大纲

- **前言**
  - 为什么需要“大规模计算” [HPC, DL, Business platform system, Cloud已经合流]
    - 导入 – 科学计算(天气预报)，DL, 互联网平台(Google, Amazon, Alibaba, MeiTuan, …)
- **基础篇**
  - 并发程序的样子 – Divide & Conquer, Model & Challenges, PCAM, Data/Task, …
    - 天气预报的计算
  - 运行环境
    - 硬件 – 自己梳理的3个方案 – Shared/Unshared Memory, Hybrid
    - 系统软件 – 协议栈, Modern OS, Distributed Job Scheduler, GTM等
- **算法级篇**
  - OpenMP, MPI, CUDA (DL的实现), Big Data 中的MR/Spark等 (只涉及在Big Data SDK之上的编程；大数据本身的介绍放到后一部分)
- **系统级篇 – 互联网平台的实现**
  - “秒杀”的技术架构
  - 计算广告
  - 系统架构 (HTAP等)
    - Flink, ClickHouse, MaxCompute, ELK …

Chapter 3: Large Scale Computing Systems

Faster for larger data

- von Neumann architecture
  - Foundation of modern computers
  - 1960 2 CPUs
- 1962 Channel
  - Origin of concurrent programming
- Parallel
  - Vector processor, Multi-core, later GPU/CUDA
- Distributed
  - Cluster, Grid, …
- Now Clouding – Virtualizing computer systems for so-called Big Data
  - IaaS, PaaS, SaaS, …

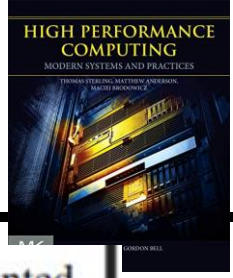
Early History (-1969)

Year	Name	Peak speed	Location
1943	Colossus	5000 char/sec	Bletchley Park, England
1945	Manchester Mark I	500 inst/sec	Manchester, England
1950	MIT Whirlwind	20 kIPS	MIT, USA
1956	IBM 704	20 kIPS 12 kFLOPS	
1959	IBM 7090	210 kFLOPS	USAF, USA
1960	LARC	500 kFLOPS (2 CPUs)	Lawrence Livermore Lab. USA
1961	IBM 7030	1.2 MIPS 600 kFLOPS	Los Alamos Lab. USA
1965	CDC 6600	10 MIPS 3 MFLOPS	Lawrence Livermore Lab. USA
1969	CDC 7600	36 MFLOPS	Lawrence Livermore Lab. USA

The Early Days – 1950

- **MIT Whirlwind** was the first computer that operated in real time, used video displays for output and was the first digital flight simulator!
  - All previous computers ran in batch mode, i.e. a series of paper tapes/cards were set up as input in advance, fed into the computer to calculate and print results.
  - For simulating an aircraft control panel, Whirlwind need to operate continually on an ever-changing series of inputs ➔ need high-speed stored-program computer.
  - Original design was too slow due to the Williams tubes and so core memory (ferrite rings that store data in polarity of magnetic field) was created ➔ doubled speed ➔ design successfully mass-produced by IBM

WHIRLWIND—THE FIRST SUPERCOMPUTER



The Whirlwind vacuum-tube-based digital electronic computer was the first modern computer architecture and represented the state of the art in high-speed calculation. It may be considered the first general-purpose supercomputer. Developed at Massachusetts Institute of Technology (MIT) under successive projects sponsored by the US Navy and then the US Air Force, its intended applications stressed performance initially for flight simulation and ultimately for radar-based air defense. Whirlwind employed a bit parallel logic design with 16-bit words performed and simultaneously implemented with vacuum tubes. It stored and controlled access to 2048 words using electrostatic storage tubes with an original (never achieved) bit density of 1024 bits per unit. The control structure incorporated an innovative diode matrix for speed as well as simplicity and flexibility of design. Its initial design was completed in 1947 by Jay Forrester and Robert Everett; it became operational in 1951 and consisted of 5000 vacuum tubes. Whirlwind was upgraded in 1953 with a new kind of memory developed by Forrester that used arrays of magnetic cores in stacks, replacing the slow and less reliable vacuum-tube storage. The resulting performance of up to 40 K instructions per second made Whirlwind the fastest computer of its time, dramatically increased its reliability and reducing its cost of operation.

The Whirlwind computer and its many innovations had far-reaching impacts on the field of computing. The invention of core memory redefined computer architecture for the next 2 decades, and is one of the main reasons why digital computers became commercially practical. Bit-parallel logic units became the norm for data processing. The diode—matrix control unit inspired Maurice Wilkes to conceive of microcontrollers and microprogramming, upon which future computers would be based at least until the microprocessor era. Whirlwind was the prototype for the first major parallel computer system, SAGE, employed as the original US air defense system. A spinoff of the Whirlwind project was the founding of DEC, which invented the minicomputer and rose to become the world's second-largest computer company in the 1980s. A second spinoff, MITRE, a major defense research contractor, can also be attributed to Whirlwind. With the final operational deployment of the Whirlwind computer, the future direction of high performance computer architecture was established.

ETH zürich

The beginnings of electronic computing

1939-42: Atanasoff-Berry Computer - Iowa State Univ.

1938: Konrad Zuse's Z1 - Germany

1943/44: Colossus Mark 1&2 - Britain

Zuse and Z3 (**1941**)

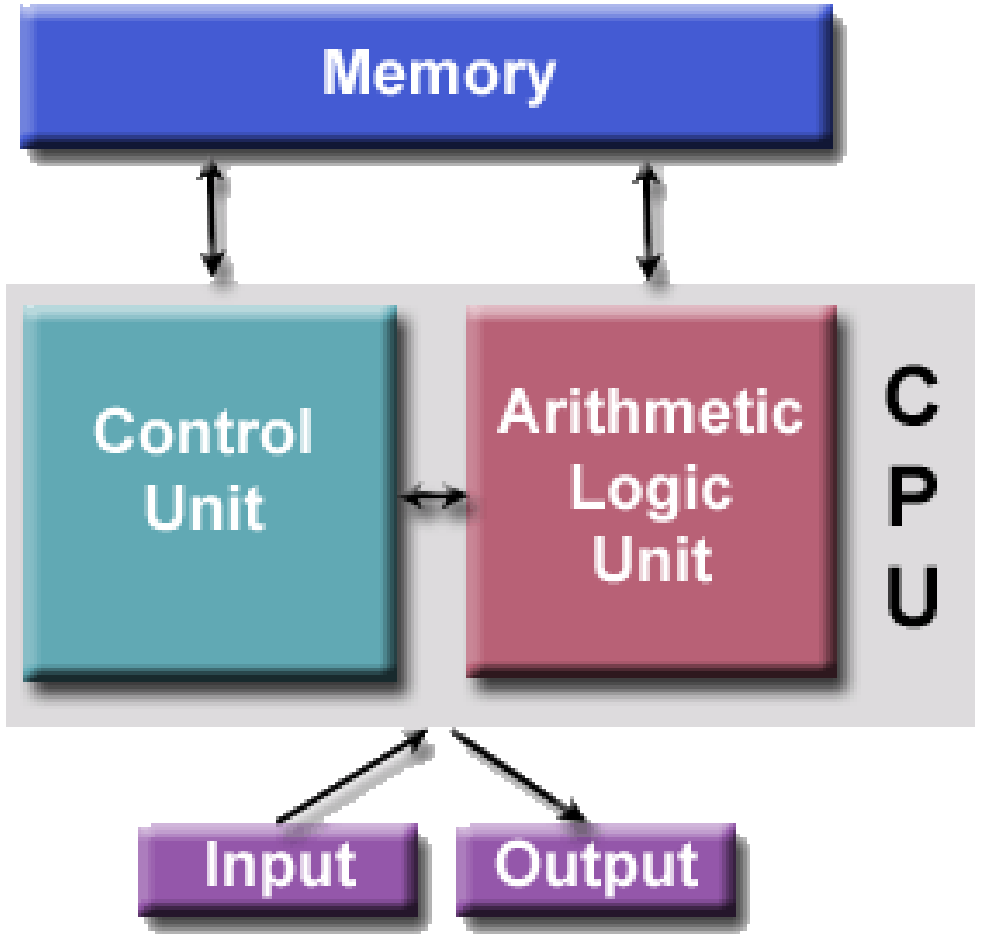
Z4 @ ETH (1950-54)

1945-51: UNIVAC I
Eckert & Mauchly - "first commercial computer"

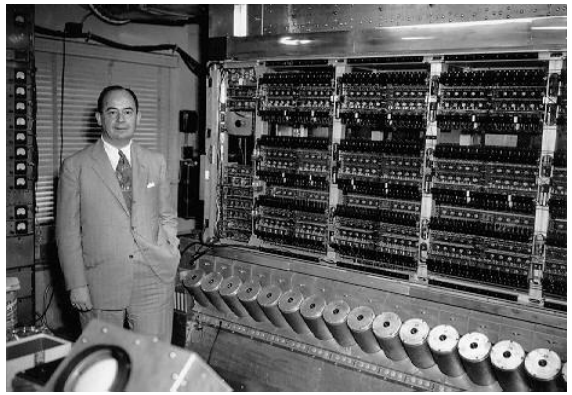
1945: John von Neumann report that defines the "von Neuman" abstraction

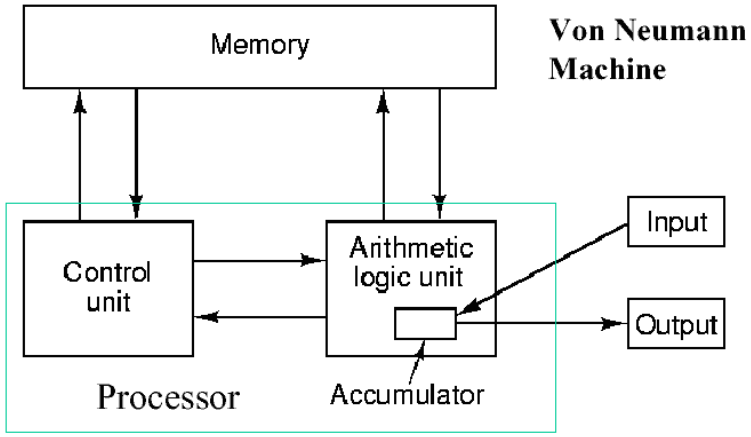
CSCS

von Neumann Architecture



Still von Neumann is the basis now





Walk-Through: *c=a+b*

1. Get next instruction
2. Decode: Fetch ***a***
3. Fetch ***a*** to internal register
4. Get next instruction
5. Decode: fetch ***b***
6. Fetch ***b*** to internal register
7. Get next instruction
8. Decode: add ***a*** and ***b*** (***c*** in register)
9. Do the addition in ALU
10. Get next instruction
11. Decode: store ***c*** in main memory
12. Move ***c*** from internal register to main memory

Note: Some units are idle while others are working…waste of cycles.
Pipelining (modularization) & Cashing (advance decoding)…parallelism

Chapter 3: Large Scale Computing Systems

Faster for larger data

- von Neumann architecture
  - Foundation of modern computers
- 1962 Channel
  - Origin of concurrent programming
- Parallel
  - Vector processor, Multi-core, later GPU/CUDA
- Distributed
  - Cluster, Grid, …
- Now Clouding – Virtualizing computer systems for so-called Big Data
  - IaaS, PaaS, SaaS, …

The Early Days – 1960

LARC was designed for multiprocessing, but never implemented

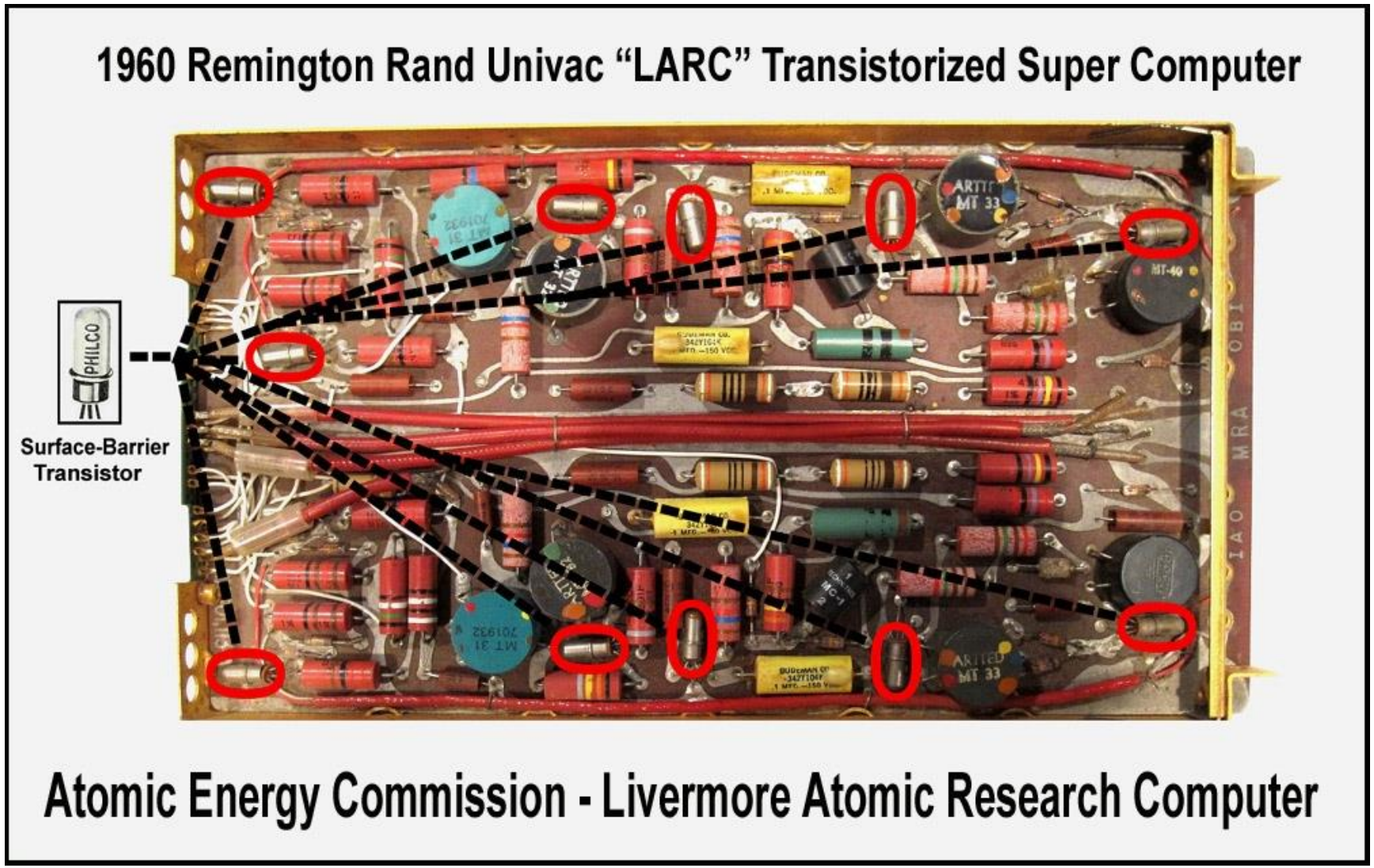
- **The LARC (Livermore Advanced Research Computer) was the first true supercomputer – it was designed for multiprocessing, with 2 CPUS and a separate I/O processor.**
  - Used 48 bits per word with a special form of decimal arithmetic ➔ 11 digit signed numbers. Had 26 general purpose registers with an access time of 1 microsecond.
  - The I/O processor controlled 12 magnetic drums, 4 tape drives, a printer and a punched card reader.
  - Had 8 banks of core memory (20000 words) with an access time of 8 microseconds and a cycle time of 4 microseconds.

Both examples had only one *Computer*, so no multiprocessor LARCs were ever built.

- The UNIVAC **LARC**, short for the *Livermore Advanced Research Computer*, is a mainframe computer designed to a requirement published by Edward Teller in order to run hydrodynamic simulations for nuclear weapon design. It was one of the earliest supercomputers.
- LARC supported multiprocessing with two CPUs (called *Computer*s) and an Input/output (I/O) Processor (called the *Processor*). Two LARC machines were built, the first delivered to Livermore in June 1960, and the second to the Navy's David Taylor Model Basin.
- **Both examples had only one *Computer***, so no multiprocessor LARCs were ever built.[2]

UNIVAC LARC at Livermore

LARC circuit board



The Early Days – 1961

- **IBM feared the success of the LARC so designed the 7030 to beat it.**
  - Initially designed to be 100x faster than the 7090, it was only 30x faster once built. An embarrassment for IBM ➔ big price drops and in the end only 9 were sold (but only 2 LARCs were built so not all bad!)
  - Many of the ideas developed for the 7030 were used elsewhere, e.g. **multiprogramming**, memory protection, **generalized interrupts**, **the 8-bit byte**, **instruction pipelining**, **prefetching and decoding**, and memory interleaving were used in many later supercomputer designs.
  - Ideas also used in modern commodity CPUs!

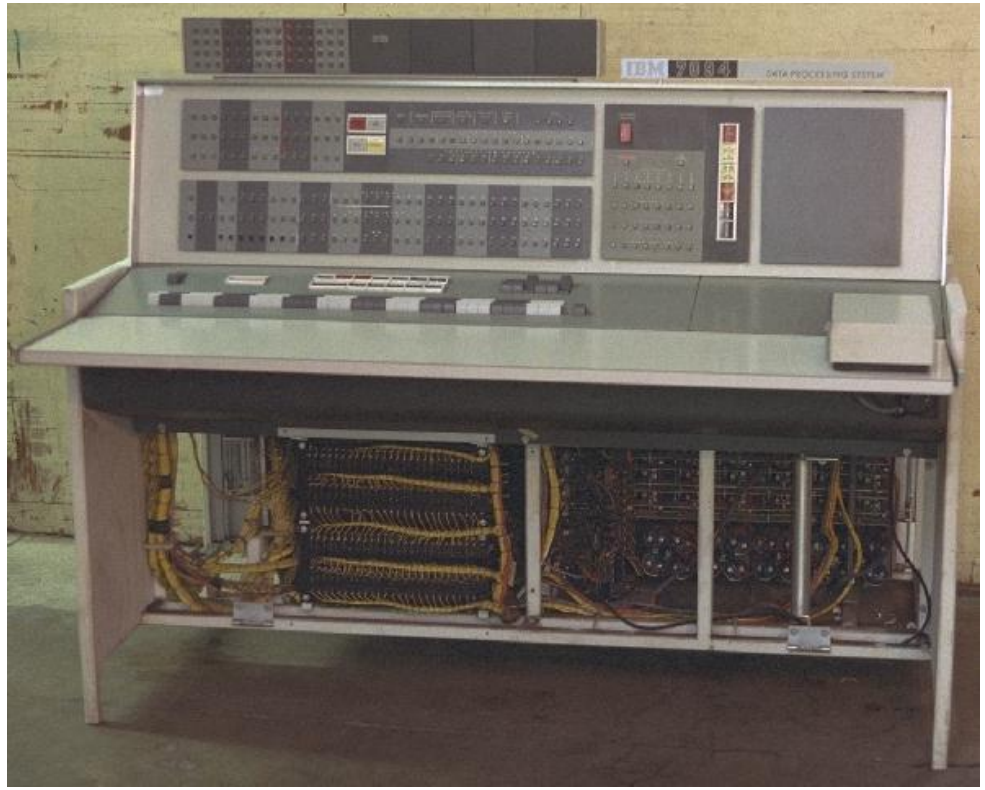
IBM 7030

http://computer-history.info/Page4.dir/pages/IBM.7030.Stretch.dir/images/7030.operator.console.jpg

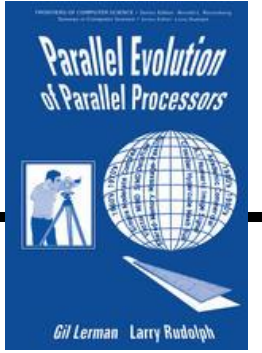
The IBM 7094 – 1962

http://gunkies.org/wiki/IBM_7094

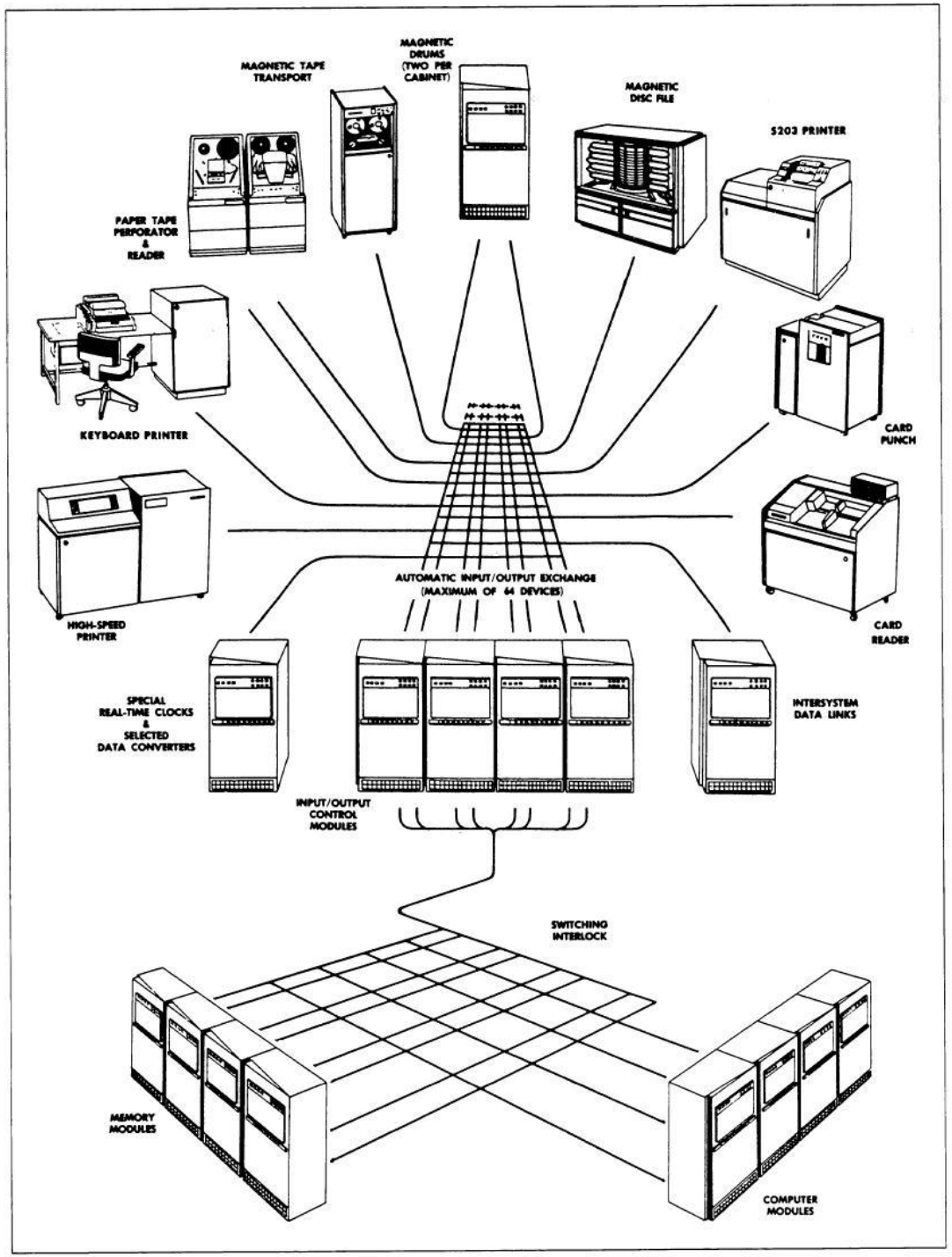
- The **IBM 7094** was IBM's last commercial scientific mainframe (built at a time when computers for scientific and business computing used separate instruction sets).



1962 – Burroughs proposed D825 (4 processors connected via a crossbar)



Although parallel processors are currently in vogue, their construction dates back some 30 years. In 1960 Burroughs announced the D825, a machine using four processors, connected via a crossbar to four memory banks, so that the memory in those banks could be accessed by each of the processors. Two years later, the first machines were delivered to the military [13] to serve in military command and control applications. In that same year (1962), yet another parallel processor was planned: the "SOLOMON" machine, designed by Slotnick [403]. The basic ideas behind this design can be traced back even further, to Unger's article of 1958 (see [183]). It was to be the first machine in which the processors would work concurrently toward the completion of the same task, under the supervision of a single sequencer. Sadly, that machine never reached operational status.

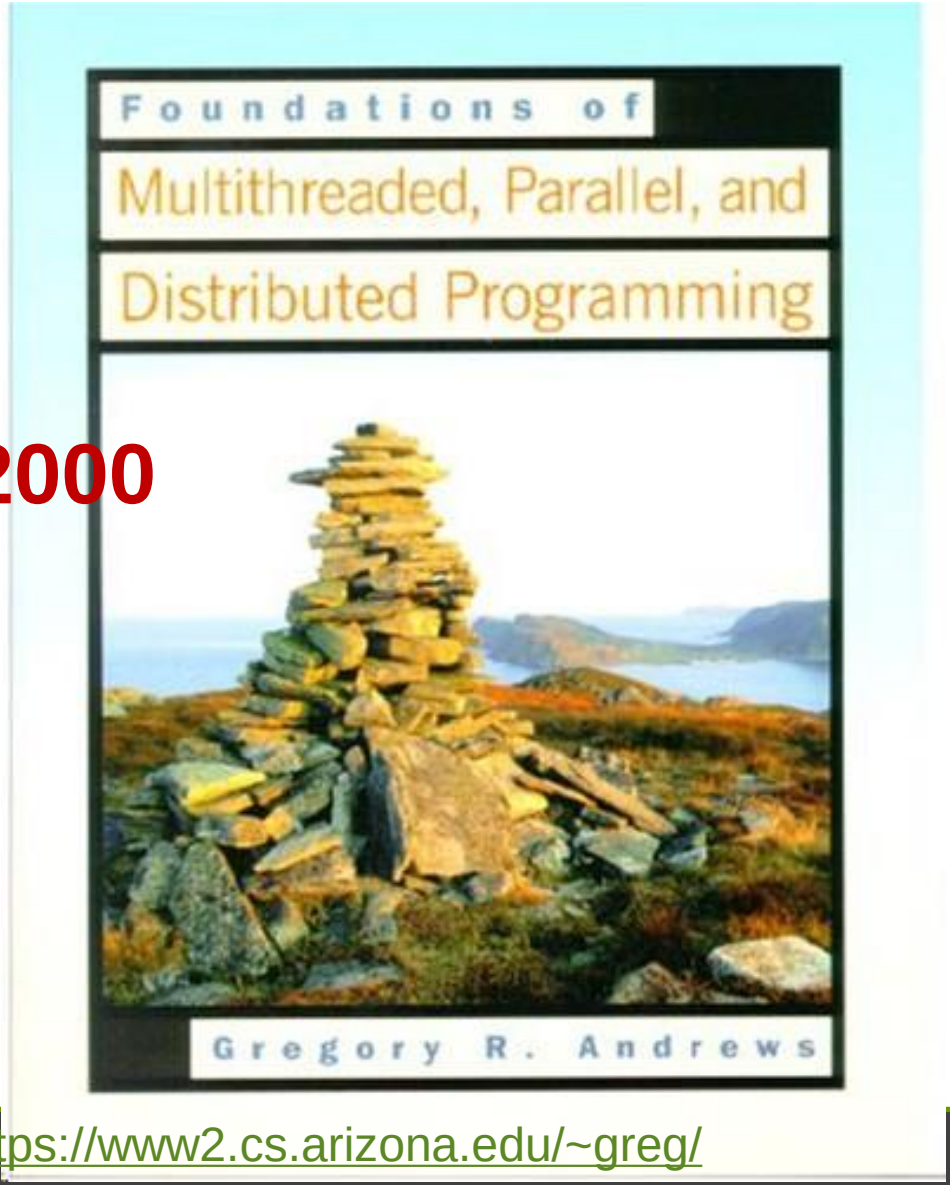


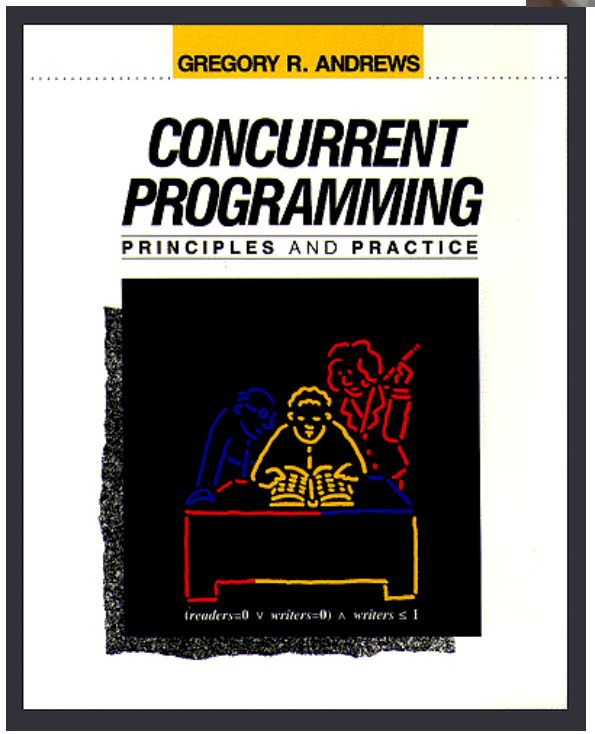
System organization, Burroughs D825 modular data processing system.

Burroughs D-285

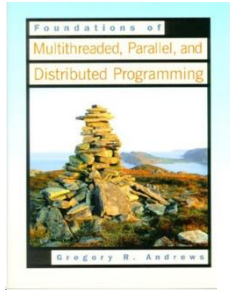
Channel – 1962

2000





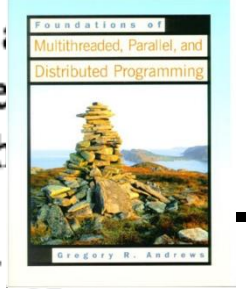
https://www2.cs.arizona.edu/~greg/



Concurrent programming originated in 1962 with the invention of channels, which are independent device controllers that make it possible to have a CPU execute a new application program at the same time that I/O operations are being executed on behalf of other, suspended application programs. Hence, concurrent programming—the word *concurrent* means happening at the same time—was initially of concern to operating systems designers. In the late 1960s, hardware designers developed multiple processor machines. This presented not only a challenge for operating systems designers but also an opportunity that application programmers could exploit.

The first major concurrent programming challenge was to solve what is now called the critical section problem. It and related problems (dining philosophers, readers/writers, etc.) led to a spate of papers in the 1960s. To harness the challenge, people developed synchronization primitives such as semaphores and monitors to simplify the programmer's task. By the mid-1970s, people came to appreciate the necessity of using formal methods to help control the inherent complexity of concurrent programs.

Computer networks were introduced in the late 1970s and early 1980s. The Arpanet supported wide-area computing, and the Ethernet established local-area networks. Networks gave rise to distributed programming, which was a major topic of the 1980s and became even more important in the 1990s. The essence of distributed programming is that processes interact by means of message passing rather than by reading and writing shared variables.



client/server computing, the Internet, and the World Wide Web. Finally, we
beginning to see multiprocessor workstations and PCs. Concurrent hardware
more prevalent than ever, and concurrent programming is more relevant th
ever.

This is my third book, another attempt to capture a part of the history
concurrent programming. My first book—*Concurrent Programming: Principles and Practice*, published in 1991—gives a broad, reference-level coverage of the period between 1960 and 1990. Because new problems, programming mechanisms, and formal methods were significant topics in those decades, the book focuses on them.

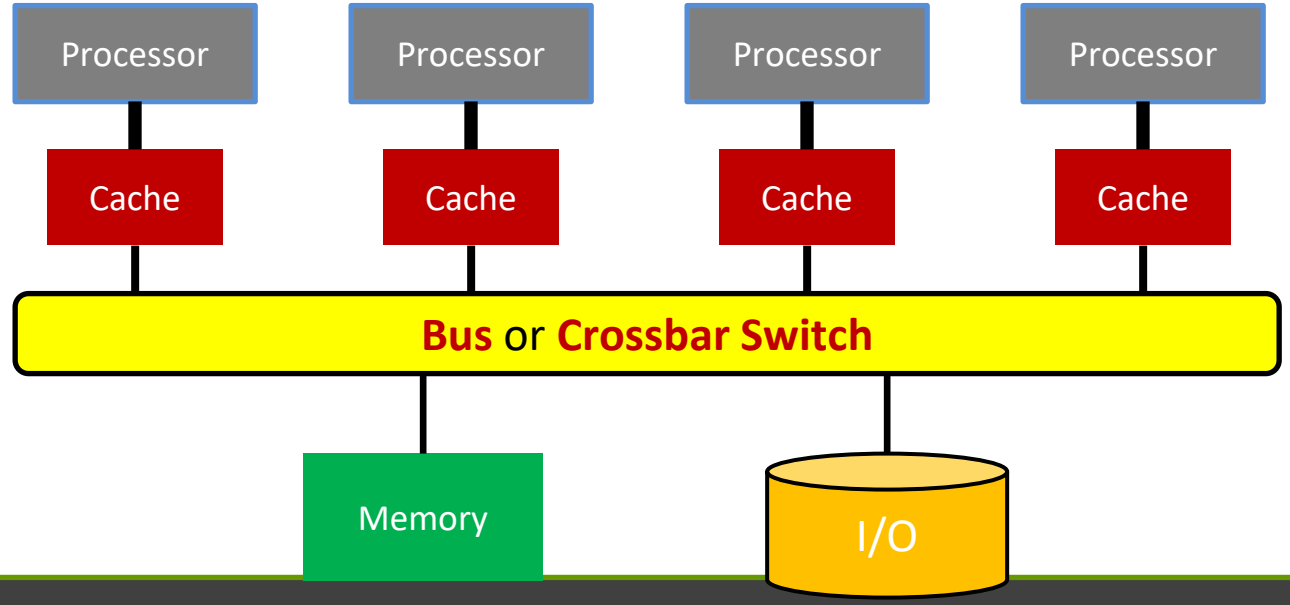
My second book—*The SR Programming Language: Concurrency in Practice*, published in 1993—summarizes my work with Ron Olsson in the late 1980s and early 1990s on a specific language that can be used to write concurrent programs for both shared- and distributed-memory machines. The SR book is pragmatic rather than formal, showing how to solve numerous problems in a single programming language.

This book is an outgrowth of my past and a reflection of what I think is important now and will be in the future. I have drawn heavily from material in the *Concurrent Programming* book, but I have completely rewritten every section that I have retained and have rewritten examples to use pseudo-C rather than SR. I have added new material throughout, especially on parallel scientific programming. I have also included case studies of the most important languages and software libraries, with complete sample programs. Finally, I have a new vision for the role of this book—in classrooms and in personal libraries.

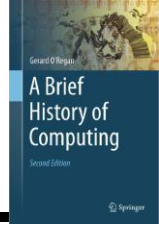
https://en.wikipedia.org/wiki/Symmetric_multiprocessing

1961 SMP (Symmetric multiprocessing)

- **Symmetric multiprocessing** (**SMP**) involves a multiprocessor computer hardware and software architecture where two or more **identical** processors are connected to a single, shared main memory, have full access to all input and output devices, and are controlled by a single operating system instance that treats all processors equally, reserving none for special purposes. Most multiprocessor systems today use an SMP architecture.



IBM 7094 and its OS – FMS



- **The 7090 and 7094 were operated in batch mode, controlled by the Fortran Monitor System (FMS).**
  - Batch jobs on cards were transferred to tape on an auxiliary 1401, and the monitor took one job at a time off the input tape, ran it, and captured the output on another tape for printing and punching by the 1401.
  - Each user job was loaded into core by the BSS loader (**B**inary **S**ymbolic **S**egment **loader**) along with a small monitor routine that terminated jobs that ran over their time estimates.
  - Thus, each user's job had complete control of the whole 7094, all 32K words of memory, all the data channels, everything.

The designers of the IBM 704 included John Backus and Gene Amdahl. Backus was one of the key designers of the FORTRAN programming language introduced by IBM in 1957.This was the first scientific programming language and is used by engineers and scientists. Gene Amdahl later founded the Amdahl Corporation in 1970, and this company later became a rival to IBM in the mainframe market.

And IBM was smart then

- It donated IBM 7094 to UM (Univ. of Michigan) and MIT
- IBM's managers required UM and MIT to stop their computing jobs on 7094 to finish match computing, because those managers liked sailing match very much!

Match racing - Open - Wednesday 18 April 2012

Pos	Name	Country	Crew	Events	Previous	Best	Points
1	Williams Ian	GBR		8	1	1	12074
2	Mirsky Torvar	AUS		8	3	1	11065
3	Bruni Francesco	ITA		8	2	1	10904
4	Morvan Pierre-antoine	FRA		8	4	4	10770
5	Hansen Bjorn	SWE		8	5	4	10667
6	Radich Jesper	DEN		8	6	1	10479
7	Swinton Keith	AUS		8	13	7	9866
8	Berntsson Johnie	SWE		8	7	7	9792
9	Robertson Philip	NZL		8	9	7	9757
10	Gilmour Peter	AUS		8	10	1	9472

http://uk.boats.com/boat-content/2012/04/ian-williams-and-anna-tunnicliffe-top-match-racing-table/

http://en.wikipedia.org/wiki/University_of_Michigan_Executive_System

- **UM provided UMES (University of Michigan Executive System)**
  - To overcome that indignant requirement so that **the intermediate result could be stored**!
  - It's a more complex batch operating system developed at the University of Michigan in **1958**, was widely used at many universities.
    - It was in use at the University of Michigan until 1967
- **IBM 7090/94 and IBSYS (1960)**
  - IBSYS was the tape based operating system that IBM supplied with its IBM 7090 and IBM 7094 computers. http://en.wikipedia.org/wiki/IBM_7090/94_IBSYS
    - IBSYS was based on **F**ORTRAN **M**onitor **S**ystem (**FMS**) and **SHARE** Operating System.
    - IBSYS itself was really a basic monitor program
      - ✓ read control card images placed between the decks of program and data cards of individual jobs.

- **FMS, UMES and IBSYS belong to so-called simple batch (process) system [单道批处理系统]**
  - Two sub-types: offline and online
  - No matter which subtype, SBS can only store one job/program in computer's main memory
    - All the resources are occupied by that job which is now in main memory!
- **This is a kind of waste: if there are many IO operations in that job, the CPU is idle!!**

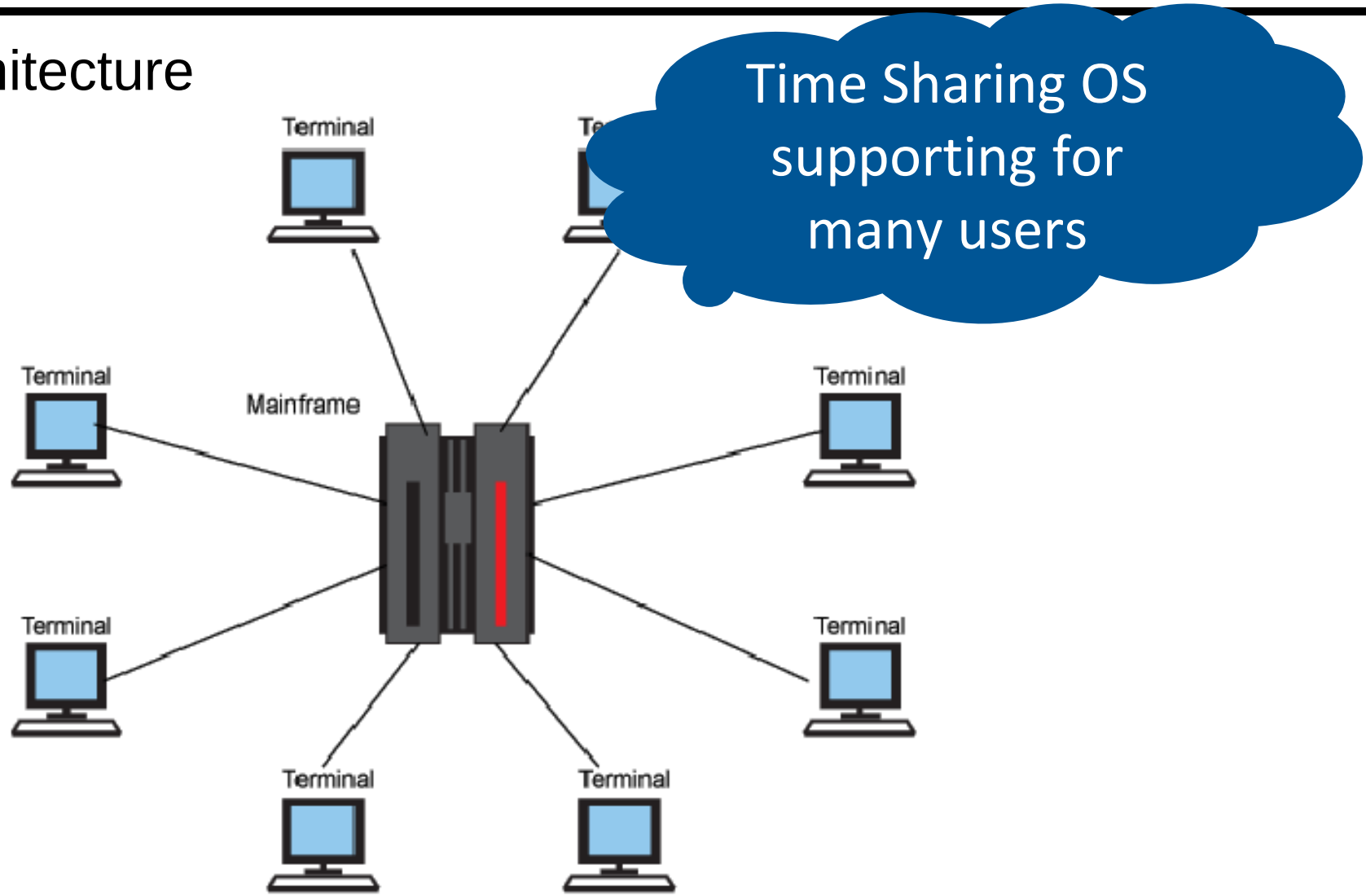
➔ **Multi-programmed batch (process) system [多道批处理系统] and Time-sharing system**

➔ **CTSS (**Compatible **Time-Sharing System)**

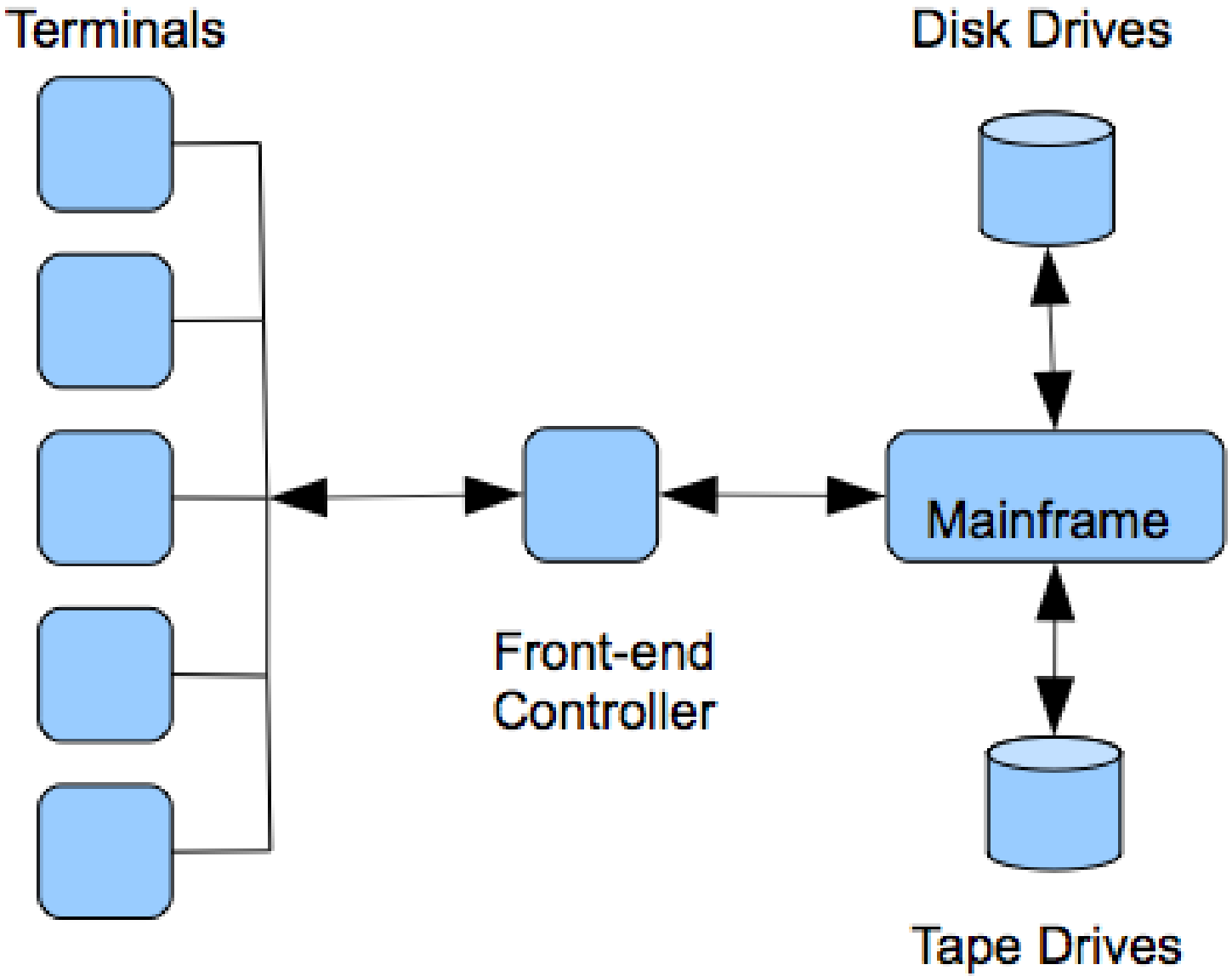
https://baike.baidu.com/item/CTSS/10533936

https://www.researchgate.net/figure/Mainframe-Architecture_fig2_275405572

- Mainframe Architecture



https://userpages.umbc.edu/~jianwu/is651/651book/is651-strapdown.php?f=is651-Chapter02.md

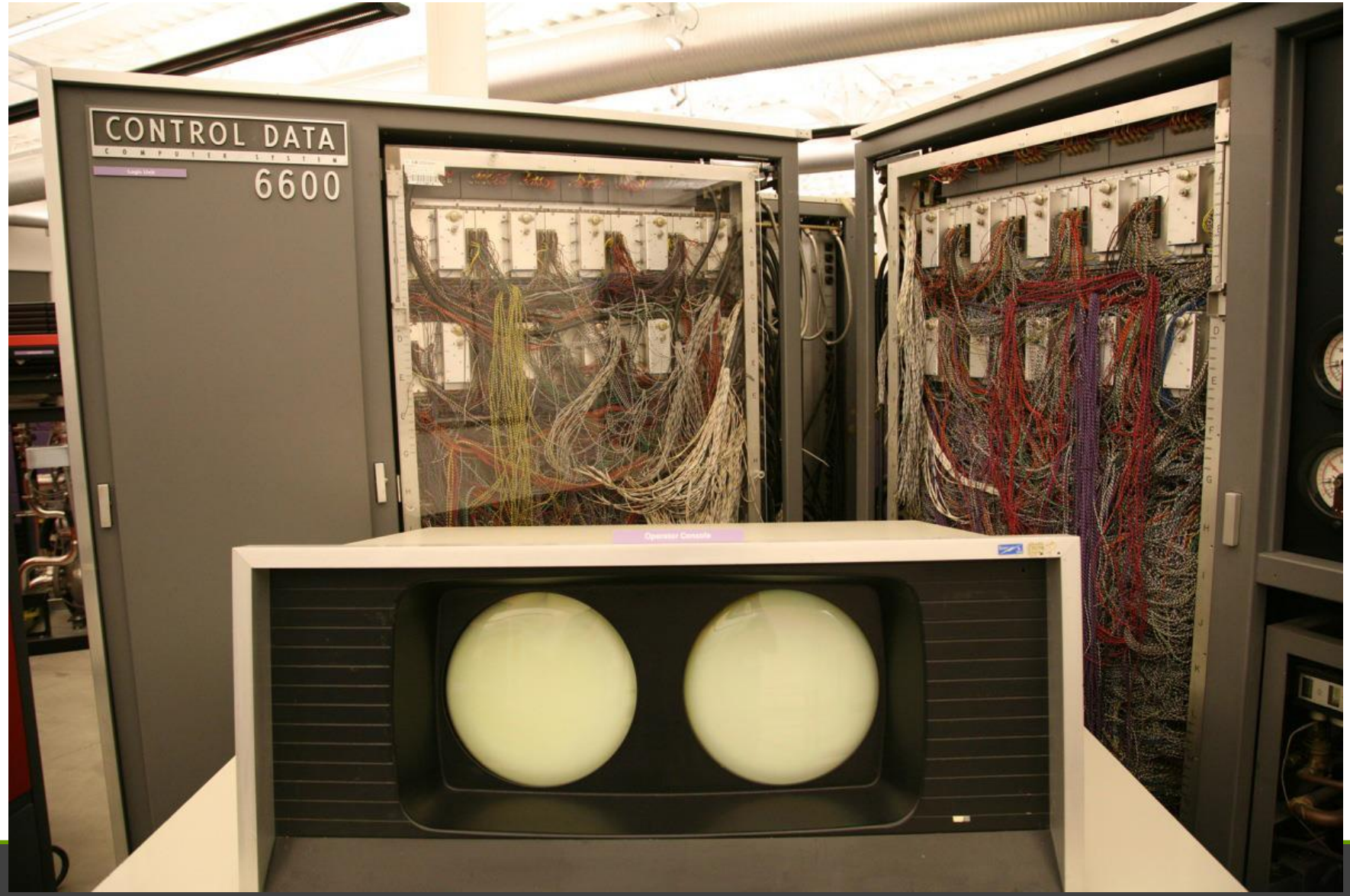



User
Terminal
Mainframe
Mainframe

The Early Days – 1965

- **CDC (Control Data Corporation) employed Seymour Cray to design the CDC 6600 which was 10x faster than any other computer when built.**
  - **First ever RISC system** – simple instruction set – which simplified timing within the CPU and allowed for instruction pipelining leading to higher throughput and a higher clock speed, 10 MHz.
  - Used logical address translation to map addresses in user programs and restrict to using only a portion of contiguous core memory. Hence user program can be moved around in core memory by the operating system.
  - System contained 10 other “Peripheral Processors” to handle I/O and run the operating system.

