallization

Optical interconnect



Detectors, lasers,
QWM, waveguides

Self-assembly

Interconnects
and contacts for
nanodevices

Nanotechnology

Nanotube

Molecular devices

100 nm

Feature Size

2 nm

Computing efficiency

Computations per kilowatt-hour



Source: Jonathan Koomey

An example for a planned 2012 machine: Blue Waters

- 10 PFlop (10^{16}) sustained operations
- 300'000 compute cores = 37'500 CPU chips = 9375 QCM = 1172 drawers = 98 racks
- 800W / QCM \rightarrow 7.5 MW in CPUs
- New building being finished
- 24 transformers@2 MW

- Blue waters PUE = 1.1

- <http://www.ncsa.illinois.edu/BlueWaters/>



Rack

- 990.6w x 1828.8d x 2108.2
- 39" w x 72" d x 83" h
- ~2948kg (~6500lbs)

Data Center In a Rack

- Compute
- Storage
- Switch
- 100% Cooling
- PDU Eliminated

Input: 8 Water Lines, 4 Power Cords
Out: ~100TFLOPs / 24.6TB / 153.5TB
192 PCI-e 16x / 12 PCI-e 8x



BPA

- 200 to 480Vac
- 370 to 575Vdc
- Redundant Power
- Direct Site Power Feed
- PDU Elimination

Storage Unit

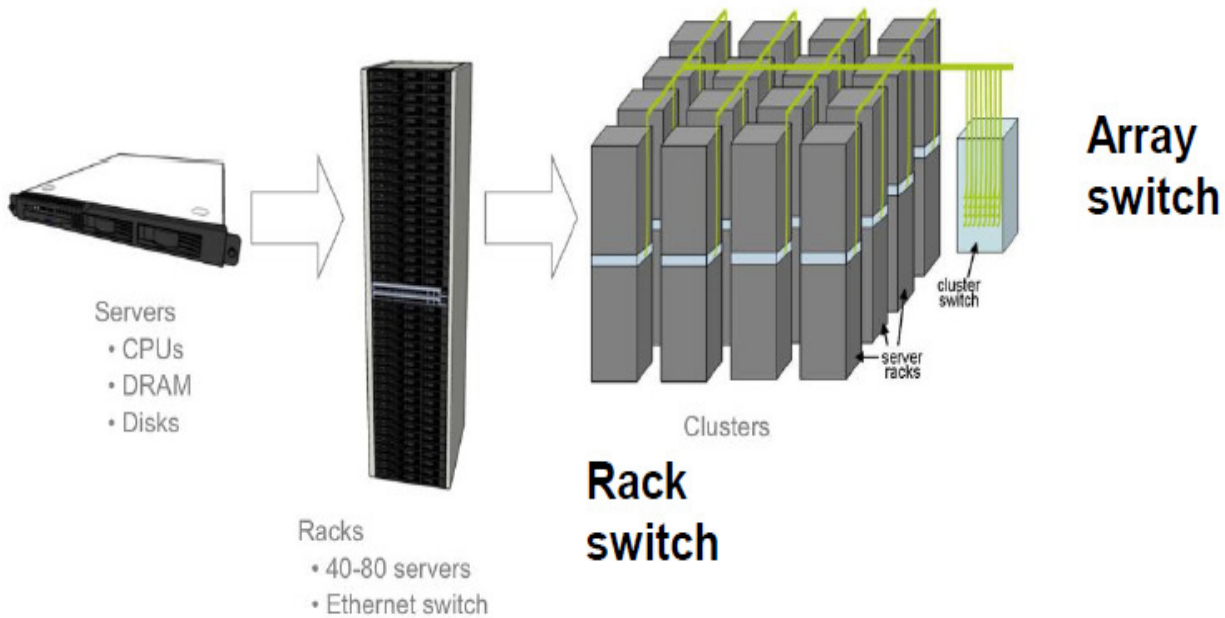
- 4U
- 0-6 / Rack
- Up To 384 SFF DASD / Unit
- File System

CECs

- 2U
- 1-12 CECs/Rack
- 256 Cores
- 128 SN DIMM Slots / CEC
- 8,16, (32) GB DIMMs
- 17 PCI-e Slots
- Imbedded Switch
- Redundant DCA
- NW Fabric
- Up to:3072 cores, 24.6TB (49.2TB)

WCU

- Facility Water Input
- 100% Heat to Water
- Redundant Cooling
- CRAH Eliminated



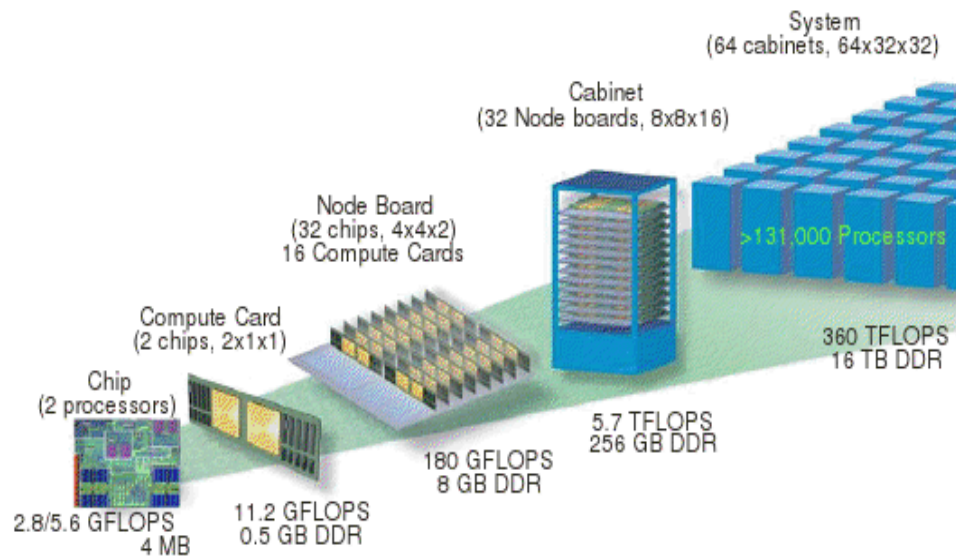
Connecting 50000 servers challenging

- High bandwidth at low costs

Hierarchy of network

- Rack switch, array switch, L3 switch, border routers

BlueGene/L – Holistic Design in Practice

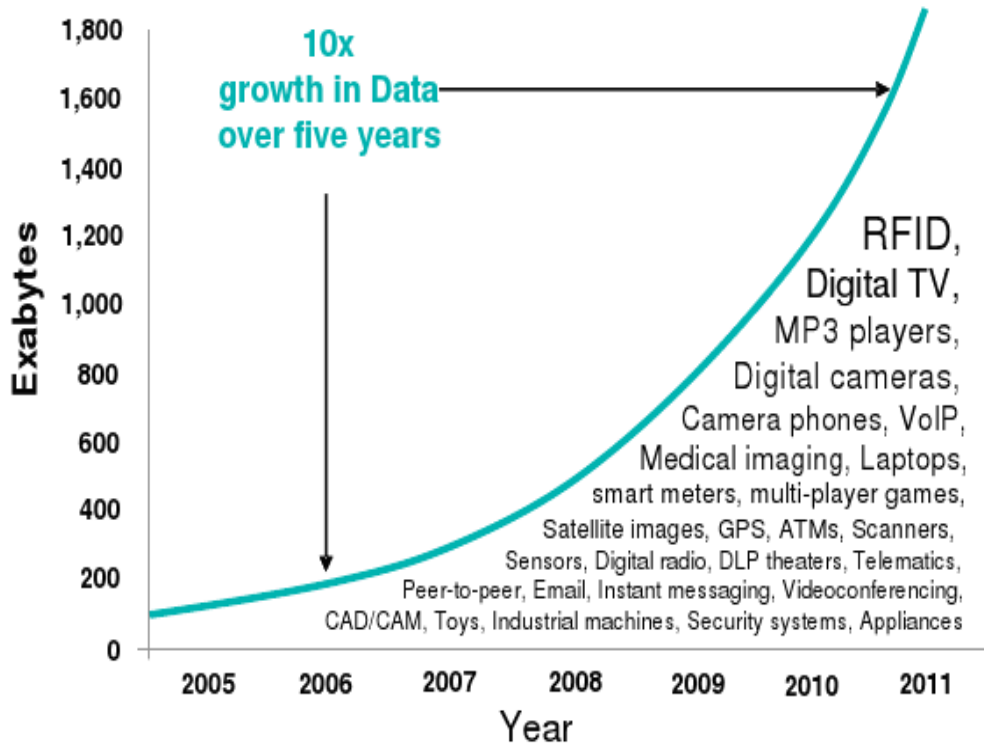


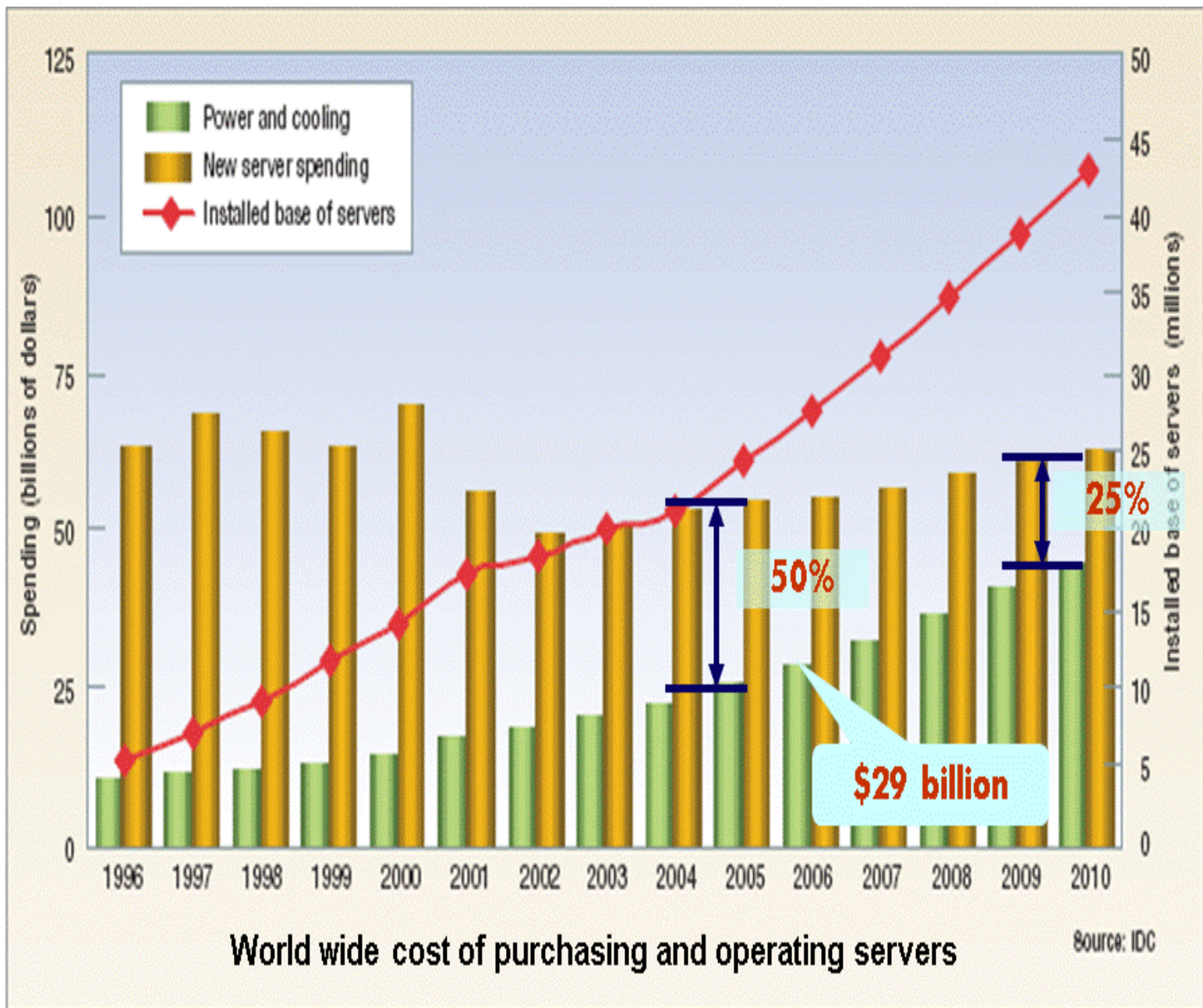
- Using the industry-standard LINPACK benchmark, the IBM Blue Gene/L system attained a sustained performance of **70.72 Teraflops**, eclipsing the three year old top mark of **35.86 Teraflops** for the Japanese Earth Simulator and the recent mark of **42.7 Teraflops** at the NASA's Ames research center.
- The BlueGene/L system is **1/100th** the physical size (**320 vs 32,500 square feet**) and consumes **1/28th** the power (**216KW vs 6,000KW**) as compared to the Earth Simulator.