CDC 6600



DEC PDP-7 1965

https://www.soemtron.org/pdp7.html
https://gunkies.org/wiki/PDP-7

So-called Minicomputer

- The PDP-7 is a minicomputer produced by DEC, introduced in 1965; with a low cost, it was cheap but powerful. There were two models, the second being the -7/A, but the difference is not yet clarified.
- The PDP-7 was the third of Digital's 18-bit machines, with essentially the same instruction set and architecture as the predecessor PDP-4 and successor PDP-9. It was the first wire-wrapped PDP. It was the first to use their Flip-Chip® technology, but also included the older System Modules.
- In 1969, Ken Thompson wrote the first UNIX system in assembly language on a PDP-7, then named Unics as a somewhat treacherous pun on Multics, as the operating system for Space Travel, a game which required graphics to depict the motion of the planets. A PDP-7 was also the development system used during the development of MUMPS at MGH in Boston a few years earlier.
- There are a few remaining PDP-7's still in operable condition. One under restoration in Oslo, Norway, has been thrown away.

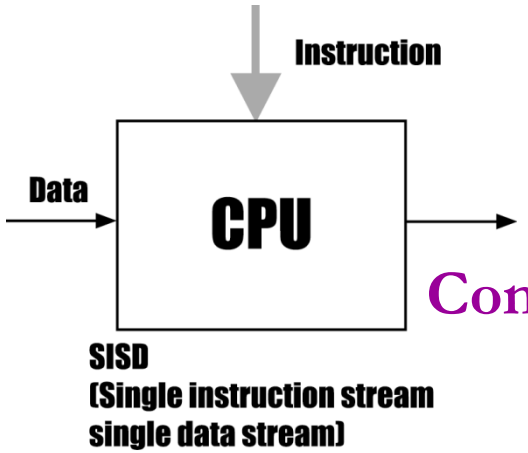
PDP-7

A PDP-7 in Oslo, Norway

Manufacturer:	Digital Equipment Corporation
Year Introduced:	1965
Form Factor:	minicomputer
Word Size:	18 bits
Logic Type:	PNP Transistor FLIP CHIPs
Memory Speed:	1.75 µsec
Physical Address Size:	15 bits (32K words)
Virtual Address Size:	13 bits (direct), 15 bits (extended)
Operating System:	DECSYS-7
Predecessor(s):	PDP-4
Successor(s):	PDP-9
Price:	US$72K

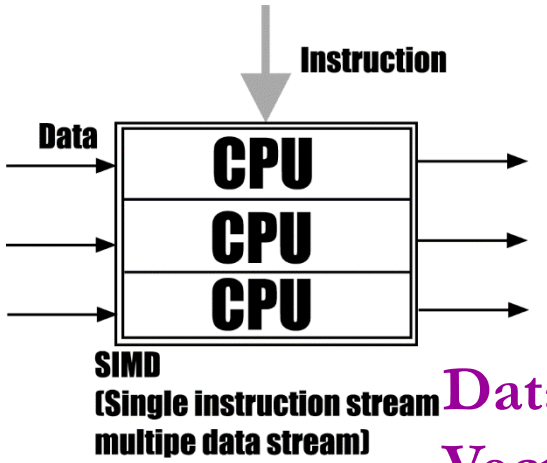
Flynn's Taxonomy (1966)

	Single Data	Multiple Data
Single Instruction	**SISD** uniprocessors	**SIMD** processor arrays pipelined vector processors
Multiple Instruction	**MISD** systolic arrays	**MIMD** multiprocessors multicomputers



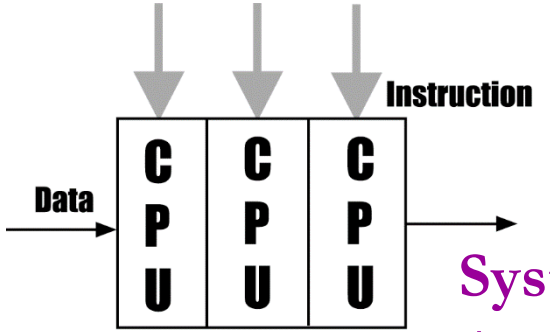
SISD
(Single instruction stream
single data stream)

Conventional



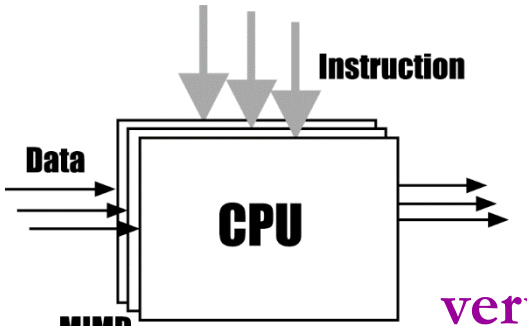
SIMD
(Single instruction stream
multipe data stream)

Data Parallel,
Vector Computing



MISD
(Multipe instruction stream
single data stream)

Systolic
Arrays

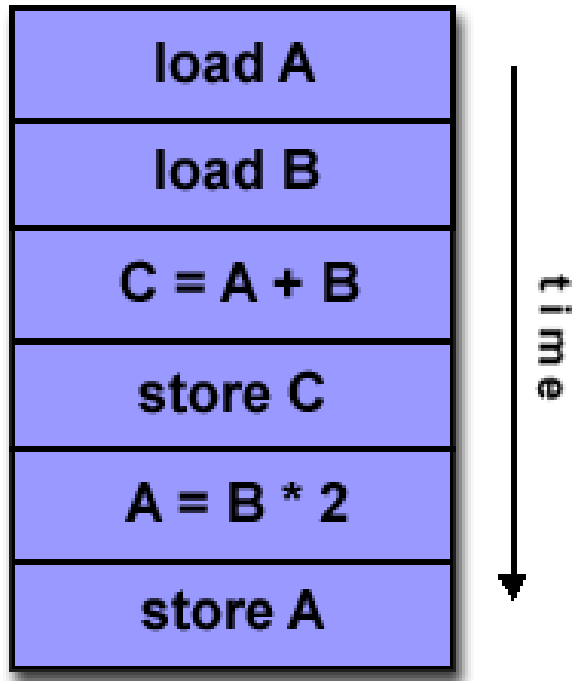


MIMD
(Multipe instruction stream
mingle data stream)

very general,
multiple approaches

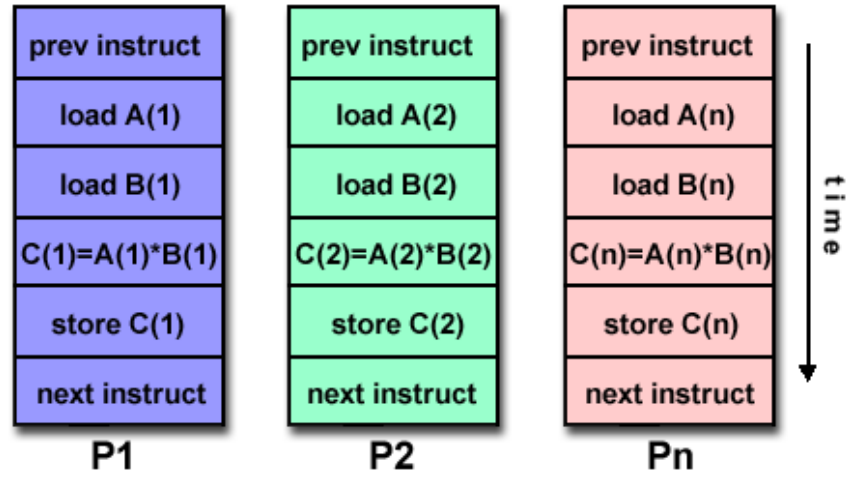
Single Instruction, Single Data (SISD)

- A serial (non-parallel) computer
- Single instruction: only one instruction stream is being acted on by the CPU during any one clock cycle
- Single data: only one data stream is being used as input during any one clock cycle
- Deterministic execution
- This is the oldest and until recently, the most prevalent form of computer
- Examples: most PCs, single CPU workstations and mainframes

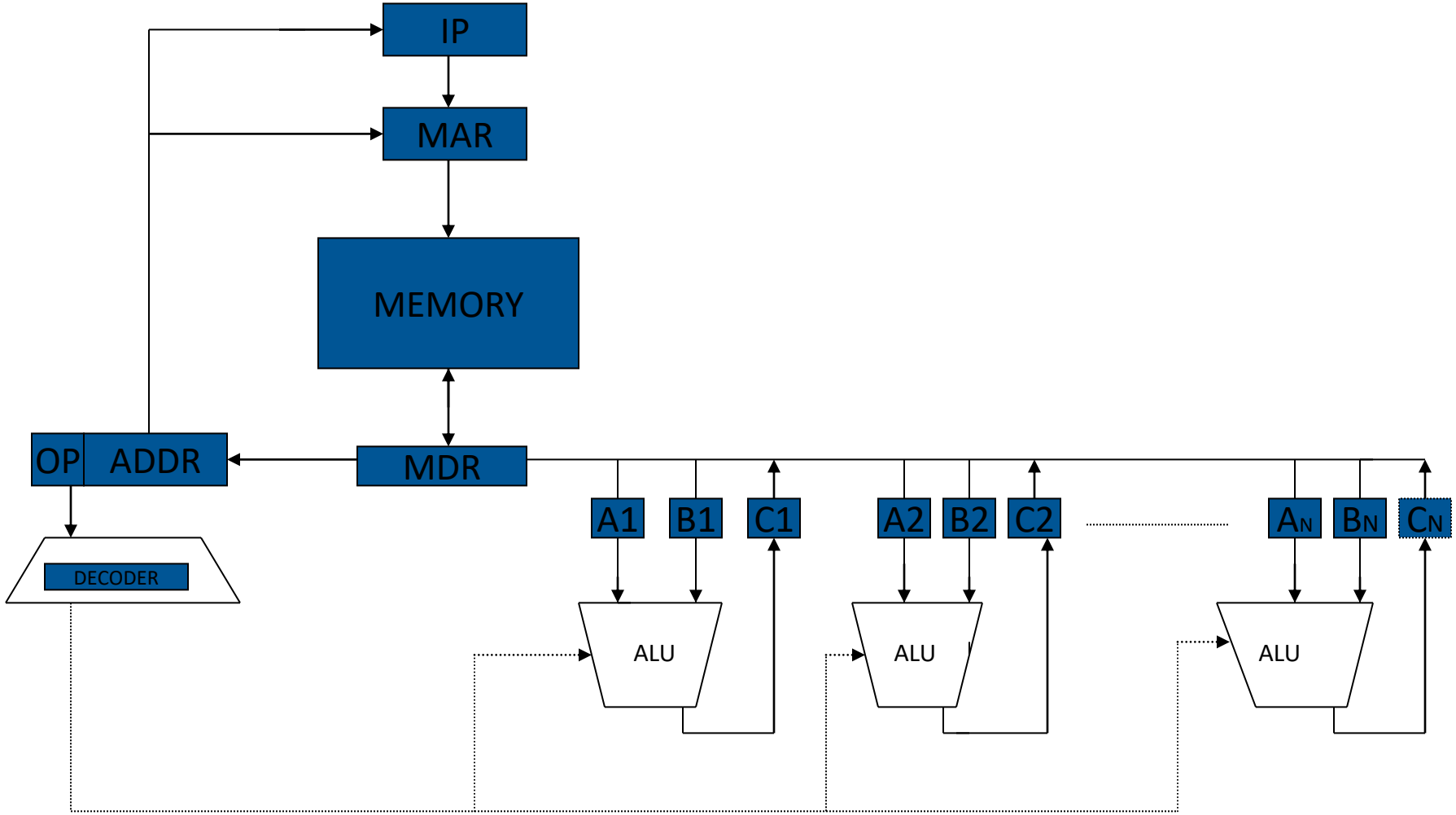


Single Instruction, Multiple Data (SIMD)

- A type of parallel computer
- Single instruction: All processing units execute the same instruction at any given clock cycle
- Multiple data: Each processing unit can operate on a different data element
- This type of machine typically has an instruction dispatcher, a very high-bandwidth internal network, and a very large array of very small-capacity instruction units.
- Best suited for specialized problems characterized by a high degree of regularity, such as image processing.
- Synchronous (lockstep) and deterministic execution
- Two varieties: Processor Arrays and Vector Pipelines
- Examples:
  - Processor Arrays: Connection Machine CM-2, Maspar MP-1, MP-2
  - Vector Pipelines: IBM 9000, Cray C90, Fujitsu VP, NEC SX-2, Hitachi S820

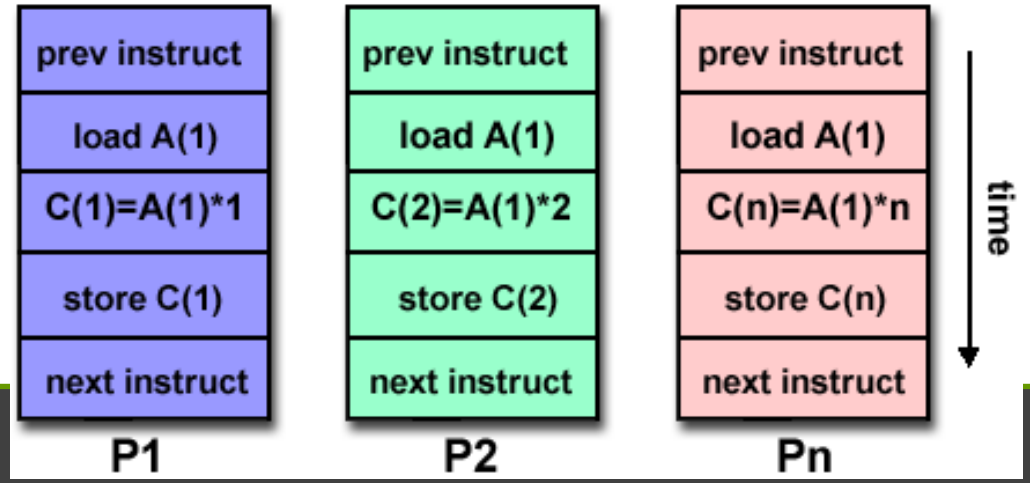


Array processor (SIMD)



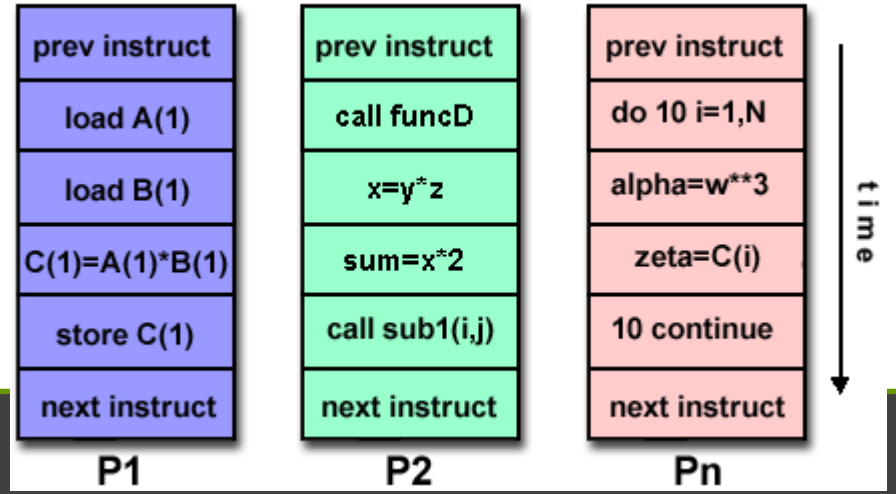
Multiple Instruction, Single Data (MISD)

- **A single data stream is fed into multiple processing units.**
- **Each processing unit operates on the data independently via independent instruction streams.**
- **Few actual examples of this class of parallel computer have ever existed. One is the experimental Carnegie-Mellon C.mmp computer (1971).**
- **Some conceivable uses might be:**
  - multiple frequency filters operating on a single signal stream
- **multiple cryptography algorithms attempting to crack a single coded message.**



Multiple Instruction, Multiple Data (MIMD)

- Currently, the most common type of parallel computer. Most modern computers fall into this category.
- Multiple Instruction: every processor may be executing a different instruction stream
- Multiple Data: every processor may be working with a different data stream
- Execution can be synchronous or asynchronous, deterministic or non-deterministic
- Examples: most current supercomputers, networked parallel computer "grids" and multi-processor SMP computers - including some types of PCs.



Other taxonomies

SPMD — Single Program, Multiple Data. This is similar to SIMD but indicates that a single program is used for the parallel application i.e. every node runs the same program.

MPMD — Multiple Program, Multiple Data. Like MIMD except this explicitly uses more than one program for the parallel application. Typically one is a *master* or *control* program and the others are *slave* or *compute* programs.

Modern clusters typically follow one of these two models. It is often convenient to use the SPMD model but have the program behave as either a *master* or *slave* based on some criteria determined at run-time (often the node on which the program is running).

ARPANET – the origin of the Internet

- 1967: Lawrence Roberts of ARPA publishes plan for the first computer network system – the ARPANET
- Packet switches were needed. Called Interface Message Processors (IMP), the contract was awarded to BBN (a Boston-area computer company)
- Oct 1969: IMPs installed in UCLA, Stanford, UCSB and Utah

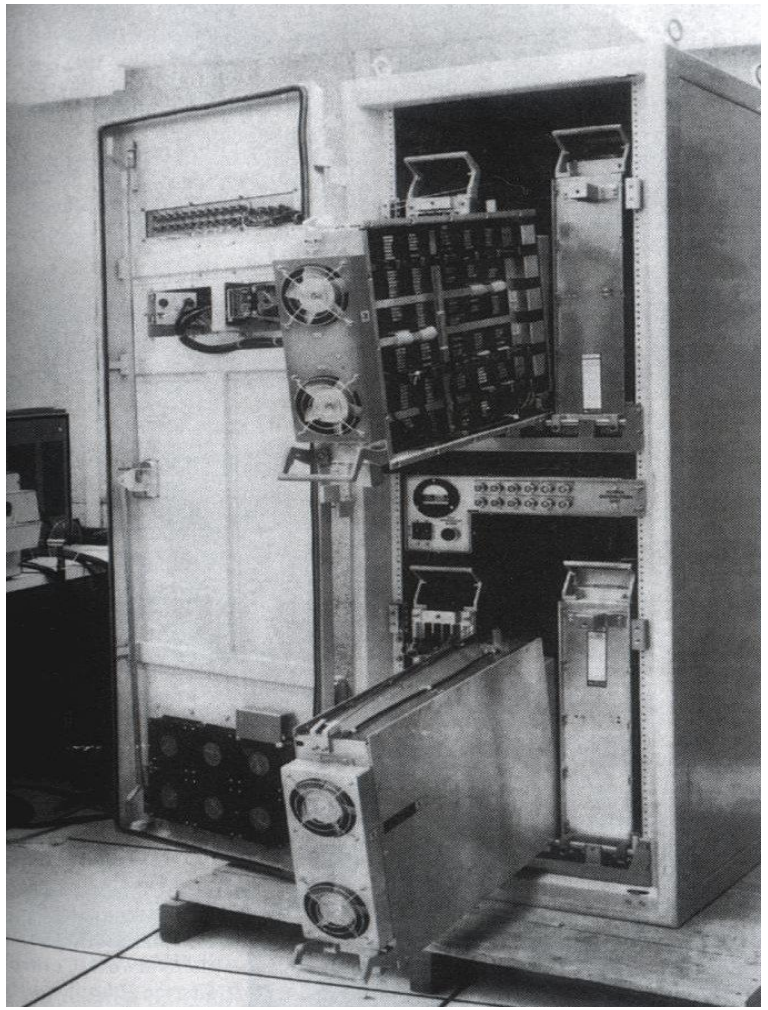


Image Source:http://aleph.llull.net/wp-content/files/imp.jpg

Interface Message Processor

Packet Switching

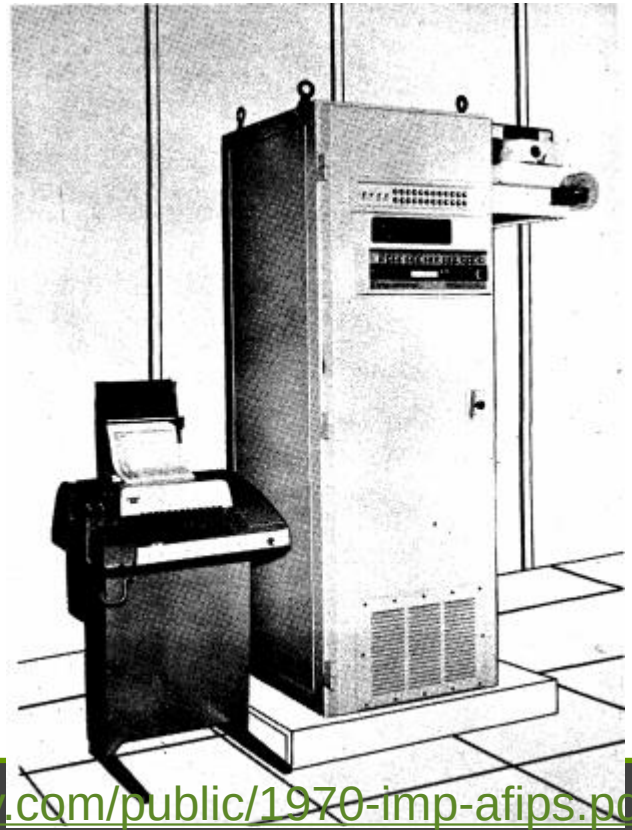
Photo by Louis Bachrach

- 1961: Leonard Kleinrock uses queuing theory, proposes packet switched networks
  - More bandwidth efficient
  - Robust – not reliant on single route

Widely circulated photo of the IMP Team (L to R): Truett Thatch, Bill Bartell (Honeywell), Dave Walden, Jim Geisman, Robert Kahn, Frank Heart, Ben Barker, Marty Thrope, Will Crowther, Severo Ornstein. Not pictured: Bernie Cosell.

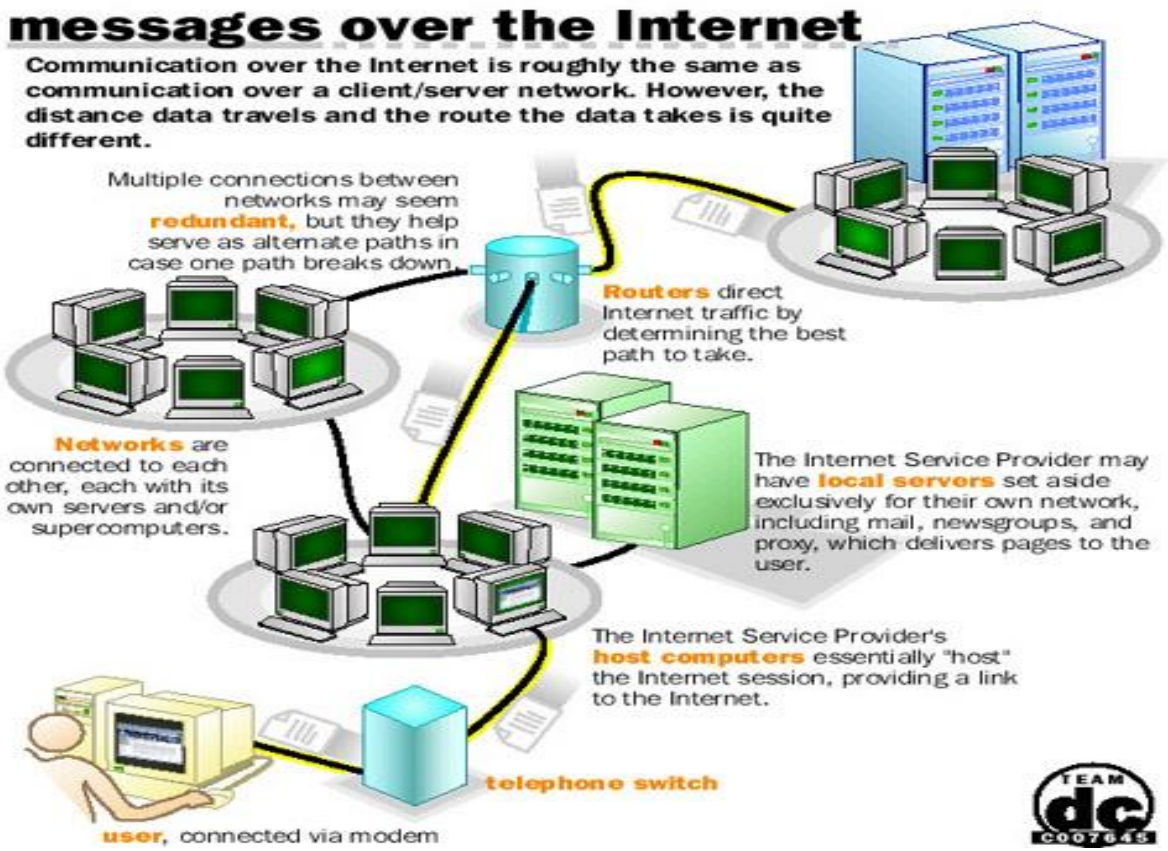
https://www.livinginternet.com/i/ii_imp.htm

- The Interface Message Processor provided a system independent interface to the *ARPANET* that could be used by any computer system, thereby opening the *Internet* network *architecture* from the very beginning.



http://walden-family.com/public/1970-imp-afips.pdf

- **The Internet is a global system of interconnected computer networks that use the standardized Internet Protocol Suite (TCP/IP).**



Chapter 3: Large Scale Computing Systems

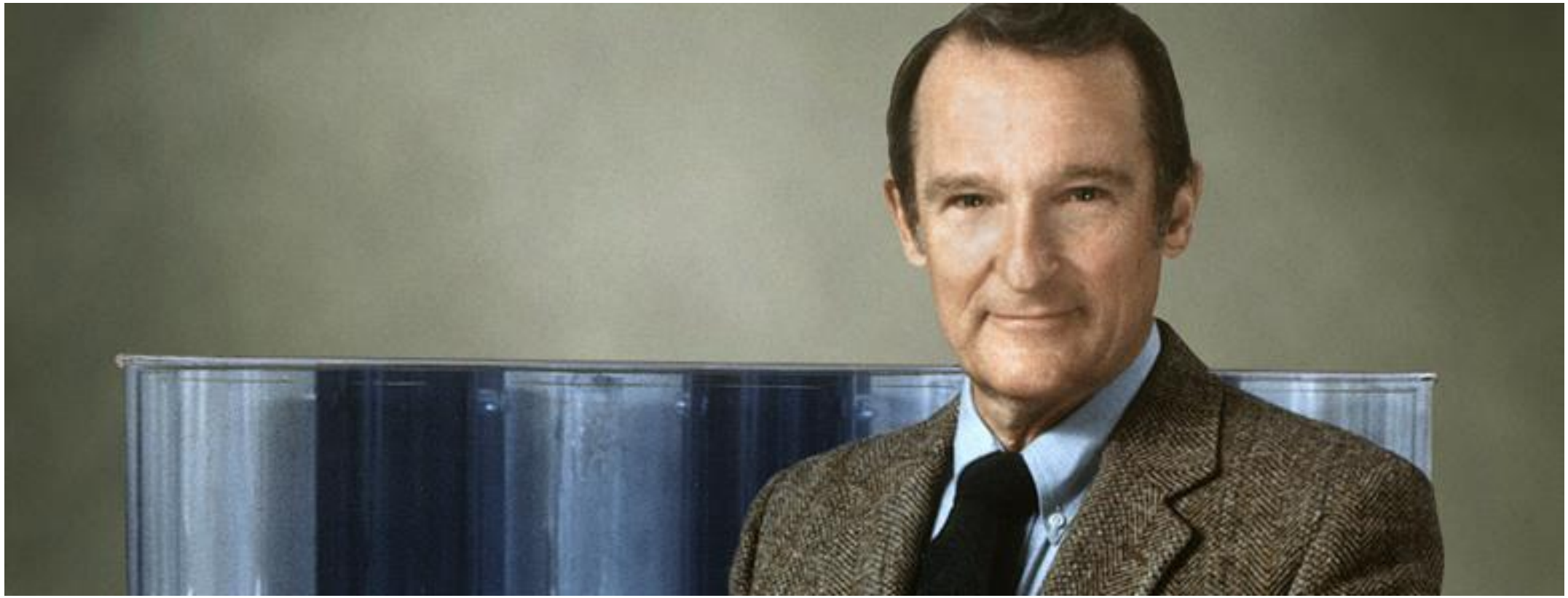
Faster for larger data

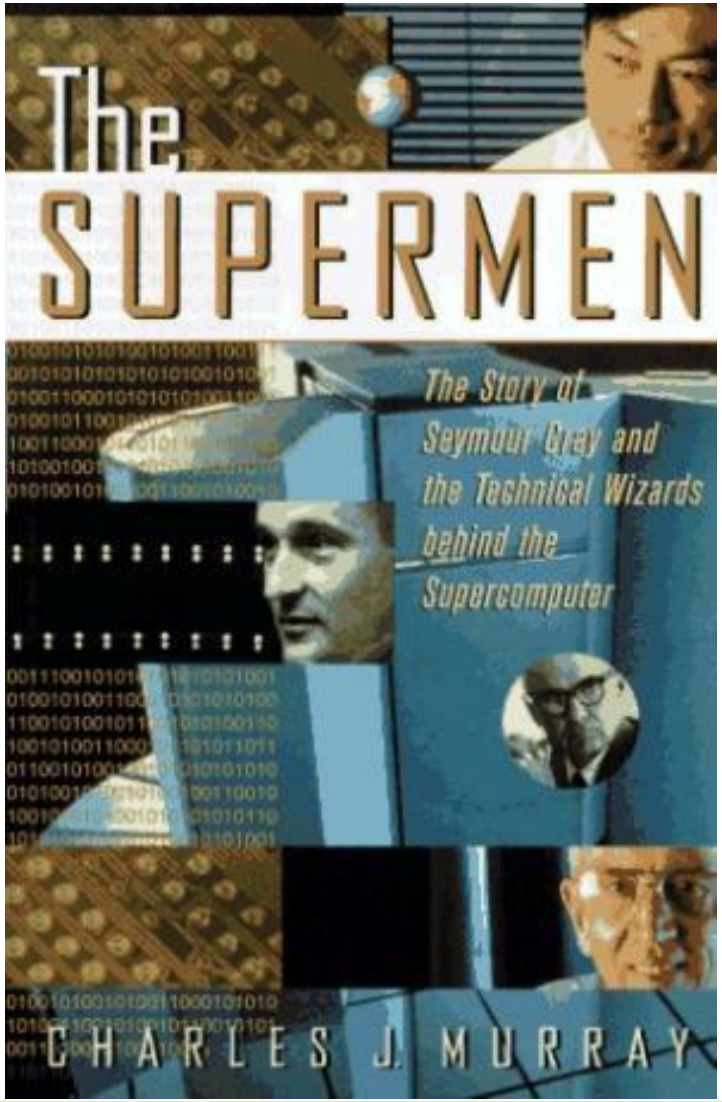
- von Neumann architecture
  - Foundation of modern computers
  - 1960 2 CPUs
- 1962 Channel
  - Origin of **concurrent programming**
- Parallel
  - Vector processor, Multi-core, later GPU/CUDA
- Distributed
  - Cluster, Grid, …
- Now Clouding – Virtualizing computer systems for so-called Big Data
  - IaaS, PaaS, SaaS, …

History (1970-1990)

Year	Name	Peak speed	Location
1974	CDC Star-100	100 MFLOPS (vector) ~2 MFLOPS (scalar)	Lawrence Livermore Lab. USA
1975	Cray-1	80 MFLOPS (vector) 72 MFLOPS (scalar)	Los Alamos Lab. USA
1981	CDC Cyber-205	400 MFLOPS (vector) peak, avg much lower	
1983	Cray X-MP	500 MFLOPS (4 CPUs)	Los Alamos Lab. USA
1985	Cray-2	1.95 GFLOPS (4 CPUs) 3.9 GFLOPS (8 CPUs)	Lawrence Livermore Lab. USA
1989	ETA-10G	10.3 GFLOPS (vector) peak, avg much lower (8 CPUs)	
1990	Fujitsu Numerical Wind Tunnel	236 GFLOPS	National Aerospace Lab, Japan

Seymour Cray





- The supermen: the story of Seymour Cray and the supercomputer
- Charles J. Murray
- 1997
- Wiley

The Vector Years – 1974

- **The CDC Star-100 was one of the first machines to use a vector processor**
  - Used “deep” pipelines (25 vs. 8 on 7600) which need to be "filled" with data constantly and had high setup cost. The vector pipeline only broke even with >50 data points in each set.
  - But the number of algorithms that can be effectively vectorized is very low and need careful coding otherwise the high vector setup cost dominates.
  - And the basic scalar performance had been sacrificed in order to improve vector performance ➔ machine was generally considered a failure.
  - Today almost all high-performance commodity CPU designs include vector processing instructions, e.g. SIMD.

Vector program (using AVX intrinsics)

Intrinsics available to C programmers

```
#include <immintrin.h>

void sinx(int N, int terms, float* x, float* result)
{
   float three_fact = 6;  // 3!
   for (int i=0; i<N; i+=8)
   {
       __m256 origx = _mm256_load_ps(&x[i]);
      __m256 value = origx;
      __m256 numer = _mm256_mul_ps(origx, _mm256_mul_ps(origx, origx));
      __m256 denom = _mm256_broadcast_ss(&three_fact);
      int sign = -1;

      for (int j=1; j<=terms; j++)
      {
         // value += sign * numer / denom
         __m256 tmp = _mm256_div_ps(_mm256_mul_ps(_mm256_set1ps(sign), numer), denom);
         value = _mm256_add_ps(value, tmp);

         numer = _mm256_mul_ps(numer, _mm256_mul_ps(origx, origx));
         denom = _mm256_mul_ps(denom, _mm256_broadcast_ss((2*j+2) * (2*j+3)));
         sign *= -1;
      }
      _mm256_store_ps(&result[i], value);
   }
}
```

Vector program (using AVX intrinsics)

```
#include <immintrin.h>
void sinx(int N, int terms, float* x, float* sinx)
{
   float three_fact = 6;  // 3!
   for (int i=0; i<N; i+=8)
   {
       __m256 origx = _mm256_load_ps(&x[i]);
       __m256 value = origx;
       __m256 numer = _mm256_mul_ps(origx, _mm256_mul_ps(origx, origx));
       __m256 denom = _mm256_broadcast_ss(&three_fact);
       int sign = -1;

       for (int j=1; j<=terms; j++)
       {
          // value += sign * numer / denom
          __m256 tmp = _mm256_div_ps(_mm256_mul_ps(_mm256_broadcast_ss(sign),numer),denom);
          value = _mm256_add_ps(value, tmp);

          numer = _mm256_mul_ps(numer, _mm256_mul_ps(origx, origx));
          denom = _mm256_mul_ps(denom, _mm256_broadcast_ss((2*j+2) * (2*j+3)));
          sign *= -1;
       }
       _mm256_store_ps(&sinx[i], value);
   }
}
```

```
vloadps  xmm0, addr[r1]
vmulps   xmm1, xmm0, xmm0
vmulps   xmm1, xmm1, xmm0
...
...
...
...
...
...
vstoreps  addr[xmm2], xmm0
```

Compiled program:

Processes eight array elements simultaneously using vector instructions on 256-bit vector registers

CDC Star-100 (1974)

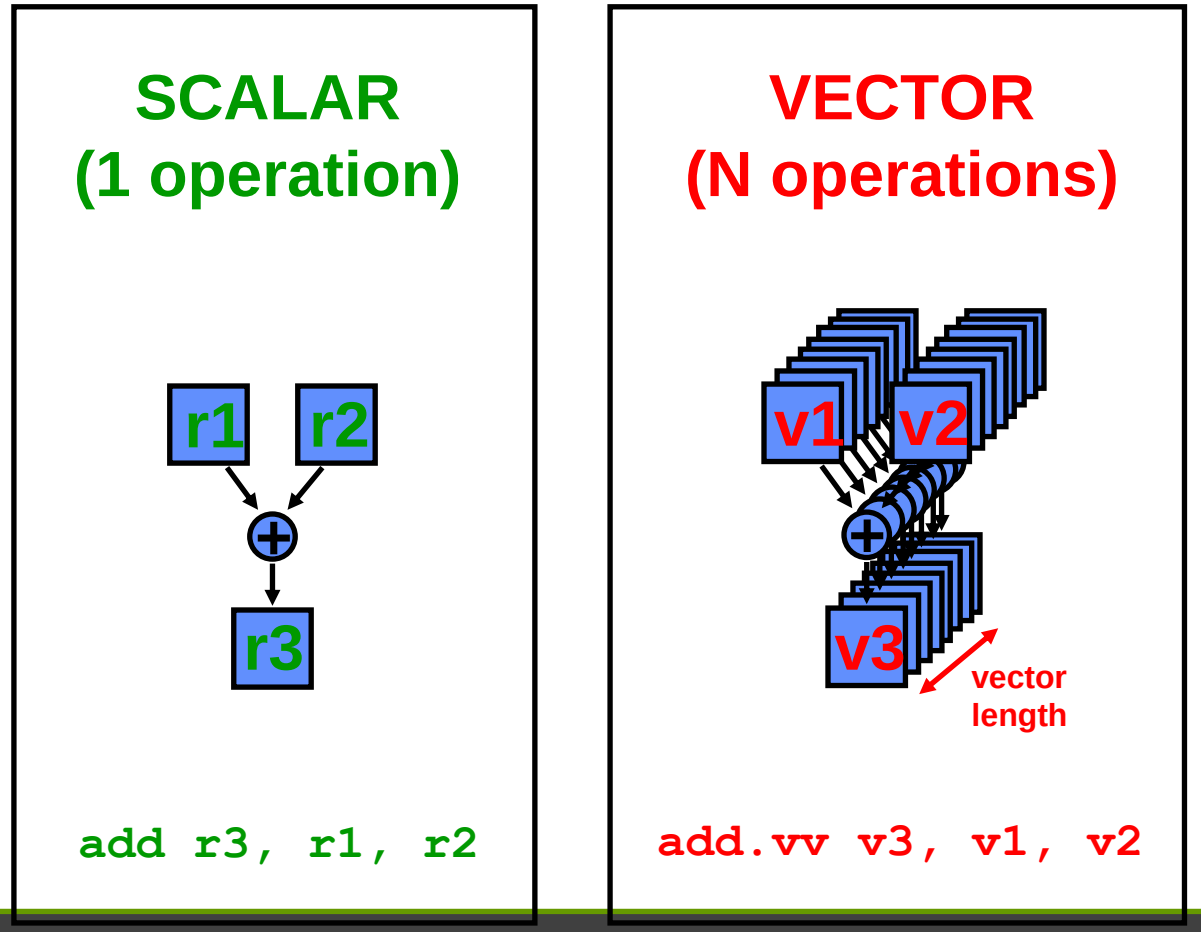
Vector Machines

- Vector architectures are based on a single processor
  - Multiple functional units
  - All performing the same operation
  - Instructions may specific large amounts of parallelism (e.g., 64-way) but hardware executes only a subset in parallel (Later inherited by GPGPU)
- Historically important
  - Overtaken by MPPs in the 1990s
- Re-emerging in recent years
  - At a large scale in the Earth Simulator (NEC SX6) and Cray X1
  - At a small scale in SIMD media extensions to microprocessors
    - SSE, SSE2 (Intel: Pentium/IA64)
    - Altivec (IBM/Motorola/Apple: PowerPC)
    - VIS (Sun: Sparc)
  - At a larger scale in GPUs
- Key idea: Compiler does some of the difficult work of finding parallelism, so the hardware doesn't have to

Vector Processing

http://bwrcs.eecs.berkeley.edu/Classes/CS252/Notes/Lec07-vector.ppt

- Vector processors have high-level operations that work on linear arrays of numbers: "vectors"



Vector operations

- **Vector addition Z = X + Y**

  for (i=0; i<n; i++) z[i] = x[i] + y[i];

$$\begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} + \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix} = \begin{pmatrix} x_1 + y_1 \\ x_2 + y_2 \\ \dots\dots \\ x_n + y_n \end{pmatrix}$$