Challenge; The Data Tsunami

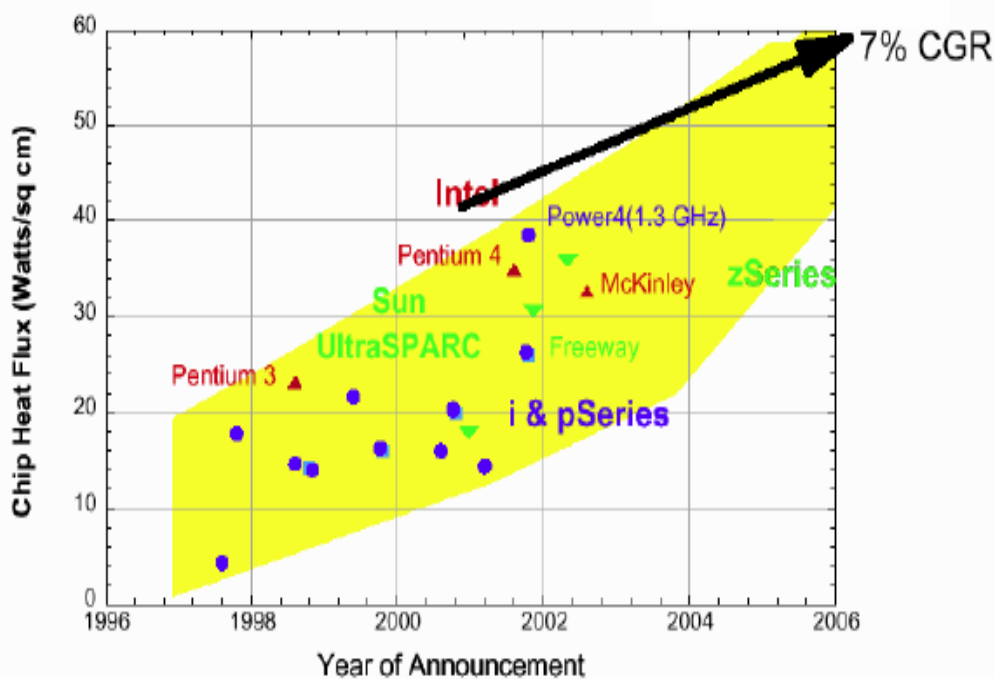


By 2011, the world will store 10X the Data stored in 2006, BUT; Internet connected devices will grow by **2000X**, from 500M to 1 Trillion, and each will demand that someone “listen”.





2002 International Technology Roadmap For Semiconductors



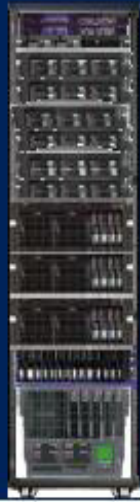
Traditional servers

Theoretical rack-mount density (42 x 2P 1U)

High blade density (48 x 2P blades)

Theoretical blade density (96 x 2P blades)

Theoretical blade density (next gen) (96 x 2P blades)



Average power /cooling

- 6-8 kW

- 16 kW

- 14 kW

- 34 kW

- 55 kW

- 27k BTU/hr

- 55k BTU/hr

- 48k BTU/hr

- 116k BTU/hr

- 188k BTU/hr

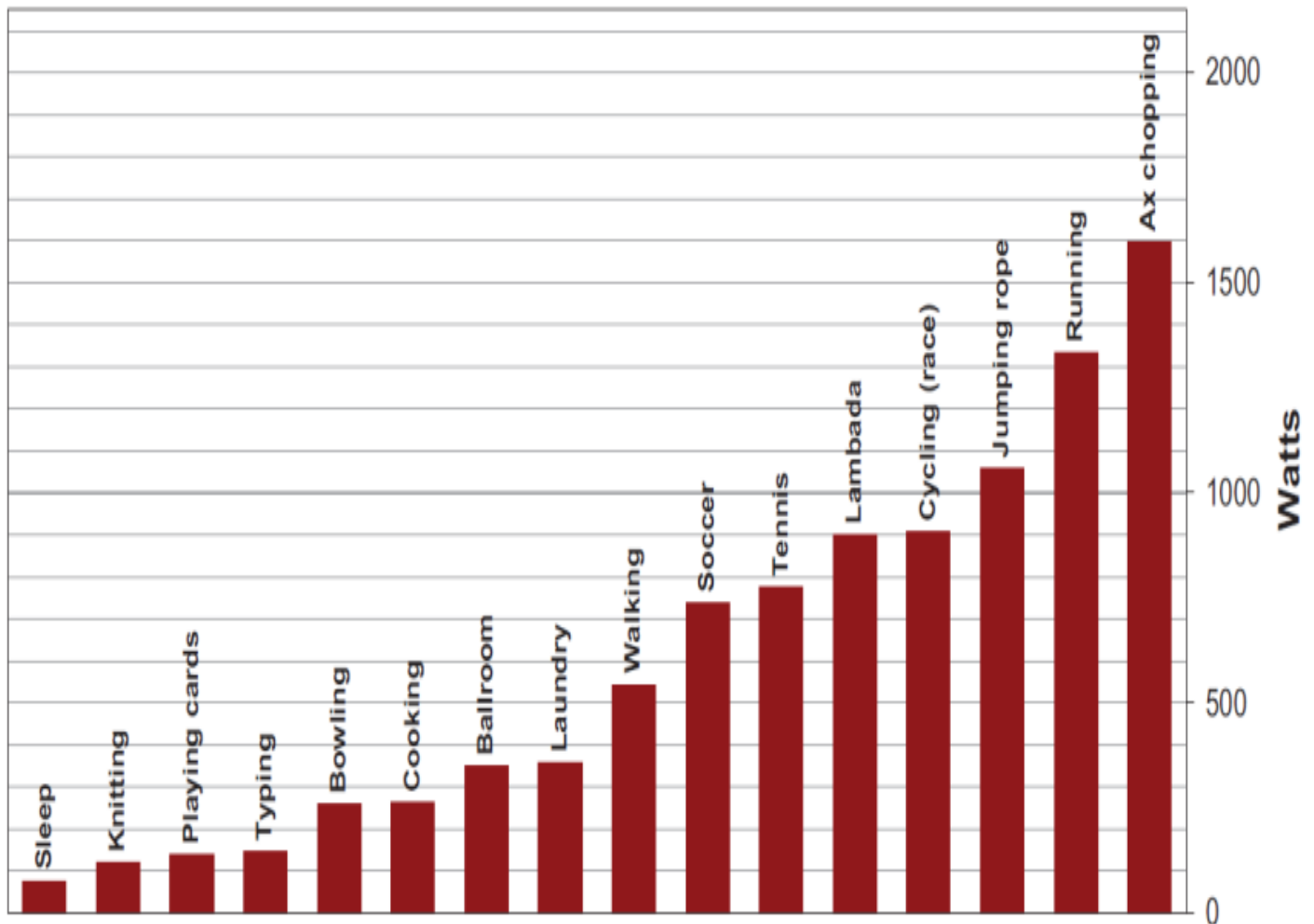


FIGURE 5.7: Human energy usage vs. activity levels (adult male) [52].

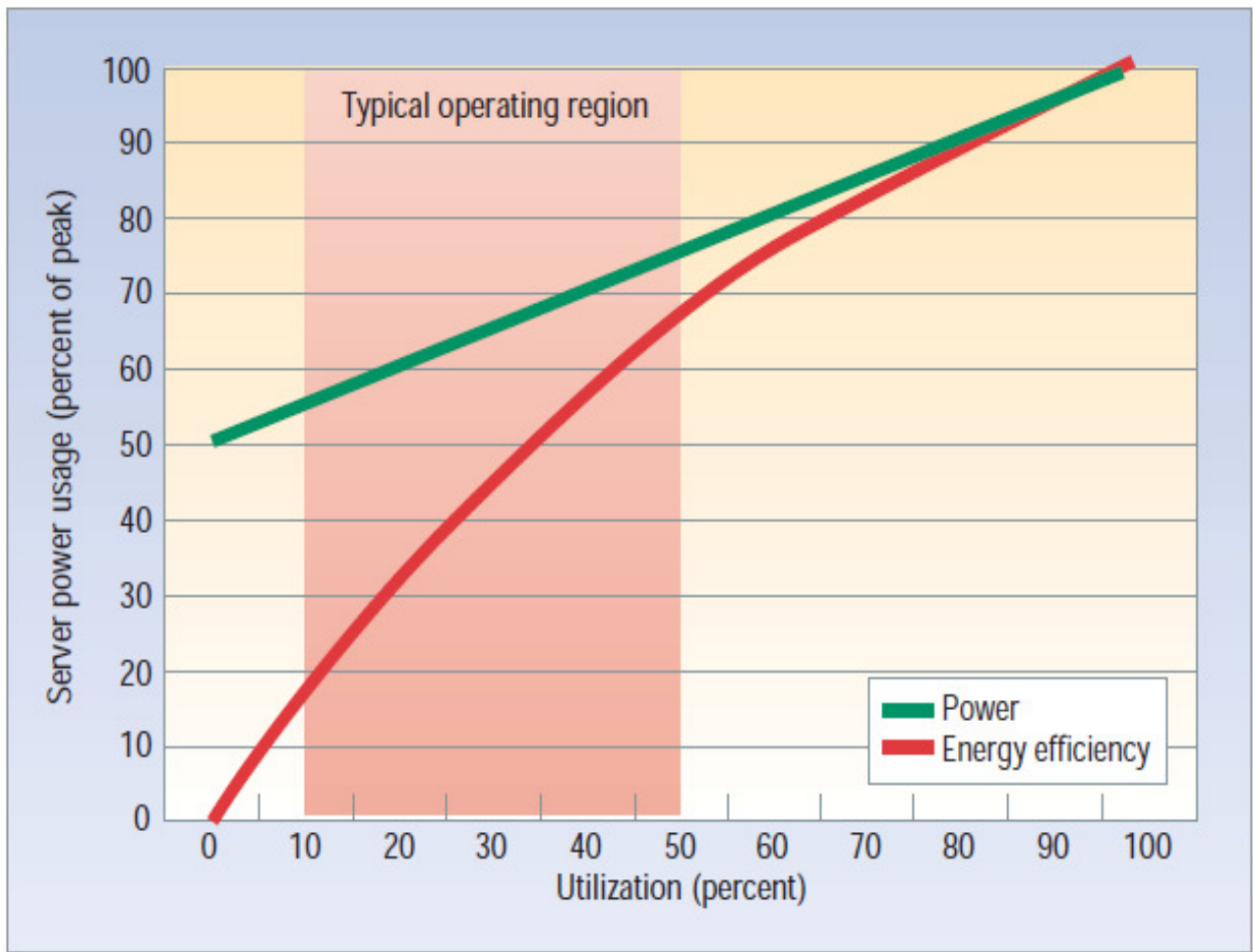


Figure 2. Server power usage and energy efficiency at varying utilization levels, from idle to peak performance. Even an energy-efficient server still consumes about half its full power when doing virtually no work.

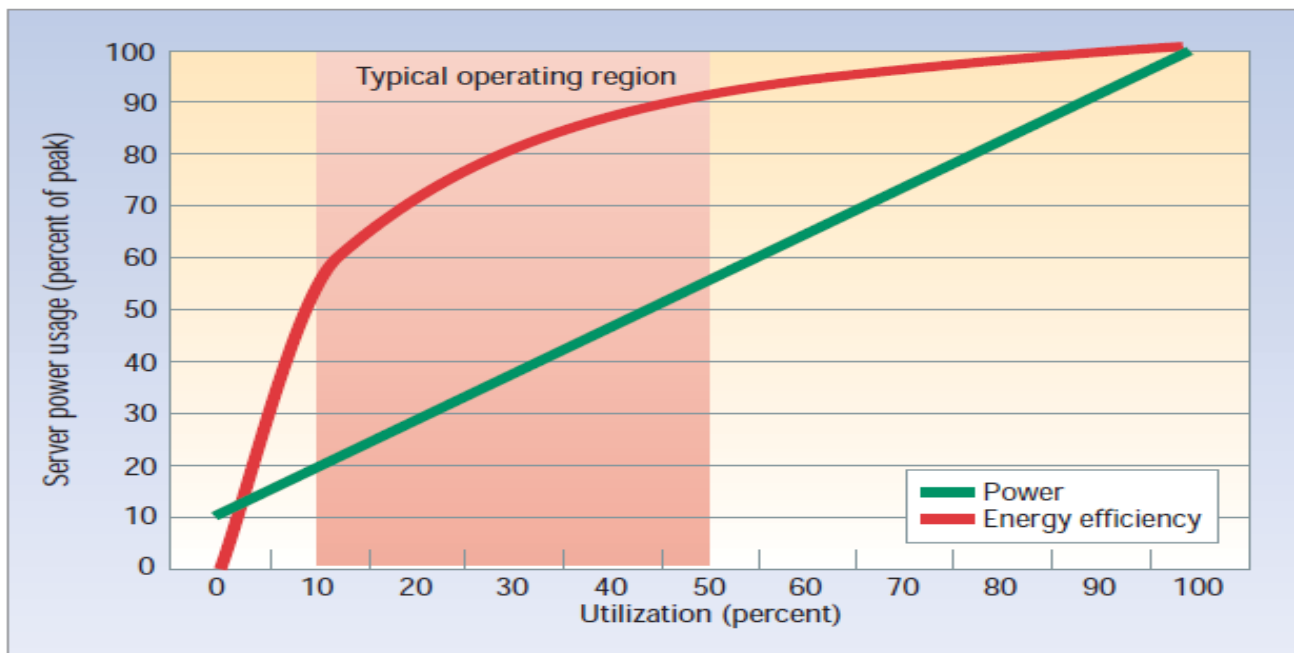


Figure 4. Power usage and energy efficiency in a more energy-proportional server. This server has a power efficiency of more than 80 percent of its peak value for utilizations of 30 percent and above, with efficiency remaining above 50 percent for utilization levels as low as 10 percent.

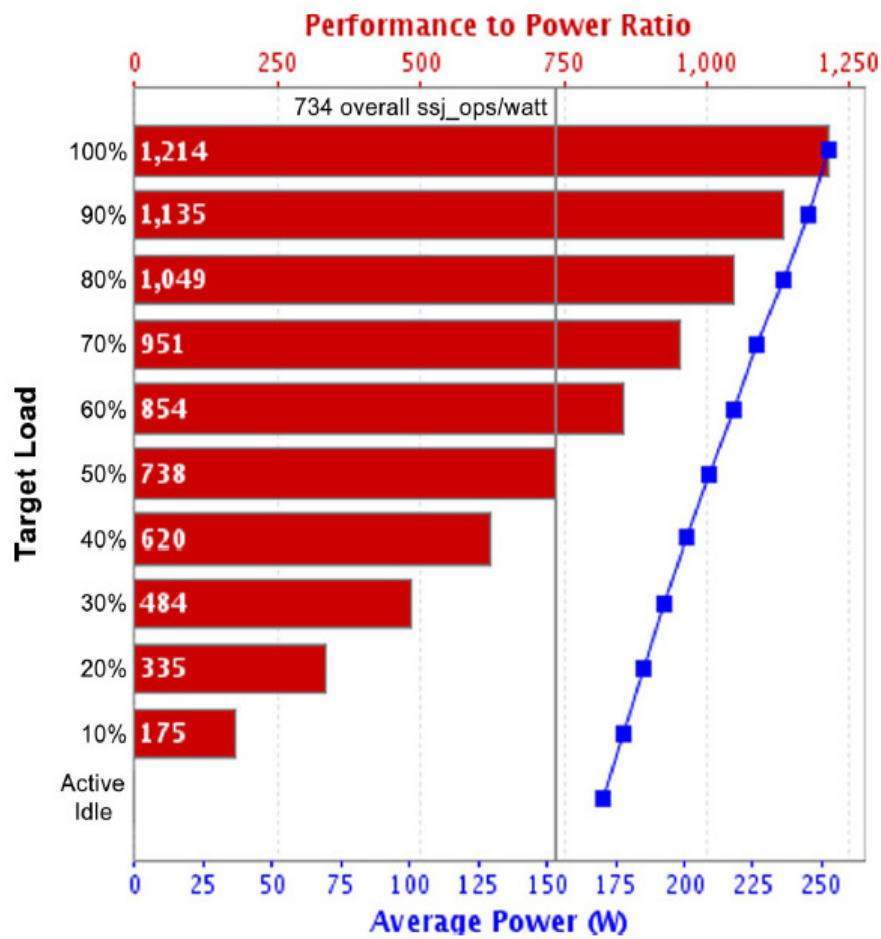
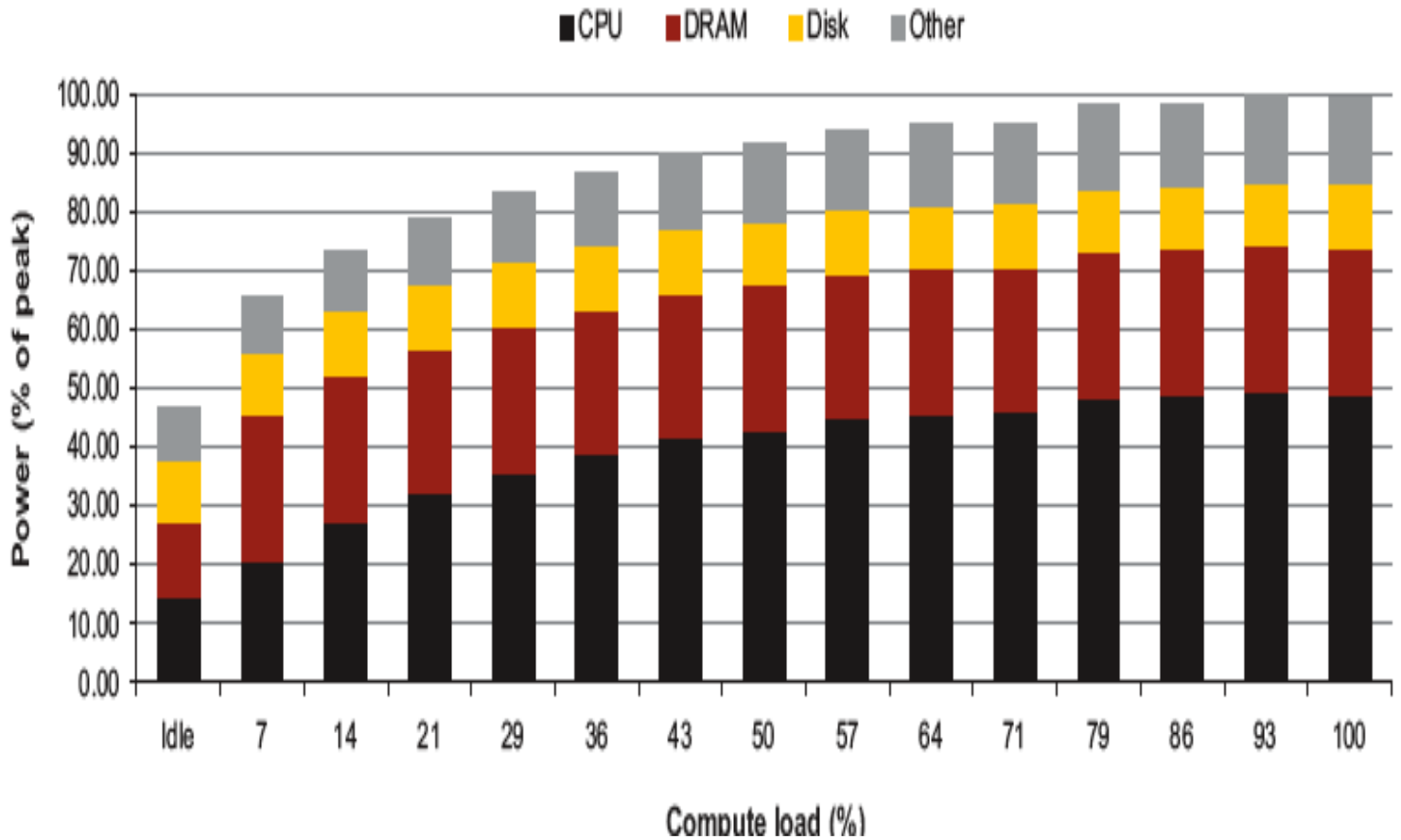


FIGURE 5.3: An example benchmark result for SPECpower_ssj2008; energy efficiency is indicated by bars, whereas power consumption is indicated by the line. Both are plotted for a range of utilization levels, with the average metric corresponding to the vertical dark line. The system has a single-chip 2.83 GHz quad-core Intel Xeon processor, 4 GB of DRAM, and one 7.2 k RPM 3.5" SATA disk drive.



Multi-Level Model Equation Examples

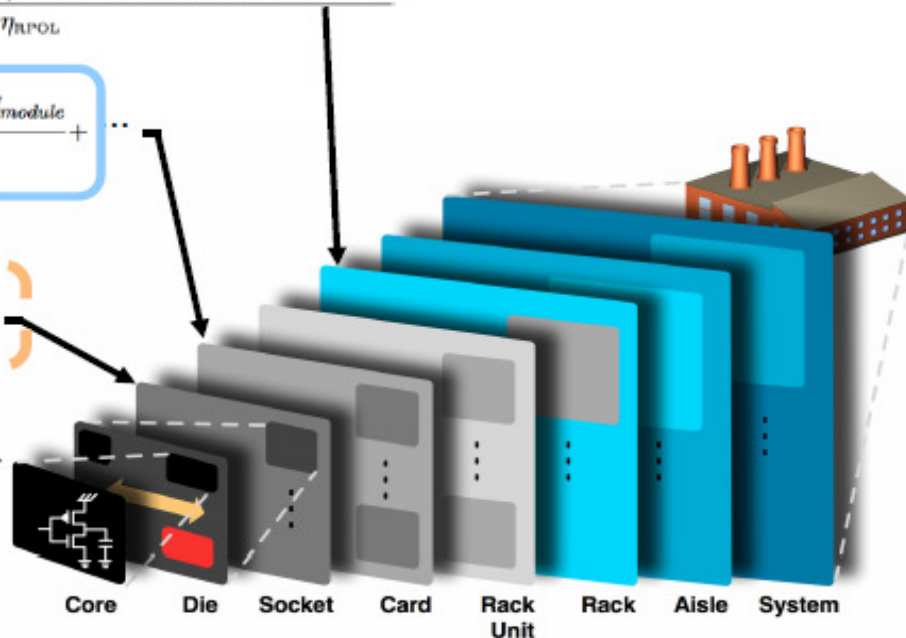
$$P_{rack} = \frac{n4_u \cdot P_{rackunit} + \sum_{x \in \{netw, XLS\}} P_{accx} + P_{netw}(B_{netw}, T_{netw}, n4_u)}{\eta_{RPOU}}$$

$$C_{card} = \frac{A2_{sw} \cdot K_{io} \cdot \left(\frac{n2_m}{A2_{app} \cdot O_{acc}(n2_m) + n2_m - A2_{app} \cdot n2_m} \right) \cdot C_{module}}{1 + \max \left(Biobus_{app} - \frac{B_{periph}(job_{bus}, 0)}{n2_m + n2_{ato}}, 0 \right)} + \dots$$

$$P_{module} = \left(\frac{n1_d \cdot P_{die}}{n_{pads} \cdot V_{dd}} \right)^2 \cdot n_{pads} \cdot R_{pads} + n1_d \cdot P_{die}$$