- **Vector scaling Y = a * X**

  for(i=0; i<n; i++) y[i] = a*x[i];

$$a * \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} = \begin{pmatrix} a * x_1 \\ a * x_2 \\ \dots\dots \\ a * x_n \end{pmatrix}$$

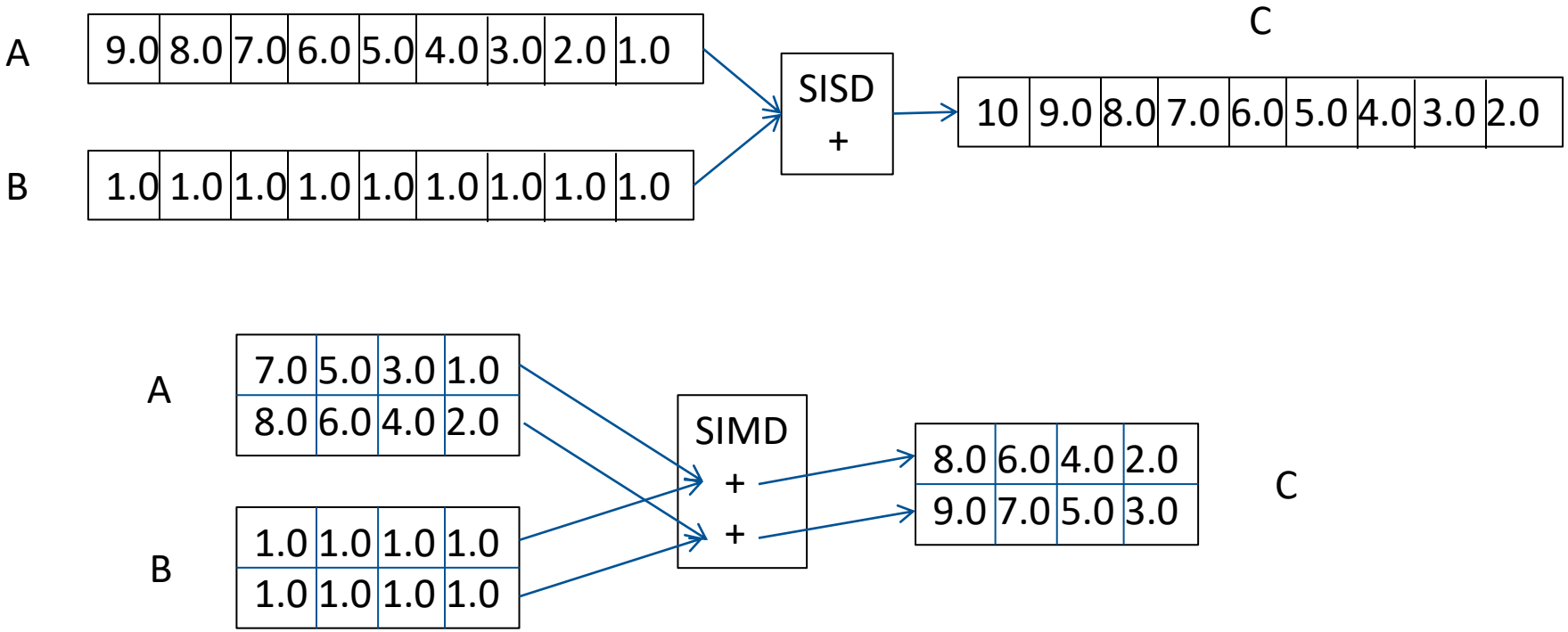
- **Dot product**

  for(i=0; i<n; i++) r += x[i]*y[i];

$$\begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} \bullet \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix} = x_1 * y_1 + x_2 * y_2 + \dots\dots + x_n * y_n$$

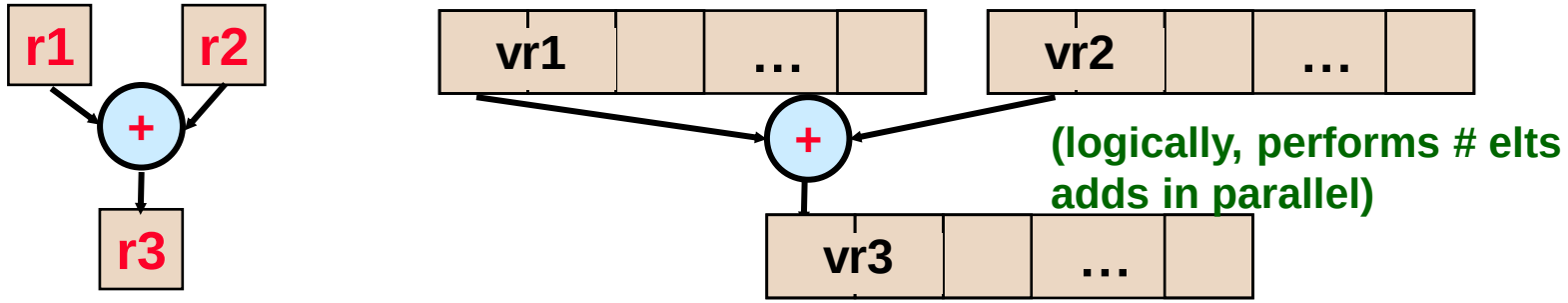
SISD and SIMD vector operations

- **C = A + B**
  - For (i=0;i<n; i++) c[i] = a[i] + b[i]

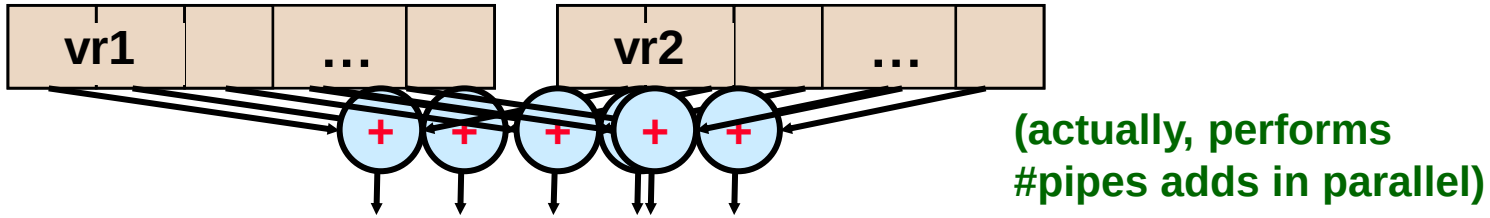


Vector Processors

- Vector instructions operate on a vector of elements
  - These are specified as operations on vector registers



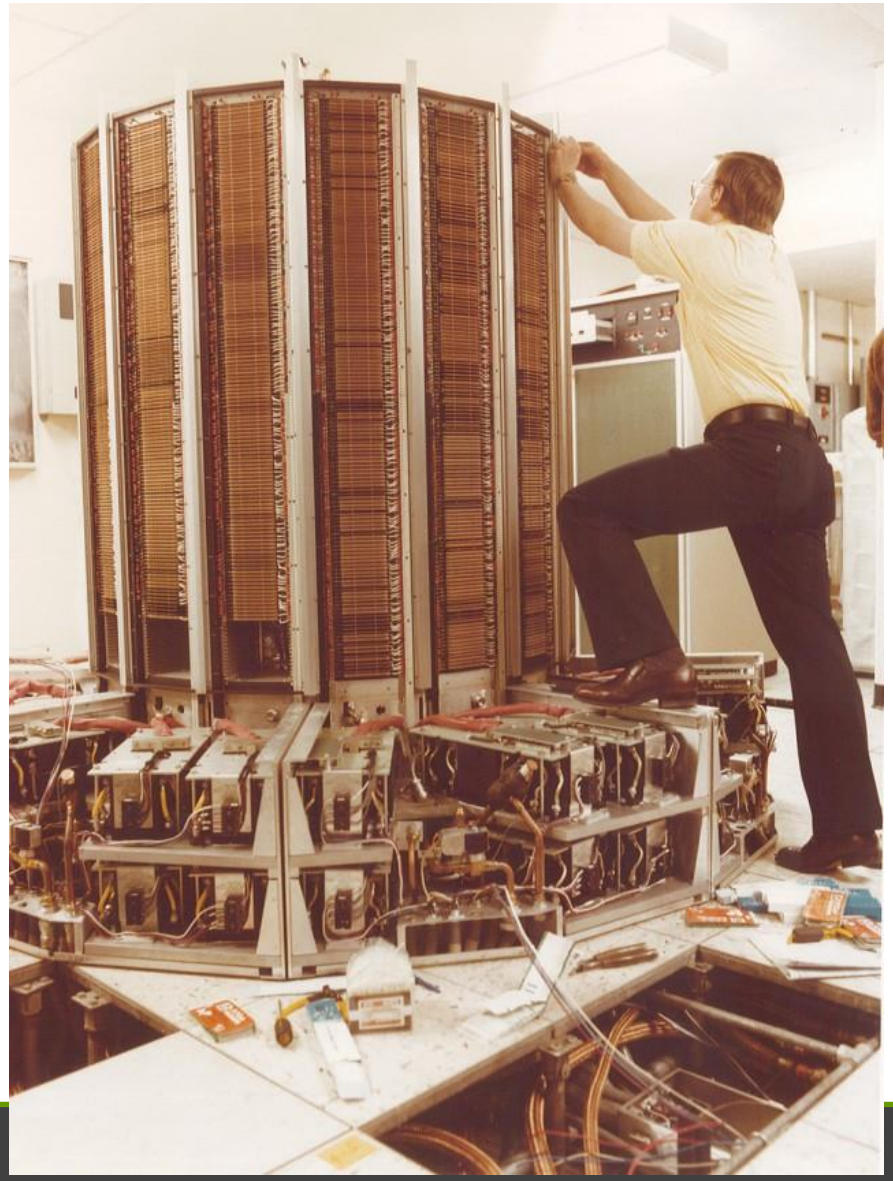
- A supercomputer vector register holds ~32-64 elts
  - The number of elements is larger than the amount of parallel hardware, called vector pipes or lanes, say 2-4
- The hardware performs a full vector operation in
  - #elements-per-vector-register / #pipes



The Vector Years – 1975

- **Seymour Cray left CDC to form Cray Research to make the Cray-1.**
  - A vector processor without compromising the scalar performance using vector registers not pipelined memory operations
  - Uses **ECL transistors**
  - No wires more than 4´ long
  - 8 MB RAM and 80 MHz clock speed
  - Cost $5-$8m, with ~80 sold worldwide
  - Ships with **Cray OS**, **Cray Assembler** and **Cray FORTRAN** – the world's first auto-vectorising FORTRAN compiler

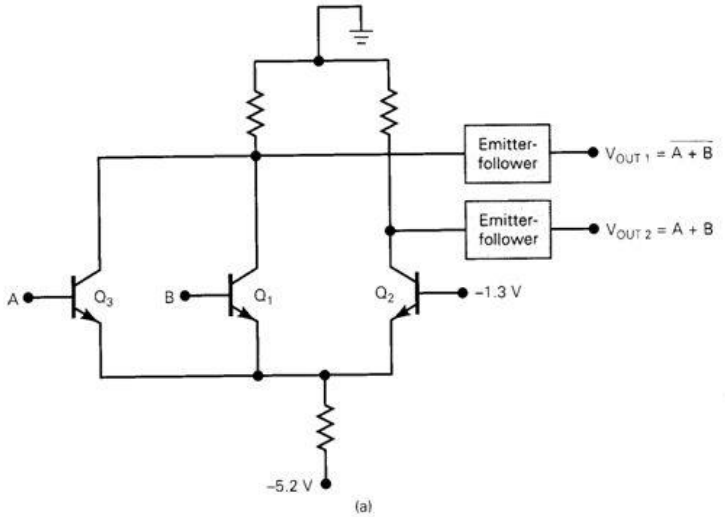
Cray – 1




ECMWF
CRAY-1

http://slideplayer.com/slide/3531418/12/images/19/ECL+EMITTER+COUPLED+LOGIC.jpg

ECL
EMITTER COUPLED LOGIC



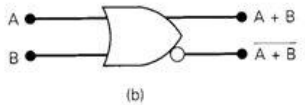


Figure 8-30 (a) ECL NOR/OR circuit; (b) logic symbol.

Cray Companies

- **Cray Research, Inc.** (CRI) 1972. Seymour Cray.
- **Cray Computer Corporation** (CCC) 1989. Spin-off. Bankrupt in 1995.
- **Cray Research, Inc.** bought by Silicon Graphics, Inc (SGI) in 1996.
- **Cray Inc.** Formed when Tera Computer Company (pioneer in multi-threading technology) bought Cray Research, Inc. in 2000 from SGI.

Seymour Cray

- Joined Engineering Research Associates (ERA) in 1950 and h create the ERA 1103 (1953), als known as UNIVAC 1103.
- Joined the Control Data Corpora (CDC) in 1960 and collaborated design of the CDC 6600 and 76
- Formed Cray Research Inc. in 1 when CDC ran into financial difficulties.
  - First product was the Cray-1 supercomputer
  - Faster than all other comput the time.
  - The first system was sold wi month for US$8.8 million.
  - Not the first system to use a vector processor but was the to operate on data on a regis instead of memory

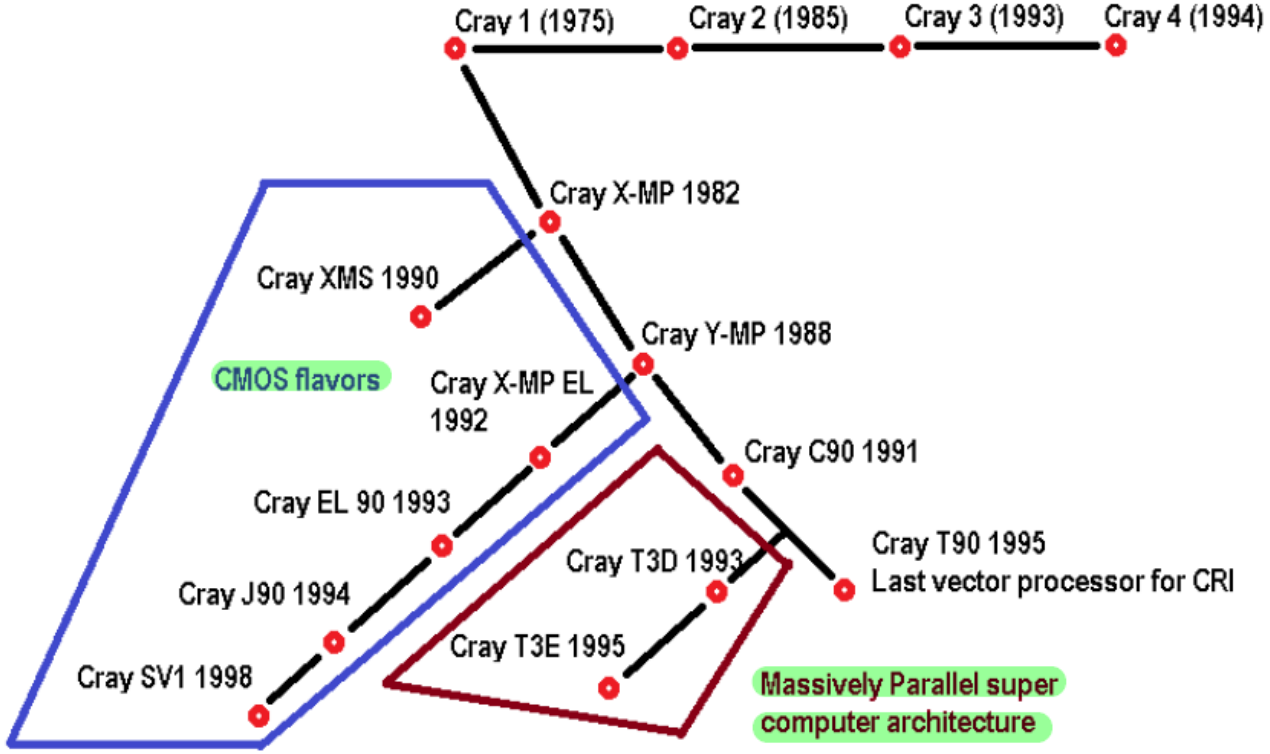
Vector Processor

- CPU that implements an instruction set that operates on one-dimensional arrays of data called *vectors*.
- Appeared in the 1970s, formed the basis of most supercomputers through the 80s and 90s.
- In the 60s the Solomon project of Westinghouse wanted to increase math performance by using a large number of simple math co-processors under the control of a single master CPU.
- The University of Illinois used the principle on the ILLIAC IV. The original design wanted a 1 GFLOP machine with 256 ALUs, but it only had 64 ALUs and could reach only 100 to 150 MFLOPS (Not bad for 1972).

Vector Processo

- CDC had the first pract vector processor with t
  - Used memory-to-memor
  - Uses vectors of any leng
  - Pipeline had to be very l instructions to make up
  - High cost when switchin located operands
  - Poor scalar performance

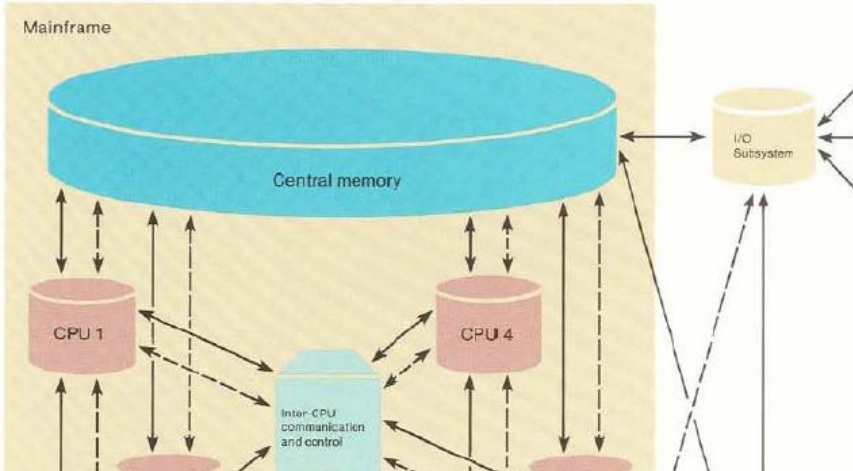
Cray Vector Processors Timeline



Cray X-MP (1982)

- Shared memory parallel vector processor (first)
- Better memory bandwidth, two read ports to memory instead of one
- Improved chaining support
- 32 memory banks
- Two read ports, one write port, dedicated port to I/O
- 400MFlops, 105MHz
- Further versions included 1, 2 or 4 processors. Up to 800MFlops

Cray X-MP



Cray Failed Projects

- **Cray 3 1993**
  - First to use gallium arsenide
  - had a foreground processing system dedicated to I/O with 32 bit processor and 4 synchronous data channels
  - Up to 16 background processors
  - Up to 16GB common memory
  - Background processor had a computation section, control section and local memory
  - 4 to 16 processors at 474MHz
- **Cray 4 1994**
  - 4 to 64 processors at 1GHz
  - 8GB of memory
  - 32 Gflops
  - Went back to B & T registers, due to local memory failure

Cray Research Sup

- APP (1992)
  - up to 84 processors (Intel i860) arrar performance of 6.7 GFLOPS
  - acted as a co-processor for SPARC
- S-MP (1992)
  - eight SPARC processors, 66 MHz
  - supported the APP co-processor
- CS6400 (1993)
  - up to 64 SPARC processors, 60 MH
  - 16 GB RAM, JTAG bus control
  - most successful of the servers, ende up sold to Sun in the acquisition, led the Sun Enterprise 10000

Cray, Inc.

- *Tera Computer Company* re Cray Research from SGI, stil
- Combined Cray Research ar designed by Tera, NEC Corp
- Received funding from the N including some classified wo

Cray, Inc.

- SX-6 (2001)

1970s – IBM VM/370

- The Conversational Monitor System(CMS) is a single-user operating system, while the OS/370 and DOS/370 are multiprogramming operating systems.
- A user process is unaware of the presence of the VM/370—it sees only the guest OS that it uses.

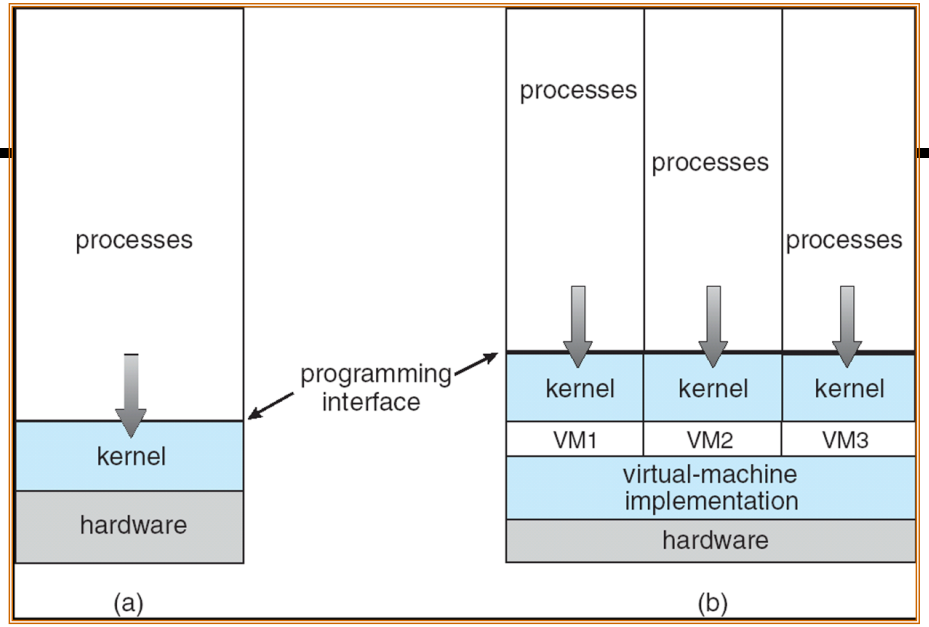


Figure 4.5 Virtual machine operating system VM/370.

https://en.wikipedia.org/wiki/History_of_personal_computers

Personal Computer – 1970s

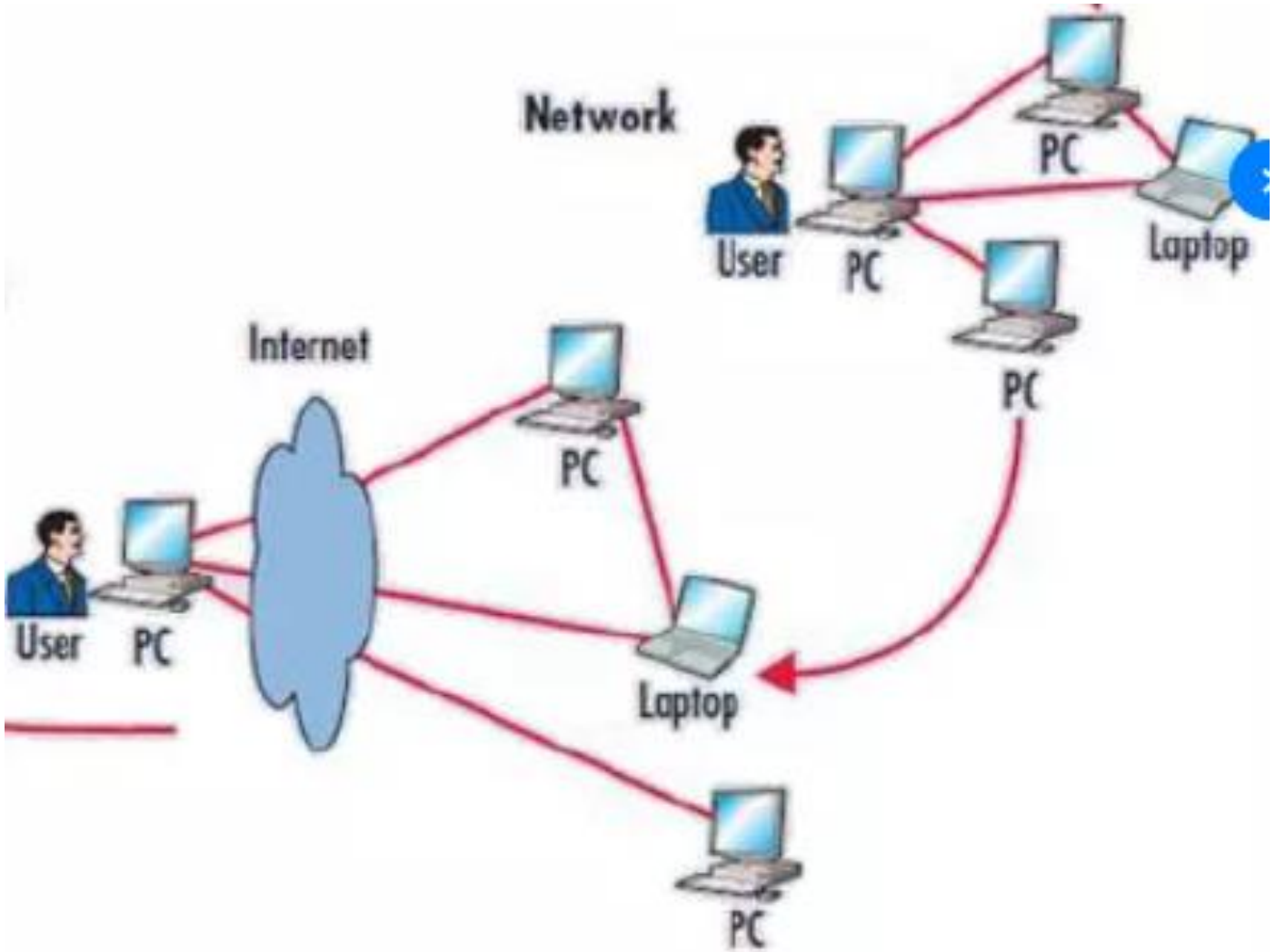
- **Xerox Alto and Star**
  - The Xerox Alto, developed at Xerox PARC in **1973**, was the first computer to use a mouse, the desktop metaphor, and a graphical user interface(GUI), concepts first introduced by Douglas Engelbart while at International.

Ethernet – 1973

https://en.wikipedia.org/wiki/Ethernet

- Ethernet was developed at Xerox PARC between 1973 and 1974.
  - On May 22, 1973, Bob Metcalfe (then at the Xerox Palo Alto Research Center, PARC, in California) wrote a memo describing the Ethernet network system he had invented for interconnecting advanced computer workstations, making it possible to send data to one another and to high-speed laser printers.
  - Probably the best-known invention at Xerox PARC was the first personal computer workstation with graphical user interfaces and mouse pointing device, called the Xerox Alto.
  - The PARC inventions also included the first laser printers for personal computers, and, with the creation of Ethernet, the first high-speed LAN technology to link everything together.

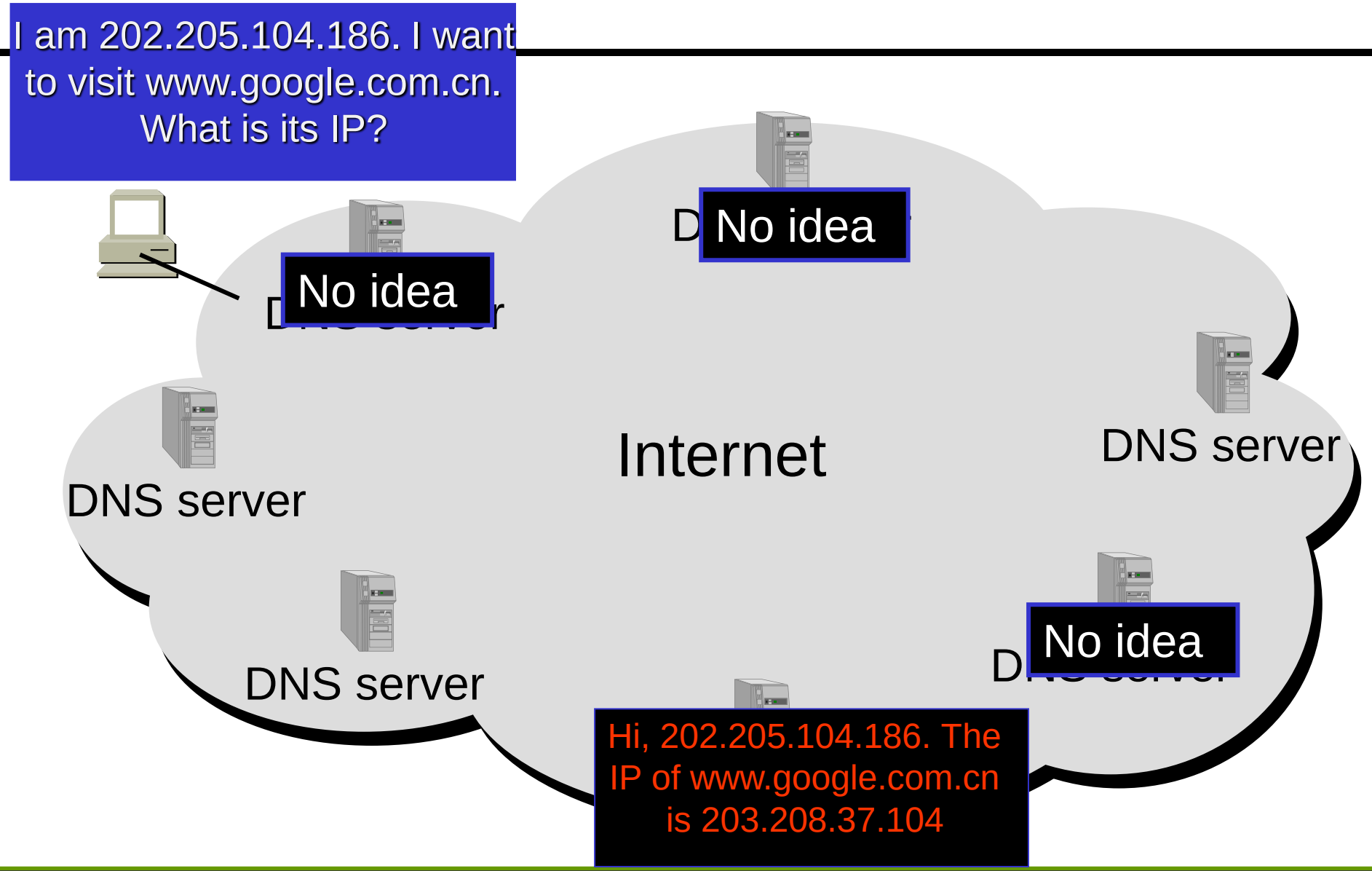
https://www.oreilly.com/library/view/ethernet-the-definitive/1565926609/ch01.html


Network
User
PC
PC
Laptop
PC
Internet
PC
User
PC
Laptop
PC

DNS system – 1983

- **DNS - Domain Name Service**
  - Proposed in 1983 by Paul Mockapetris
  - Aims to assign IP with semantic meaning

Dynamics of DNS services



The Vector Years - 1981

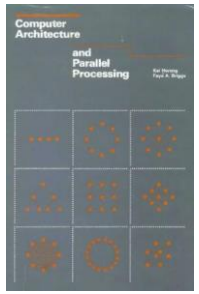
- **CDC Cyber-205**
  - CDC put right the mistakes made with the Star-100
  - **1-4 separate vector units**
  - Rarely got anywhere near peak speed except with hand-crafted assembly code
  - Used semiconductor memory and virtual memory concept

CDC Cyber-205

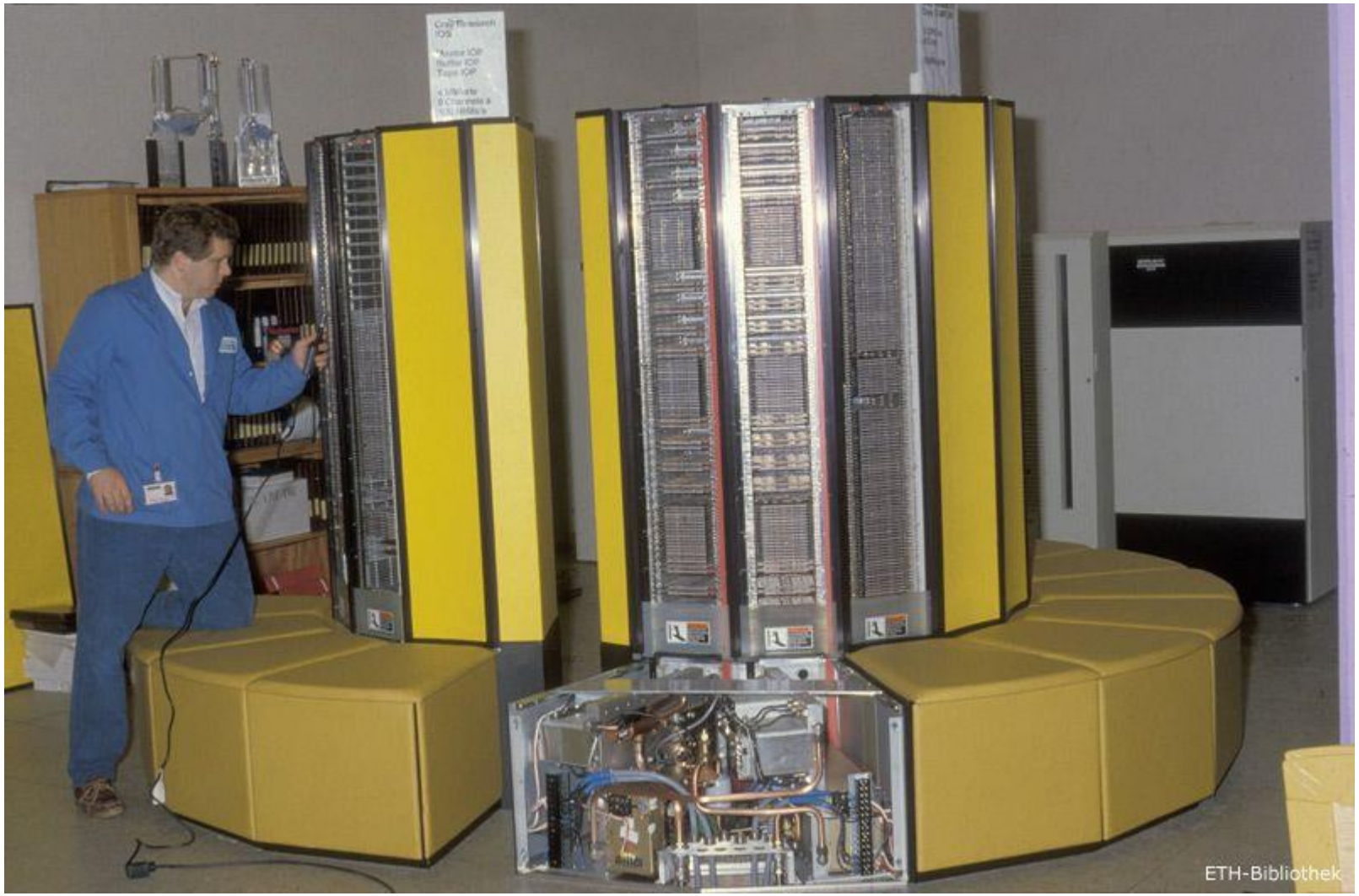
The Vector Years – 1983

- **Cray X-MP**
  - A parallel (1-4) vector processor machine with 120 MHz clock speed for ~125 MFLOPS/CPU with 8-128 MB of RAM main memory
  - Better chaining support, parallel arithmetic pipelines and shared memory access with multiple pipelines per processor.
  - Switched from Cray OS to **UniCOS** (a UNIX variant) in 1984
  - Typical cost ~$15m plus disks!

Cray Research recently announced its multiprocessor model the Cray X-MP. This is a dual-processor system highly pipelined for both scalar and vector processing at high speed. Denelcor, Inc. developed the HEP computer, which can be



Cray X-MP



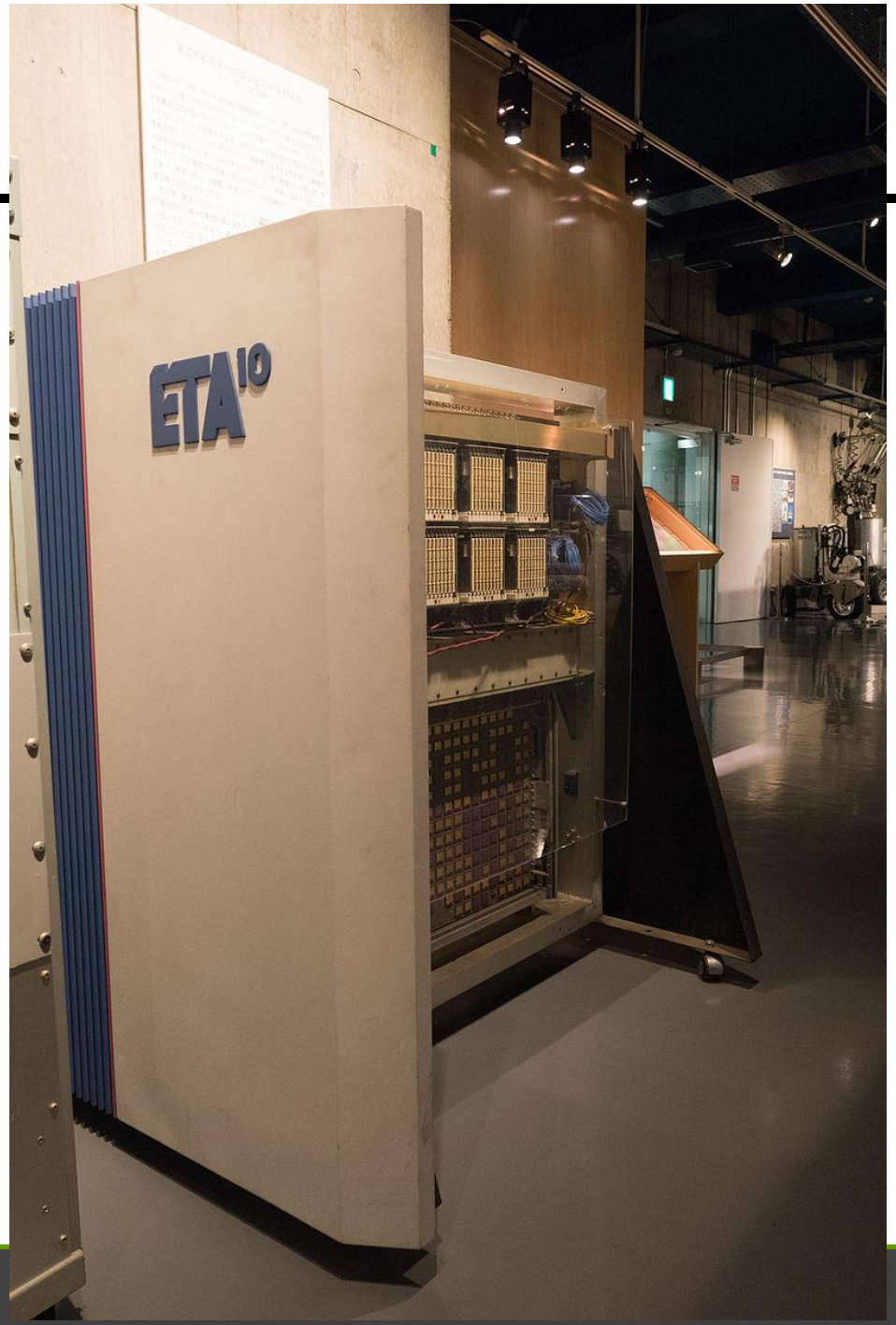
The Vector Years – 1985

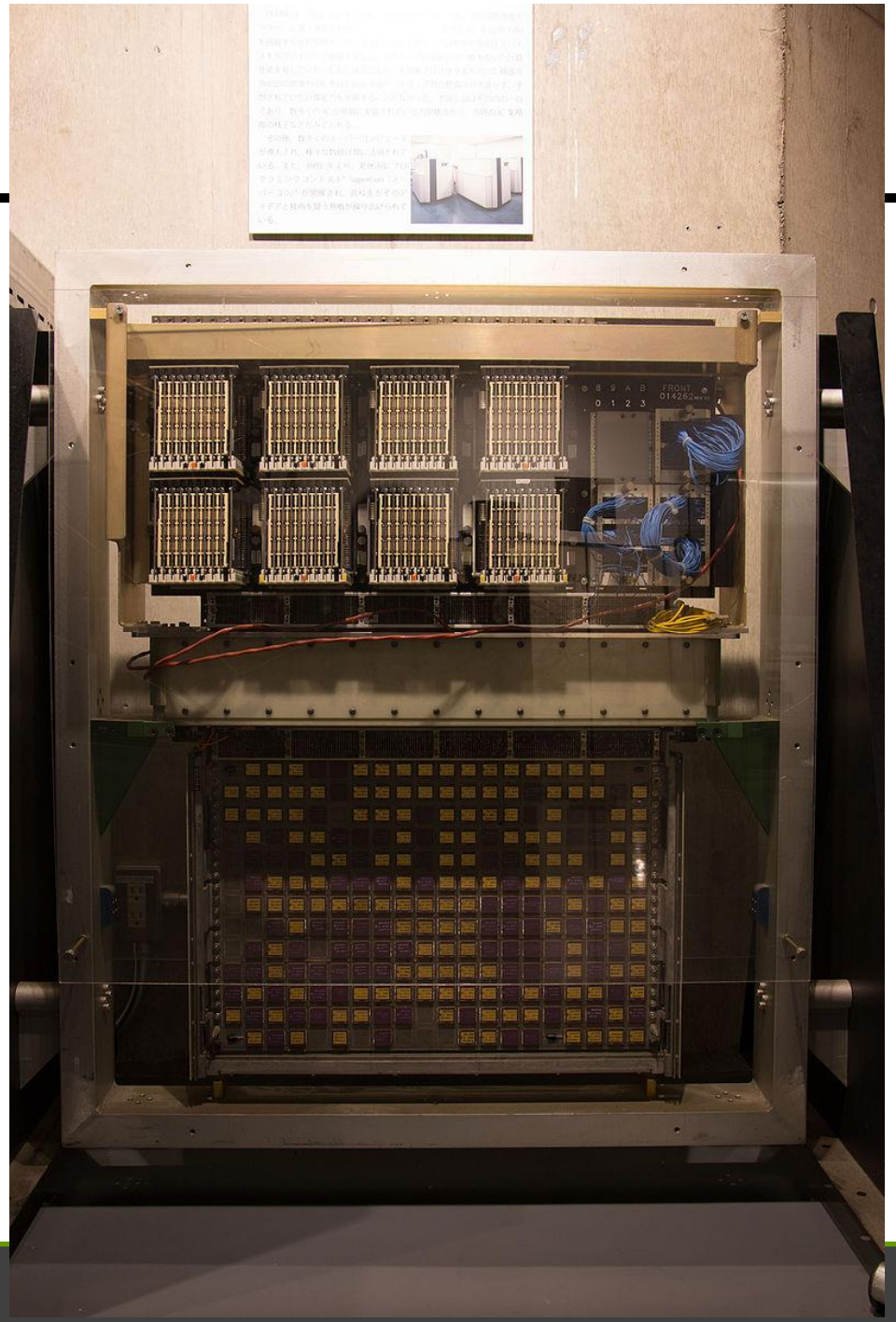
- **Cray-2**
  - A completely new, compact 4-8 processor design
  - Had 512 MB to 4 GB of main memory but with higher memory latency than X-MP
  - Hence X-MP faster than Cray-2 on certain problems – impact of memory architecture on compute speed

- **Cray Y-MP introduced in 1988 – an evolution of the X-MP with up to 16 processors (new type).**

The Vector Years – 1989

- **ETA-10G**
  - **Spin-off company** from CDC due to competition from Cray, with only one product the ETA-10.
  - Compatible with CDC Cyber-205, including pipelined memory not vector registers.
  - **Shared memory multiprocessor** (up to 8) with up to 32MB of private memory/CPU plus common access to up to 2GB of shared memory.
  - 2 variants one with liquid nitrogen cooling and the other with air cooling for CMOS components
  - 7 liq-N2 and 27 air-cooled units sold
  - A failure - remaining units given to high schools!


ETA10



1999

https://en.wikipedia.org/wiki/Grid_computing

What is Grid? – Cooperate HPCs

- **An infrastructure that dynamically couples**
  - Computers (PCs, workstations, clusters, traditional supercomp and even laptops, notebooks, mobile computers, PDA, and so on)
  - Software (e.g., renting special purpose applications on demand)
  - Databases (e.g., transparent access to human genome database)
  - Special Instruments (e.g., radio)
  - People
- **across the local/wide-area networks (enterprise, organisations, or Internet) and presents them as a unified resource or problem solving environment.**

P2P 1999 - Napster

- 5/1999: Shawn Fanning (freshman, Northeastern University) founds Napster Online (supported by Groove)
- 12/1999: First lawsuit
- 7/2001: simultaneous online users 160K
- 6/2002: file bankrupt
- …
- 10/2003: Napster 2 (Supported by Roxio) (users should pay $9.99/month)

Napster -- Shawn Fanning

IoT proposed in 1990s

- The concept of a network of smart devices was discussed as early as 1982, with a modified Coke machine at Carnegie Mellon University becoming the first Internet-connected appliance,[7] able to report its inventory and whether newly loaded drinks were cold.[8]

- Mark Weiser's 1991 paper on ubiquitous computing, "The Computer of the 21st Century", as well as academic venues such as UbiComp and PerCom produced the contemporary vision of IoT.

UBICOMP 2019

https://en.wikipedia.org/wiki/Internet_of_things#History

The Structure of IoT

The IoT can be viewed as a gigantic network consisting of networks of devices and computers connected through a series of intermediate technologies where numerous technologies like RFIDs, wireless connections may act as enablers of this connectivity.