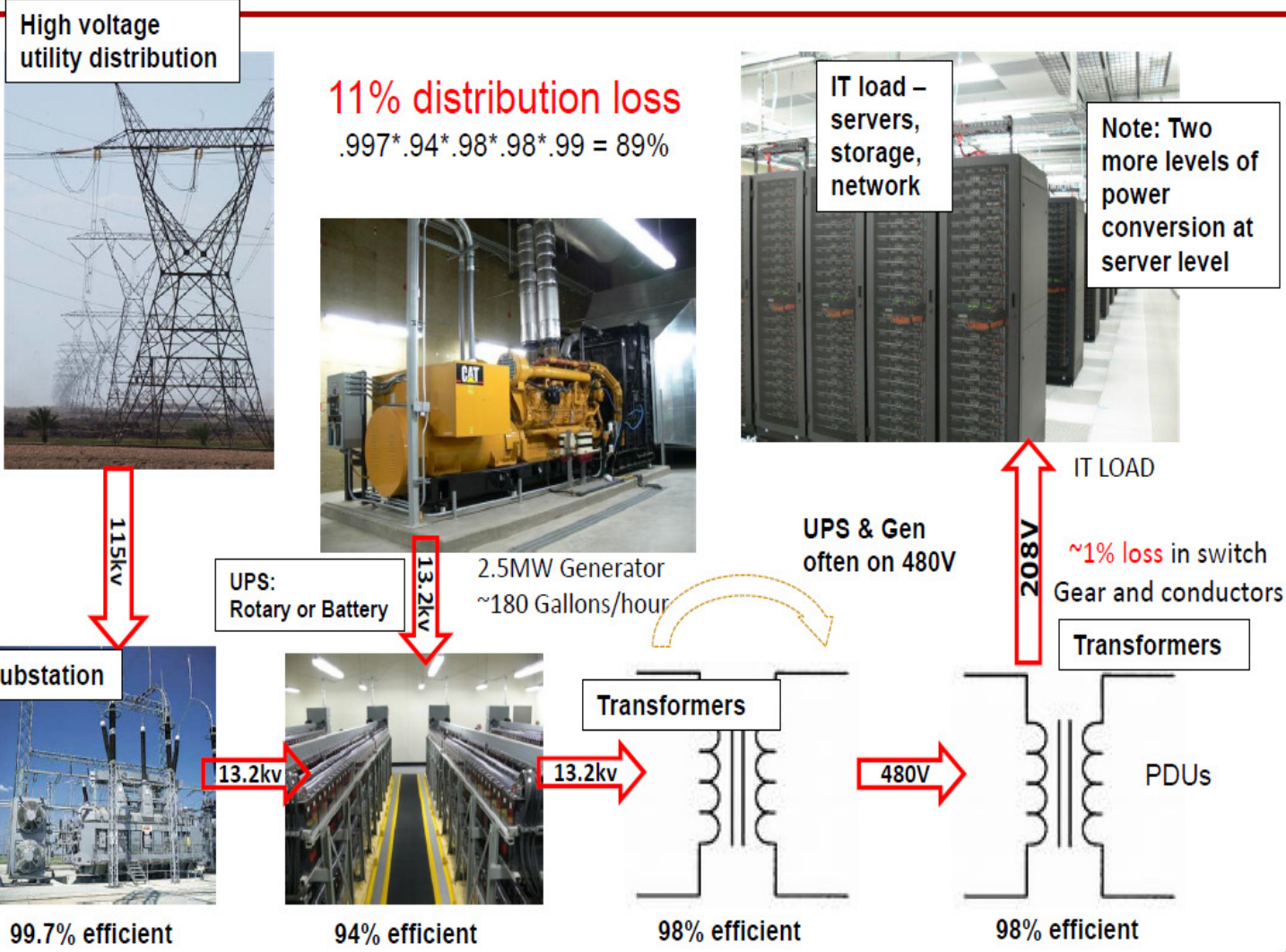
$$P_{core} = K_{pc} \cdot V_{dd}^2 \cdot f \cdot \rho_{app} + K_{is} \cdot \left(e^{\frac{q \cdot V_{bias}}{k \cdot T}} - 1 \right)$$

$$f = K_{scale} \cdot \frac{(V_{dd} - V_i)^\alpha}{V_{dd}}$$

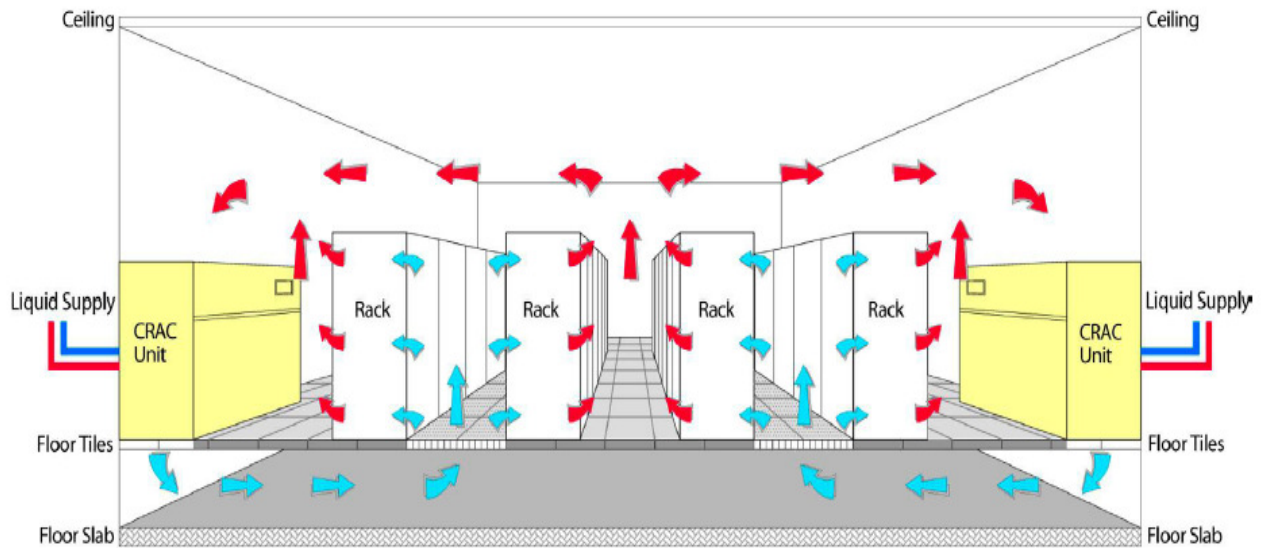


Notation: C_{xx} : computation throughput (FLOPS) P_{xx} : Power; f : clock frequency; K_{xx} : constant; V_{xx} : voltage; R_{xx} : resistance; η_{xx} : power supply efficiency; ρ : application active/idle ratio, etc.

Power Distribution (J. Hamilton)



Cooling: Cold/Hot Aisles



- CRAC = computer room air conditioning
 - Cold air goes through servers and exits in hot aisle
 - Cold aisles ~18-22C, hot aisles ~35C
 - CRAC units consume significant amount of energy!

Energy Use in a DC

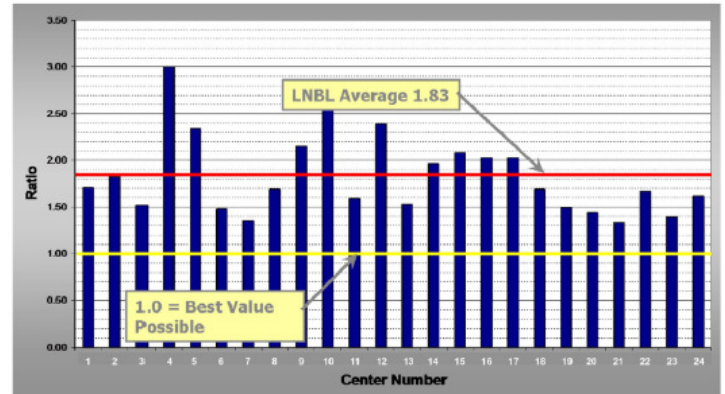
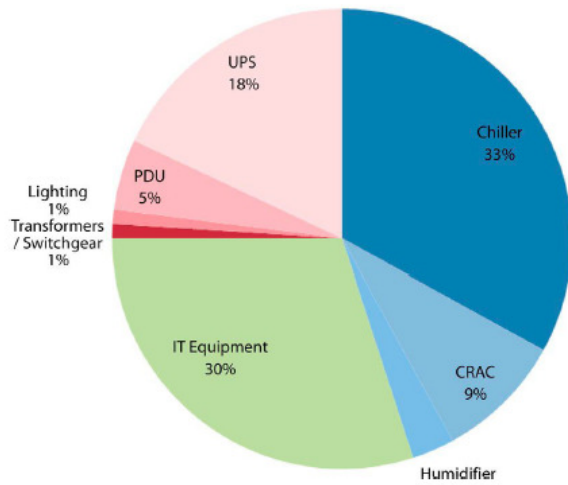


FIGURE 5.1: LBNL survey of the power usage efficiency of 24 datacenters, 2007 (Greenberg et al.)

- Cooling infrastructure is a major contributor
 - Picture from a PUE=3 data center
 - Current datacenters: PUE: 1.2 to 2

Data center infrastructure

Power Infrastructure

On-Site Power Generation

Utility

Switch Gear

UPS

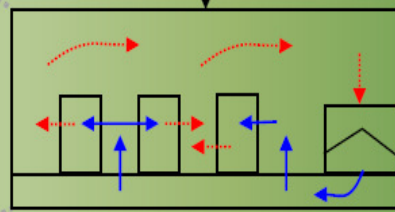
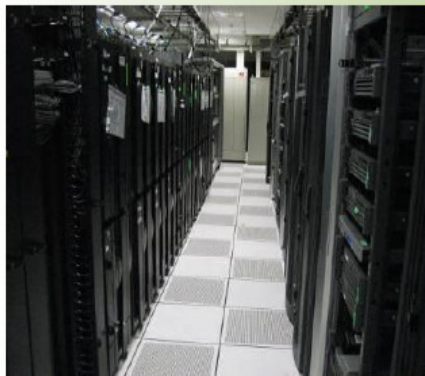
PDU

PDU

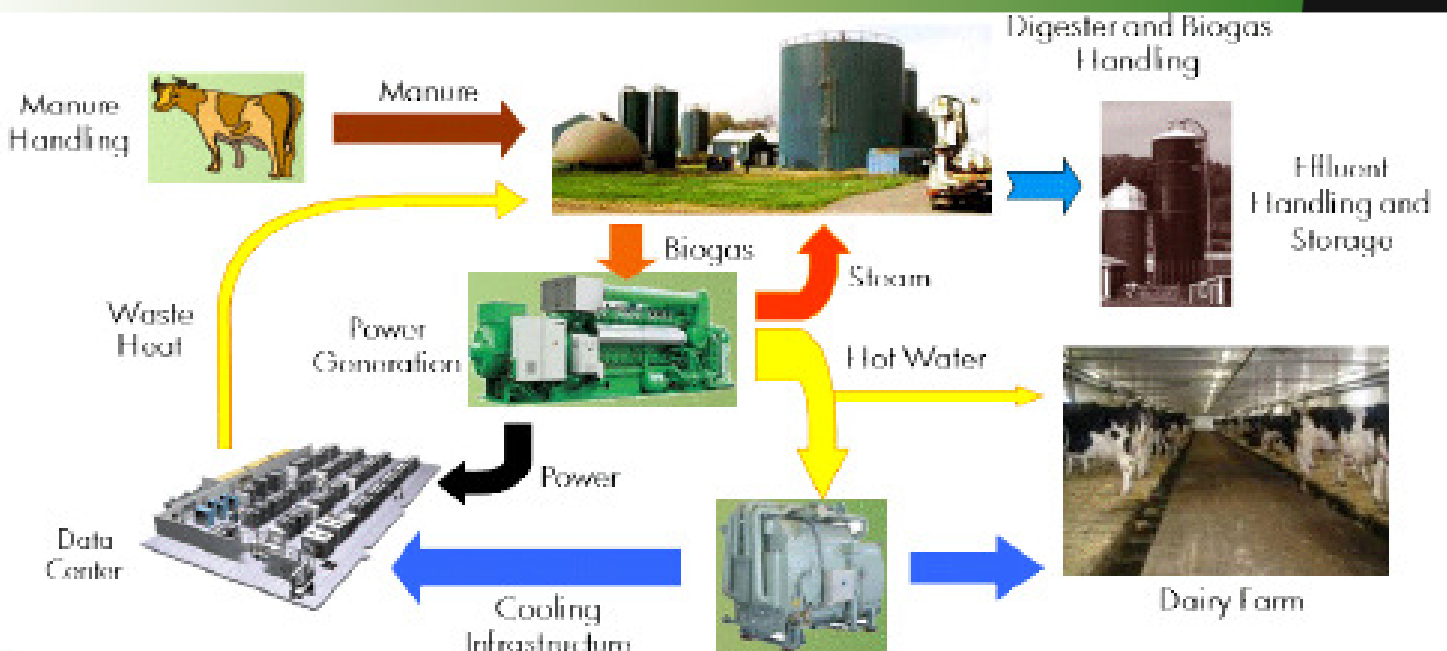
Cooling Infrastructure

Cooling Tower

Computing Infrastructure



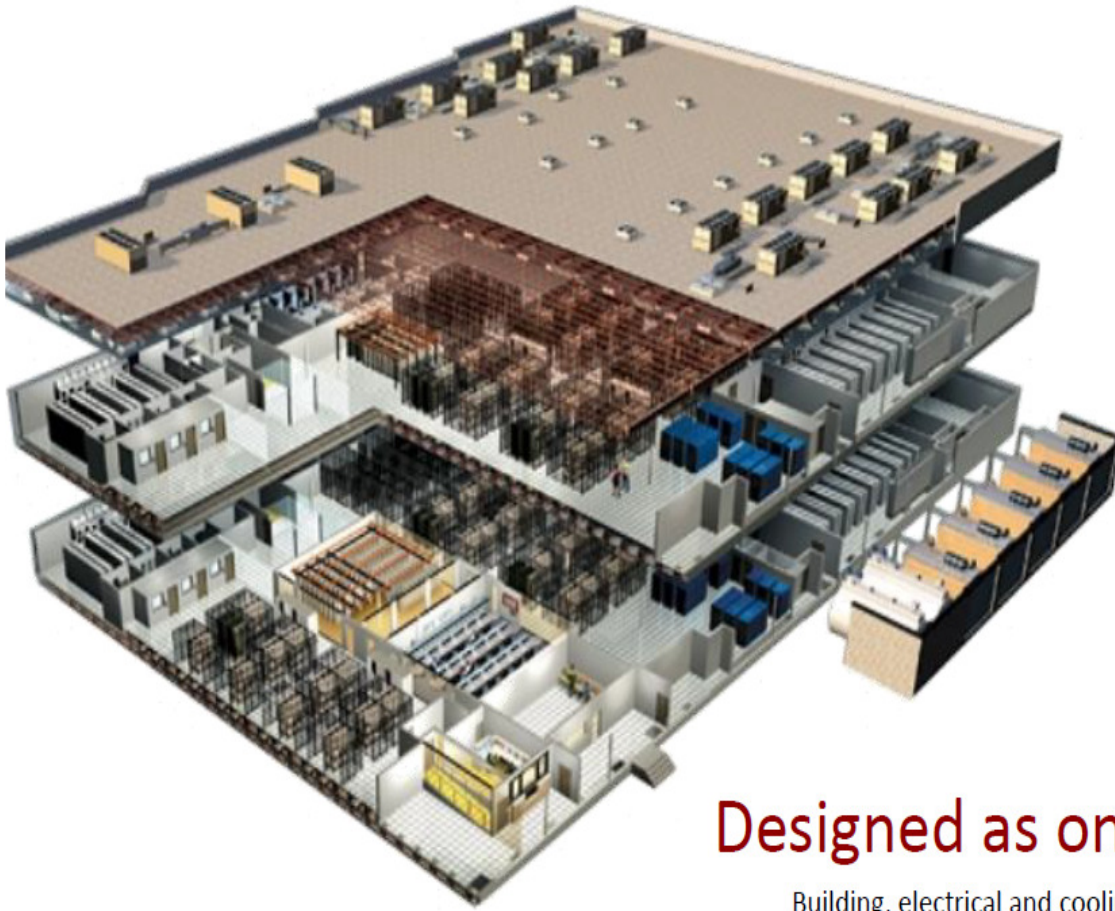
Chillers



WSC design considerations: Request-level parallelism

- Instruction-level parallelism (ILP)
 - Pipelining, speculation, OOO, ...
- Data-level parallelism (DLP)
 - Vectors, GPUs, MMX, ...
- Thread-level parallelism (TLP)
 - Multithreading, multi-cores, ...
- Request-level parallelism (RLP)
 - Parallelism among multiple decoupled tasks
 - Web servers, “map-reduce”, search, email, ...
 - Large-scale distributed systems (clusters, NOW, Grids)

WSC design considerations: The datacenter is the computer



Designed as one machine

Building, electrical and cooling infrastructure, the servers, networking, storage, (and software)

Datacenter Construction Costs



- Land: 0%-2%
- Core & Shell Costs: 5%-9%
- Architectural: 4%-7%
- Mechanical / Electrical: 70%-85%

Where the costs are:
>80% scale with power
<10% scale with space

Cost model: systems capex

■ Servers:

- 45,978 servers x \$1450 per server = \$66.7M CAPEX
- Depreciation: 3 years; cost of money = 5%
- Monthly OPEX: \$2000K

■ Networking

- Rack switches: 1150 x \$4800; Array switches: 22 x \$300K; Layer3 switch: 2 x \$500K; Border routers: 2 x \$144.8K = \$13.41M CAPEX
- Depreciation: 4 years; cost of money = 5%
- Monthly OPEX: \$309K

Cost model: opex costs

■ Power

- $[= \text{MegaWattsCriticalLoad} * \text{AveragePowerUsage} / 1000 * \text{PUE} * \text{PowerCost} * 24 * 365 / 12]$
- 0.07c/KWhr; PUE = 1.45; average power use: 80%
- \$475K OPEX (monthly)

■ People

- Security guards: $3 \times 24 \times 365 \times \20 + Facilities: $1 \times 24 \times 365 \times \30 ; Benefits multiplier: 1.3
- \$85K OPEX (monthly)

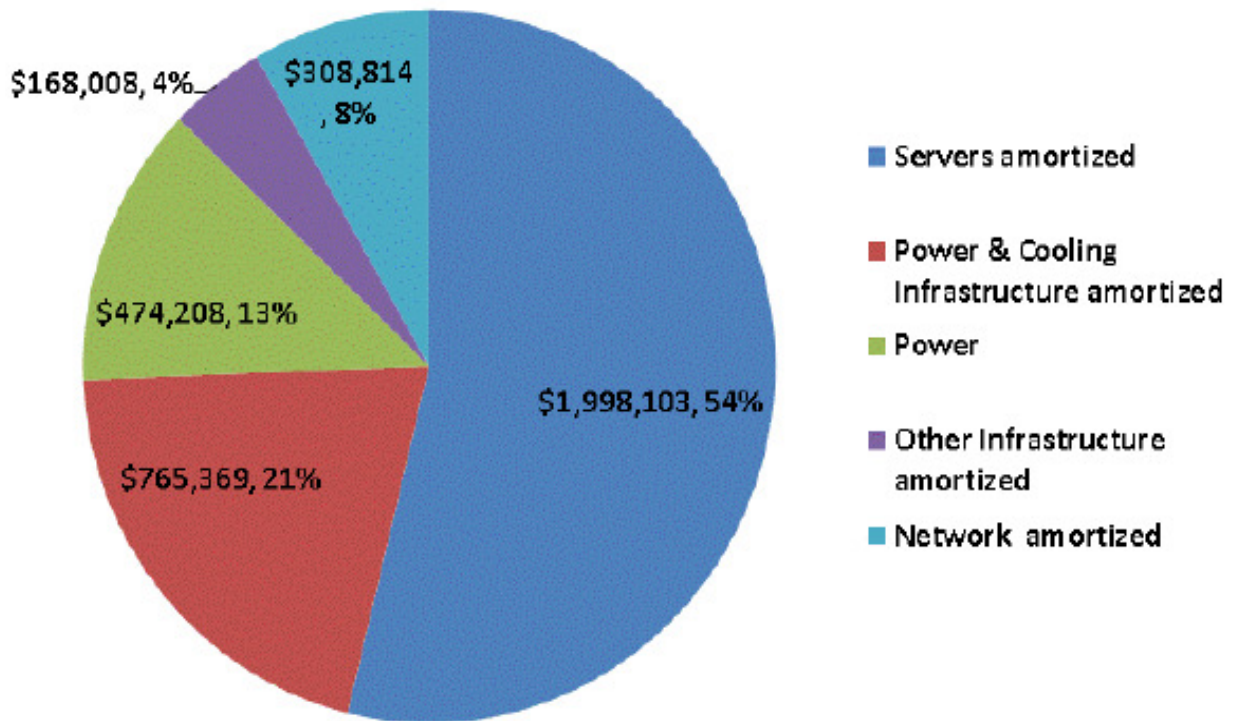
■ Network bandwidth costs to internet

- Varies by application and usage

■ Vendor maintenance fees + sysadmins

- Varies by equipment and negotiations

Monthly Costs



3yr server, 4yr network, 10yr infrastructure amortization

Enterprise Vs WSC: a Cost Perspective

- Enterprise computing approach
 - Largest cost is people -- scales roughly with servers (~100:1 common)
 - Enterprise interests focus on consolidation & utilization
 - Consolidate workload onto fewer, larger systems
 - Large SANs for storage & large routers for networking
- Internet-scale services approach
 - Largest costs is server H/W
 - Typically followed by cooling, power distribution, power
 - Networking varies from very low to dominant depending upon service
 - People costs under 10% & often under 5% (>1000+:1 server:admin)
 - Services interests centered around work-done-per-\$ (or watt)
- Observations
 - People costs shift from top to nearly irrelevant.
 - Focus instead on work done /\$ & work done/watt

Google's data center at The Dalles, OR



■ Datacenter at The Dalles, Oregon

- Moderate climate, cheap hydroelectric power, near internet backbone fiber
- 75000 square feet