- ***Tagging Things* : Real-time item traceability and addressability by *RFID*s.**
- ***Feeling Things* : *Sensors* act as primary devices to collect data from the environment.**
- ***Shrinking Things* : *M*iniaturization and *Nanotechnology* has provoked the ability of smaller things to interact and connect within the "things" or "smart devices."**
- ***Thinking Things* : *Embedded intelligence* in devices through sensors has formed the network connection to the Internet. It can make the "things" realizing the intelligent control.**

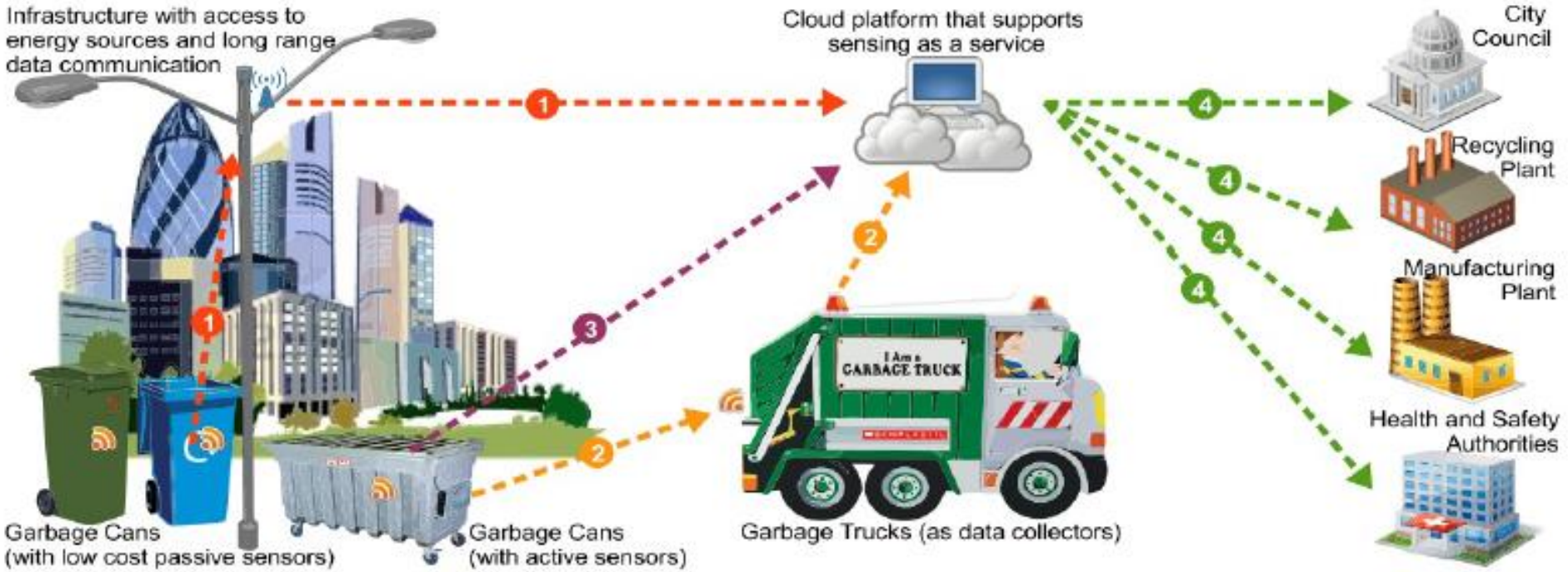
Applications of IoT

- ✓ Building and Home automation
- ✓ Manufacturing
- ✓ Medical and Healthcare systems
- ✓ Media
- ✓ Environmental monitoring
- ✓ Infrastructure management
- ✓ Energy management
- ✓ Transportation
- ✓ Better quality of life for elderly
- ✓ … … …

Smart Appliances

You name it, and you will have it in IoT!

Efficient Waste Management in Smart Cities Supported by the Sensing-as-a-Service



[Source: "Sensing as a Service Model for Smart Cities Supported by Internet of Things", Charith Perera et. al., Transactions on Emerging Telecommunications Technology, 2014]

IOT Application Scenario - Shopping

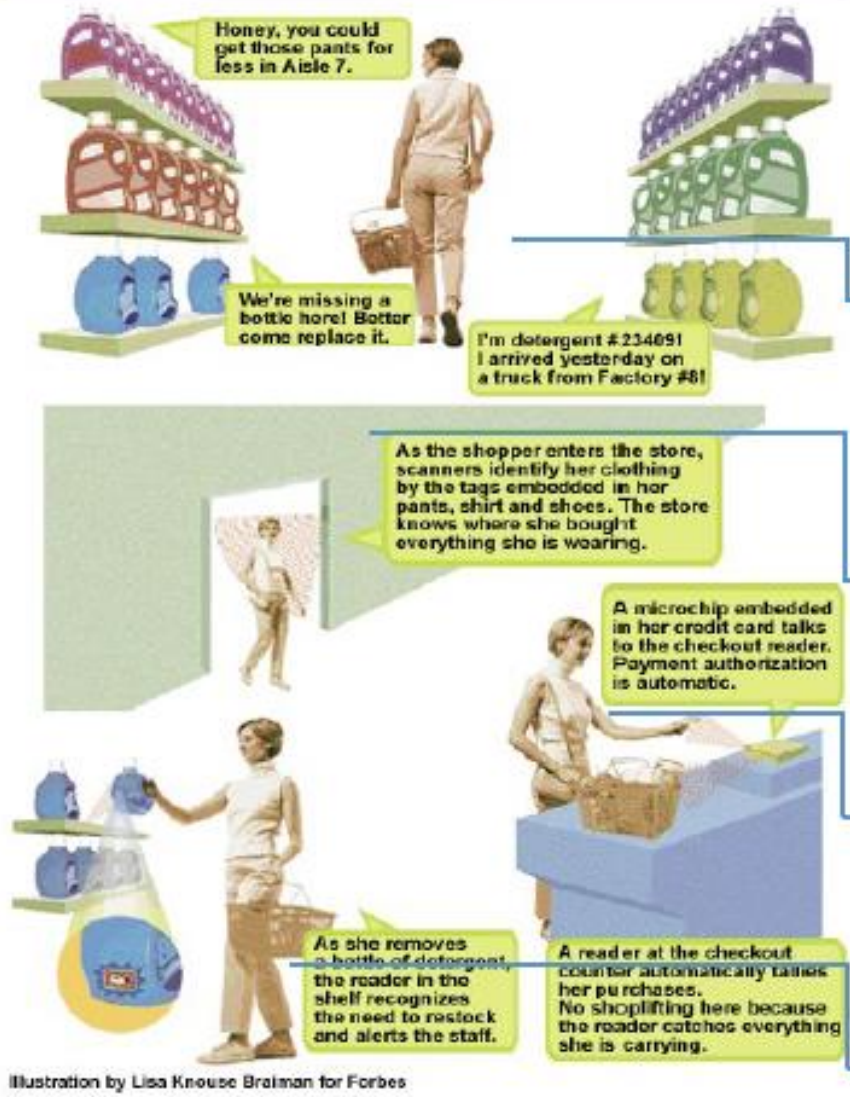


Illustration by Lisa Krouse Brahman for Forbes

(2) When shopping in the market, the goods will introduce themselves.

(1) When entering the doors, scanners will identify the tags on her clothing.

(4) When paying for the goods, the microchip of the credit card will communicate with checkout reader.

(3) When moving the goods, the reader will tell the staff to put a new one.

2018年01月03日 20:06

http://tech.sina.com.cn/i/2018-01-03/doc-ifyqkarr6869082.shtml

http://www.sohu.com/a/210722935_413980

Innovation implies in the history - GPU

- GPU (graphics processing unit) is a RISC specialized processor that offloads 3D graphics rendering from microprocessor.
- 1970s:
  - **ANTIC (Alphanumeric Television Interface Controller) and CTIA (Color Television Interface Adaptor)** chips provided for hardware control of mixed graphics and text modes, sprite positioning and display, and other operations based on Atari 8-bit computers.

https://en.wikipedia.org/wiki/ANTIC

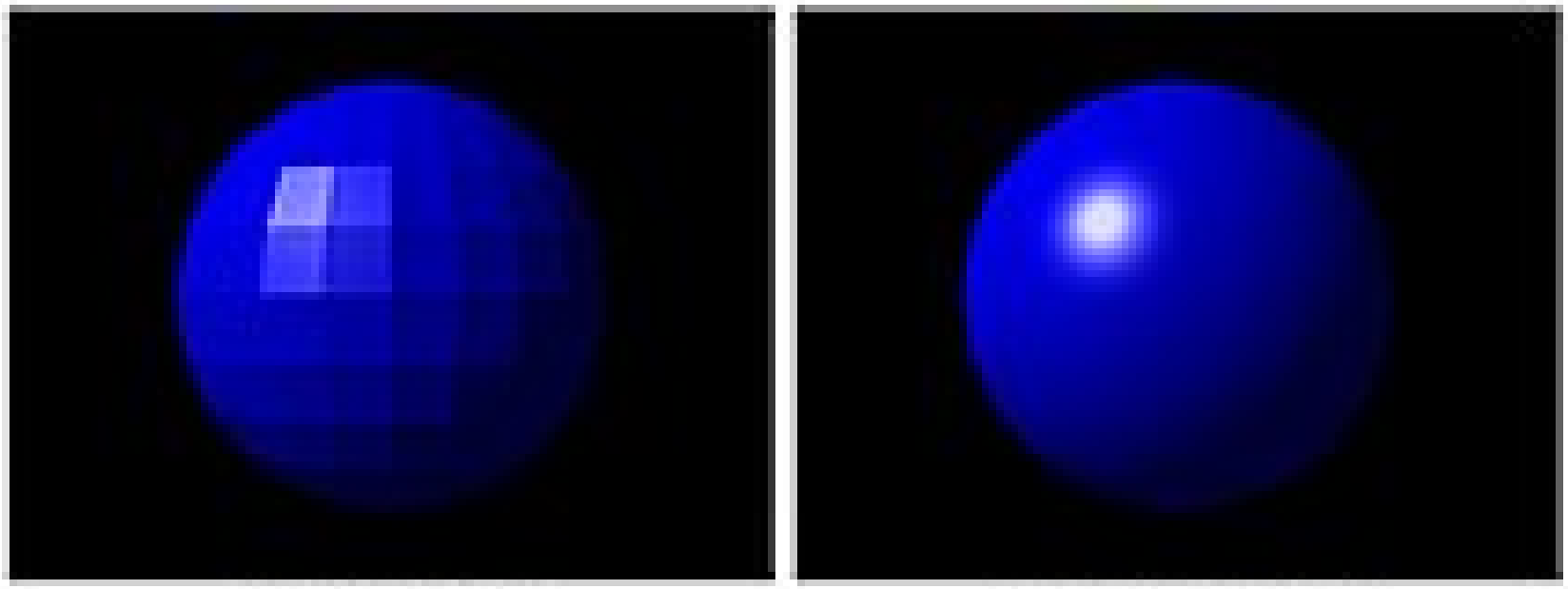
https://en.wikipedia.org/wiki/CTIA_and_GTIA#2600_and_TIA

- 1980s:
  - IBM Professional Graphics Controller was one of the very first 2D/3D graphics accelerators available for the IBM PC, released in 1984. But it was expensive, slow and lack of compatibility.

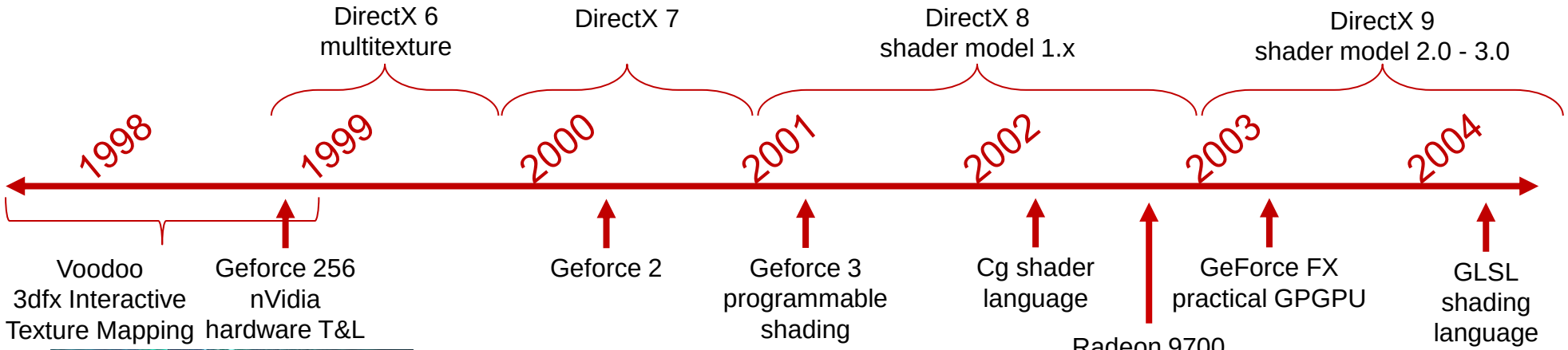
- **1990s:**
  - S3 Graphics introduced the first single-chip 2D accelerator (S3 86C911); in mid-1990s, PlayStation and Nintendo 64 developed hardware-accelerated 3D graphics for the requirement of game market.
- **2000s:**
  - **2001 NVIDIA** was first to produce a chip capable of **programmable shading (着色)**, GeForce 3 (NV20);
  - Oct. 2002, ATI Radeon 9700 (R300), was introduced as the world's first Direct3D 9.0 accelerator.
  - 2005 – Massively parallel programmable processors
  - 2007 – **CUDA** (Compute Unified Device Architecture)

https://zh.wikipedia.org/wiki/着色器

- 计算机图形学领域中，**着色器**（英语：shader）是一种计算机程序，原本用于进行图像的浓淡处理（计算图像中的光照、亮度、颜色等），但近来，它也被用于完成很多不同领域的工作，比如处理CG特效、进行与浓淡处理无关的影片后期处理、甚至用于一些与计算机图形学无关的其它领域。



Programmable GPUs



1998, Ocarina of Time

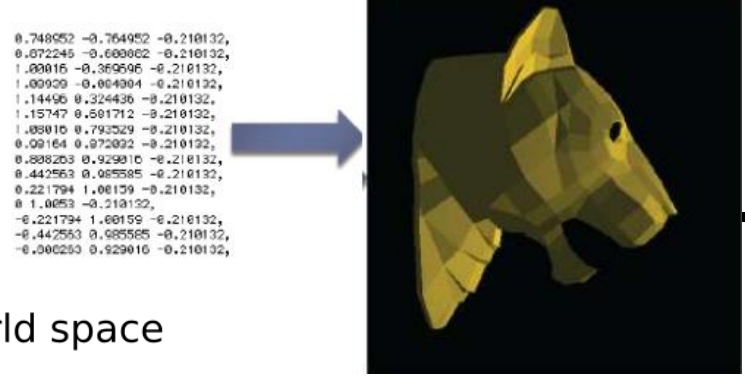
1999, System Shock II

2000, Baulder's Gate II

2002, Warcraft III

2004, Fable

They all draw pixels



Polygon mesh

Vertex Processing

Rasterization

Fragment Processing

Raster Operation

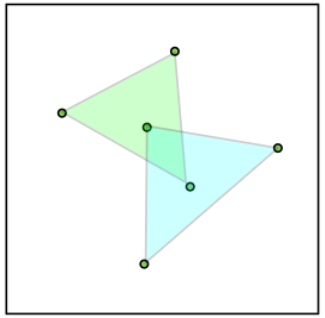
Pixels

Polygon mesh in world space

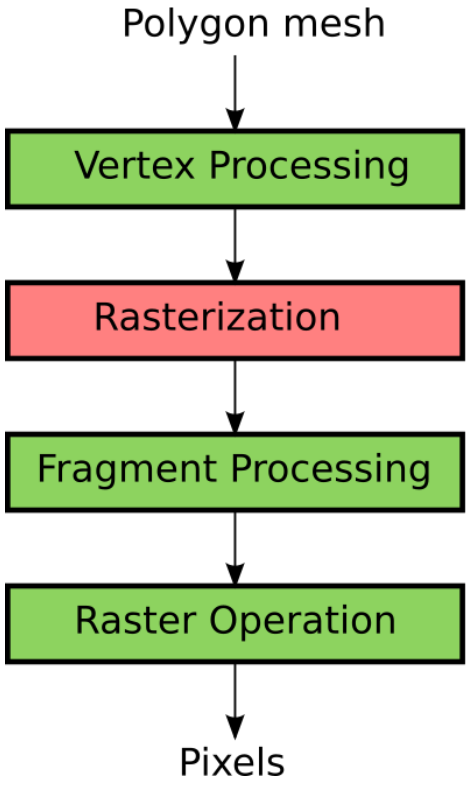
Vertex Processing

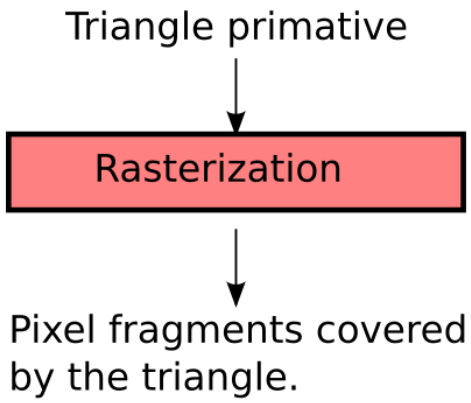
Transform from world space to camera/eye space. Provide vertices with lighting, color, etc.

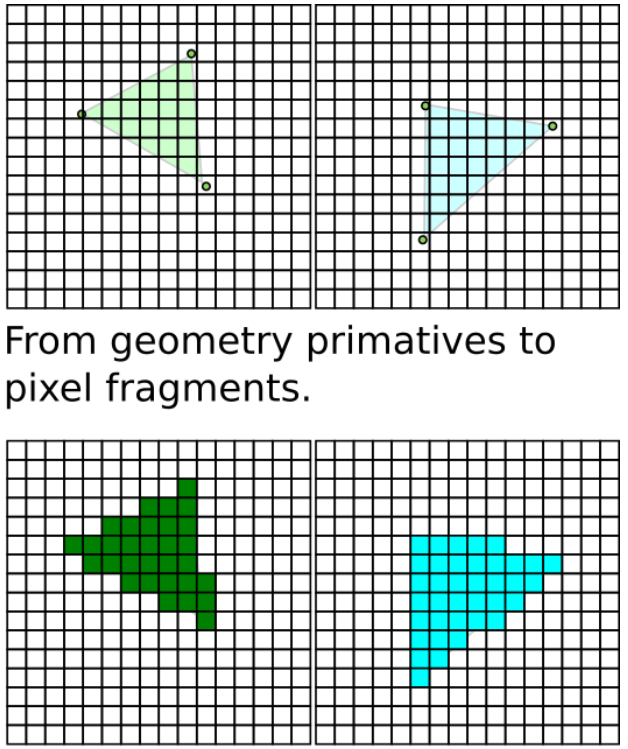
Polygons mesh in eye space. Vertices are attached with lighting, color, etc



K. Fatahalian, et al. "GPUs: a Closer Look", ACM Queue 2008, http://doi.acm.org/10.1145/1365490.1365498

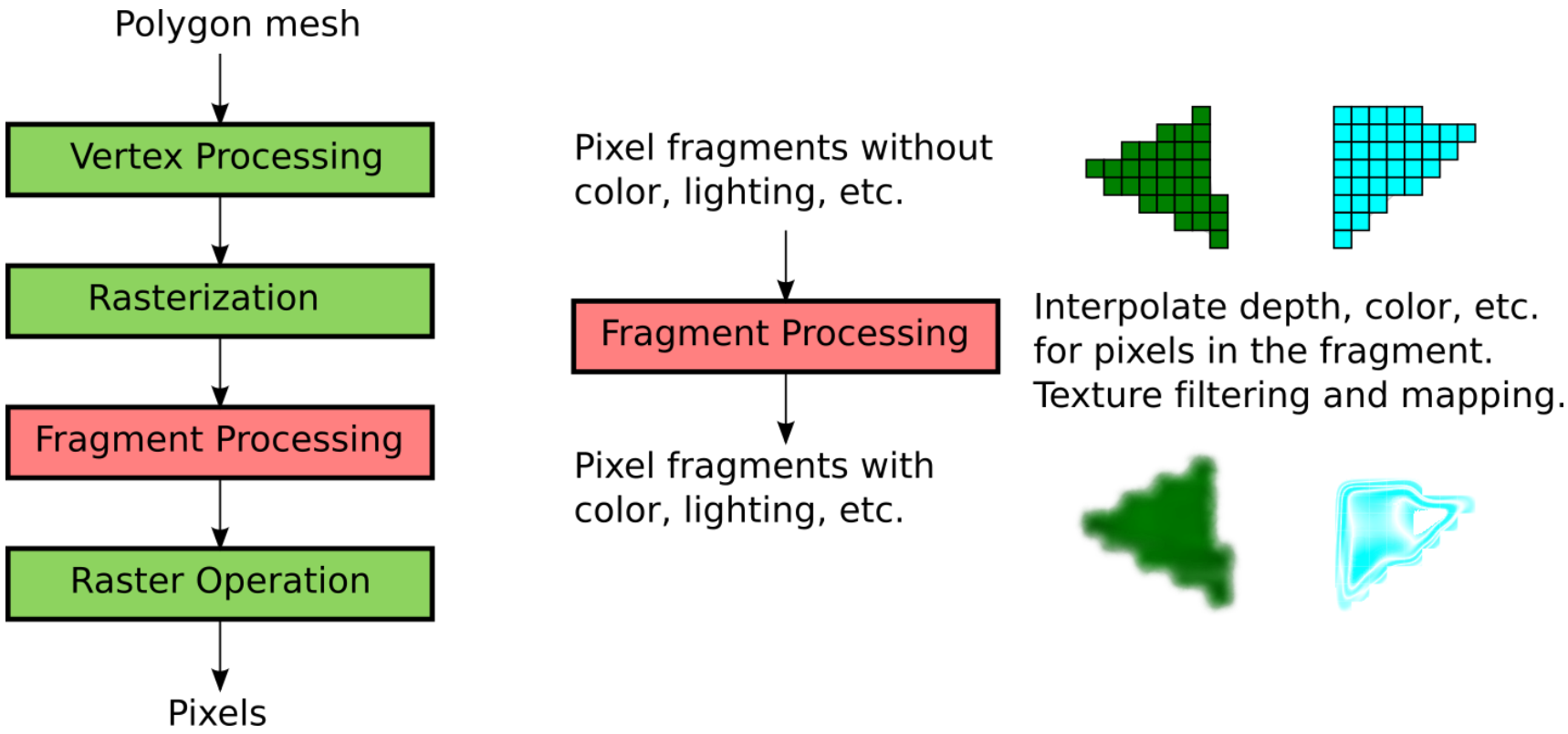




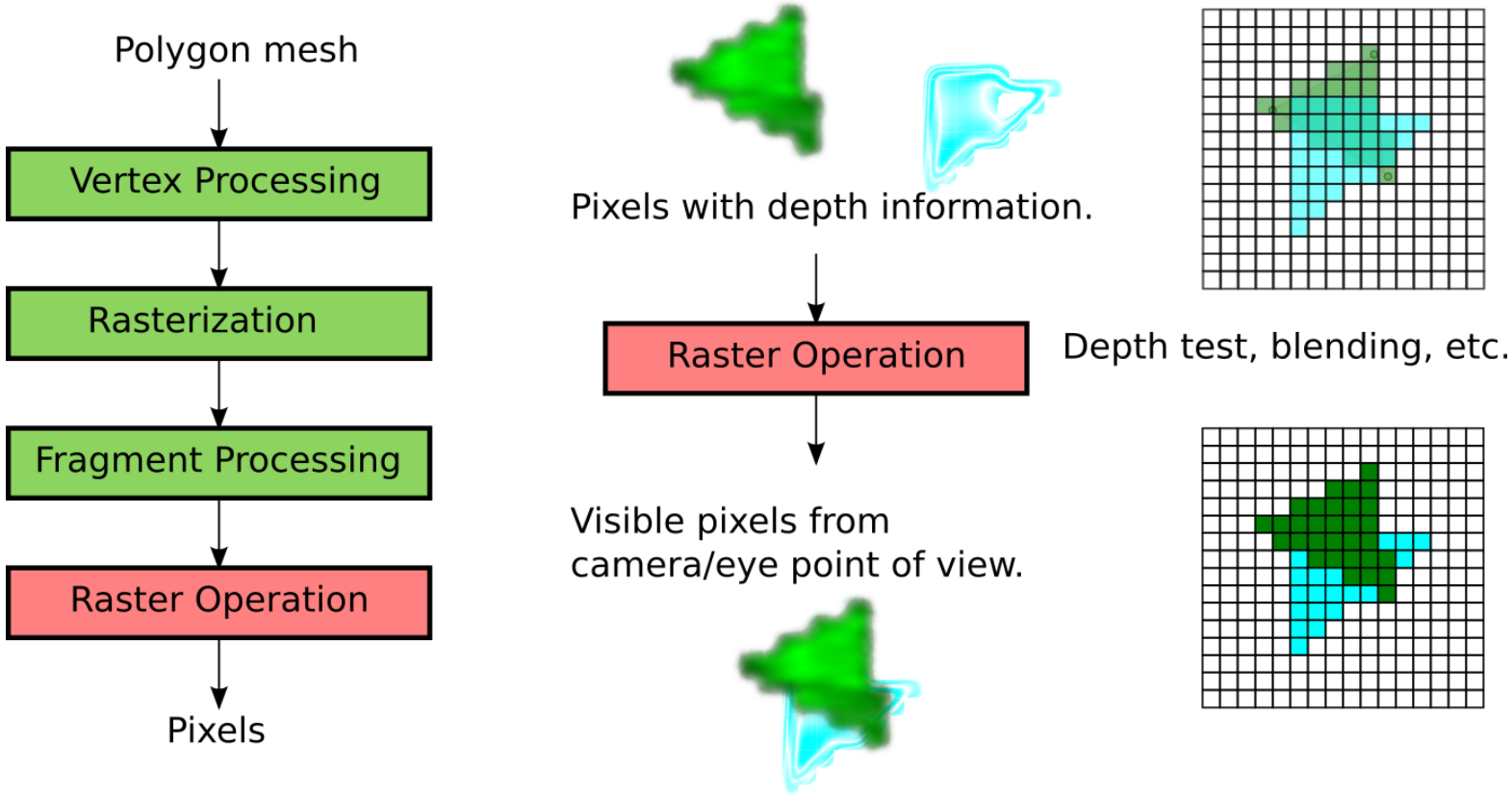


From geometry primatives to pixel fragments.

K. Fatahalian, et al. "GPUs: a Closer Look", ACM Queue 2008, http://doi.acm.org/10.1145/1365490.1365498

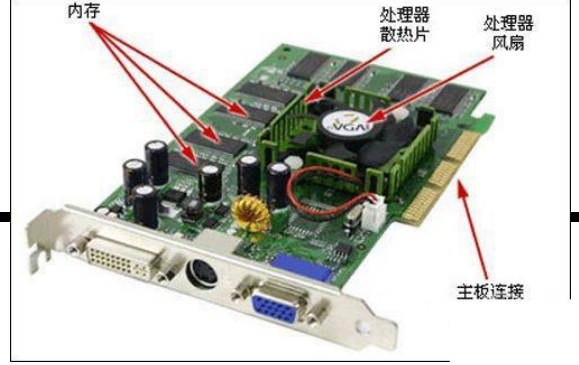


K. Fatahalian, et al. "GPUs: a Closer Look", ACM Queue 2008, http://doi.acm.org/10.1145/1365490.1365498



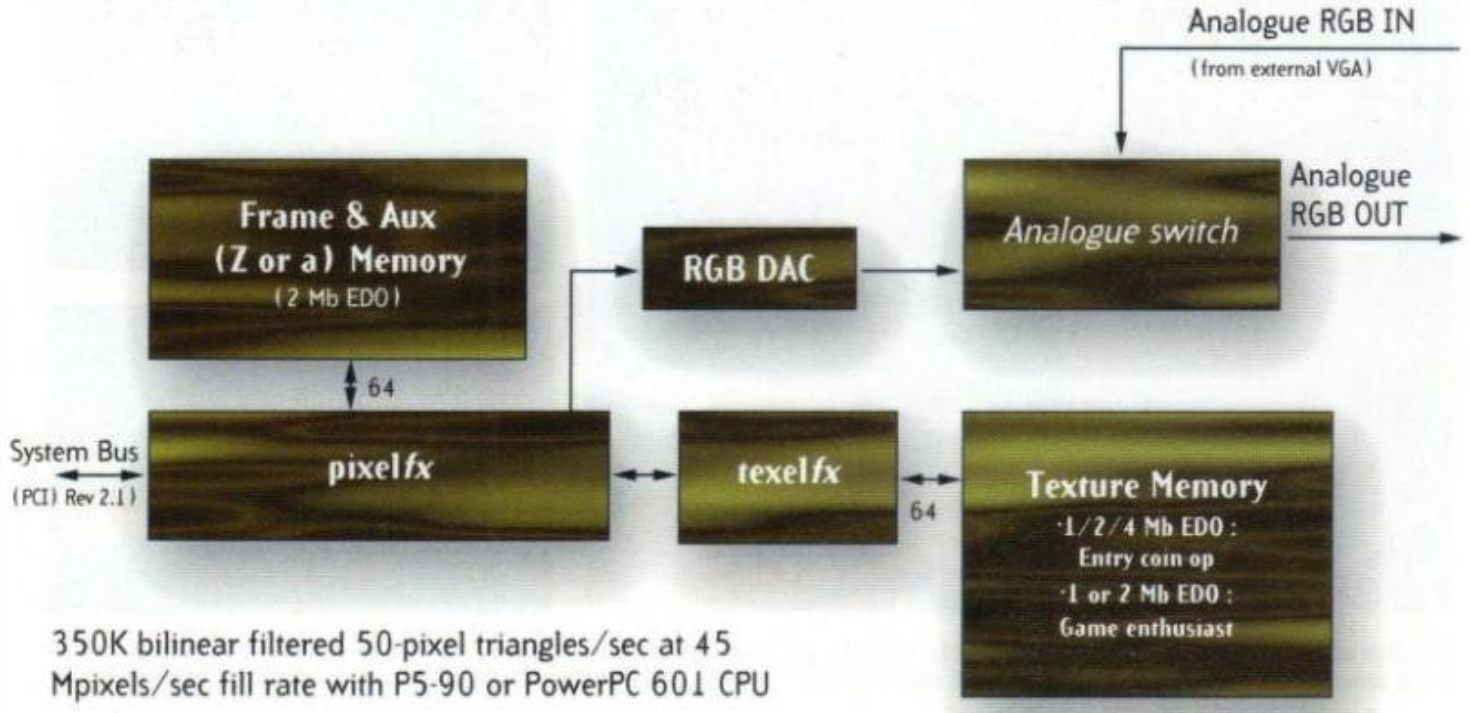
K. Fatahalian, et al. "GPUs: a Closer Look", ACM Queue 2008, http://doi.acm.org/10.1145/1365490.1365498

Video Card – old name



- Specialized RISC chip

Voodoo graphics game enthusiast board

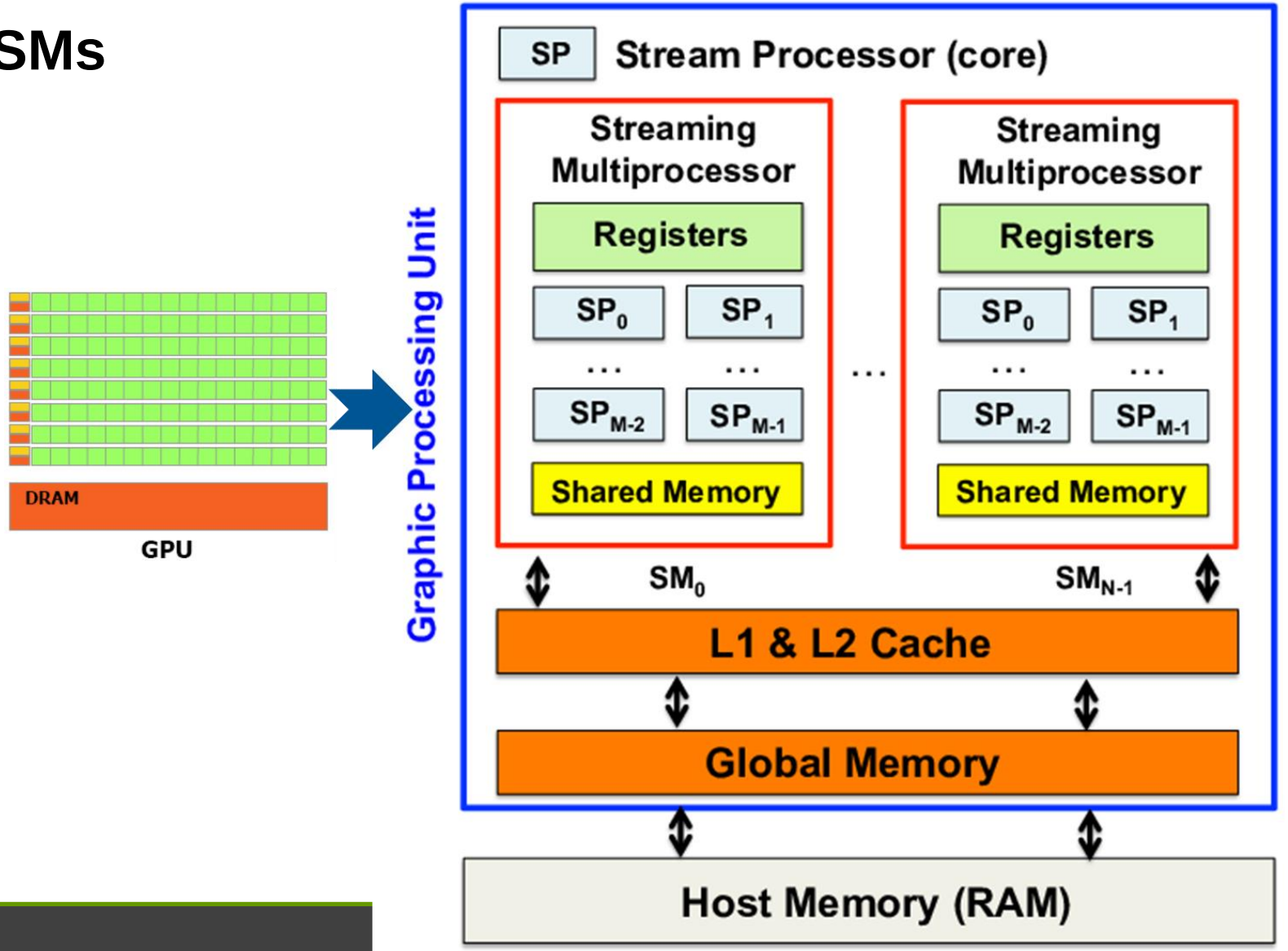


According to 3Dfx Interactive, 'the first chip, pixel*fx*, is the primary graphics controller and contains interfaces to the PCI bus and companion texture-processing unit, texel*fx*. The 3Dfx Interactive pixel*fx* graphics controller is packaged in a 240-pin PQFP. Texel*fx*, the advanced texture-processing unit, is packaged in a 208-pin PQFP'. For arcade use the board is scalable and extra texel*fx* chips and texture memory can be added in parallel

- **Before graphics-programming APIs were introduced, 3D applications issued their commands directly to the graphics hardware**
  - Fast
  - Became infeasible with increasing graphics hardware
- **Graphics APIs like DirectX and OpenGL act as a middle layer between the application and the graphics hardware**
- **Using this model, applications write one set of code and the API does the job of translating this code to instructions that can be understood by the underlying hardware**
- **A product of detailed collaboration among**
  - Application developers，Hardware designers，API/runtime architects

GPU/CUDA – current popular name

- Much more SMs

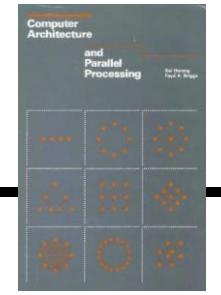


Chapter 3: Large Scale Computing Systems

Faster for larger data

- von Neumann architecture
  - Foundation of modern computers
  - 1960 2 CPUs
- 1962 Channel
  - Origin of concurrent programming
- Parallel
  - Vector processor, Multi-core, later GPU/CUDA
- Distributed
  - Cluster, Grid, …
- Now Clouding – Virtualizing computer systems for so-called Big Data
  - IaaS, PaaS, SaaS, …

The Conventional Era – 1979 – MPP



6.3 THE MASSIVELY PARALLEL PROCESSOR

A large-scale SIMD array processor has been developed for processing satellite imagery at the NASA Goddard Space Flight Center. The computer has been named *massively parallel processor* (MPP) because of the 128 × 128 = 16,384 microprocessors that can be used in parallel. The MPP can perform bit-slice arithmetic computations over variable-length operands. The MPP has a micro-programmable control unit which can be used to define a quite flexible instruction

6.3.1 The MPP System Architecture

In 1979, NASA Goddard awarded a contract to Goodyear Aerospace to construct a massively parallel processor for image-processing applications. The major hardware components in MPP are shown in Figure 6.16. The *array unit* operates with SIMD mode on a two-dimensional array of 128 × 128 PEs. Each PE is associated with a 1024-bit random-access memory. Parity is included to detect memory faults. Each PE is a bit-slice microprocessor connected to its nearest neighbors. The programmer can connect opposite array edges or leave them open so that the array topology can change from a plane to a horizontal cylinder, a vertical cylinder, or a torus. This feature reduces routing time significantly in a number of imaging applications.

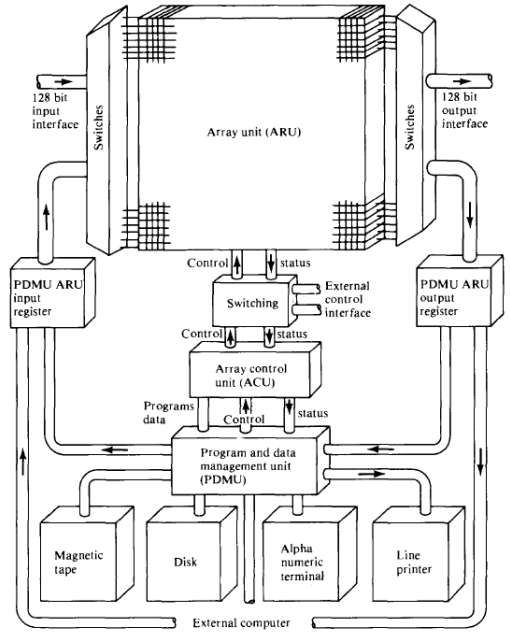
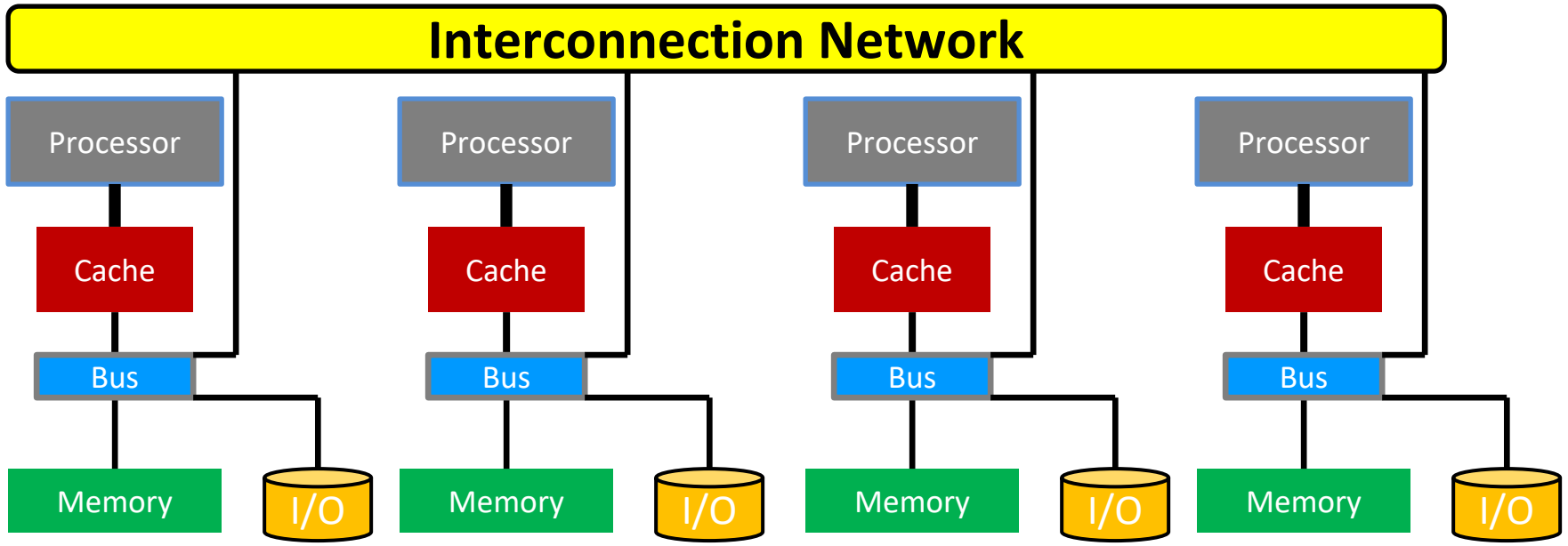


Figure 6.16 The system architecture of the MPP system. (Courtesy of *IEEE Trans. Computers*, Batcher, 1980.)