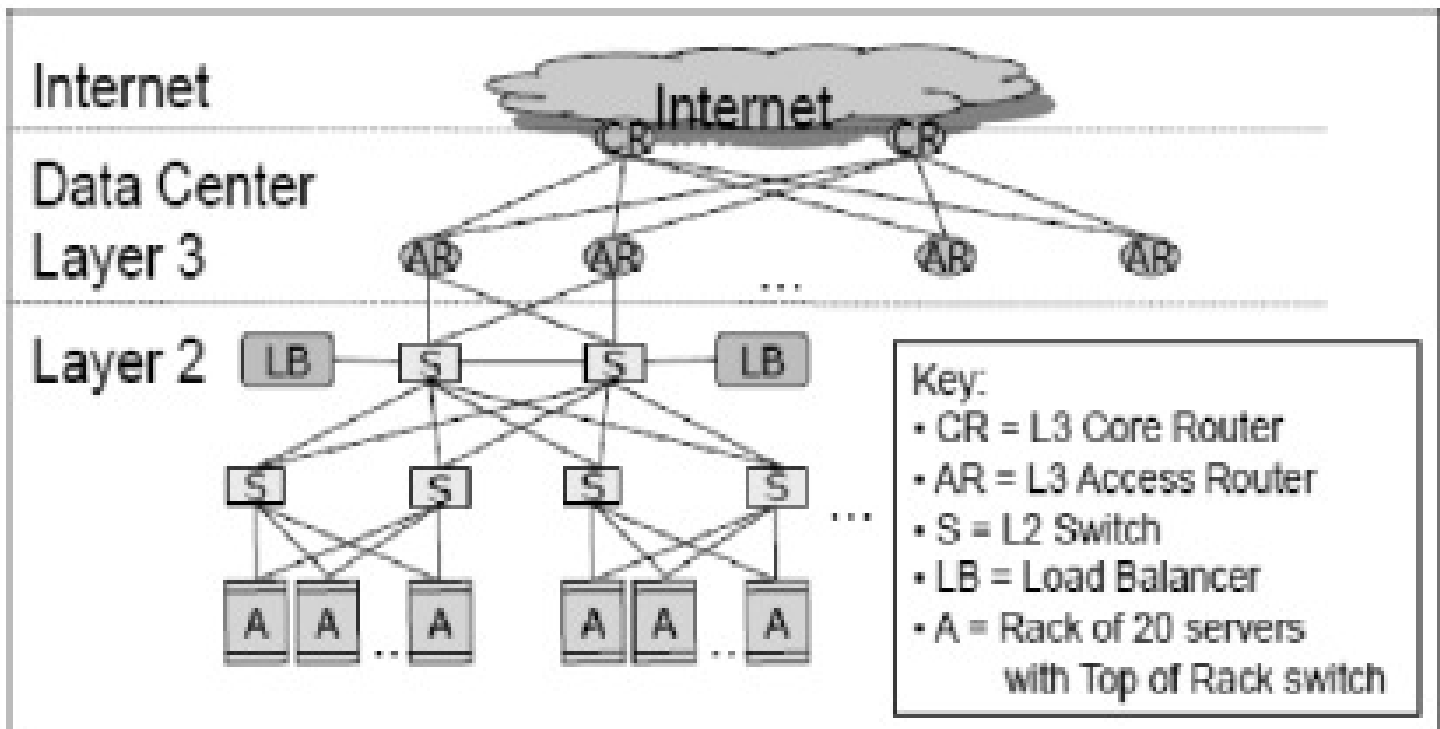
Google's data center at The Dalles, OR



- **MS Quincy Datacenter**
 - 470k sq feet (10 football fields)
 - Next to a hydro-electric generation plant
 - At up to 40 MegaWatts, \$0.02/kWh is better than \$0.15/kWh 😊
 - That's equal to the power consumption of 30,000 homes

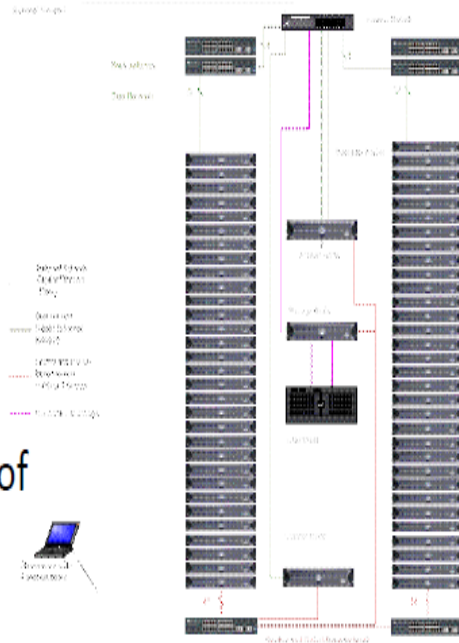


View from the east side of the facility showing the layout of the buildings and infrastructure.



■ Rack switch = 48-port ethernet 1Gig switch

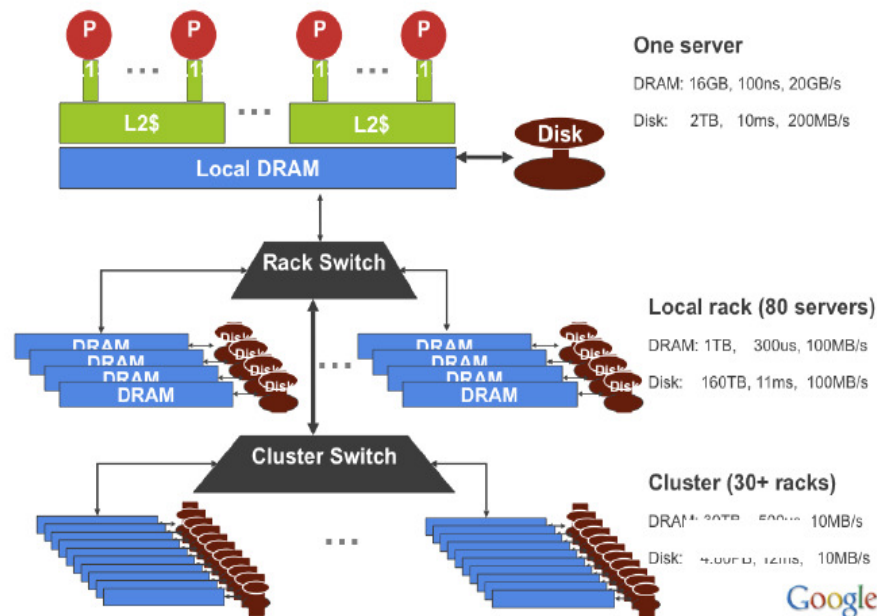
- Commodity switch \geq \$30 per port
 - Infiniband \sim \$500/port
- One Switch per two racks
- 40 server ports; 2-8 uplink ports
 - Oversubscription ratio
 - Programmer burden
- Bandwidth within rack is same irrespective of sender/receiver



■ Array switch

- More expensive: 10X more BW = 100X more \$
- High-end switches feature-rich (mgmt, inspectic CAMs, FPGAs)
- 480 1Gbit links, few 10Gbit ports to datacenter routers
- Manage oversubscription carefully

WSC Storage Hierarchy: A Programmer's Perspective

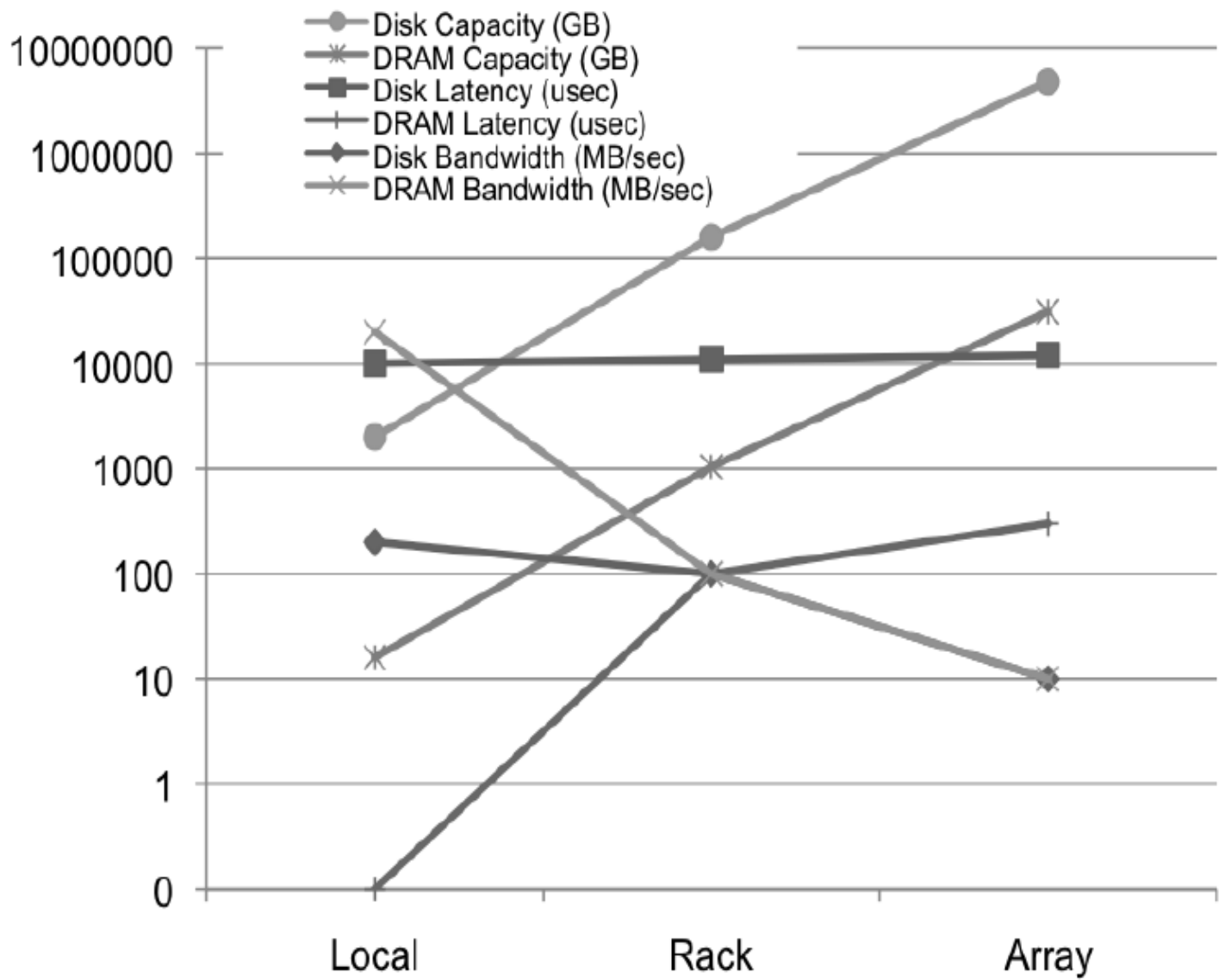


■ Interesting observations

- Remote memory is often faster than local disk
- Bandwidth bottlenecks

	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)	2,000	160,000	4,800,000

	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)	2,000	160,000	4,800,000



Useful Numbers

Courtesy of Jeff Dean, Google

■ L1 cache reference	0.5 ns
■ Branch mispredict	5 ns
■ L2 cache reference	7 ns
■ Mutex lock/unlock	25 ns
■ Main memory reference	100 ns
■ Compress 1K bytes with Zippy	3,000 ns
■ Send 2K bytes over 1 Gbps network	20,000 ns
■ Read 1 MB sequentially from memory	250,000 ns
■ Round trip within same datacenter	500,000 ns
■ Disk seek	10,000,000 ns
■ Read 1 MB sequentially from disk	20,000,000 ns
■ Send packet CA->Europe->CA	150,000,000 ns

Useful Back of the Envelope Math

Courtesy of Jeff Dean, Google



- How long to generate image results page (30 thumbnails)?

- Design 1: Read serially, thumbnail 256K images on the fly
 - $30 \text{ seeks} * 10 \text{ ms/seek} + 30 * 256\text{K} / 30 \text{ MB/s} = 560 \text{ ms}$

- Design 2: Issue reads in parallel
 - $10 \text{ ms/seek} + 256\text{K read} / 30 \text{ MB/s} = 18 \text{ ms}$
 - (Ignores variance, so really more like 30-60 ms, probably)

- Lots of other options
 - Caching (single images? whole sets of thumbnails?)
 - Pre-computing thumbnails
 - ... Back of the envelope helps identify most promising...

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%

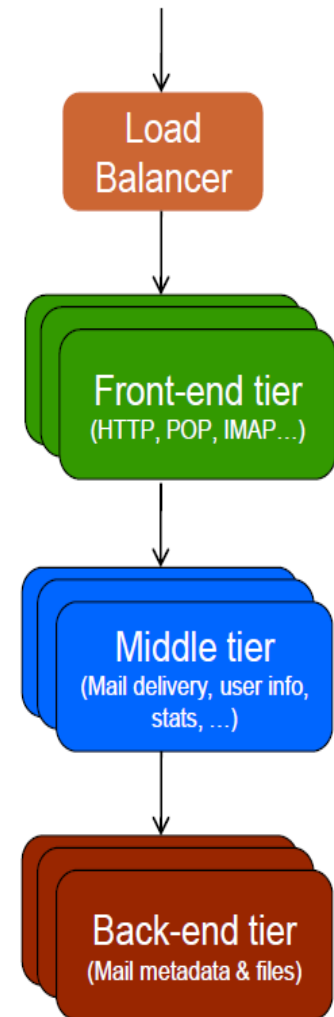
Figure 6.12 Negative impact of delays at Bing search server on user behavior [Brutlag and Schurman 2009].

- 10000 processors with 4GB per server => following rates of unrecoverable errors in 3 years of operation [IBM study]
 - Parity only: about 90,000; 1 unrecoverable failure every 17 minutes
 - ECC only: about 3500; one unrecoverable or undetected failure every 7.5 hours
 - Chipkill: about 6; one unrecoverable/undetected failure every 2 months
 - 10,000 server chipkill = same error rate as a a 17-server ECC system

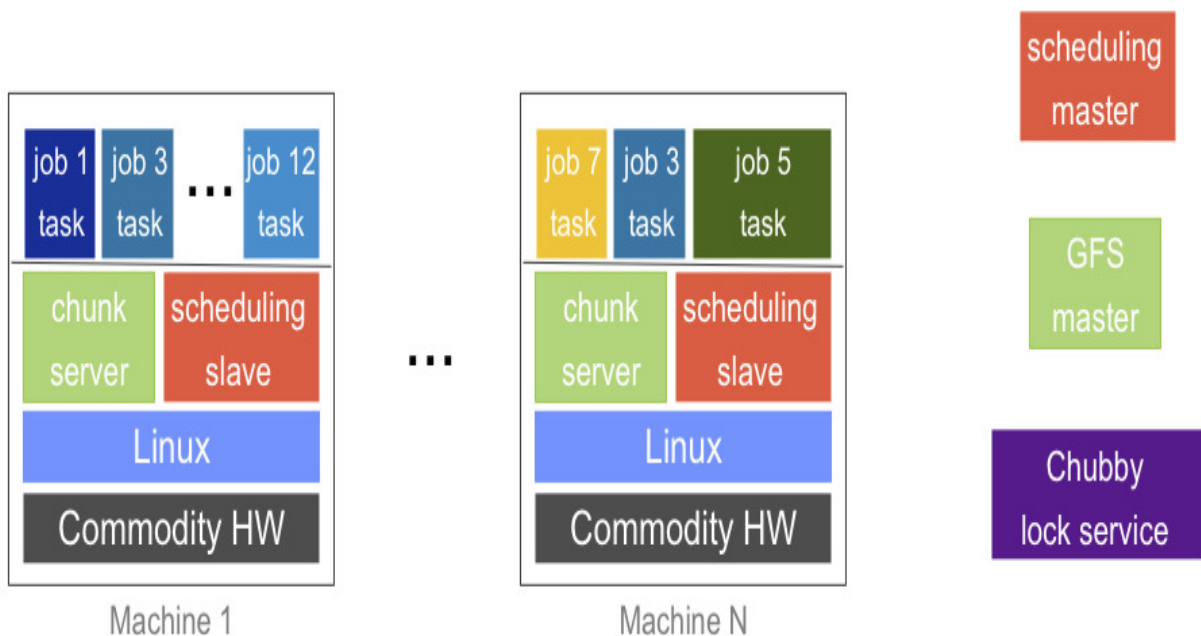
- Schroeder 2009: Google WSC error rates
 - Average DIMM had 4000 correctable errors and 0.2 uncorrectable errors per year
 - With chipkill, for one third of the servers, one memory error is corrected every 2.5 hours
 - With just parity error, one third of the machiens would spend 20% of time rebooting (5 minutes reboot time)
 - Google 2000 consistency checking in software for DRAM errors, but with cost-effective DRAM error checking, move to hardware

Example 3-tier App: WebMail

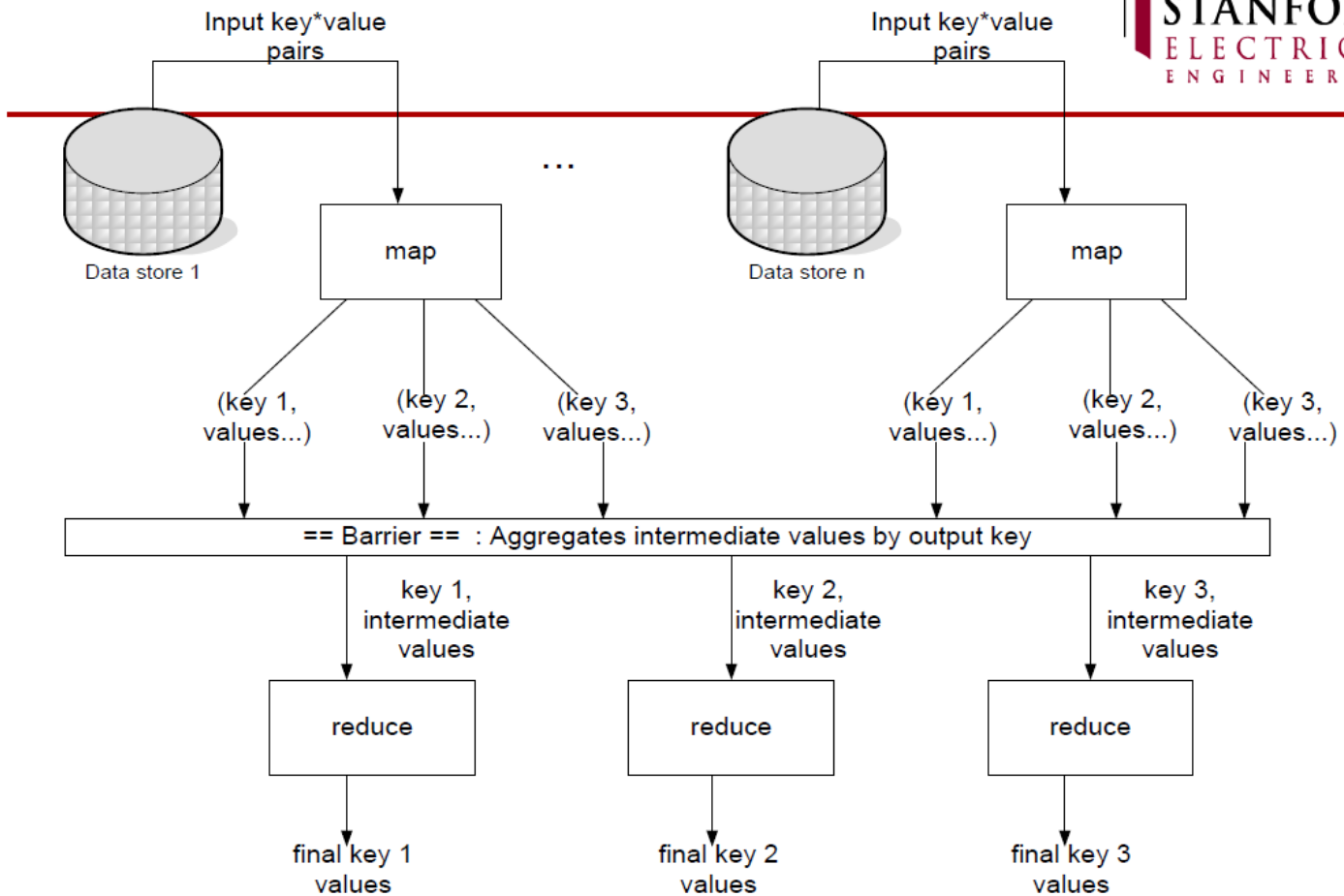
- May include thousands of machines, PetaBytes of data, and billions of users
- 1st tier: protocol processing
 - Typically stateless
 - Use a load balancer
- 2nd tier: application logic
 - Often caches state from 3rd tier
- 3rd tier: data storage
 - Heavily stateful
 - Often includes bulk of machines



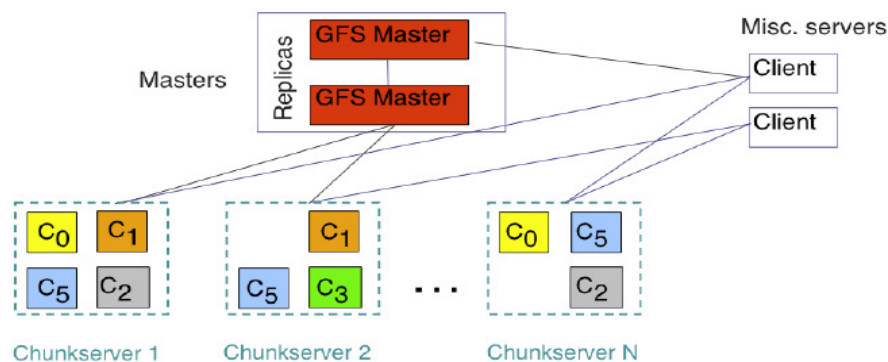
Example: Google Cluster Environment



- 1000s of machines, typically in few configurations
- File system (GFS) + Cluster scheduling system are core services
- Typically 100s to 1000s of active jobs
 - Some w/1 task, some w/1000s
 - Mix of batch and low-latency, user-facing production jobs