MPP: Massively Parallel Processors

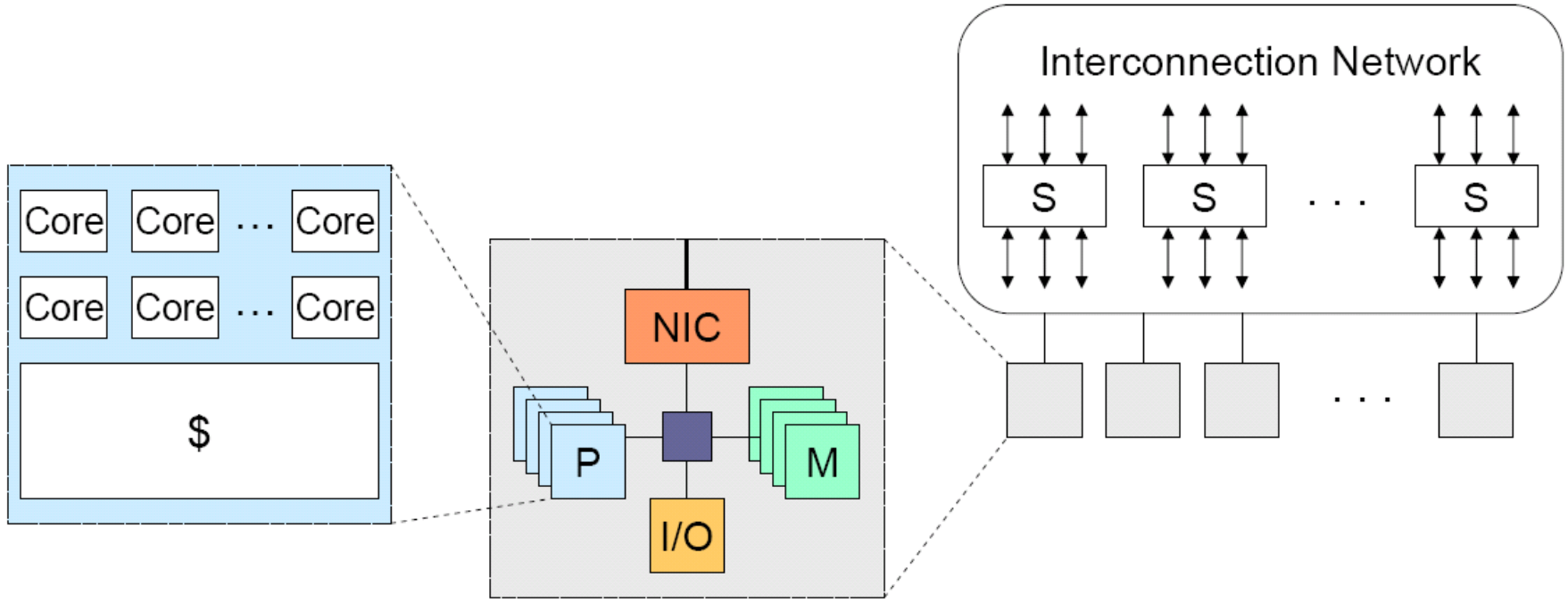
- Massively Parallel **Processors** (MPP) architecture consists of nodes with each having its own processor, memory and I/O subsystem
- An independent OS runs at each node



1995 – MPP - Intel ASCI-Red

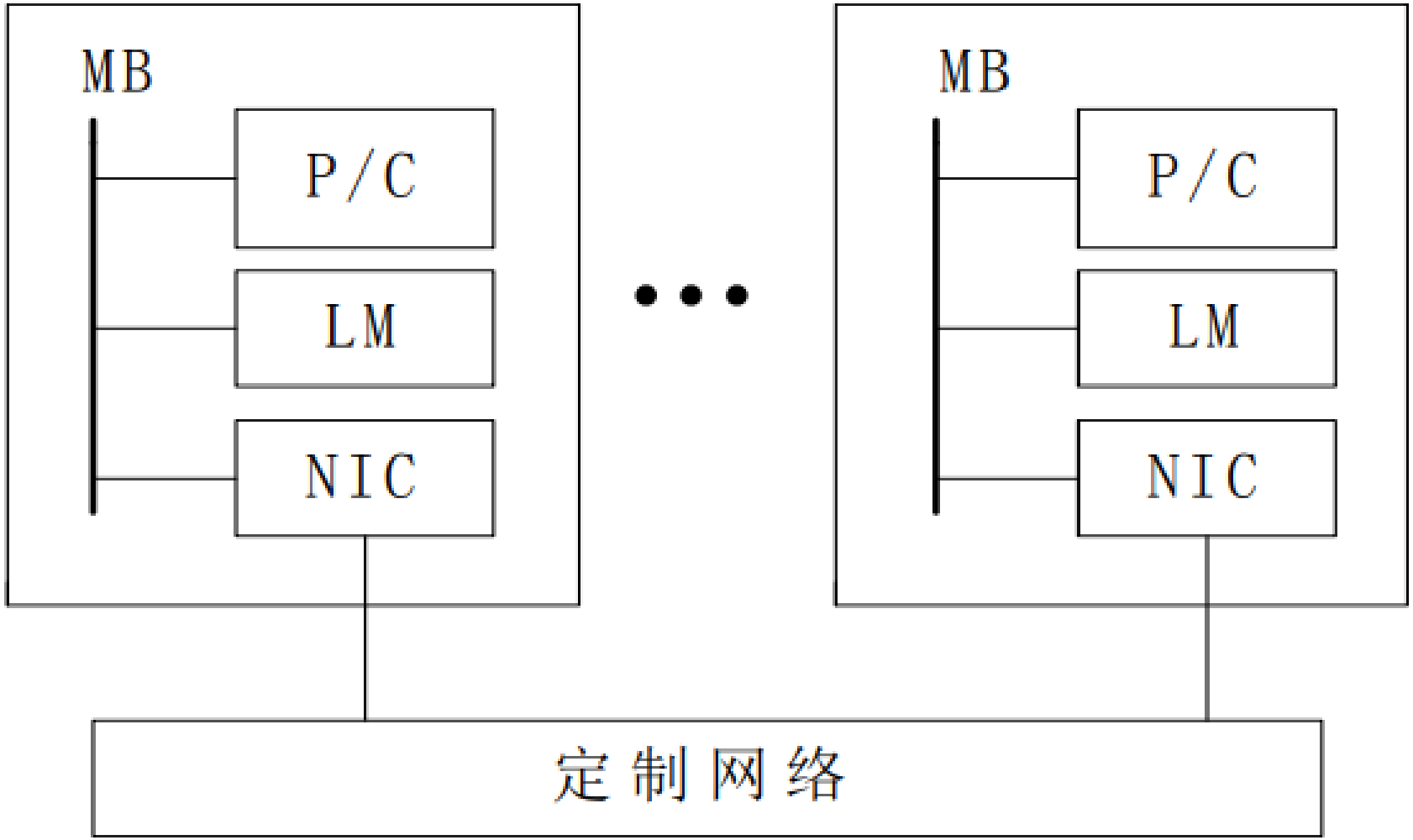
- **Intel ASCI-Red**
  - Developed under the **Accelerated Strategic Computing Initiative (ASCI)** of the DoE and NNSA (National Nuclear Security Administration) to build nuclear weapon simulators following moratorium on **nuclear weapon testing**.
  - Used commodity components for low-cost
  - Designed to be very scalable
  - A **massively-parallel processing machine** **consisting of 38x32x2 CPUs** (Pentium II Xeons) with 4510 compute nodes, 1212 GB of distributed RAM and 12.5 TB of disk storage.
  - Used MIMD (multiple instruction, multiple data) paradigm
  - See http://www.sandia.gov/ASCI/Red/RedFacts.htm

Generic scalable multiprocessor architecture

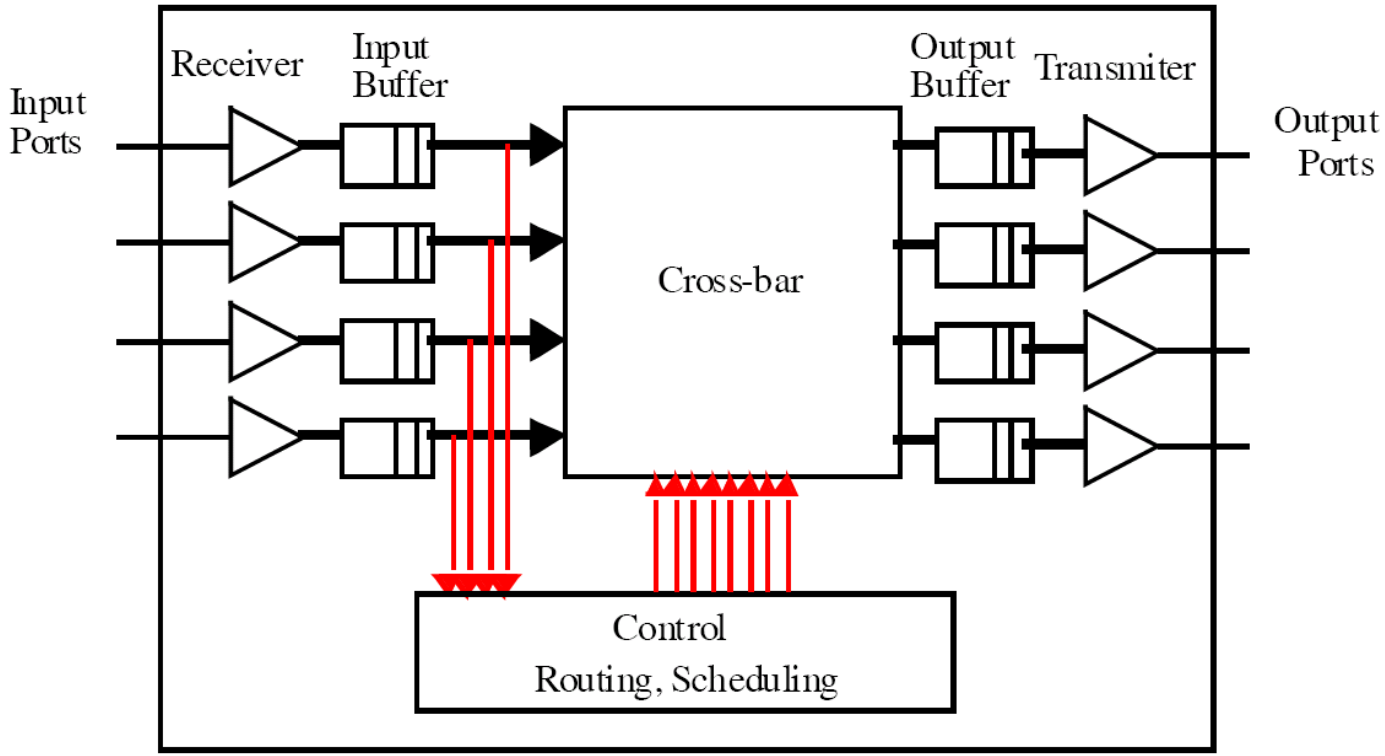


- On-chip interconnects (manycore processor)
- Off-chip interconnects (clusters of servers)
- Network characteristics: bandwidth and latency

MPP



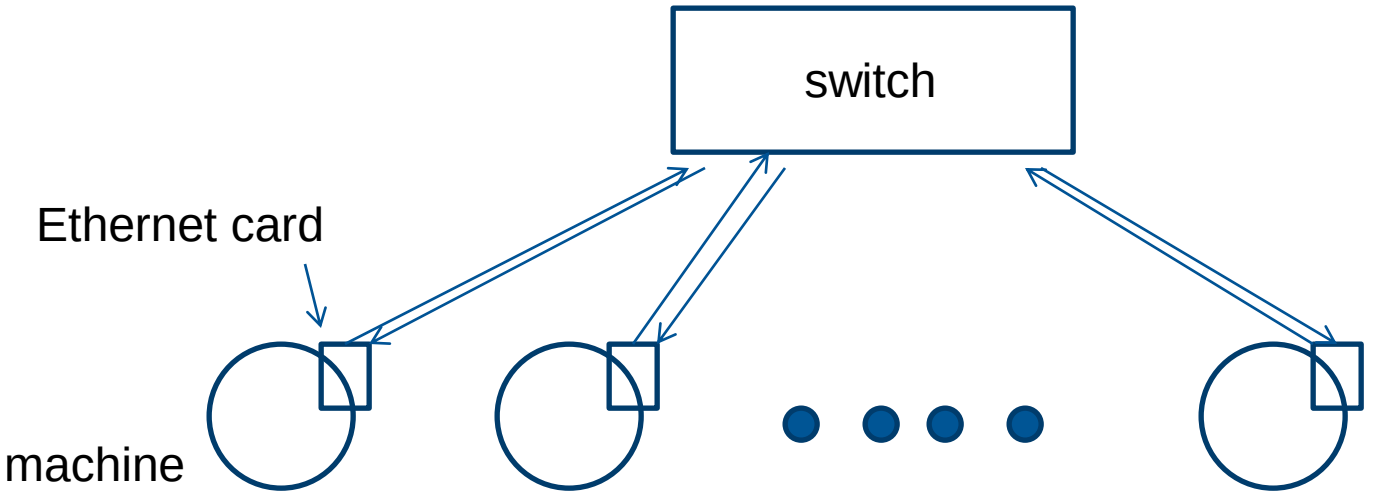
Switch



The cross-bar can realize a communication from any input port to any output port.

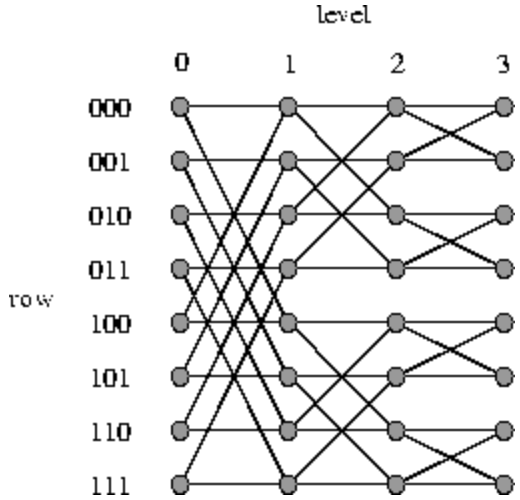
Switch example: 24-port 1Gbps Ethernet switch

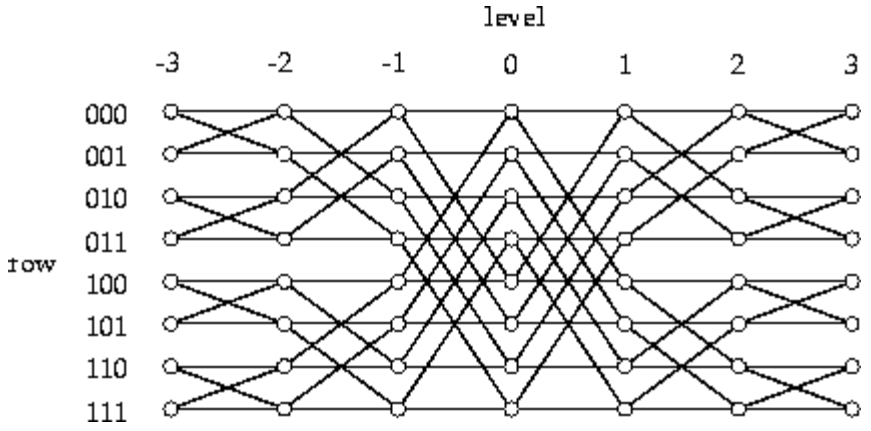
- **24 input ports and 24 output ports – each Ethernet jacket has one input port and one output port.**
  - All 24 machines can send and receive simultaneously.



Another alternative: multistage interconnection network

- **Realize all permutations without controlling O(N^2) cross-points.**
  - Clos networks, Benes networks

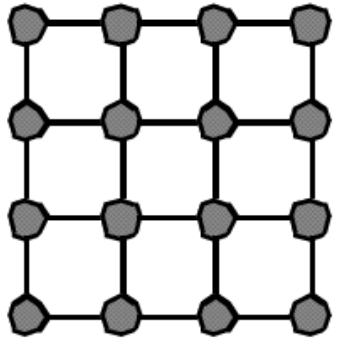




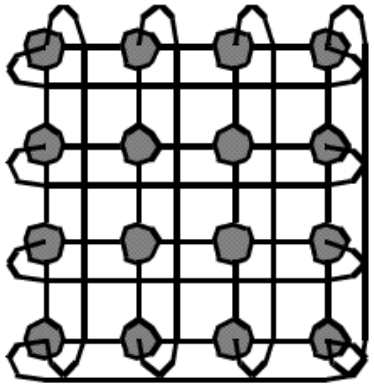
Topology

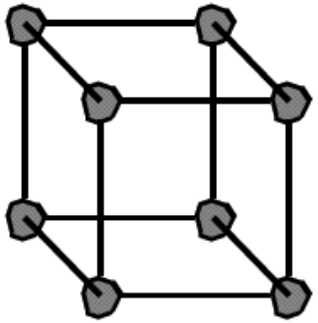
- **How the components are connected.**
- **Important properties**
  - Diameter: maximum distance between any two nodes in the network (hop count, or # of links).
  - Nodal degree: how many links connect to each node.
  - Bisection bandwidth: The smallest bandwidth between half of the nodes to another half of the nodes.
- **A good topology: small diameter, small nodal degree, large bisection bandwidth.**

Multidimensional Meshes and Tori



2D Grid

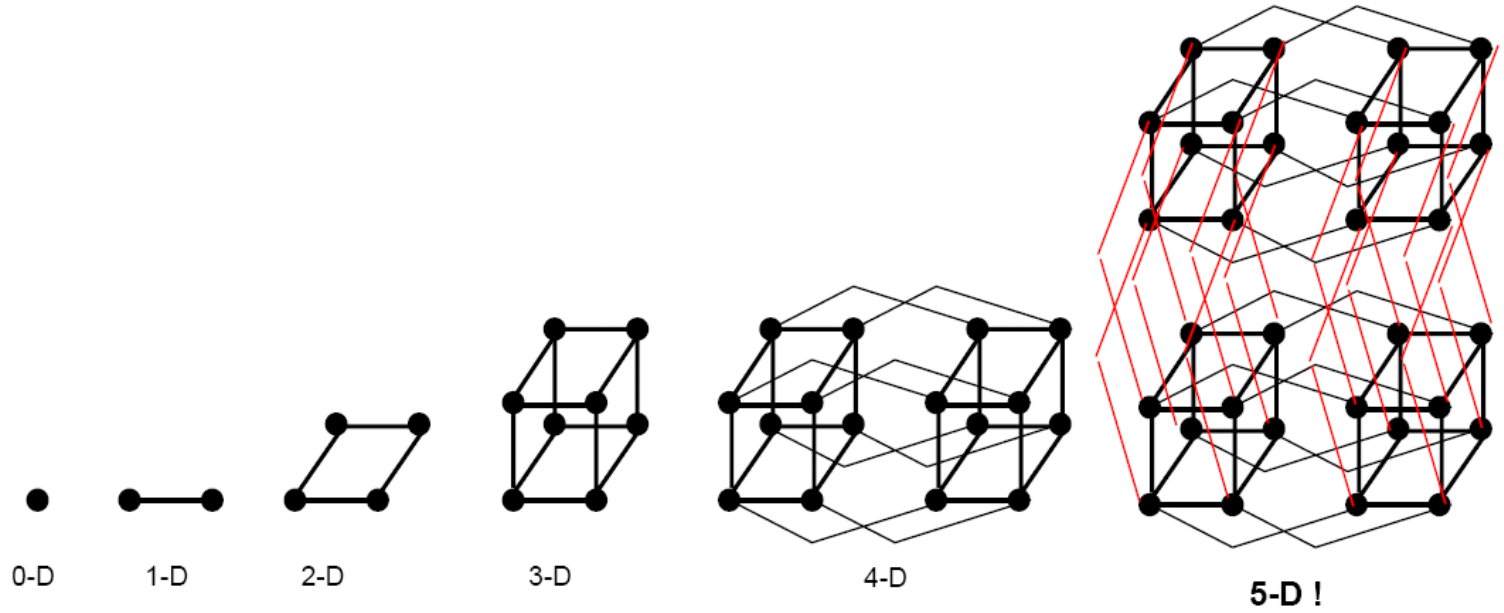




3D Cube

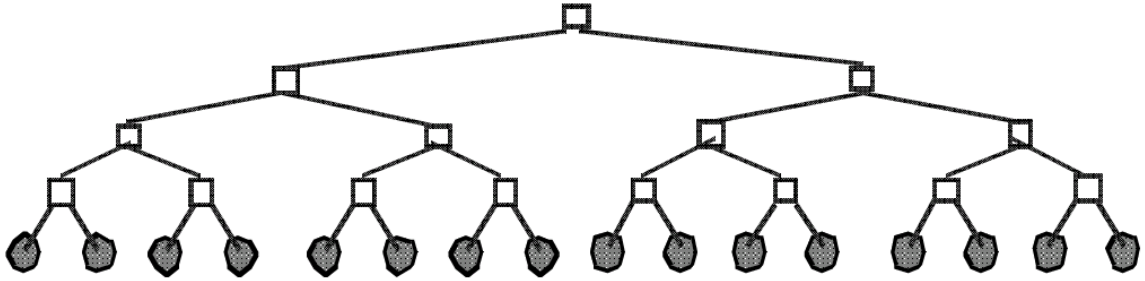
- **d-dimensional array/torus**
  - N = k_{d-1} x k_{d-2} x … x d_0
  - Each node is described by a d-vector of coordinate
  - Node (i_{d-1} x i_{d-2} x …x d_0) is connected to ???

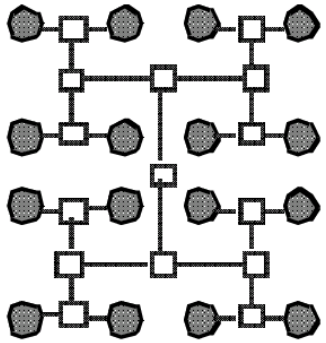
Hypercubes



- **Also call binary n-cubes. # of nodes = N = 2^n**
- **Each node is described by its binary representation.**
  - There is a link between two nodes whose binary representations differ by one bit.
- **Diameter=? Nodal degree = ? Bisection bandwidth = ?**

Trees





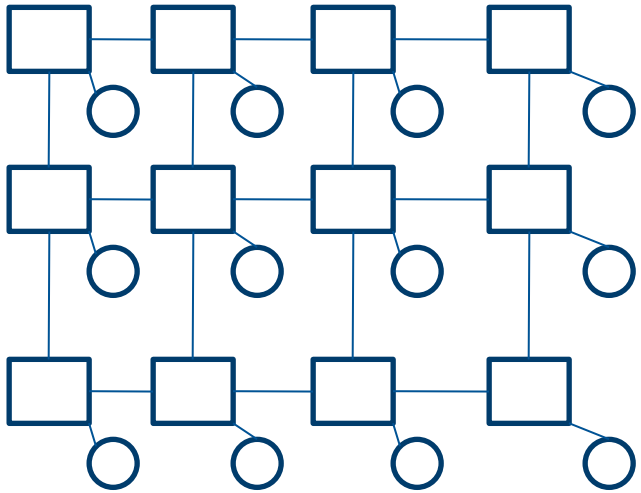
- Fixed degree, log(N) diameter, O(1) bisection bandwidth.
- Routing: up to the common ancestor than go down.

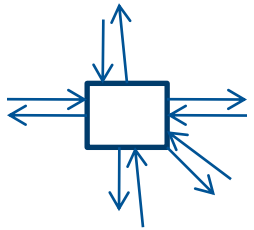
Irregular topology

- **Irregular topology does not any special mathmetic properties**
  - Can be expanded in any way.
  - No easy way for routing: routes need to be computed like in the Internet.
    - Routes can usually be determined in a regular network by using the coordinates of the source and destination.

Direct and indirect networks

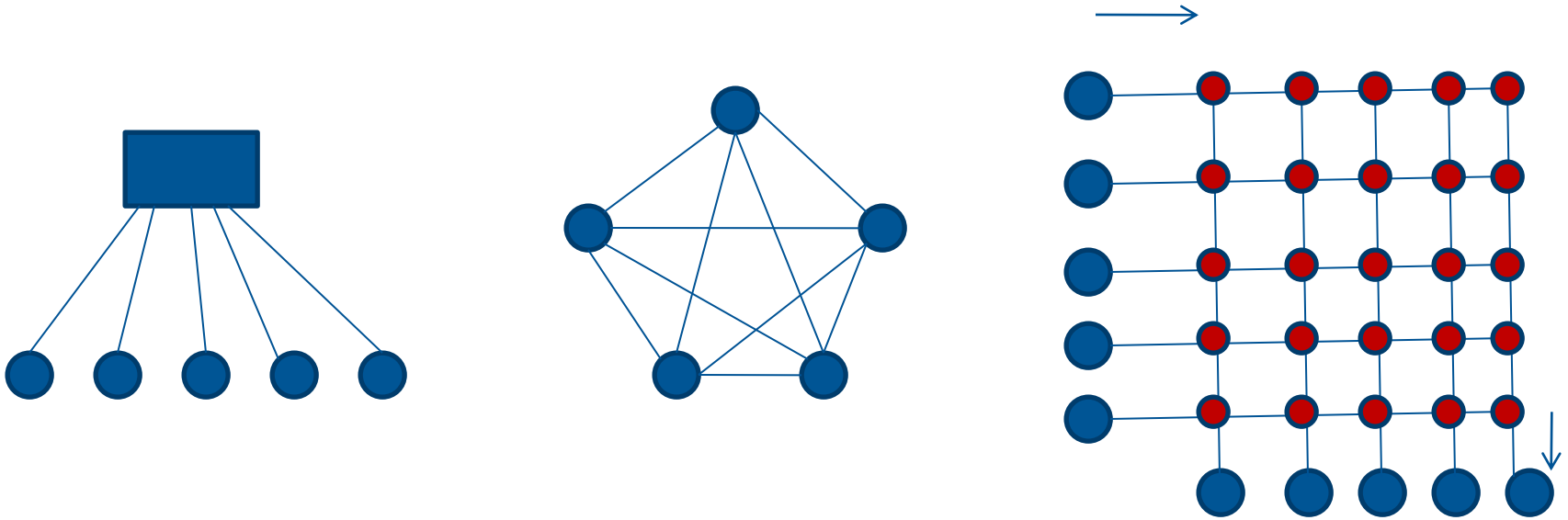
- **All the previously discussed networks are direct networks in that the compute nodes are directly attached to the nodes in the topology.**
  - An example mesh system.





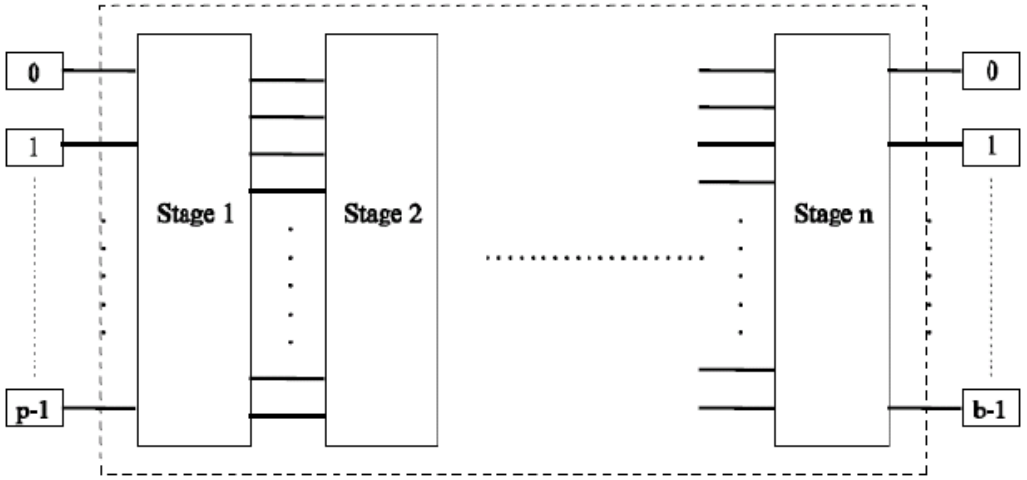
Each switch is a 5x5 switch

Fully connected network



- **Different organizations:**
  - Connected by one switch (crossbar switch), connecting all nodes, connected with a crossbar.
- **All permutation communication (each node sends one message and receives one message) can be realized.**

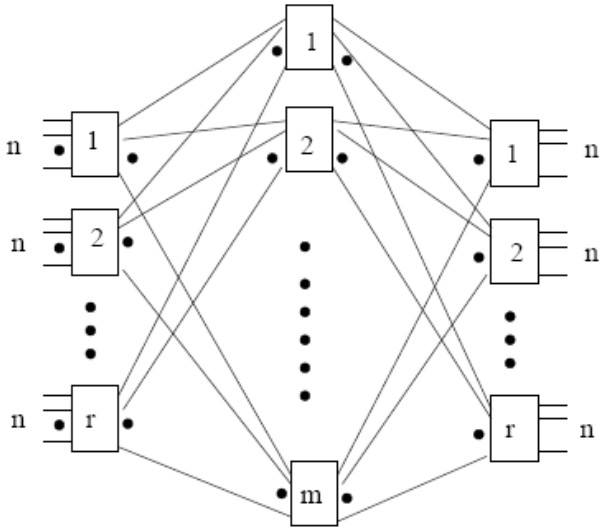
Multistage network



- **Try to emulate the cross-bar connection.**
  - Realizing permutation without blocking
  - Using smaller cross-bar(2x2, 4x4) switches as the building block. Usually O(Nlg(N)) switches (lg(N) stages.

Clos Network

- **Three stages: ingress stage, middle stage, and egress stage**
  - Ingress/egress stage has r n X m switches
  - Middle stage has m r X r switches
  - Each switch at ingress/egress stage connects to all m middle switches (one port to each switch).



(a) Clos(n, m, r)

Physical constraint on topologies

- **Number of dimensions.**
  - 2 or 3 dimensions
    - Can be layout physically
    - Short wires, easy to build
    - Many hops, low bisection bandwidth
  - >=4 dimensions
    - Harder to build, longer wires
    - Fewer hops, better bisection bandwidth
  - K-ary n-cubes provide a good framework for comparison.

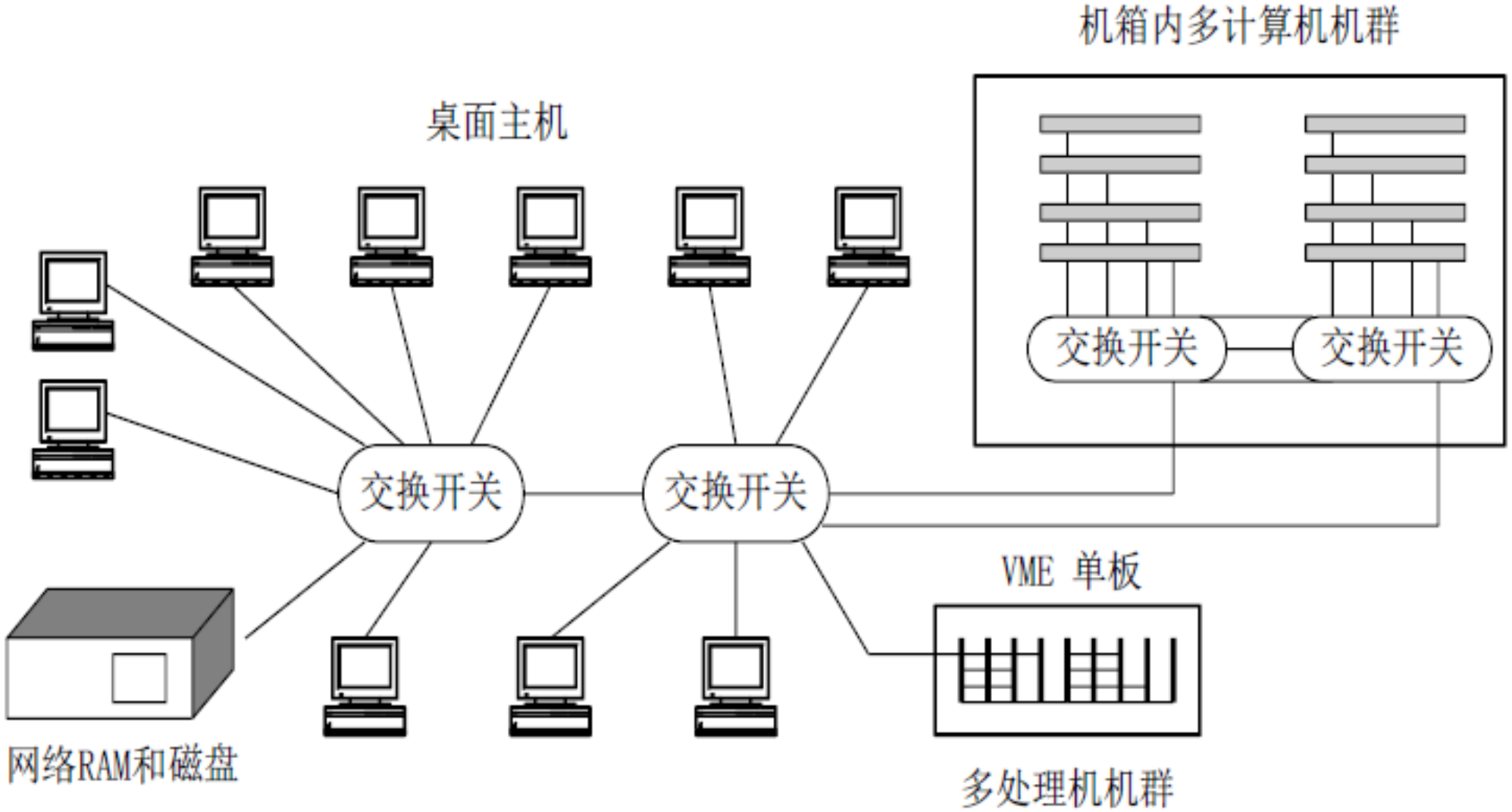
快速以太网

已经历了4代：

- 第一代，1982年引入的10Mbps
- 第二代，1994年宣布的100Mbps
- 第三代，1997年IEEE 802.3工作组宣布的1Gbps
- 第四代，2002年IEEE 802.3ae通过的10Gbps，并且2010年6月IEEE802.3ba公布了40-100Gbps

- Myrinet:
  - Myrinet是由Myricom公司设计的千兆位包交换网络，其目的是为了构筑计算机机群，使系统互连成为一种商业产品。
  - Myrinet是基于加州理工学院开发的多计算机和VLSI技术以及在南加州大学开发的ATOMIC/LAN技术。Myrinet能假设任意拓扑结构，不必限定为开关网孔或任何规则的结构。
  - Myrinet在数据链路层具有可变长的包格式，对每条链路施行流控制和错误控制，并使用切通选路法以及定制的可编程的主机接口。在物理层上，Myrinet网使用全双工SAN链路，最长可达3米，峰值速率为（1.28 + 1.28）Gbps（目前有2.56+2.56)
  - Myrinet交换开关 :8,12,16端口
  - Myrinet主机接口 : 32位的称作LANai芯片的用户定制的VLSI处理器，它带有Myrinet接口、包接口、DMA引擎和快速静态随机存取存储器SRAM。
  - 140 of the November 2002 TOP500 use Myrinet, including 15 of the top 100

LAN/Cluster connected by Myrinet

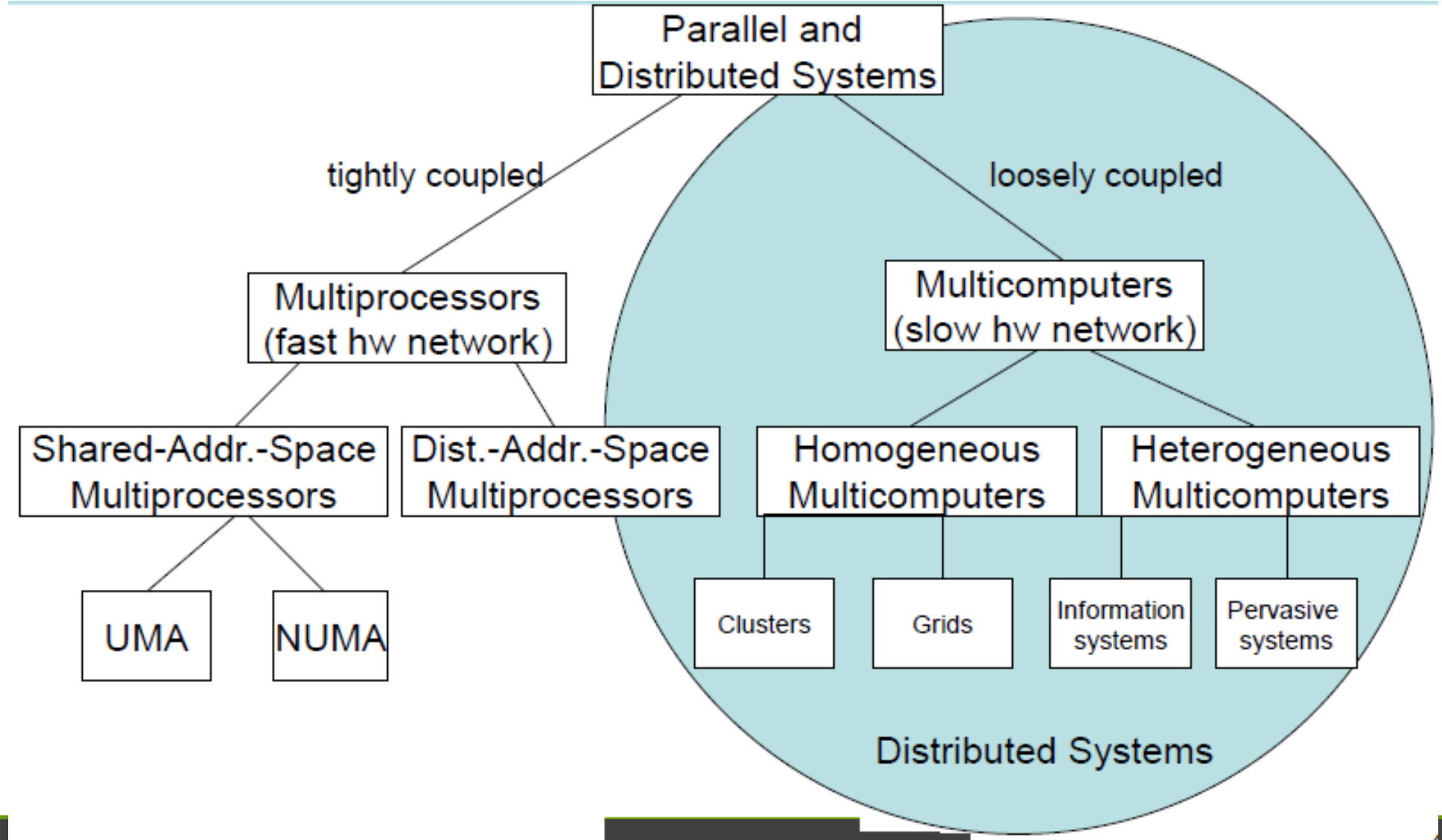


Intel ASCI-Red

ASCI Red was the last supercomputer designed and assembled solely by Intel; Intel's Supercomputer Division had already been closed down when ASCI Red was launched.

Intel ASCI-Red

Parallel or Distributed



Then Cluster based on Networking is popular

A Brief History of Clusters

- 1957 – SAGE by IBM & MIT-LL for Airforce NORAD
- 1976 -- Ethernet
- 1984 – Cluster of 160 Apollo workstations by NSA
- 1985 – M31 Andromeda by DEC, 32 VAX 11/750
- 1986 – Production Condor cluster operational
- 1990 – PVM released
- 1993 – First NOW workstation cluster at UC Berkeley
- 1993 – Myrinet introduced
- 1994 – First Beowulf PC cluster at NASA Goddard
- 1994 – MPI standard
- 1996 – >1Gflops
- 1997 – Gordon Bell Prize for Price-Performance
- 1997 – Berkeley NOW first cluster on Top-500
- 1997 -- >10 Gflops
- 1998 – Avalon by LANL on Top500 list
- 1999 -- >100 Gflops
- 2000 – Compaq and PSC awarded 5 Tflops by NSF

1957 SAGE by IBM & MIT-LL for Airforce NORAD

- The **Semi-Automatic Ground Environment** (**SAGE**) was a system of large computers and associated networking equipment that coordinated data from many radar sites and processed it to produce a single unified image of the airspace over a wide area.
- The processing power behind SAGE was supplied by the largest computer ever built, the AN/FSQ-7. Each SAGE Direction Center (DC) housed an **FSQ-7** which occupied an entire floor, approximately 22,000 square feet not including supporting equipment.
- Connecting the various sites was an enormous network of telephones, modems and teleprinters.

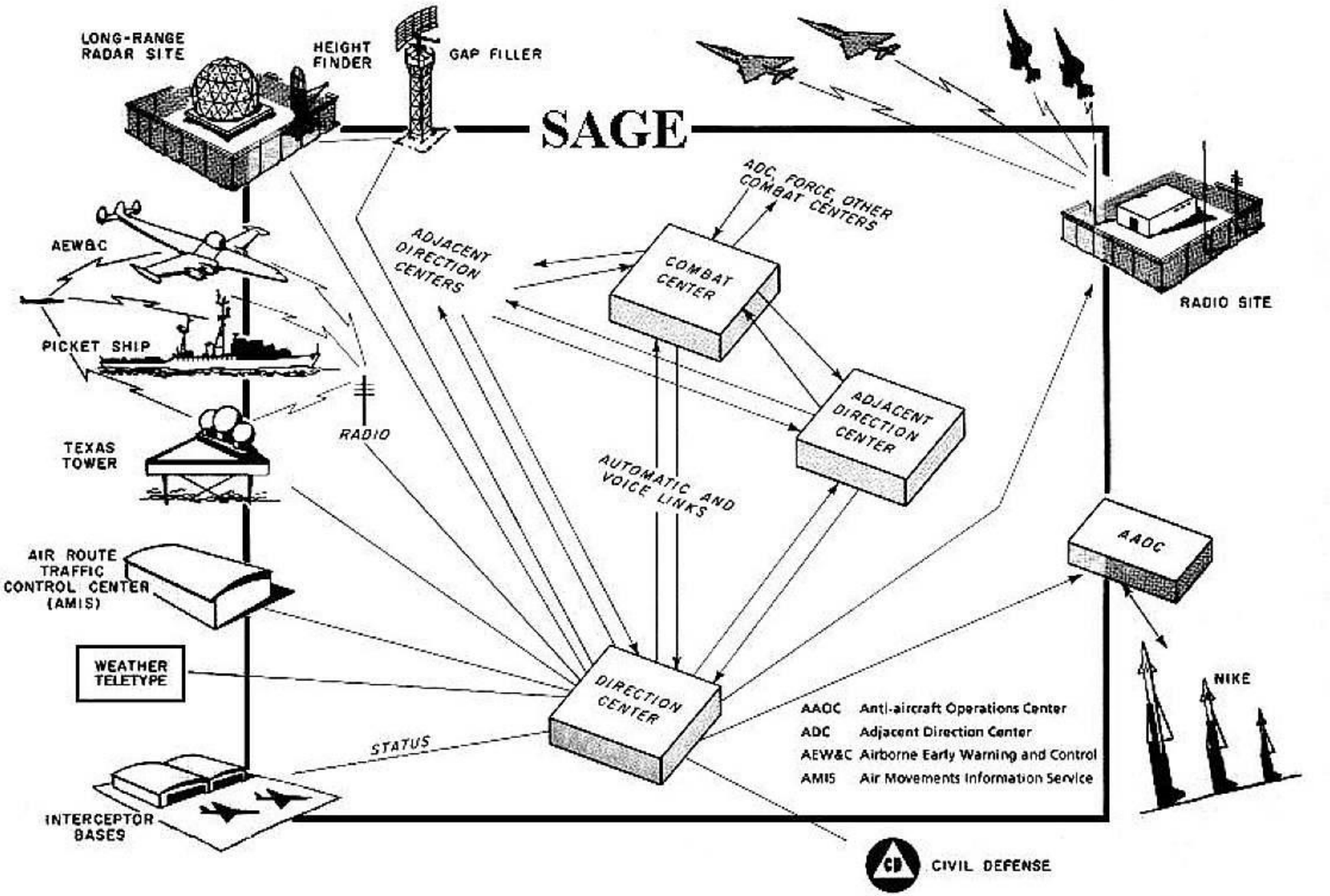
https://
7_Com

https://en.wikipedia.org/wiki/
Automatic_Ground_Environ

IBM was involved in the development of the Semi-Automatic Ground Environment (SAGE) early warning system during the Cold War. This was described in more detail in Chap. 4. IBM provided the hardware for the air defence system. The initial installation was completed in 1958, and the system was fully implemented in 1963. It remained operational until 1984.

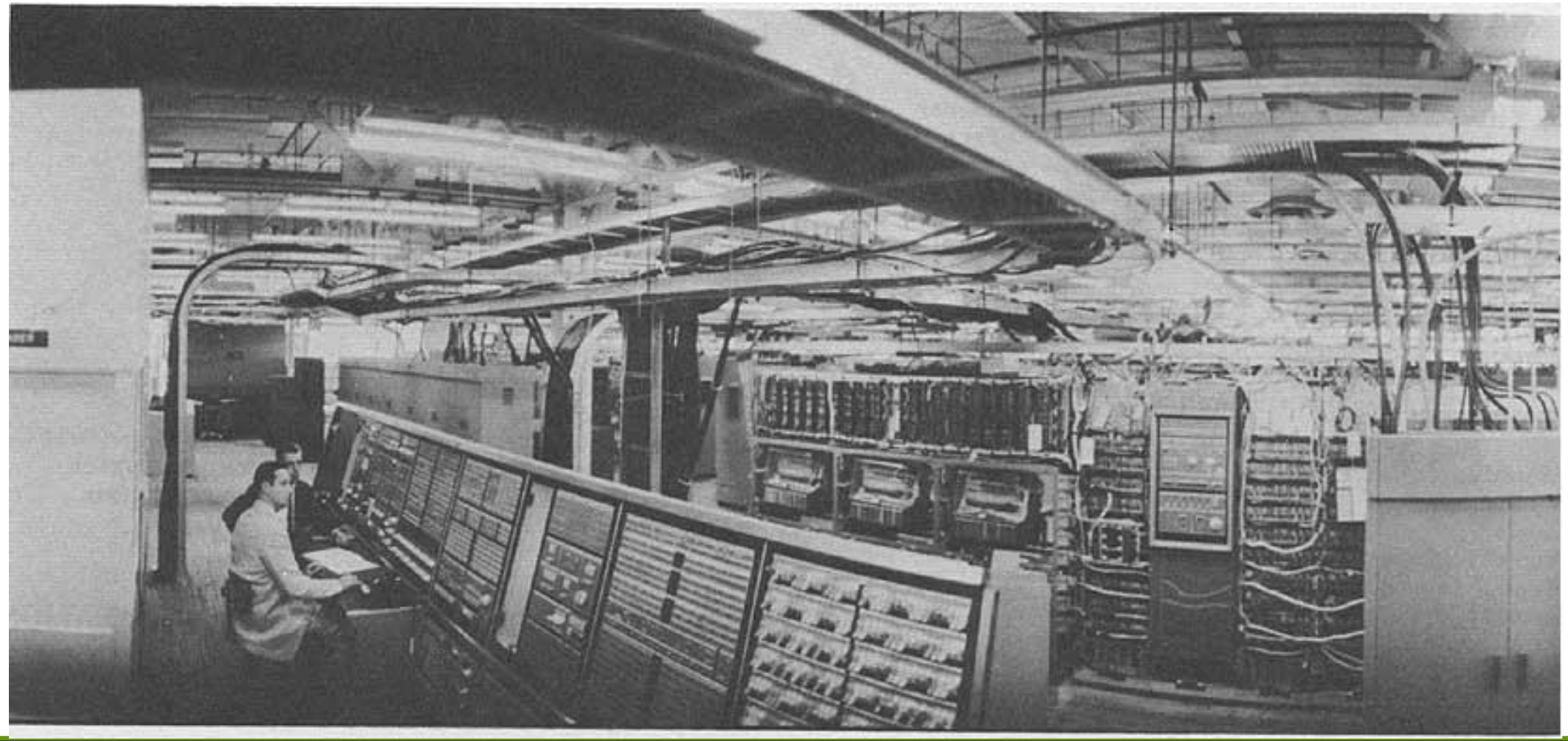
There were 24 SAGE direction centres and 3 SAGE combat centres located in the United Sates. Each centre was linked by long-distance telephone lines, and Burroughs provided the communications equipment that allowed the centres to communicate with one another. It was one of the earliest computer networks. Each centre contained a large digital computer that automated the information flow and provided real-time control information on aircraft and on weapons systems. It tracked and identified aircraft and presented the electronic information to operators on a display device (cathode ray tube).

http://ed-thelen.org/comp-hist/vs-ibm-sage.html



http://ed-thelen.org/comp-hist/vs-ibm-sage.html

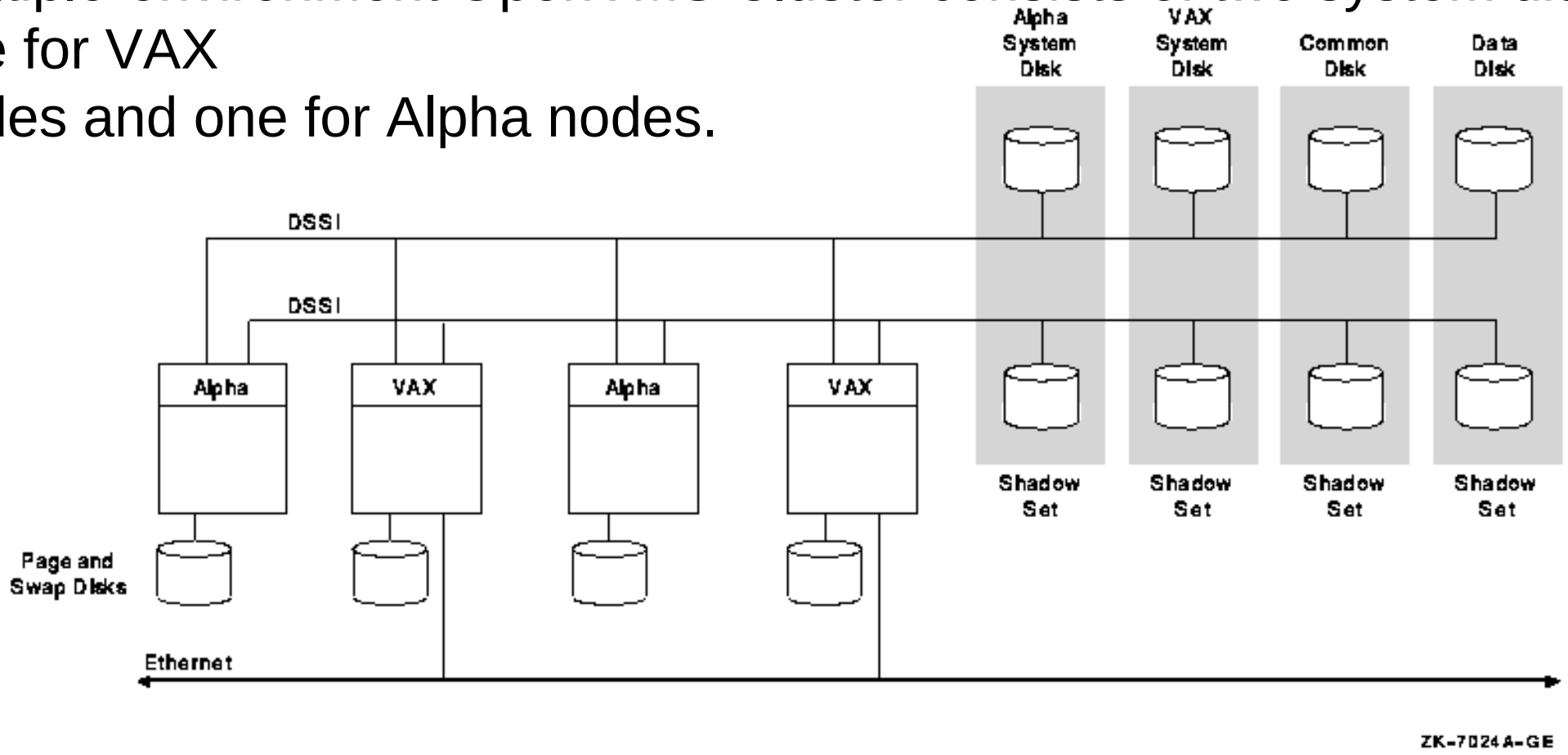
- **View of machine room, and maintenance console from (Ballistic Research Lab Report) (1961)**



Cluster Computer History – 1980's

- **Increased interest in cluster computing**
  - Ex: NSA connected 160 Apollo workstations in a cluster configuration
- **First widely used clustering product: VAXcluster**
- **Development of task scheduling software**
  - Condor package developed by UW-Madison
- **Development of parallel programming software**
  - **PVM** (Parallel Virtual Machine)

- multiple-environment OpenVMS Cluster consists of two system disks:
  one for VAX
  nodes and one for Alpha nodes.



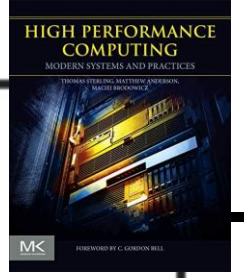
https://www.itec.suny.edu/scsys/vms/ovmsdoc073/v73/6318/6318pro_018.html

NASA Beowulf Project – 1994

- Wiglaf - 1994
- 16 Intel 80486 100 MHz
- VESA Local bus
- 256 Mbytes memory
- 6.4 Gbytes of disk
- Dual 10 base-T Ethernet
- 72 Mflops sustained
- $40K

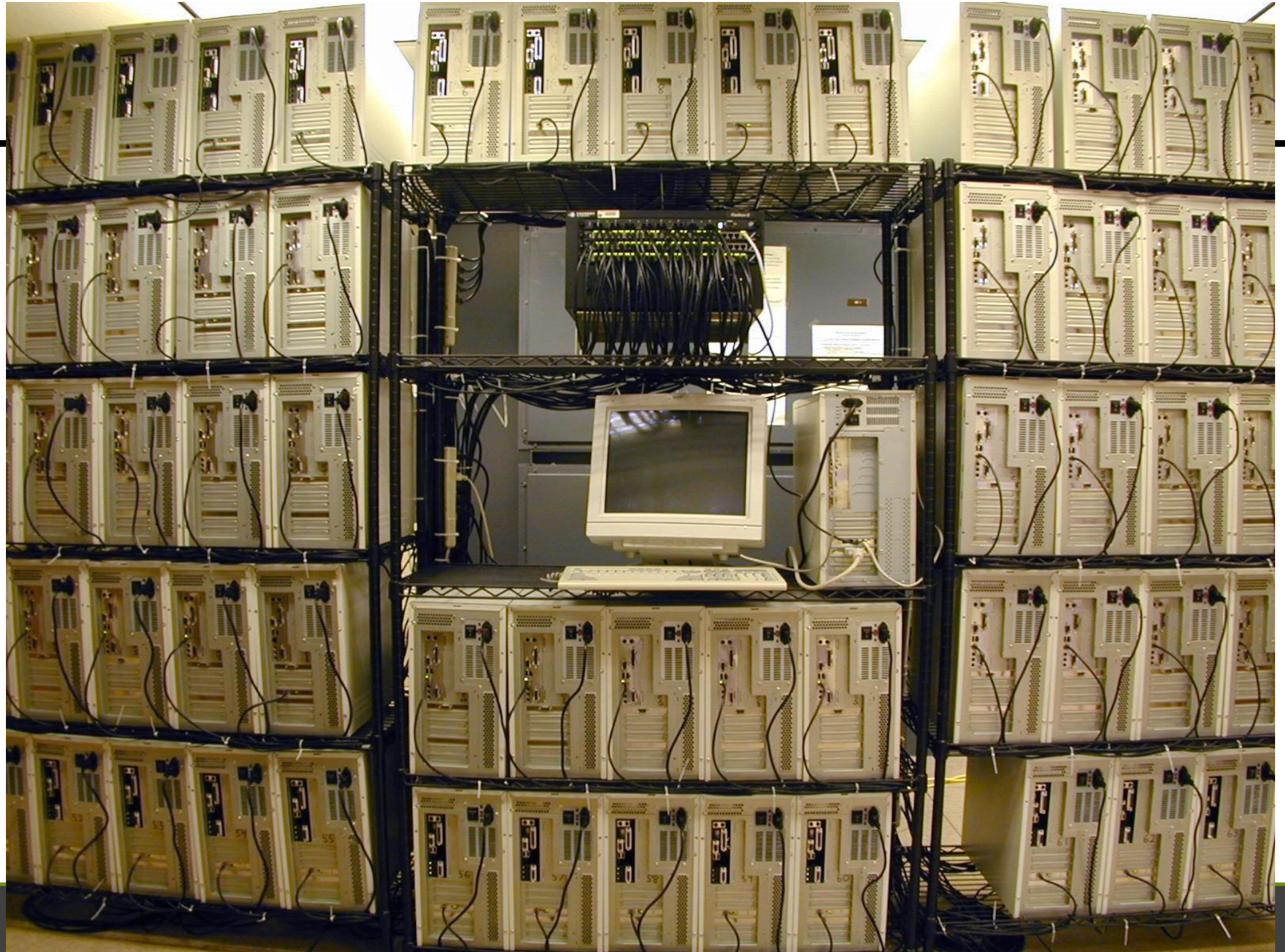
- Hrothgar - 1995
- 16 Intel Pentium100 MHz
- PCI
- 1 Gbyte memory
- 6.4 Gbytes of disk
- 100 base-T Fast Ethernet (hub)
- 240 Mflops sustained
- $46K

- Hyglac-1996 (Caltech)
- 16 Pentium Pro 200 MHz
- PCI
- 2 Gbytes memory
- 49.6 Gbytes of disk
- 100 base-T Fast Ethernet (switch)
- 1.25 Gflops sustained
- $50K



Thomas Sterling in front of a commodity cluster built as part of the Beowulf Project. Such commodity clusters are now frequently referred to as belonging to the Beowulf class of supercomputer.

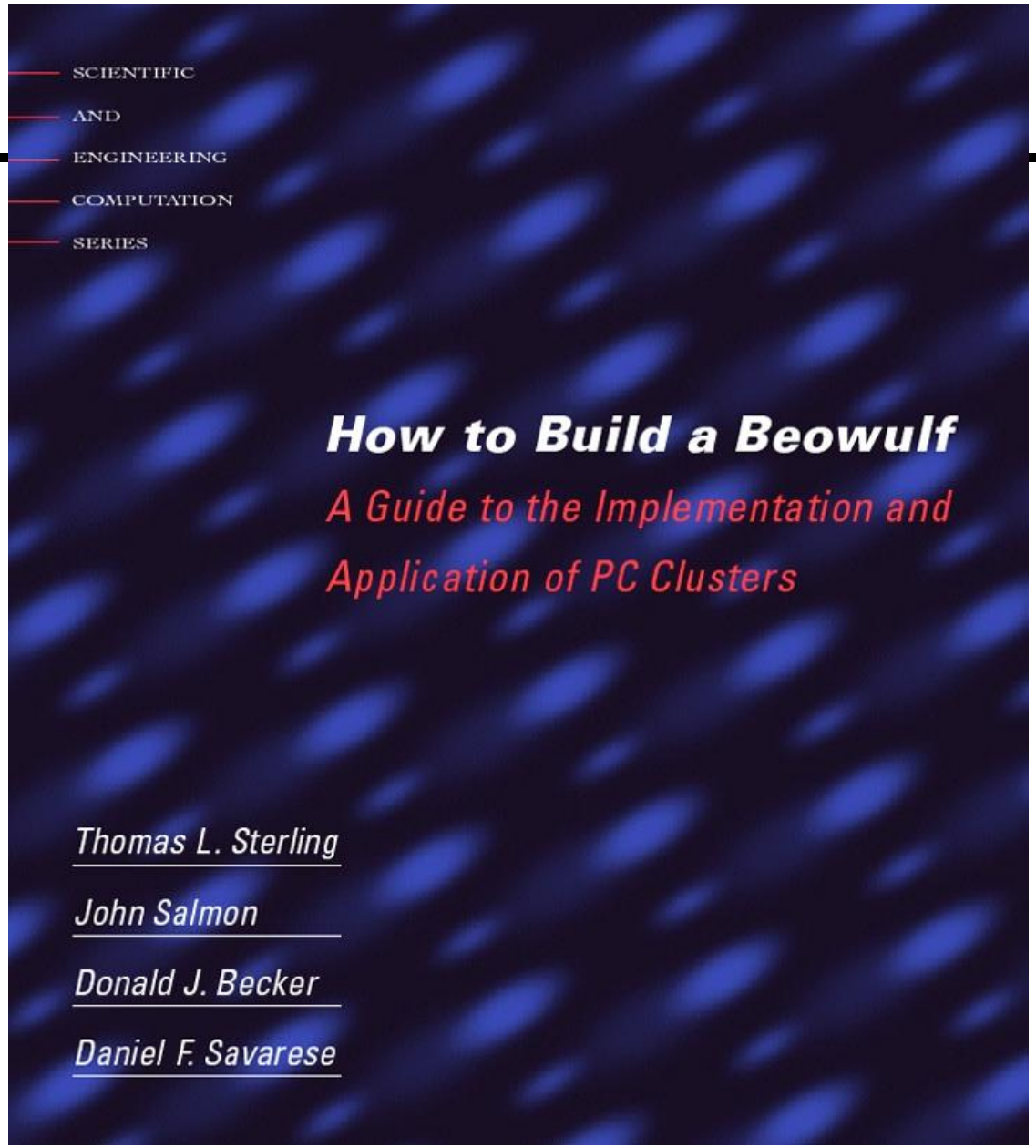
Thomas Sterling is widely known as the father of the Beowulf class of supercomputers. His pioneering work in 1994 along with Don Becker in creating a cluster comprised of commodity-grade computers, collectively referred to as a "Beowulf cluster", significantly reduced the cost of supercomputing and later resulted in the widespread adoption of commodity clusters for scientific computing. This effort resulted in Beowulf being awarded the 1997 Gordon Bell prize in the price–performance category. The Beowulf Project's adoption and software support of the Linux operating system also contributed to the widespread adoption of this operating system in supercomputing systems worldwide. Apart from being the "father of Beowulf", Thomas Sterling's contributions to the hybrid technology multithreaded architecture based on superconducting logic continue to have impacts on high-end computer system architecture design. Thomas Sterling is the recipient of the American Association for the Advancement of Science and HPC Vanguard Awards.



1st printing: May, 1999

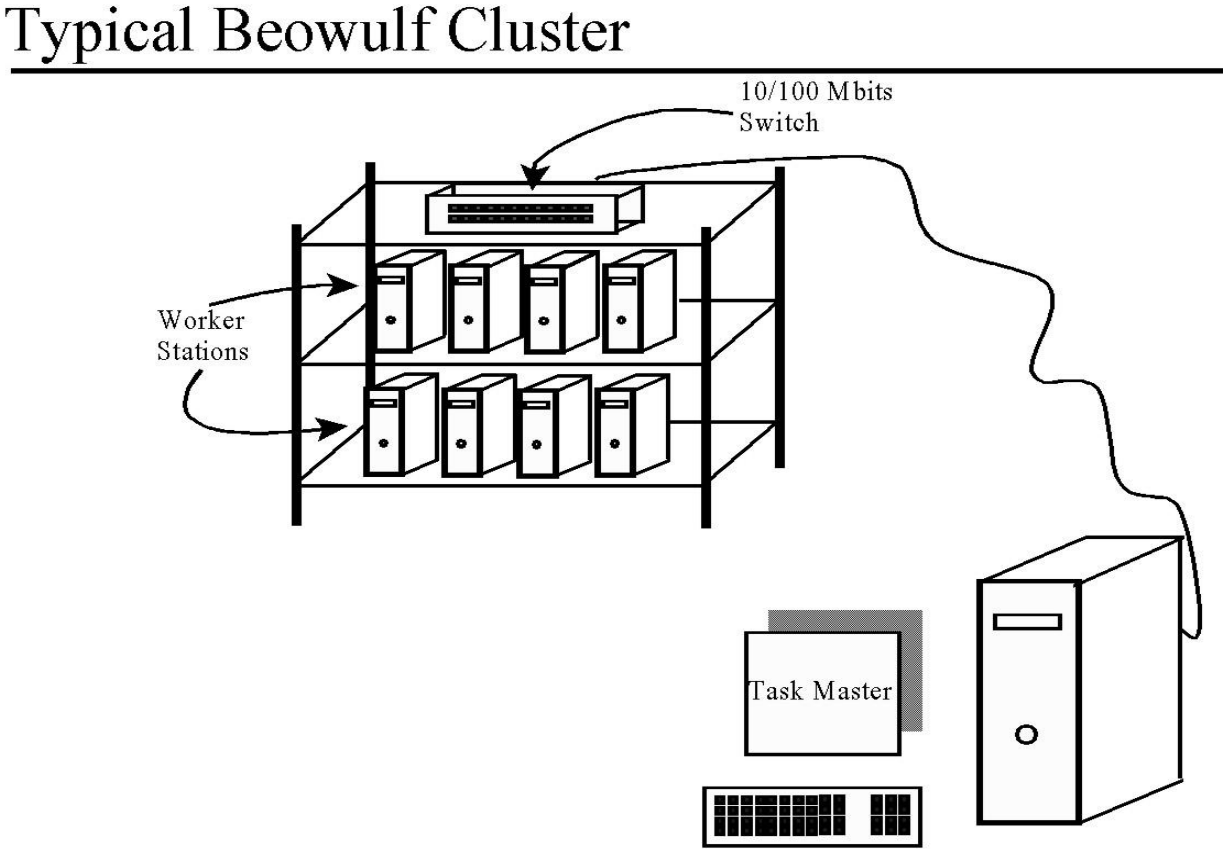
2nd printing: Aug. 1999

MIT Press



Beowulf Cluster Architecture

- **Master-Slave configuration**
- **Master Node**
  - Job scheduling
  - System monitoring
  - Resource management
- **Slave Node**
  - Does assigned work
  - Communicates with other slave nodes
  - Sends results to master node

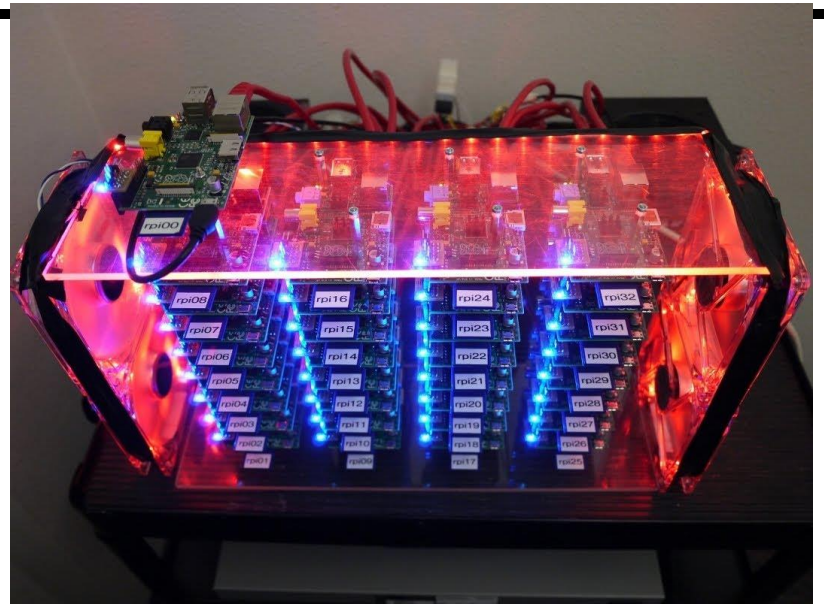


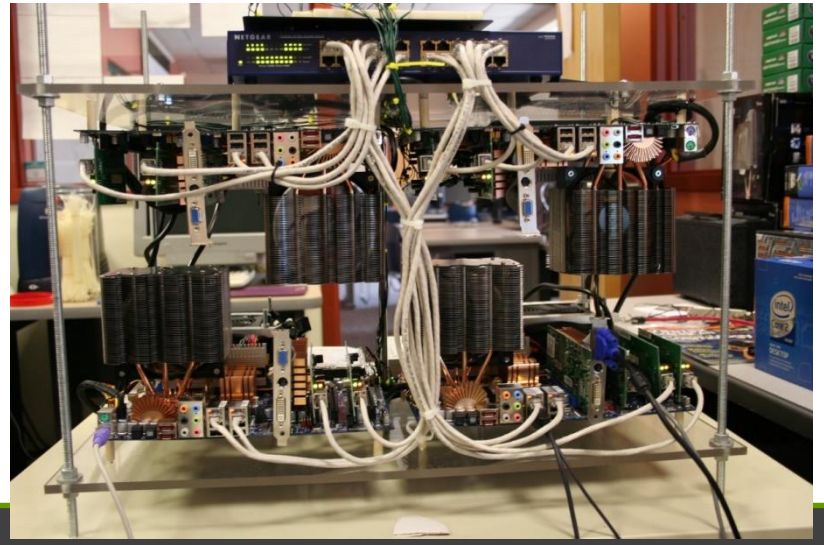
Node Hardware

- Typically desktop PC's
- Can consist of other types of computers i.e.
  - Rack-mount servers
  - Case-less motherboards
  - PS3's
  - RaspberryPi boards

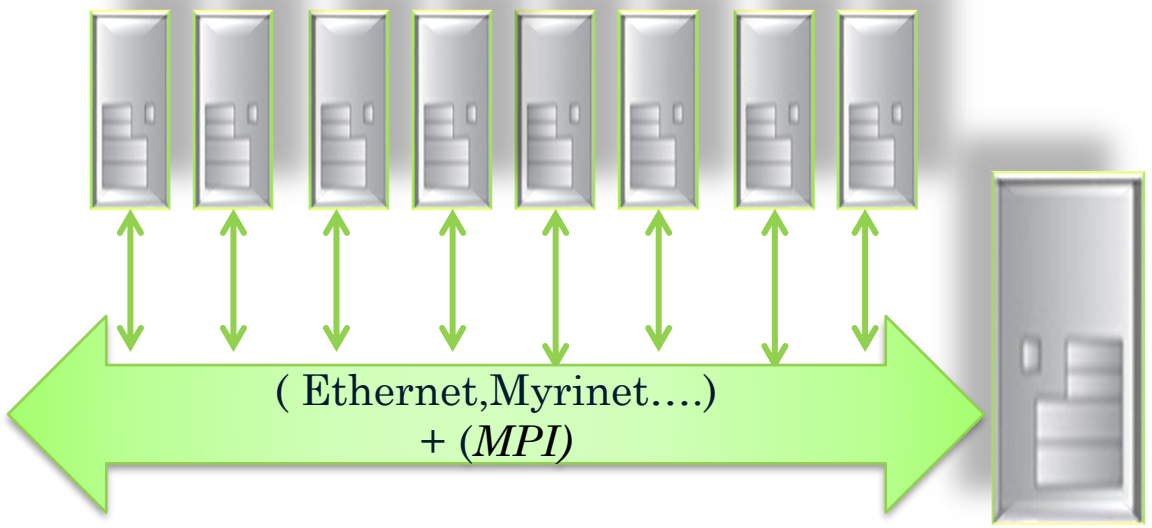
Node Software

- Operating System
- Resource Manager
- Message Passing Software





Beowulf cluster



- **Master**: ***or service node*** **or** ***front node*** **(used to interact with users and manage the cluster)**
- **Nodes : a group of computers (computing node s)(keyboard, mouse, floppy, video…)**
- **Communications between nodes on an interconnect network platform (Ethernet, Myrinet….)**
- **In order for the master and node computers to communicate, some sort message passing control structure is required. MPI (Message Passing Interface) is the most commonly used such control.**

Linux cluster computing
Hong Kong Baptist University (BUHK)

- PII 4-node clusters started in 1999

Linux cluster computing
Hong Kong Baptist University (BUHK)

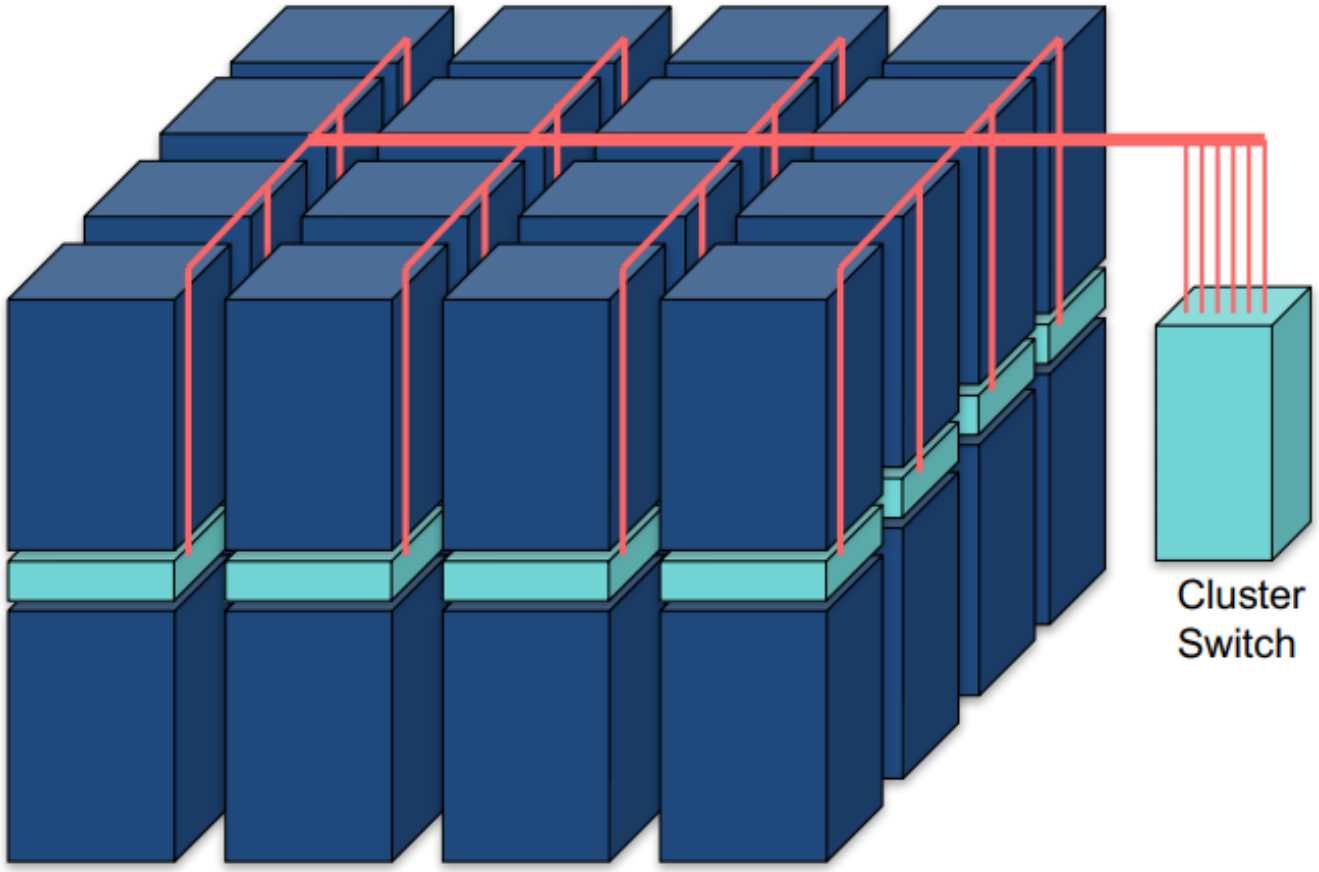
- PIII 16 node cluster purchased in 2001.

Plan for grid
For test base

Linux cluster computing
Hong Kong Baptist University (BUHK)

- 64-nodes P4-Xeon cluster at #300 of top500

https://www.cs.rutgers.edu/~pxk/417/notes/content/22-clusters-slides.pdf



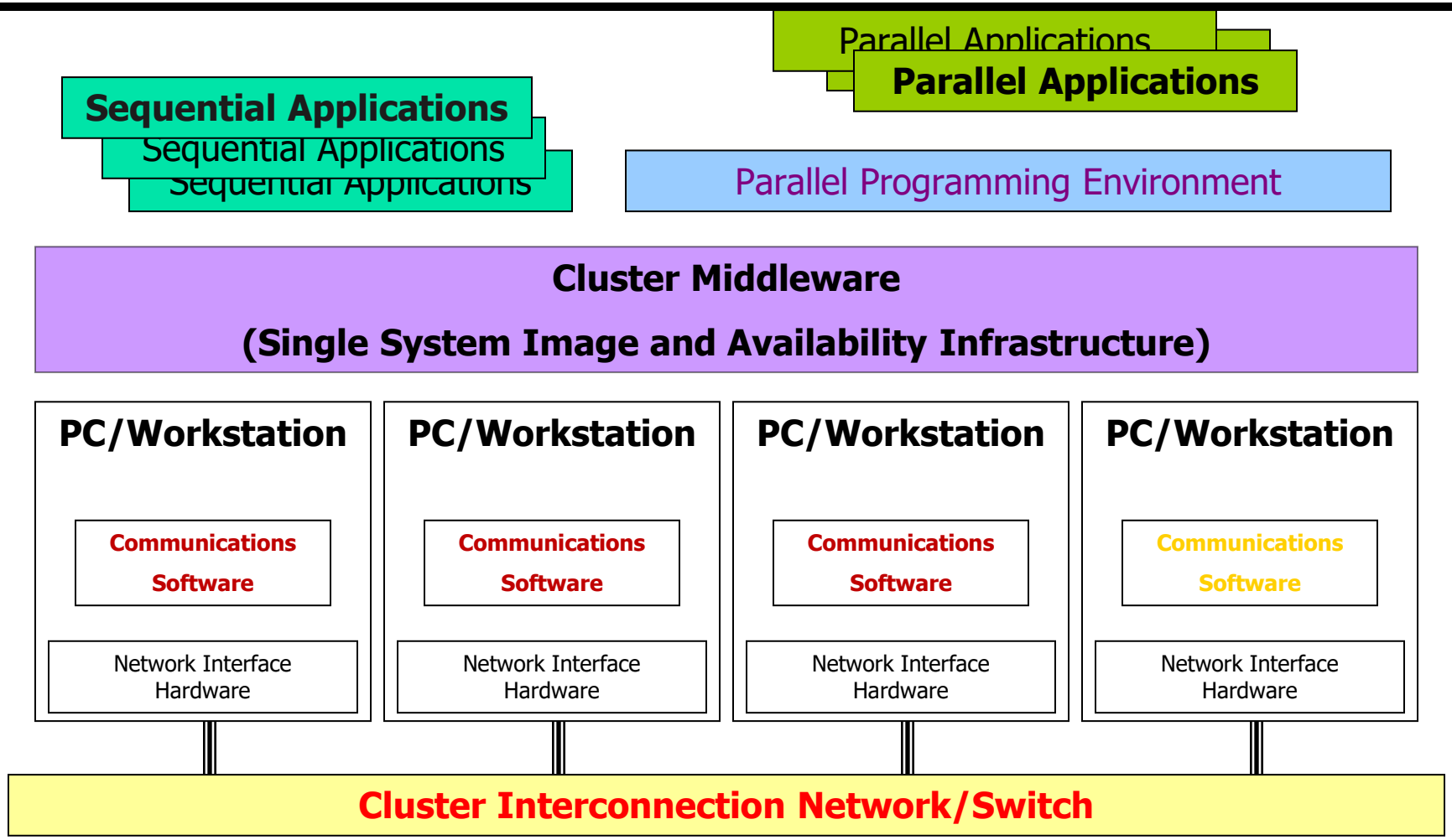
Cluster of 4×4 racks

Assume:
- 10 Gbps per server
- 40 servers per rack
- ⇒ 400 Gbps/rack

16 racks
- ⇒ 8 Tbps

Max switch capacity currently ~ 5 Tbps
- ⇒ Need at least two cluster switches

Cluster Architecture



UC-Berkeley NOW (Network of workstations)

- **NOW-1 1995**
  - 32-40 SparcStation 10s and 20s
  - originally ATM
  - first large myrinet network

- **NOW-2 1997**
  - 100+ Ultra Sparc 170s
  - 128 MB, 2 2GB disks, ethernet, myrinet
  - largest Myrinet configuration in the world
  - First cluster on the TOP500 list

Backrub (Google) 1997

Google 2001

Commodity CPUs

Lots of disks

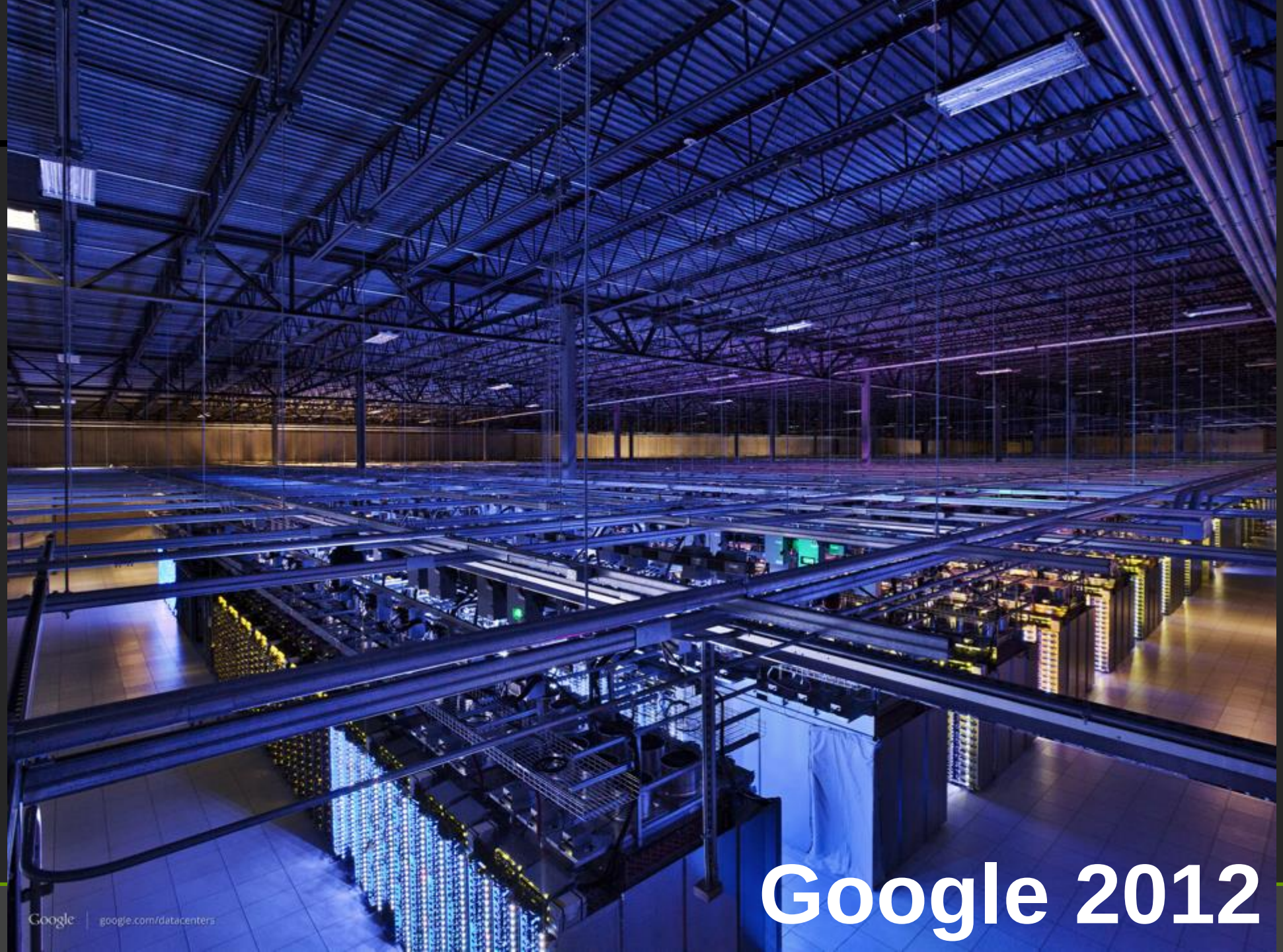
Low bandwidth network

Cheap !

Datacenter Evolution

Google data centers in The Dalles, Oregon

- 200K SF + 164K sf in (3 buildings total)
- $1.2 B investment in site
- 175 people employed on site
- 70 Megawatts (when it was 200K SF)



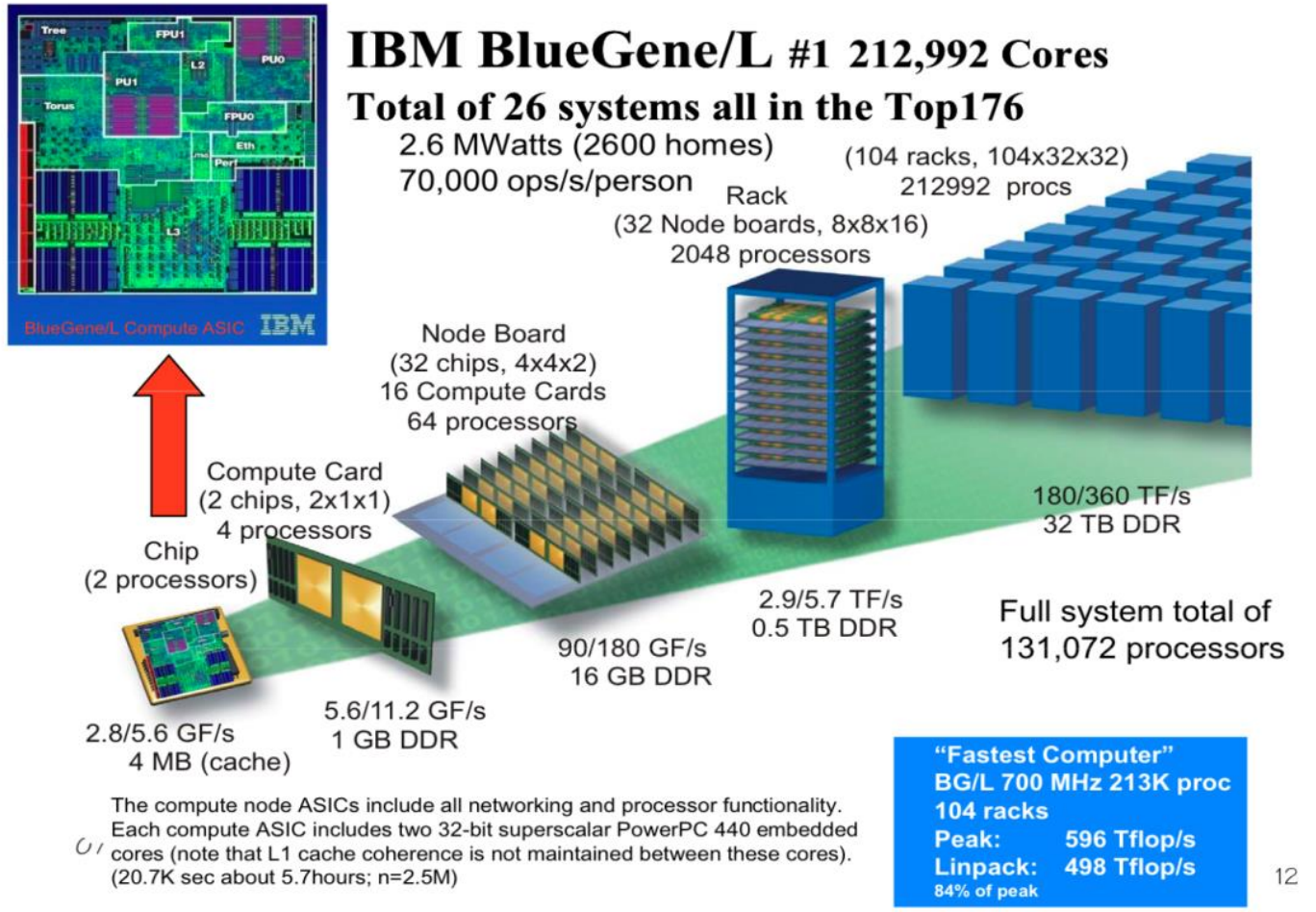
Google 2012

The Cluster Era – 2000

- **IBM ASCI-White**
  - A **cluster** computer based upon the commercial IBM RS/6000 SP computer
  - 512 machines, each containing 16 CPUs, in the cluster for a total of 6 TB of RAM and 160 TB of disk
  - Also had 28 node “Ice” and the 68 node “Frost”
  - Consumed 3 MW electricity to run and additional 3 MW to cool
  - Ran AIX (UNIX variant)
  - Cost $110 million
  - See https://www.llnl.gov/str/Seager.html

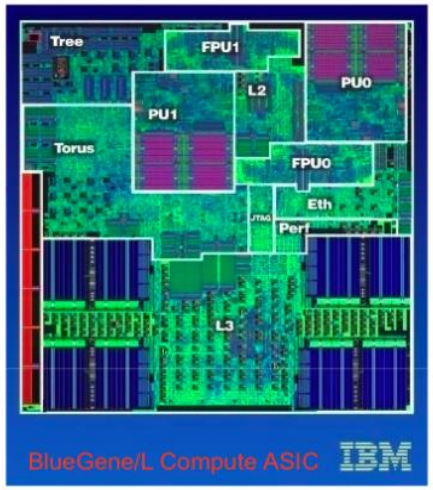
BlueGene/L

- IBM, 2004.
- First supercomputer ever to run over 100 TFLOPS sustained on a real world application, namely a three-dimensional molecular dynamics code (ddcMD).