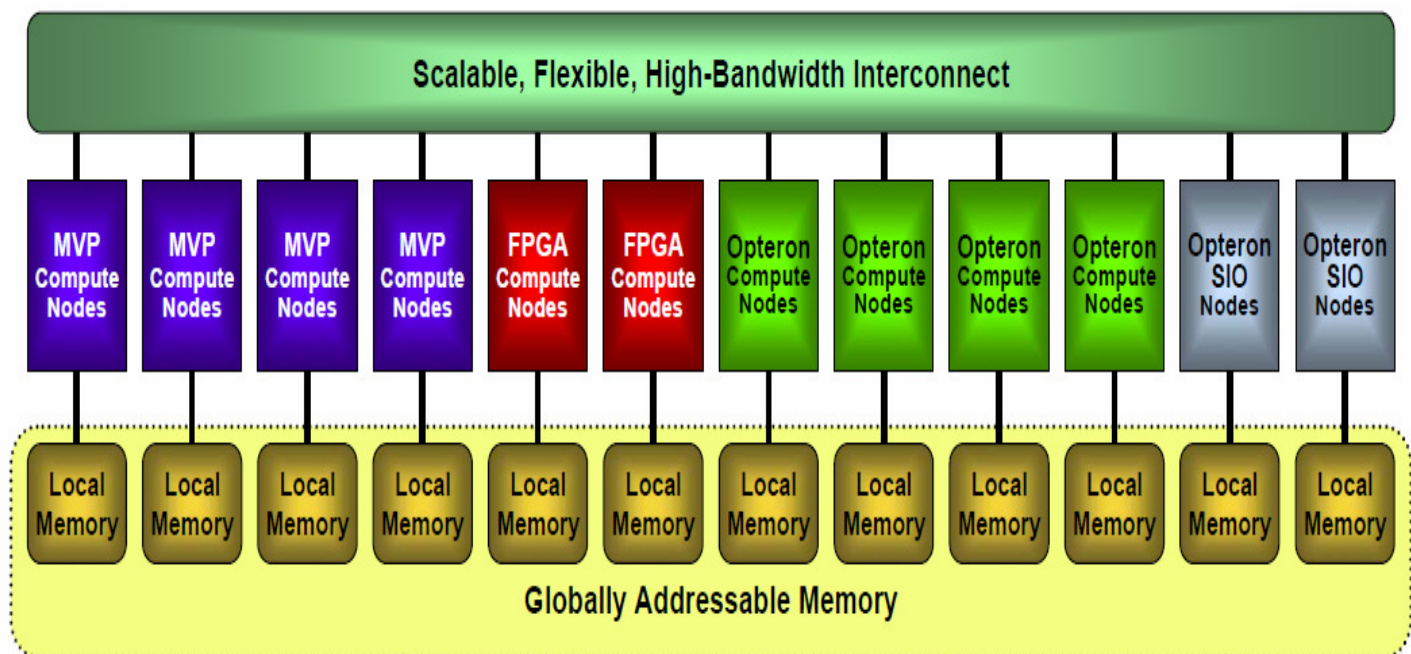


Example: Google File System



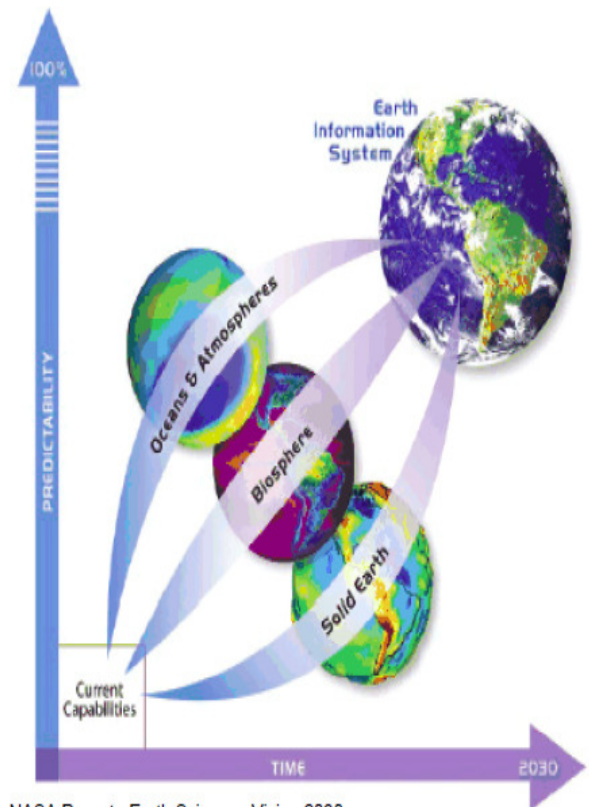
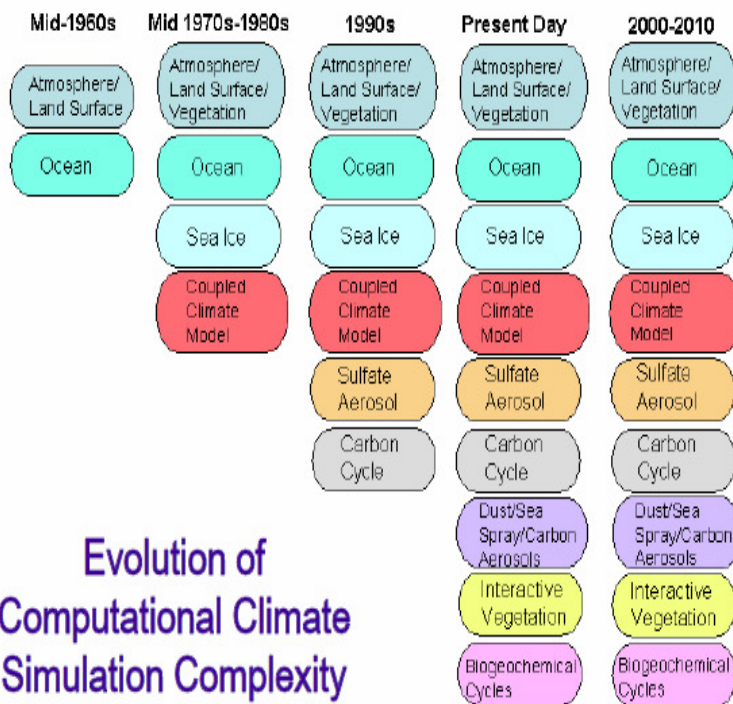
- Distributed file system using server disks
 - Master provides a naming service
 - Clients access data directly
- Replication support for availability & throughput
 - E.g. replicate across racks to survive node/switch failures

Cascade System Architecture



- Globally addressable memory with unified addressing architecture
- Configurable network, memory, processing and I/O
- Heterogeneous processing across node types, and within MVP nodes
- Can adapt at **configuration** time, **compile** time, **run** time

Increasingly Complex Application Requirements Earth Sciences Example



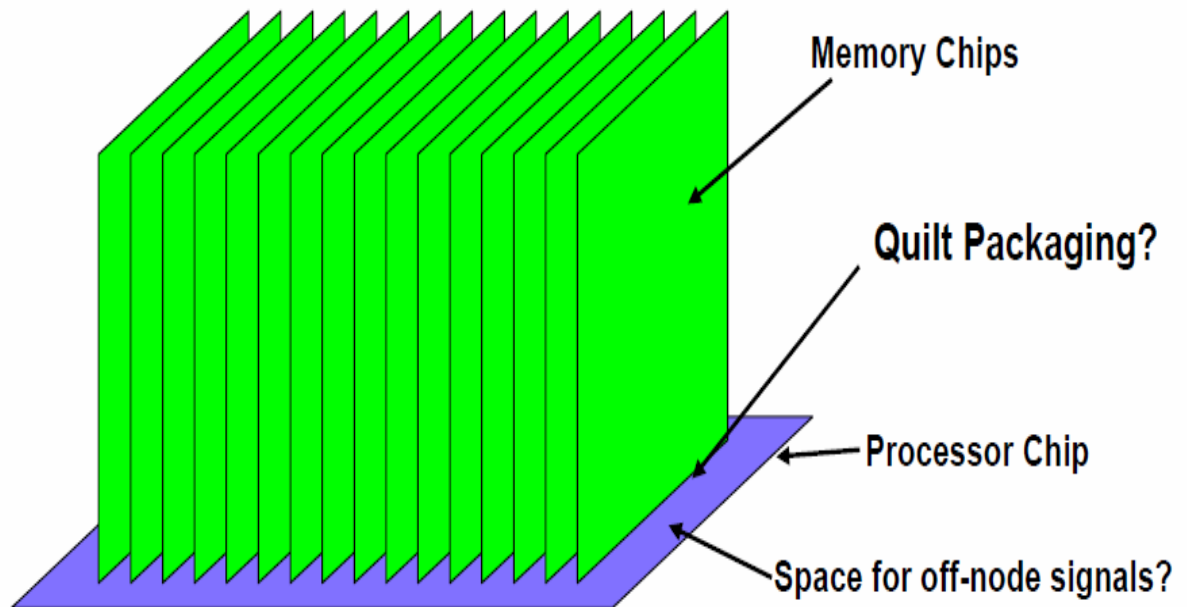
NASA Report: Earth Sciences Vision 2030

International Intergovernmental Panel on Climate Change, 2004, as updated by Washington, NCAR, 2005

Increased complexity and number of components lends itself well to a variety of processing technologies

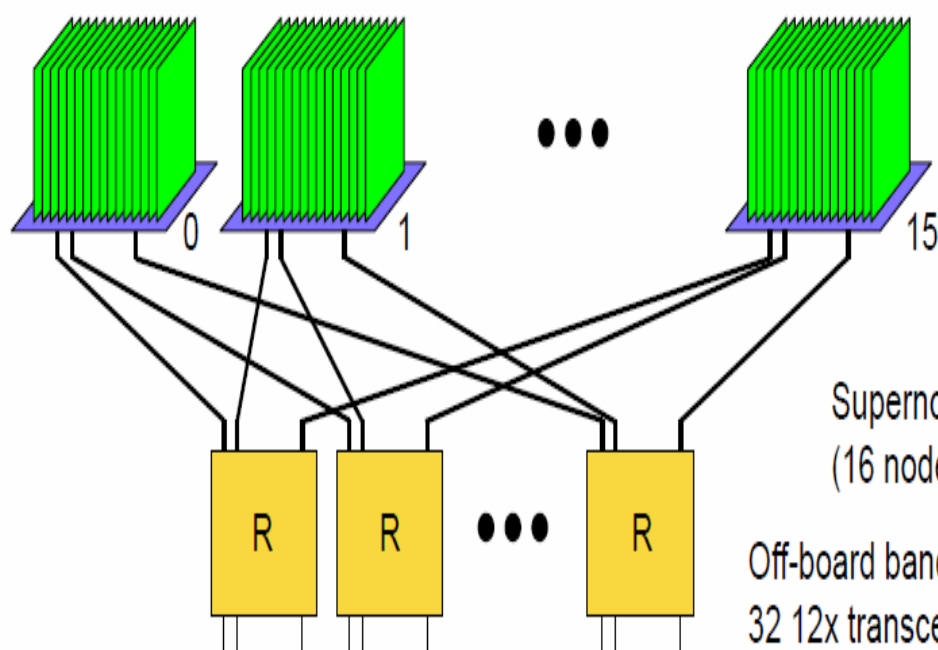
- Similar trends in astrophysics, nuclear engineering, CAE, etc.
 - Higher resolution, multi-scale, multi-science

3D Node: Processor + Orthogonal Memory Chips



- Capacity
 - 8-32 memory chips @ 1 GB each = 8-32 GB per node
- Bandwidth
 - 5 μm pitch wires (10 μm per diff signal), 15mm edge \Rightarrow 1500 signals per memory chip
 - Need to keep signaling rates to < 10 Gbps with memory periphery transistors
 - Assume 512 bits/dir @ 8.25 Gbps, packetized protocol, 80% read efficiency
 - \Rightarrow 320 GB/s read bandwidth per memory chip (1.28W at 0.5 pJ/bit)
 - \Rightarrow 2.5-10 TB/s read bandwidth per node with 8-32 memory chips
 - Could nicely feed a 5-10 TF node
 - Probably still too much power in memory chips to support this...

Example Board Architecture



Shown as fat-tree. Could consider flattened butterfly or other topology for on-board or off-board links.

Aggregate node bandwidth:
 $(16 \text{ nodes}) * (4 \text{ TB/s/node}) = 80 \text{ TB/s}$

Supernode bandwidth:
 $(16 \text{ nodes}) * (64 \text{ sigs/node}) * (25 \text{ Gbps}) = 6.4 \text{ TB/s}$

Off-board bandwidth:
32 12x transceivers @ 16 Gbps = 768 GB/s

- Can treat as 16 nodes for highest aggregate memory bandwidth
- Could combine into 2, 4, 8 or 16-node “super-nodes”
 - Flat addressing, latency and bandwidth
 - Hashed to avoid bank conflicts
 - Would still want compiler to exploit locality within a single node
 - Either via explicit local segments or via caching (possibly in main memory)
- Inter-node signaling shown using conservative technology extrapolations
 - Could also consider high-bandwidth on-board technologies (quilting, capacitive coupling, optics?, etc.) to boost super-node bandwidth even further

One Last Exascale Challenge (4)

- Need to build systems for *tomorrow's* applications
 - Irregular, dynamic, sparse, heterogeneous....
 - Codes that don't exhibit locality, or that have limited per-thread concurrency
 - Need to start, stop, move and synchronize computation efficiently
 - Let's not solve the scaling problem for the easy apps and declare success

- "Leave no application behind"

Key Challenges to Get to the Zettascale

- I accept that CMOS won't get there due to power and other reasons.
- New computing technologies will likely require new architectures, new execution models and new programming models
 - Exploitation of locality will be key
 - Very likely to involve massive threading and lightweight thread migration
- Architects need to understand the technological sandbox within the next dozen years or so...
- Absolutely must have better programming models where humans don't have to coordinate all the data distribution and communication
 - Would be nice if those were the same programming models used at Exascale
- Need to have much more sophisticated and automated tools for performance and correctness analysis
 - Presumably involving pervasive introspection

- I am an optimist. I think we *will* get to zettaflop computing using some interesting post-CMOS technology by ~2030. It will look different than any of us imagine today. Good occasion to retire.