IBM BlueGene/L #1 212,992 Cores

Total of 26 systems all in the Top176



2.6 MWatts (2600 homes)
70,000 ops/s/person

(104 racks, 104x32x32)
212992 procs

Rack
(32 Node boards, 8x8x16)
2048 processors

Node Board
(32 chips, 4x4x2)
16 Compute Cards
64 processors

Compute Card
(2 chips, 2x1x1)
4 processors

Chip
(2 processors)

180/360 TF/s
32 TB DDR

2.9/5.7 TF/s
0.5 TB DDR

90/180 GF/s
16 GB DDR

5.6/11.2 GF/s
1 GB DDR

2.8/5.6 GF/s
4 MB (cache)

Full system total of
131,072 processors

The compute node ASICs include all networking and processor functionality. Each compute ASIC includes two 32-bit superscalar PowerPC 440 embedded cores (note that L1 cache coherence is not maintained between these cores). (20.7K sec about 5.7hours; n=2.5M)

"Fastest Computer"
BG/L 700 MHz 213K proc
104 racks
Peak: 596 Tflop/s
Linpack: 498 Tflop/s
84% of peak

12

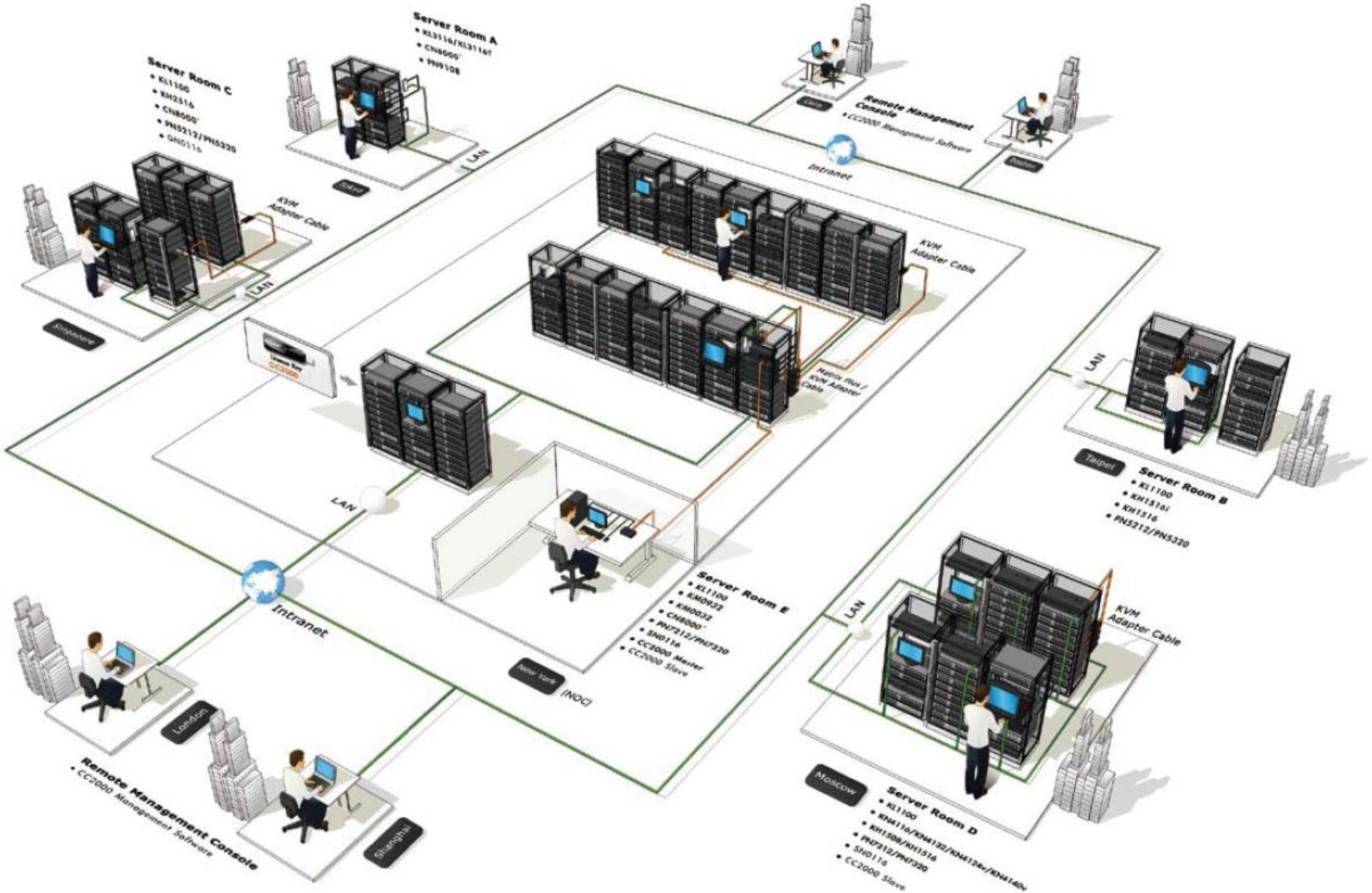
IBM Roadrunner - 2008

- **IBM Roadrunner**
  - Had 6,480 dual-core Opteron CPUs to handle O/S, interconnect, scheduling etc. and **12,960 PowerXCell 8i CPUS** one per Opteron core to handle computation
  - An Opteron cluster with Cell accelerators
  - TriBlade design 2 dual Opterons with 16 GB and 4 PowerXCell 8i also with 16 GB
  - 3 TriBlades per chassis; 180 TriBlades per Connected Unit; 18 Connected Units in total
  - A unique hybrid architecture that required all software to be specially written
  - A BIG challenge to program
  - Cost $133m and 2.35 MW to operate
  - See http://www.lanl.gov/roadrunner

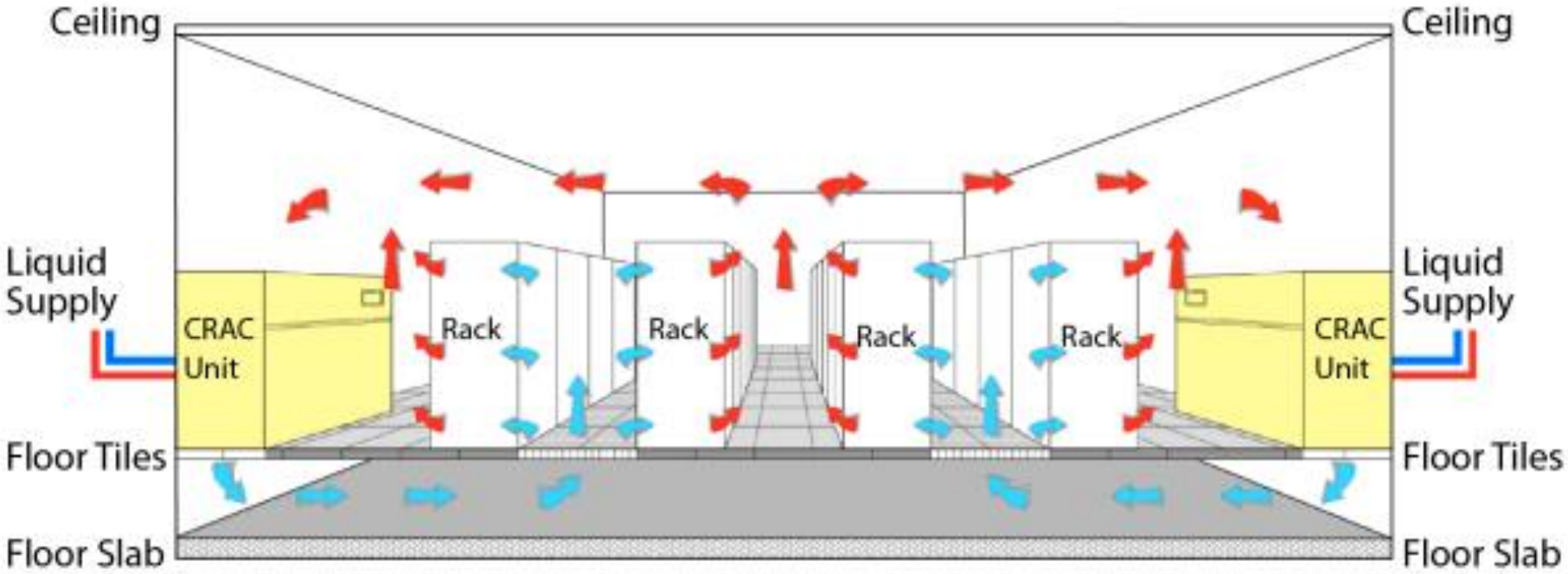
Cray Jaguar – 2009

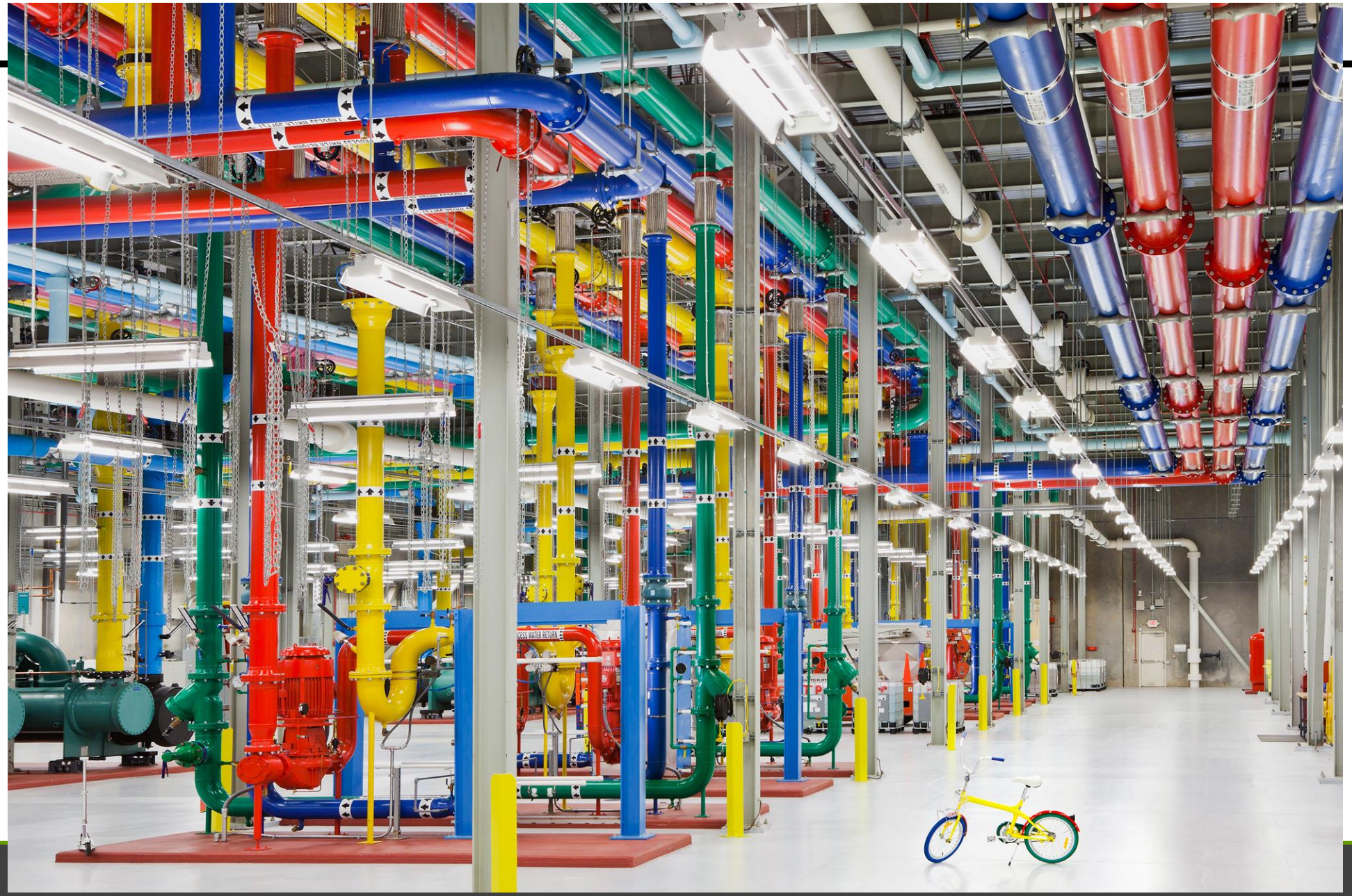
- $20m upgrade in 2009 from quad-core AMD based XT4 to Cray XT5 and AMD hex-core CPUs now got 224,256 cores!
- Requires 6.9MW to operate
- Another US Govt lab supercomputer
- See http://www.nccs.gov/computing-resources/jaguar for more details

Google

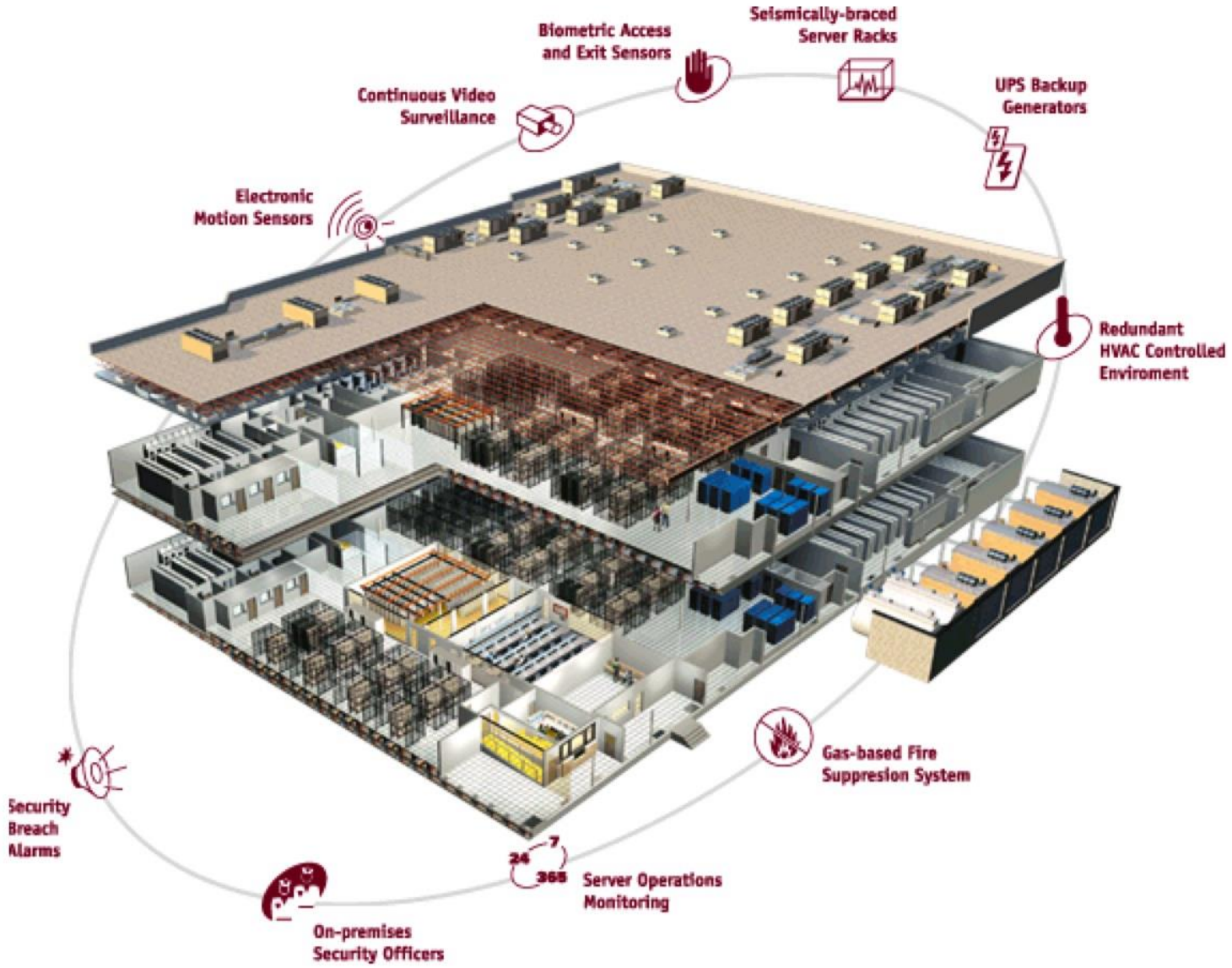

Server Room A
• KL3116/KL3116T
• CN8000
• PN9108
Server Room C
• KL1100
• KH2516
• CN8000
• PN5212/PN5320
• GN0116
Remote Management Console
• CC2000 Management Software
Intranet
KVM Adapter Cable
LAN
Server Room E
• KL1100
• KM0932
• KM0032
• CN8000
• PN7212/PN7320
• SN0116
• CC2000 Master
• CC2000 Slave
(NOC)
Server Room B
• KL1100
• KH1516i
• KH1516
• PN5212/PN5320
Server Room D
• KL1100
• KN4116/KN4132/KN4134v/KN4140v
• KH1508/KH1516
• PN7212/PN7320
• SN0116
• CC2000 Slave
Remote Management Console
• CC2000 Management Software
London
Shanghai
Moscow
Taipei

Anatomy of a Datacenter

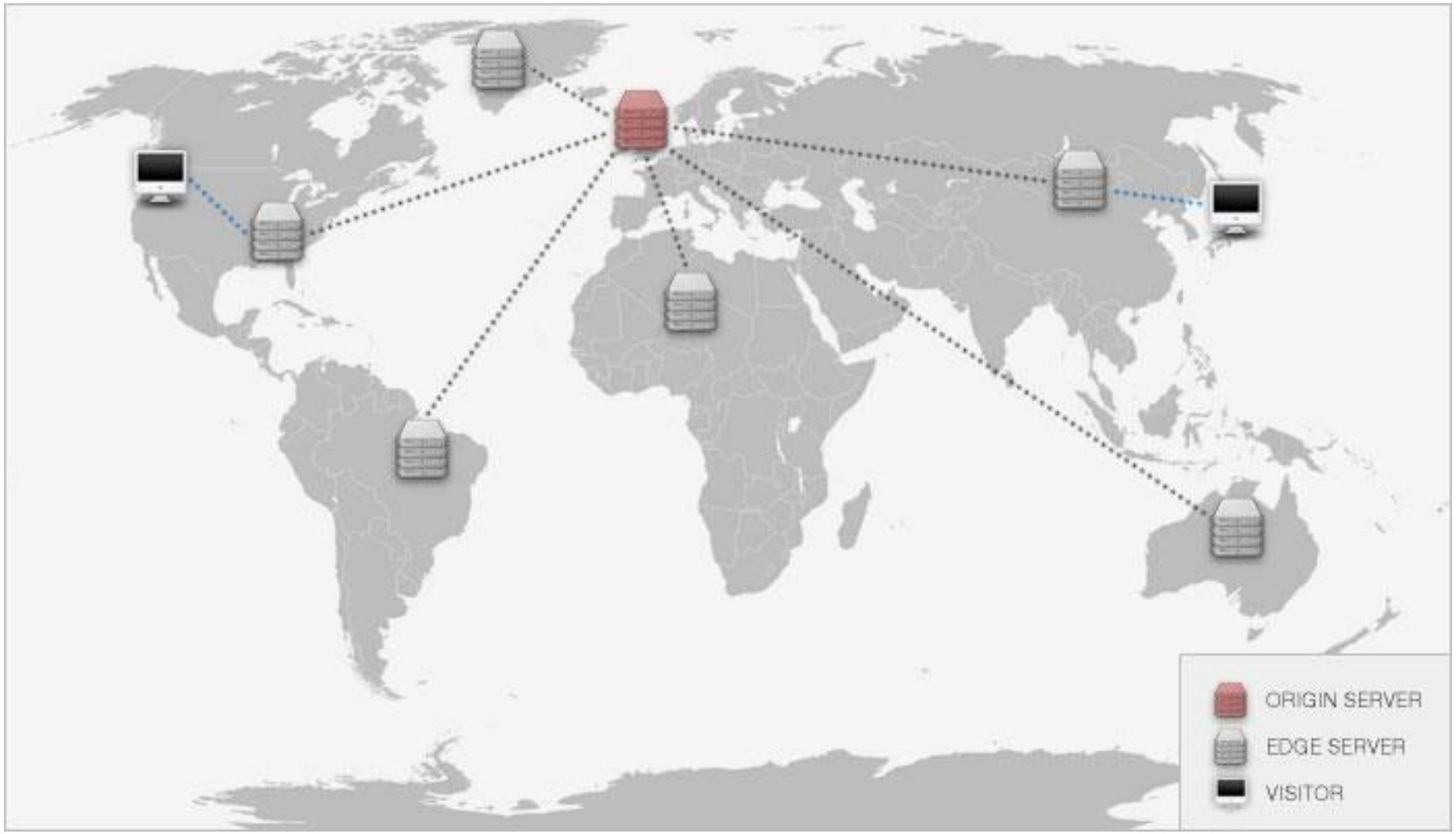




Warehouse Scale Computer



Google data centers in The Dalles, Oregon

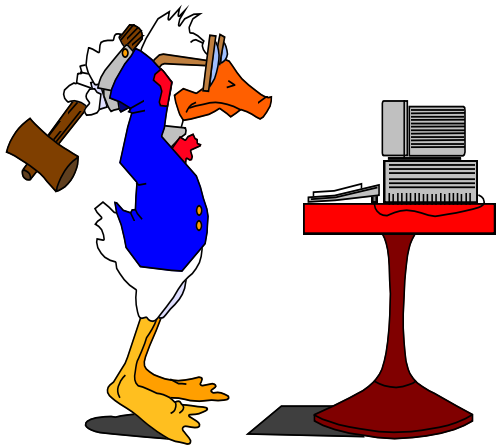

ORIGIN SERVER
EDGE SERVER
VISITOR


DATACENTER NORTH AMERICA
Memory
Disk
Server Resources
Processor
NIC
DB Servers
App Servers
Switch
File Servers
Index Servers
Clients
DATACENTER EUROPE
Switch
File Servers
Clients
DATACENTER ASIA
Switch
File Servers
Clients
Switch
File Servers
Clients
DATACENTER SOUTH AMERICA
Switch
File Servers
DATACENTER AFRICA
Clients
Switch
File Servers
DATACENTER AUSTRALIA

The datacenter *is* the computer

- **It's all about the right level of abstraction**
  - Moving beyond the von Neumann architecture
  - What's the "instruction set" of the datacenter computer?
- **Hide system-level details from the developers**
  - No more race conditions, lock contention, etc.
  - No need to explicitly worry about reliability, fault tolerance, etc.
- **Separating the *what* from the *how***
  - Developer specifies the computation that needs to be performed
  - Execution framework ("runtime") handles actual execution

Cluster Design Issues



- Enhanced Performance (performance @ low cost)
- Enhanced Availability (failure management)
- Single System Image (look-and-feel of one system)
- Size Scalability (physical & application)
- Fast Communication (networks & protocols)
- Load Balancing (CPU, Net, Memory, Disk)
- Security and Encryption (clusters of clusters)
- Distributed Environment (Social issues)
- Manageability (admin. and control)
- Programmability (simple API if required)
- Applicability (cluster-aware and non-aware app.)

The Modern Era – 2012

- **IBM BlueGene/Q**
  - 1,572,864 cores in 98,304 IBM Power CPUs
  - Very energy efficient = only 7.9 MW
    - 2066 GFLOP/ kW
  - Achieved 16.32 PFLOPs vs peak=20 PFLOP
    - Design can scale to 100 PFLOP …
  - Cost $97m

Also the GPU Era (2010-)

Year	Name	Peak speed	Location
2010	Tianhe-1A	2.57 PFLOPS	National Supercomputer Centre, Tianjin, China
2011	Fujitsu K computer	8.2 – 10.5 PFLOPS	RIKEN Advanced Institute for Computational Science, Japan
2012	IBM Sequoia	20.1 PFLOPS	Lawrence Livermore Lab. USA
2013	Tianhe-2	54.9 PFLOPS	National Super Computer Center in Guangzhou, China
2015	Sunway TaihuLight	125.4 PFLOPS	National Supercomputing Center in Wuxi, China
2018	Summit	187.6 PFLOPS	Oak Ridge National Lab, USA

Chapter 3: Large Scale Computing Systems

Faster for larger data

- von Neumann architecture
  - Foundation of modern computers
- 1962 Channel
  - Origin of concurrent programming
- Parallel
  - Vector processor, Multi-core, later GPU/CUDA
- Distributed
  - Cluster, Grid, …
- Now Clouding – Virtualizing computer systems for so-called Big Data
  - IaaS, PaaS, SaaS, …

Datacenters → Cloud Computing

Above the Clouds: A Berkeley View of Cloud Computing

Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz,
Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia
(Comments should be addressed to abovetheclouds@cs.berkeley.edu)

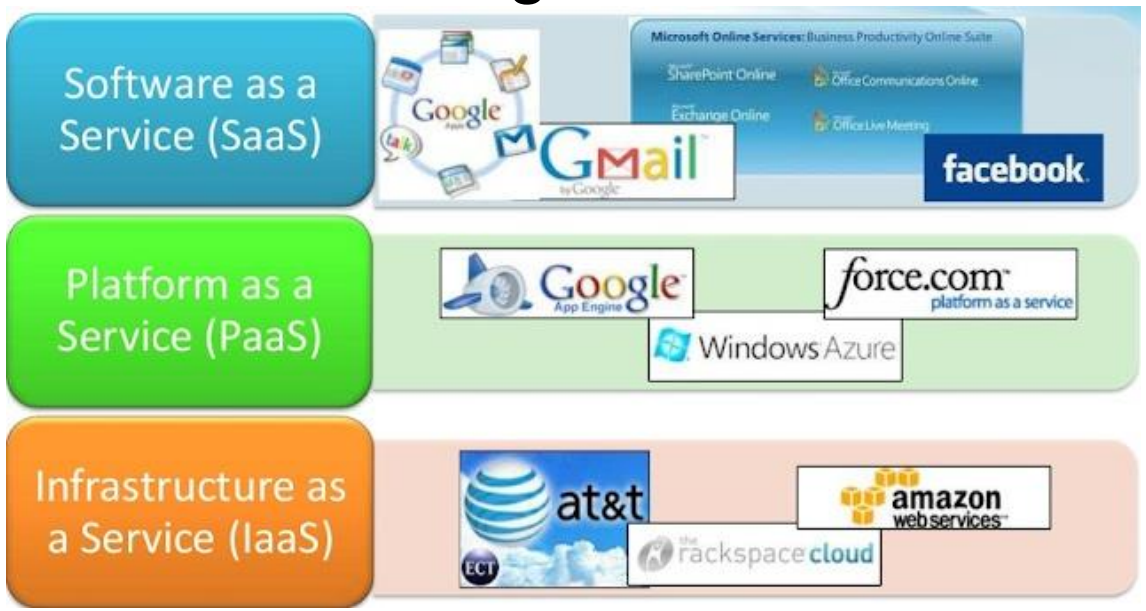
UC Berkeley Reliable Adaptive Distributed Systems Laboratory *
http://radlab.cs.berkeley.edu/

"…long-held dream of computing as a utility…"

Cloud – 1993-2002

- The term *cloud* was used for platforms for distributed computing as early as 1993
- July 2002, Amazon created subsidiary Amazon Web Services, with the goal to "enable developers to build innovative and entrepreneurial applications on their own."

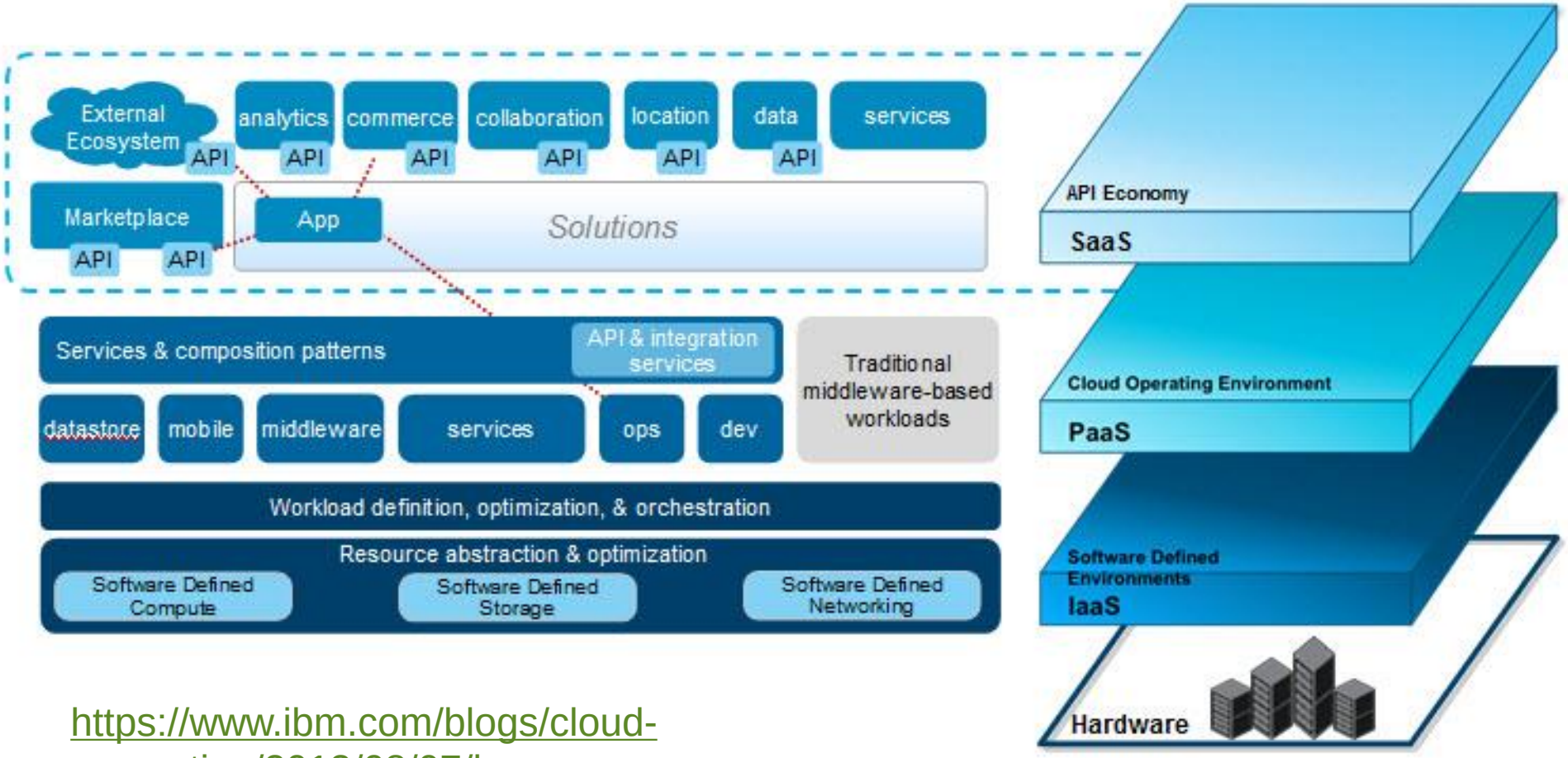
- **Originated in the business domain**
  - Outsourcing services; Pay for what you use
- **Provided by data centers built on computer and storage virtualization technologies.**



- **Scientific applications often have different requirements**
  - MPI, Shared file system, Support for many dependent jobs

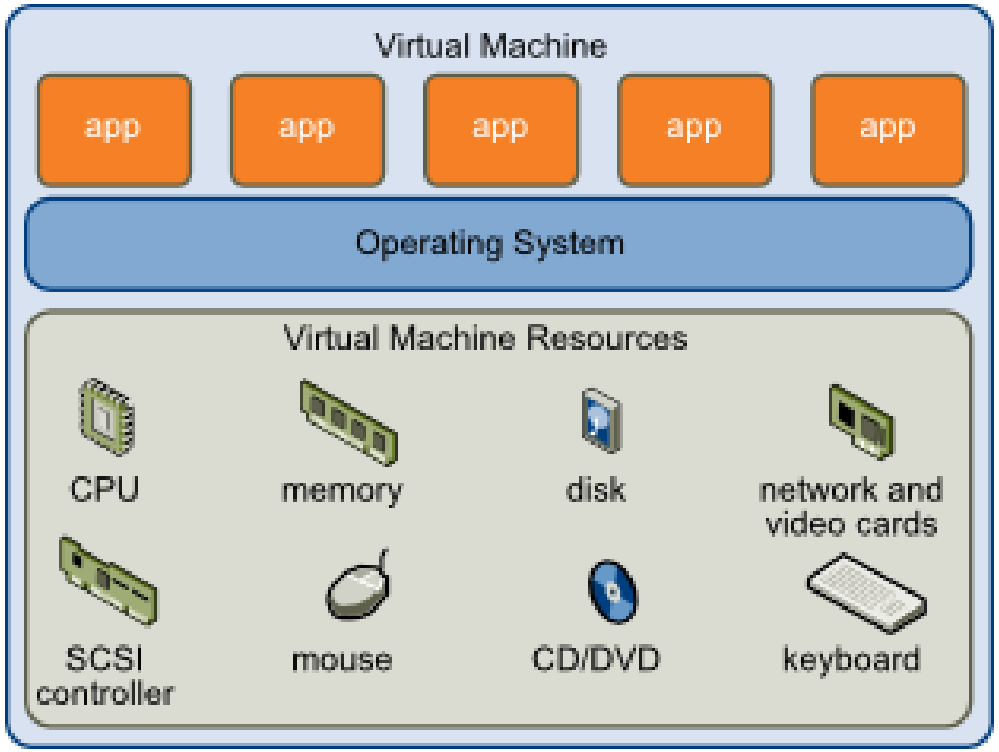
- Large scale
- Application-specific architectures
- Developed for in-house use

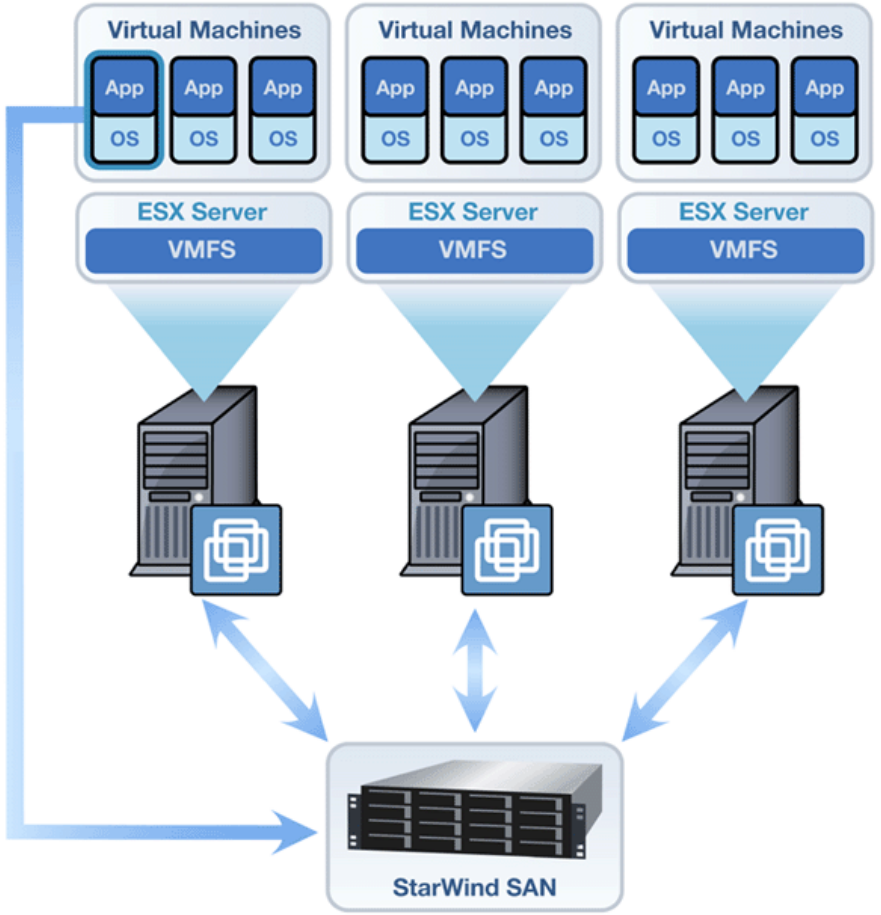
- Available for general usage
- Inexpensive, even for small or medium scale deployments



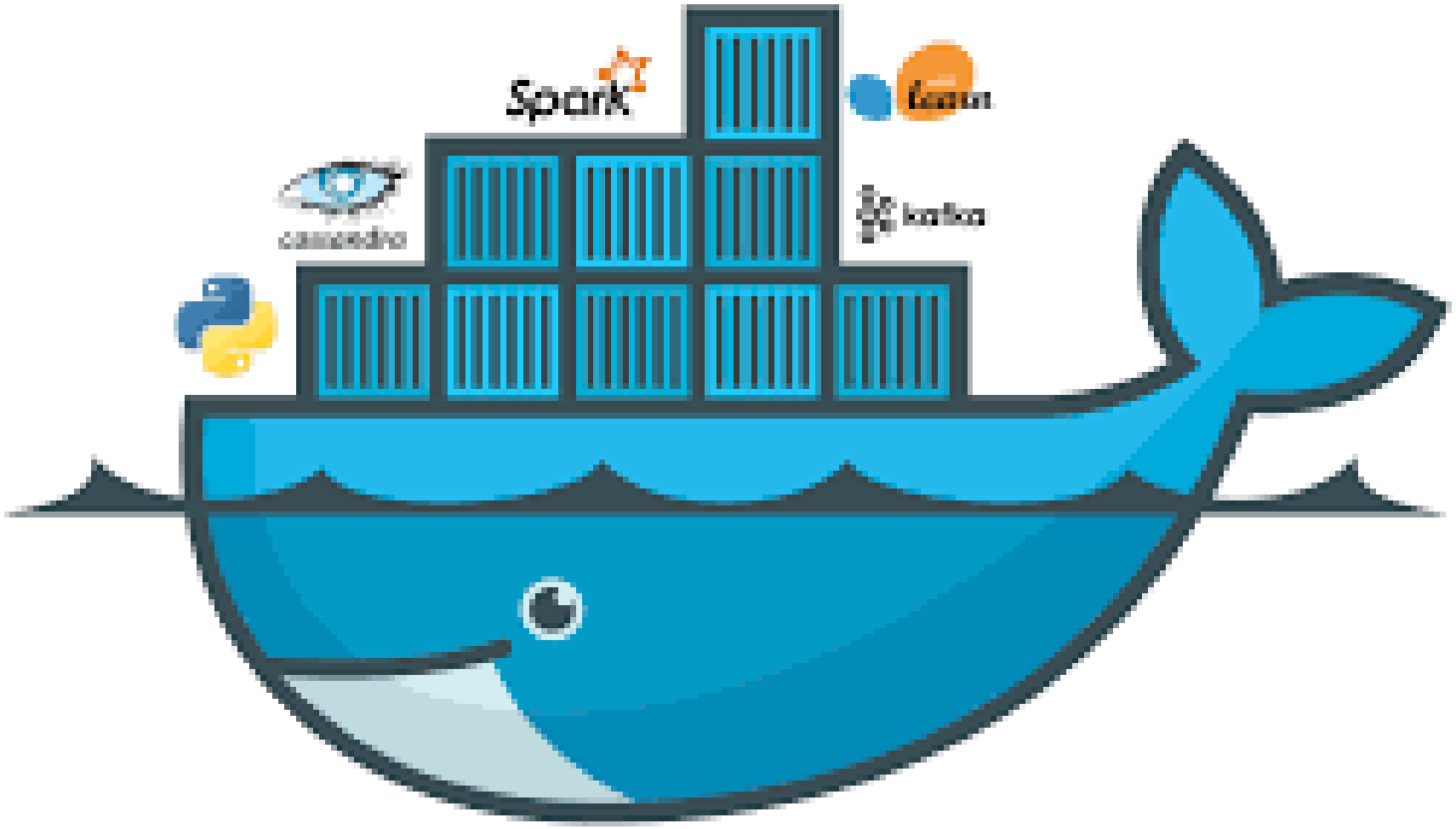
https://www.ibm.com/blogs/cloud-computing/2013/08/07/how-openpower-consortium-will-help-shape-the-open-cloud/

Virtual machines

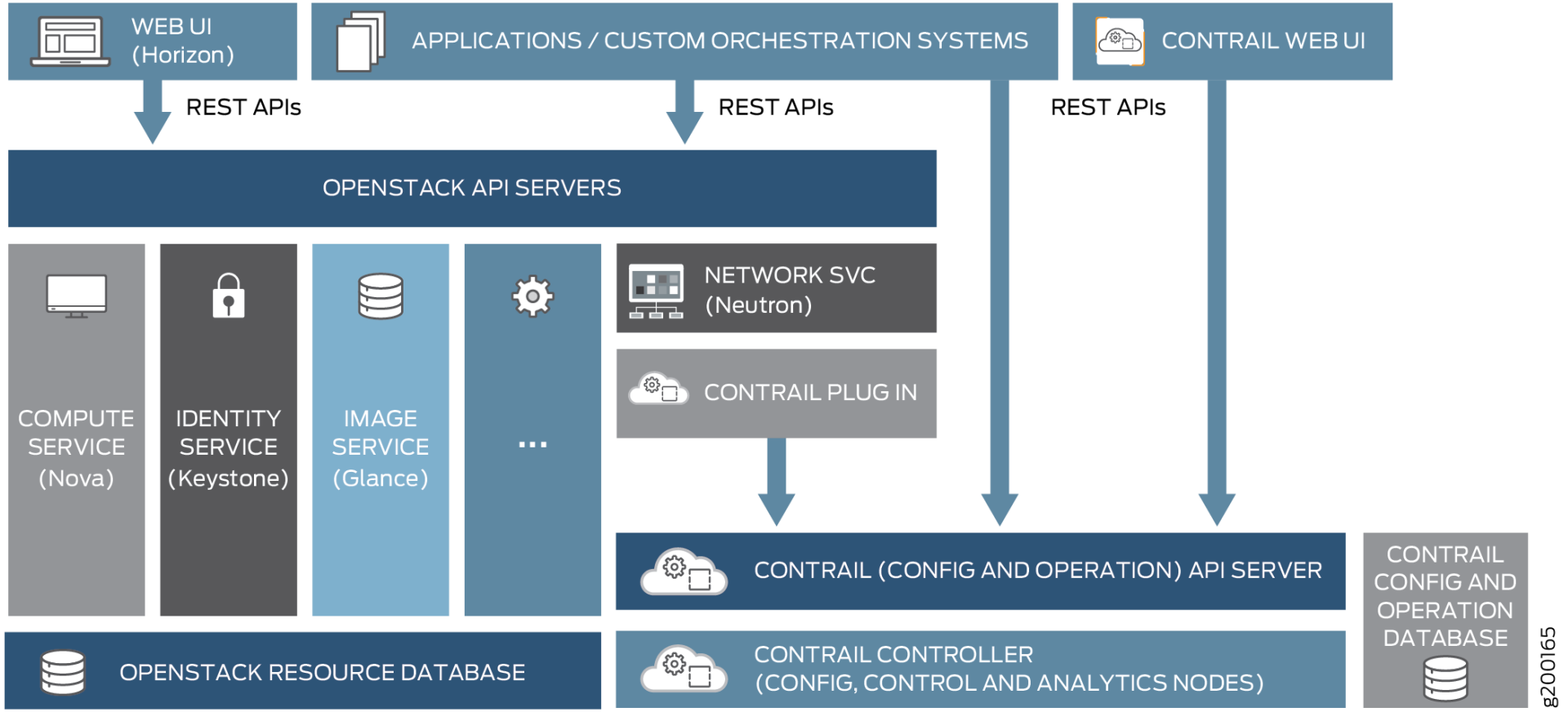




Containers



OpenStack


WEB UI
(Horizon)
APPLICATIONS / CUSTOM ORCHESTRATION SYSTEMS
CONTRAIL WEB UI
REST APIs
REST APIs
REST APIs
OPENSTACK API SERVERS
COMPUTE SERVICE (Nova)
IDENTITY SERVICE (Keystone)
IMAGE SERVICE (Glance)
...
NETWORK SVC (Neutron)
CONTRAIL PLUG IN
CONTRAIL (CONFIG AND OPERATION) API SERVER
CONTRAIL CONFIG AND OPERATION DATABASE
OPENSTACK RESOURCE DATABASE
CONTRAIL CONTROLLER
(CONFIG, CONTROL AND ANALYTICS NODES)
g200165

Summary of Traditional Differences

(both are changing)

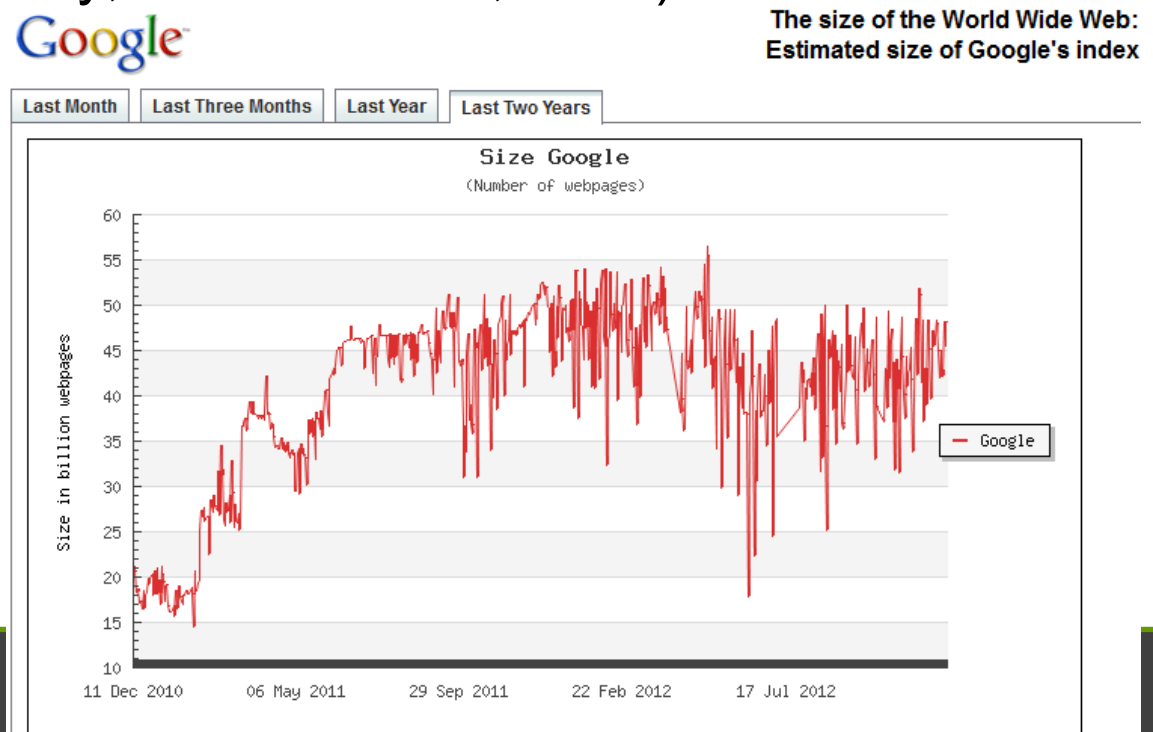
Cloud	HPC
Focus on storage	Focus on computing (flop/s)
Cheap(est) commodity component	High end components (some specialization)
Commodity networks	High performance networks
Pay as you go	Purchased for mission; pay in non-fungible “hours”
< 50% utilization	> 90% utilization
On-demand access	Large jobs wait in queues
On-node disks (air cooled)	Separate storage (compute is liquid cooled)

Big Data On Cluster or Cloud

- **Ambitious to manage huge and diverse data – 3Vs: Volume, Variety, Velocity**
  - 2.5 quintillion bytes of data are generated every day!
    - A quintillion is 10^{18}
  - Coming from many quarters, like Social media sites, Sensors, Digital photos, Business transactions, Location-based data, Web data, e-commerce, Bank/Credit Card, …
  - For information: Google processes 20 PB a day (2008)
    - http://www.worldwidewebsize.com/
      - ✓ "The Indexed Web contains **at least 9.18 billion pages** (Sunday, 09 December, 2012)."
  - For science: NASA & Hubble scope

HUGE data

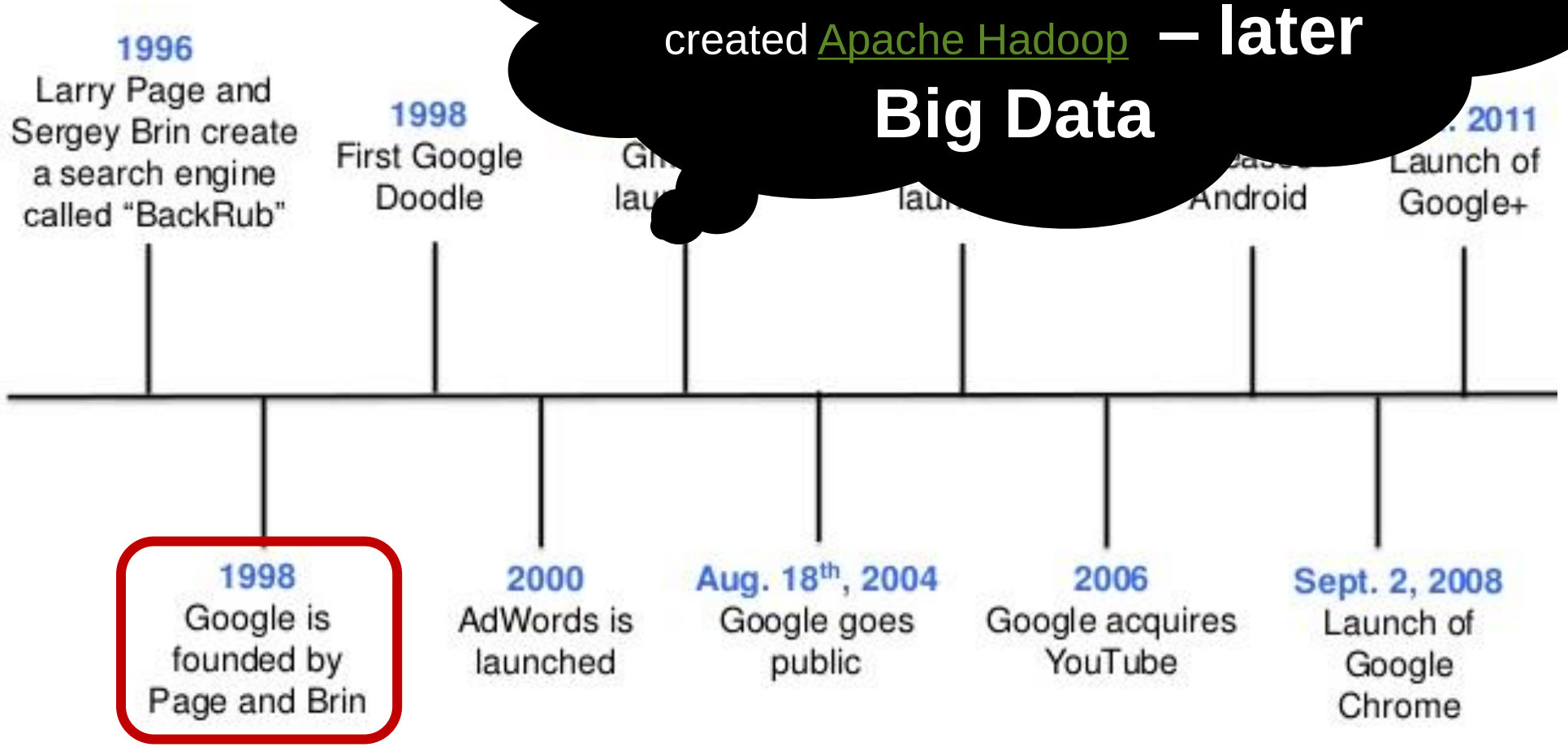
- **Making decisions!**
  - For better information: Google processes 20 PB a day (2008)
  - http://www.worldwidewebsize.com/
    - "The Indexed Web contains **at least 9.18 billion pages** (Sunday, 09 December, 2012)."



Google triggers other sea

- CBIR, XML, …

This year Google published a white paper describing the MapReduce framework, Doug Cutting and Mike Cafarella created Apache Hadoop – later **Big Data**



- **Later Hadoop (HDFS, MapReduce) project provides OPEN-SOURCE codes so that every one could uses the powerful tools to process HUGE/BIG Data!**
  - Apache top level project, open-source implementation of frameworks for reliable, scalable, distributed computing and data storage
  - Designed to answer the question: **"How to process big data with reasonable cost and time?"**

Doug Cutting

2005: Doug Cutting and Michael J. Cafarella developed Hadoop to support distribution for the Nutch search engine project.

The project was funded by Yahoo.

2006: Yahoo gave the project to Apache Software Foundation.

2003

The Google File System

Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung

Google*

2004

MapReduce: Simplified Data Processing on Large Clusters

Jeffrey Dean and Sanjay Ghemawat

jeff@google.com, sanjay@google.com

Google, Inc.

2006

Bigtable: A Distributed Storage System for Structured Data

Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach
Mike Burrows, Tushar Chandra, Andrew Fikes, Robert E. Gruber

{fay,jeff,sanjay,wilsonh,kerr,m3b,tushar,fikes,gruber}@google.com

Google, Inc.

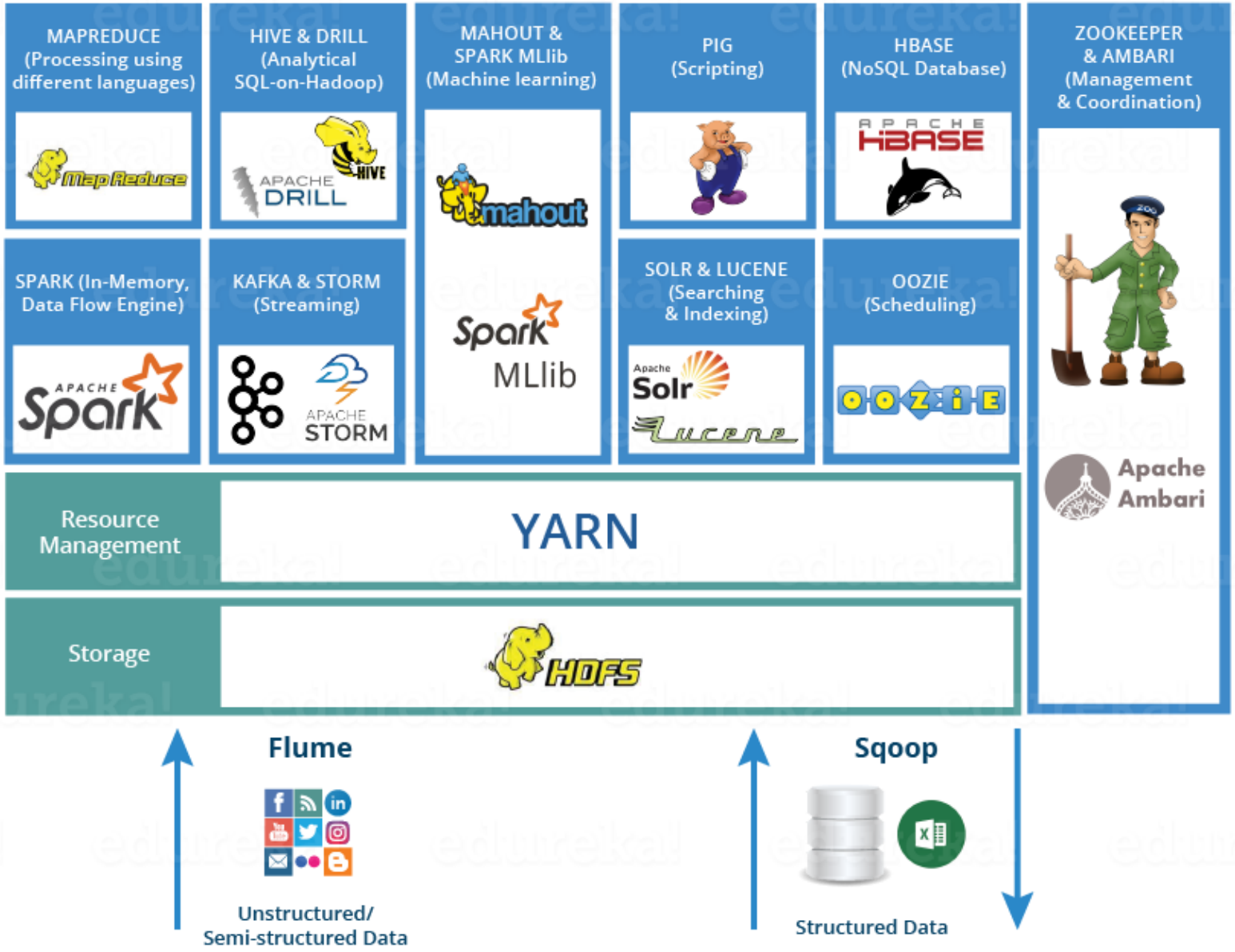
Abstract

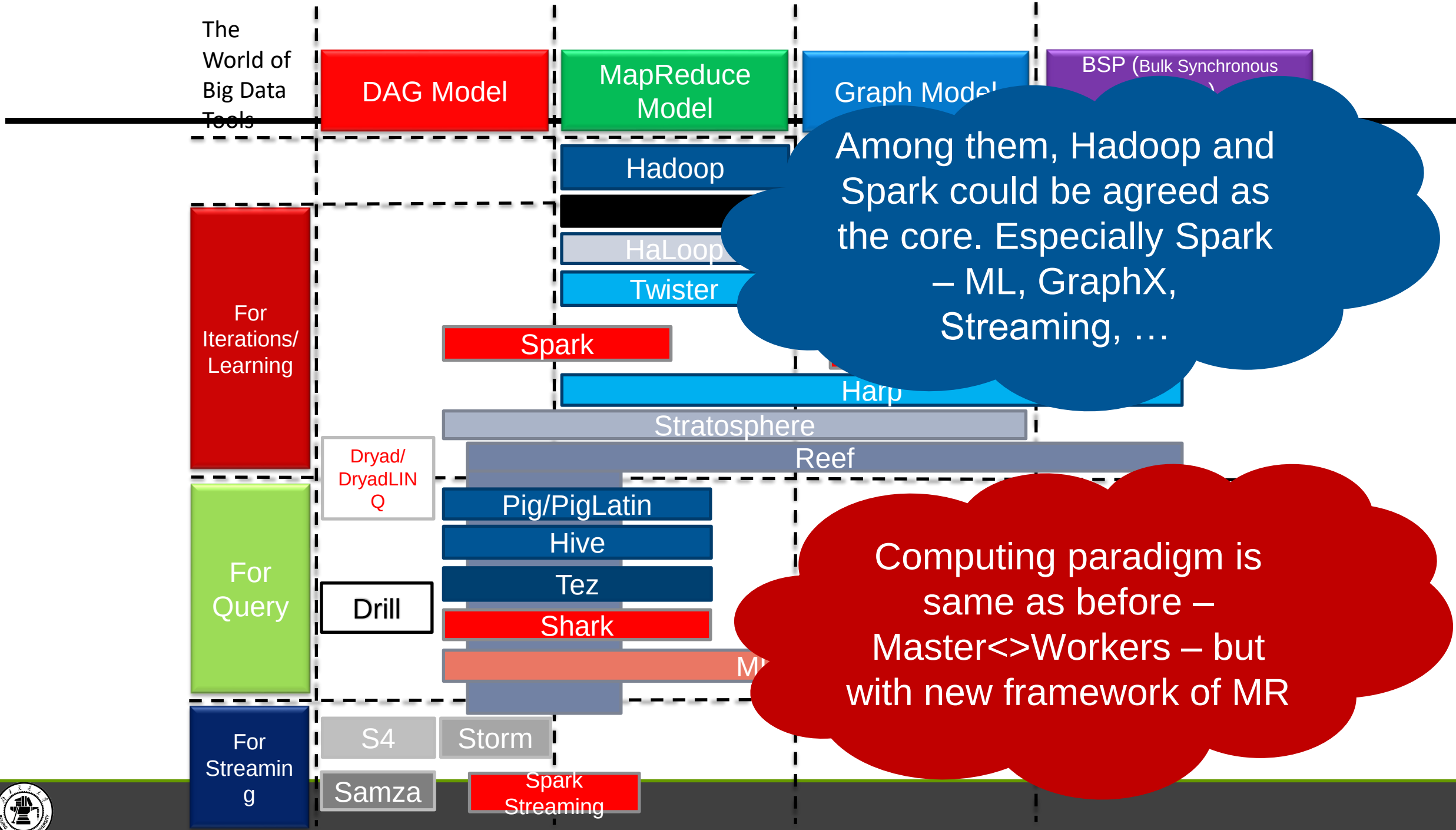
igtable is a distributed storage system for managing
tured data that is designed to scale to a very large
petabytes of data across thousands of commodity
rs. Many projects at Google store data in Bigtable,
ding web indexing, Google Earth, and Google Fi-
e. These applications place very different demands
igtable, both in terms of data size (from URLs to

achieved scalability and high performance, but Big
provides a different interface than such systems. Big
does not support a full relational data model; inste
provides clients with a simple data model that sup
dynamic control over data layout and format, an
lows clients to reason about the locality properties o
data represented in the underlying storage. Data i
dexed using row and column names that can be arbi
strings. Bigtable also treats data as uninterpreted st

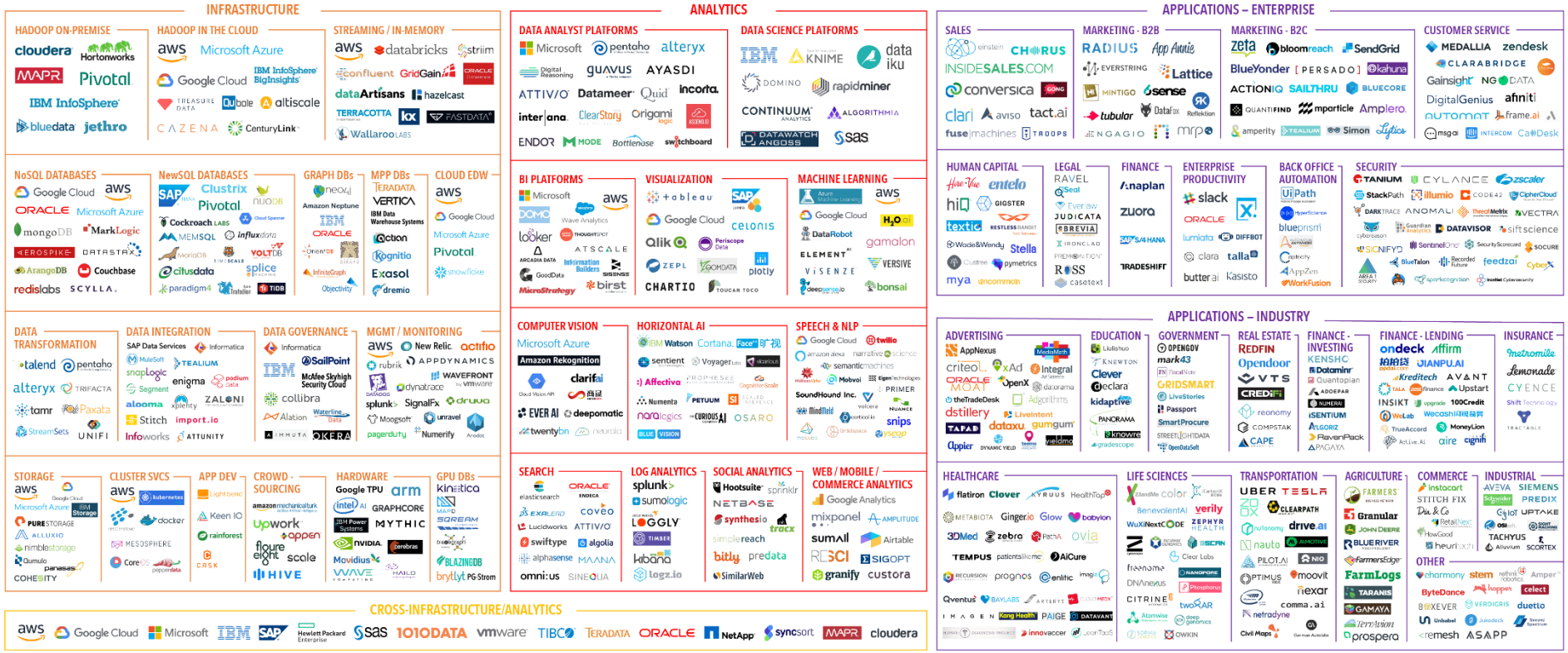
- 2008 - Hadoop Wins Terabyte Sort Benchmark (sorted 1 terabyte of data in 209 seconds, compared to previous record of 297 seconds)
- 2009 - Avro and Chukwa became new members of Hadoop Framework family
- 2010 - Hadoop's Hbase, Hive and Pig subprojects completed, adding more computational power to Hadoop framework
- 2011 - ZooKeeper Completed
- 2013 - Hadoop 1.1.2 and Hadoop 2.0.3 alpha.
  - Ambari, Cassandra, Mahout have been added

Below are the Hadoop components, that together form a Hadoop (2.X) ecosystem

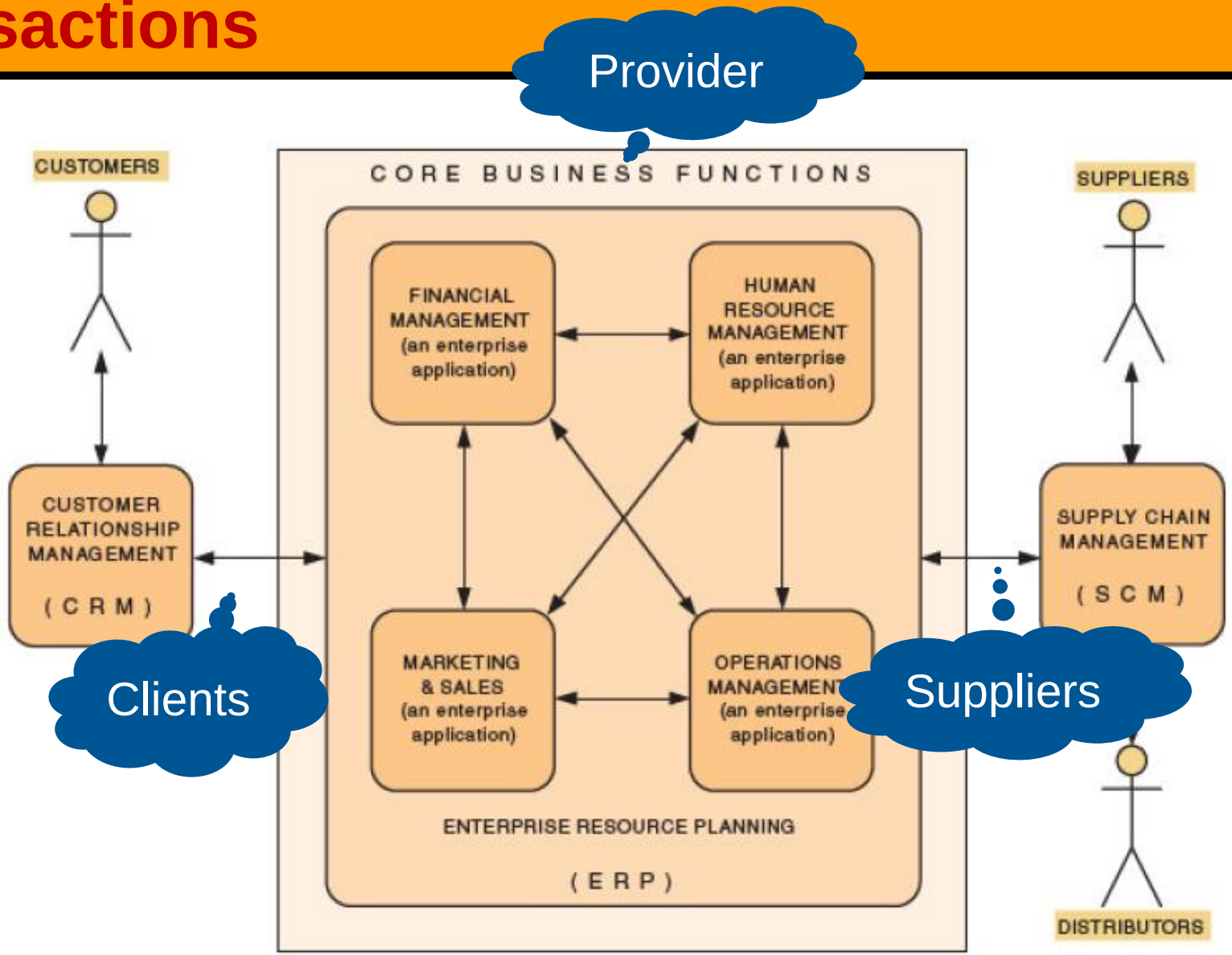



The World of Big Data Tools
DAG Model
MapReduce Model
Graph Model
BSP (Bulk Synchronous
Hadoop
HaLoop
Twister
Spark
Harp
Stratosphere
Reef
Dryad/ DryadLINQ
Pig/PigLatin
Hive
Tez
Drill
Shark
S4
Storm
Samza
Spark Streaming
For Iterations/ Learning
For Query
For Streaming
Among them, Hadoop and Spark could be agreed as the core. Especially Spark – ML, GraphX, Streaming, …
Computing paradigm is same as before – Master<>Workers – but with new framework of MR

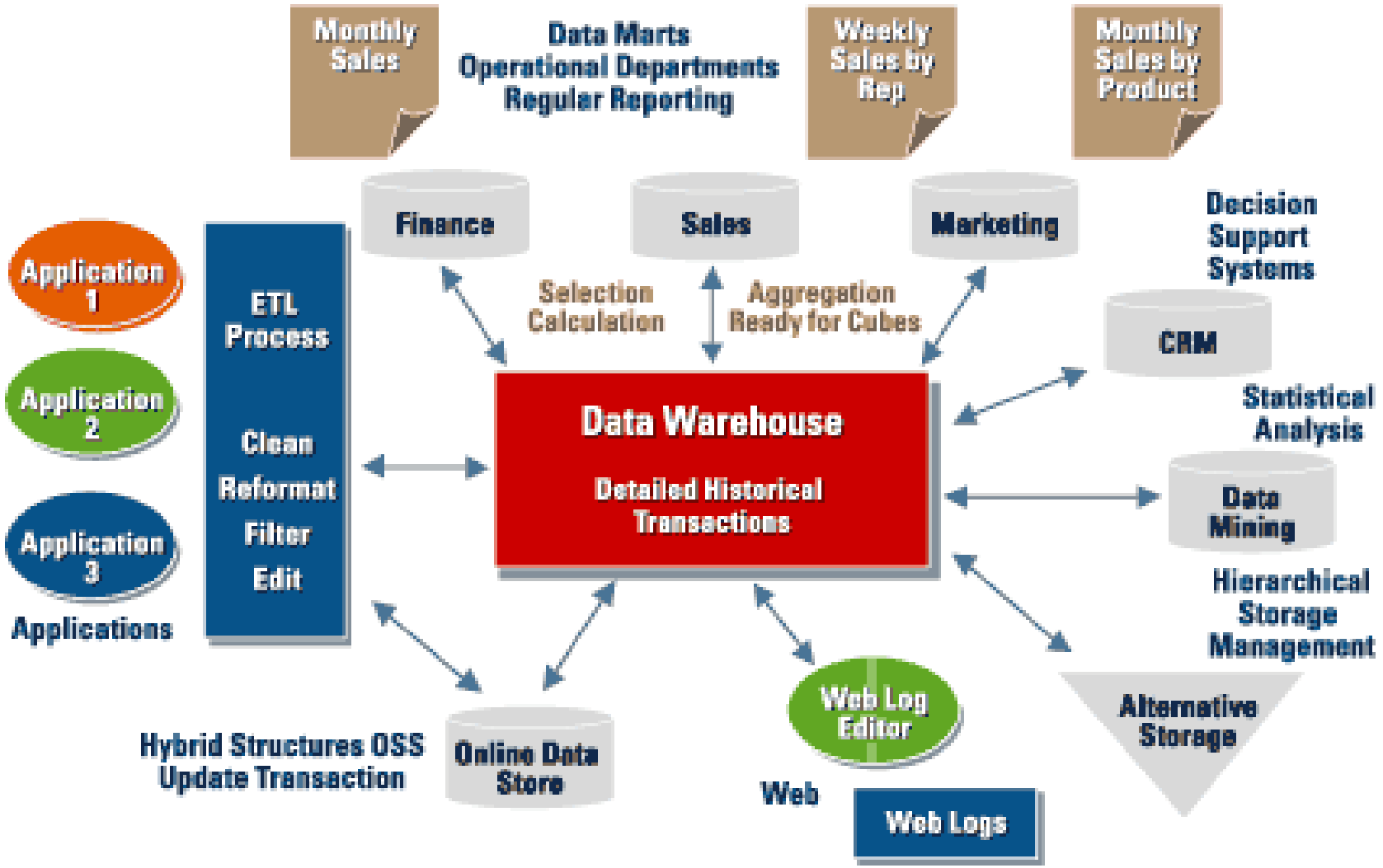
BIG DATA & AI LANDSCAPE 2018



But never forget the business system – transactions



DW also tries to integrate Data Analytics



Hadoop runs on commodity with high fault-tolerance

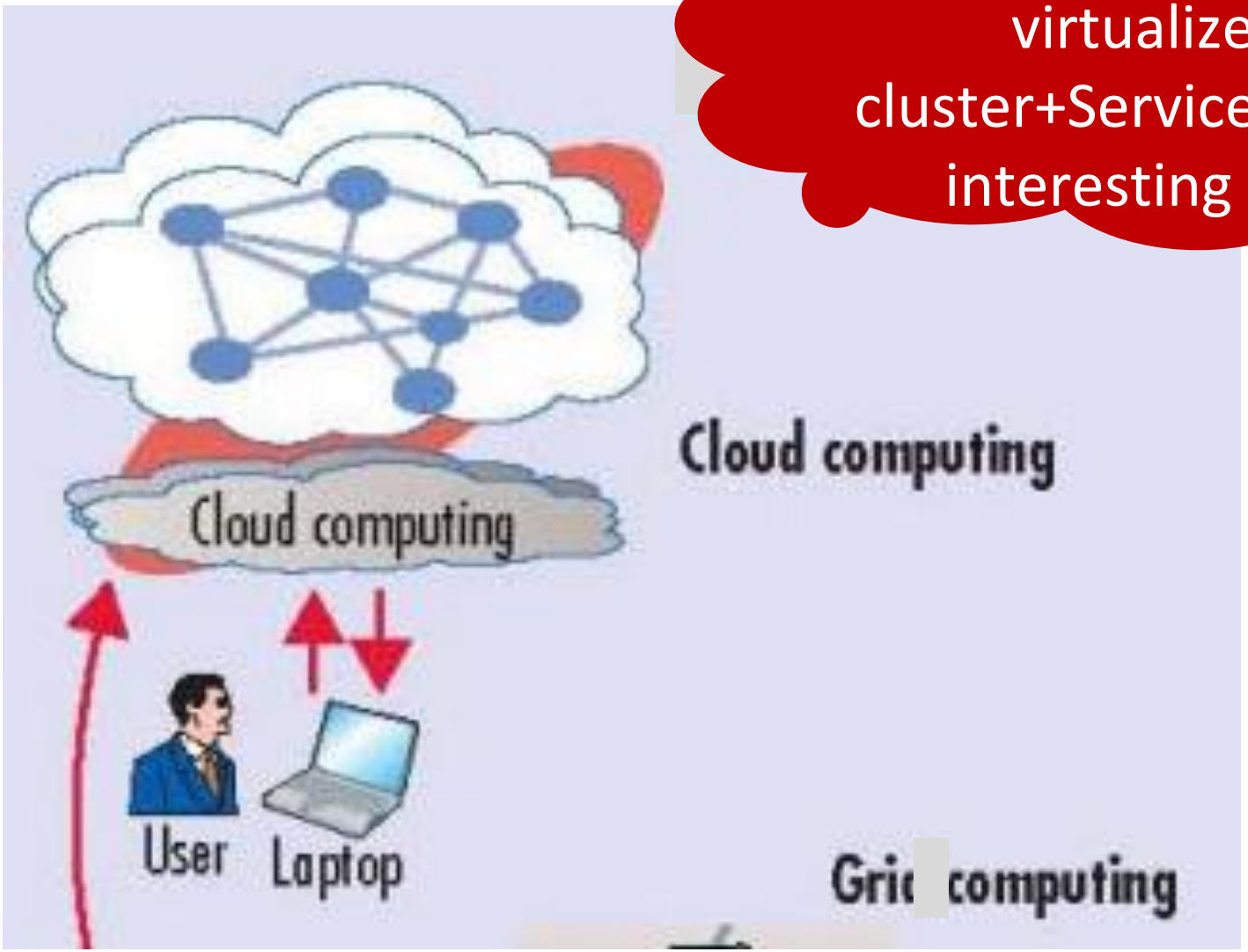
Yahoo's Hadoop cluster

Apache Hadoop is a most popular open-source software platform for data-intensive applications.

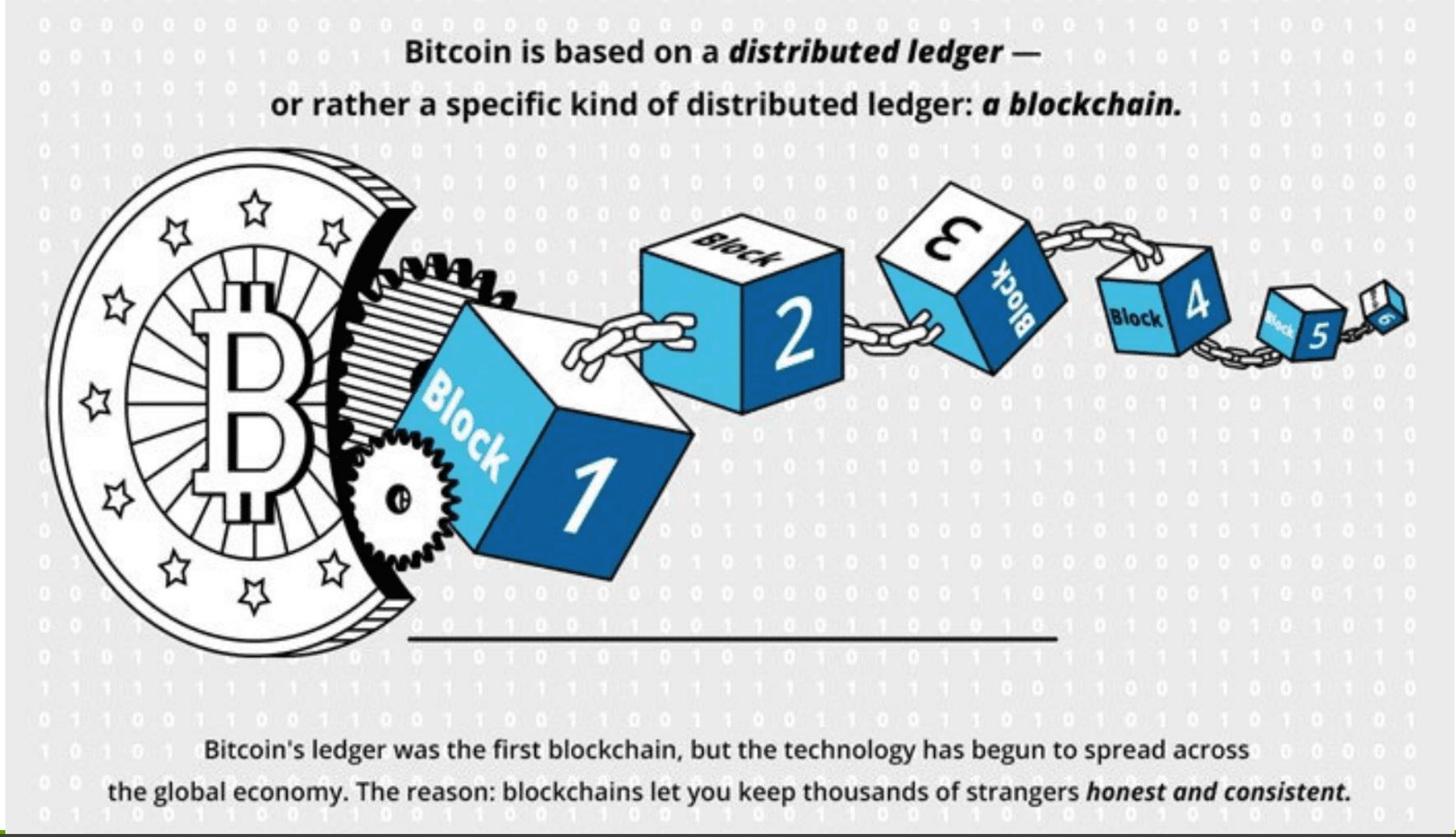
Hadoop runs on commodity clusters with high fault-tolerance.

Hadoop's primary modules are MapReduce (MR) and the Hadoop Distributed File System (HDFS). MR implements a high-level, implicit parallel programming model. HDFS provides high-throughput access to big data.

Cloud could be understood as virtualized cluster+Services ~~ an interesting way



Blockchain


Bitcoin is based on a *distributed ledger* —
or rather a specific kind of distributed ledger: *a blockchain.*
Block 1
Block 2
Block 3
Block 4
Block 5
Bitcoin's ledger was the first blockchain, but the technology has begun to spread across
the global economy. The reason: blockchains let you keep thousands of strangers *honest and consistent.*

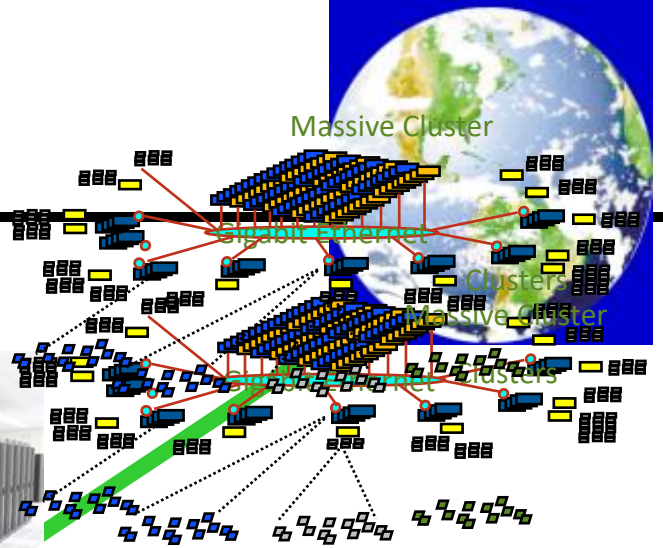
Chapter 3: Large Scale Computing Systems

- **Faster for larger data**
  - von Neumann architecture
  - 1962 Channel
  - Parallel
  - Distributed
  - Now Clouding – Virtualizing computer systems
- **Review**

Summary

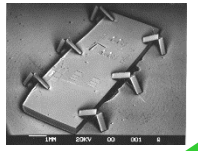
Societal Scale Information Systems (Or the "Internet of Things"?)

- **The world is a large distributed system**
  - Microprocessors in everything
  - Vast infrastructure behind them



Scalable, Reliable, Secure Services

Internet Connectivity

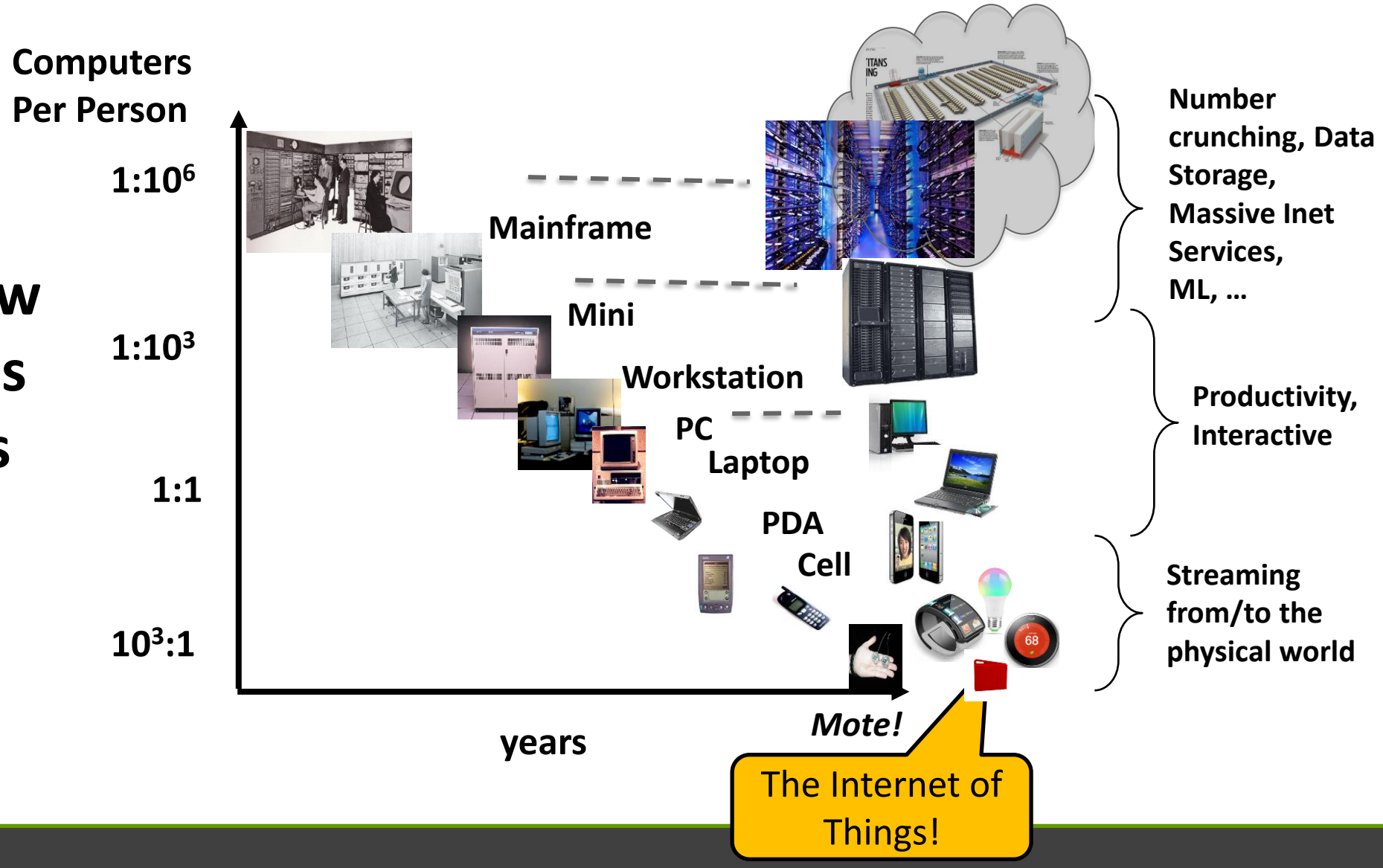


Databases
Information Collection
Remote Storage
Online Games
Commerce
…

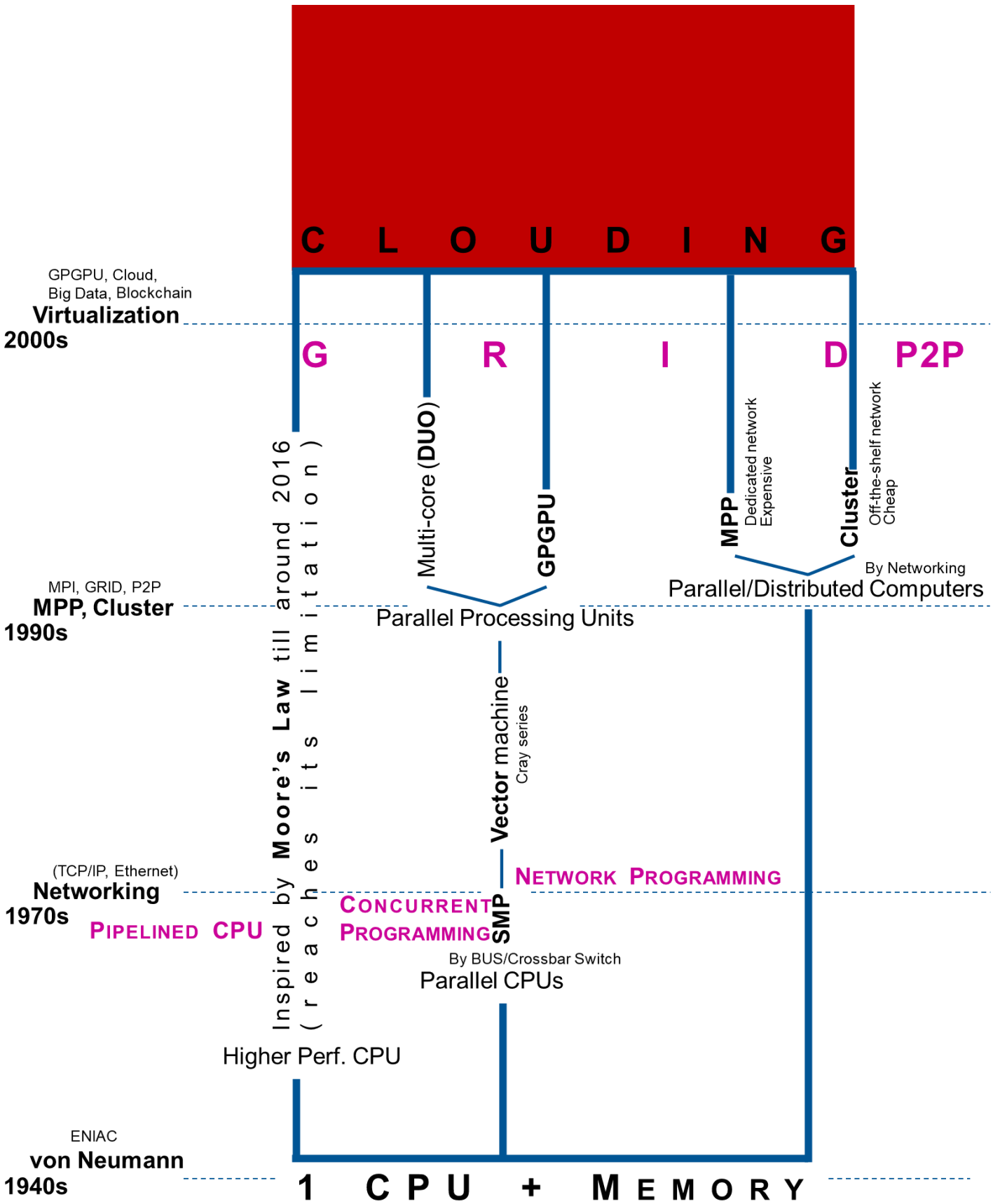
MEMS for Sensor Nets

Across Incredible Diversity

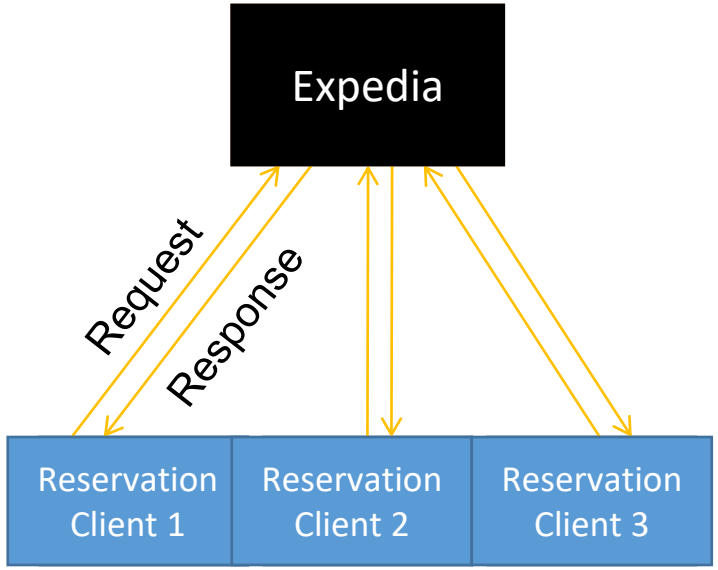
Bell's Law: New computer class every 10 years



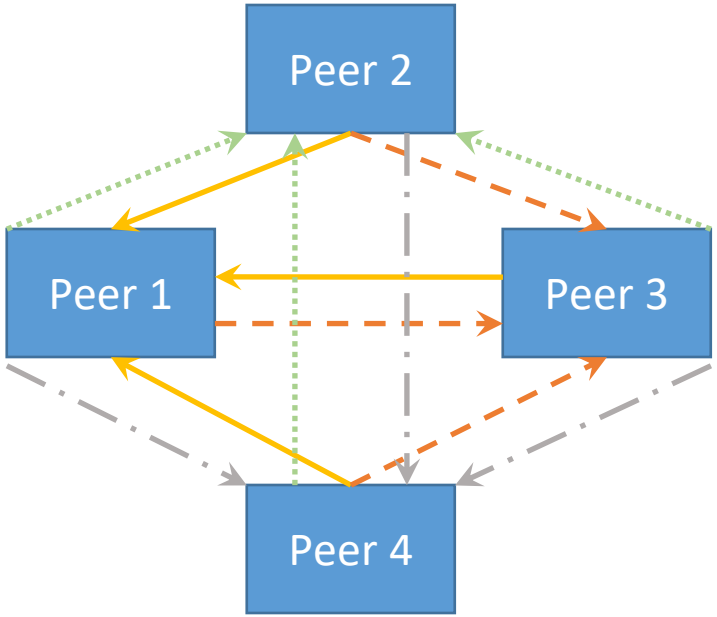
Spark	2014
Deep Learning (Champion in ImageNet test)	2012
Hadoop (Apache project with **MR**)	2011
CUDA (Compute Unified Device Architecture)	2007
CLOUD (Amazon introduced its Elastic Compute Cloud (EC2))	2007
Map/Reduce (Paper)	2004
GPU (NVIDIA was 1st to produce a chip capable of programmable shading, GeForce 3 (NV20) But, Graphic accelerator (Parallel to CPU) has long history from 1970s	2000s
P2P (famous Napster in 1999, Bittorent in 2001)	1999
GRID (by Ian Foster and Carl Kesselman)	1999
MPP (Intel ASCI-Red for nuclear weapon testing (DoE and NNSA))	1995
Cluster (NASA Beowulf project)	1994
MPI (Message Passing Interface)	1992
Python	1990
DNS (a popular distributed computing system)	1983
vector processor (CDC Star-100 was one of the first machines to use a vector processor) **Ethernet** was developed at Xerox PARC between 1973 and 1974	1974
Start of **Internet** (ARPANET)	1969
Flynn's Taxonomy	1966
Parallel (between CPU and IO by Channel) **SMP** (1st is Burroughs B5000 1961)	1962
First Numerical **Forecast** (Charney barotropic model run on ENIAC)	1950
ENIAC (von Neumann)	1946


C L O U D I N G
GPGPU, Cloud,
Big Data, Blockchain
Virtualization
2000s
G
R
I
D
P2P
Multi-core (DUO)
GPGPU
MPP
Dedicated network
Expensive
Cluster
Off-the-shelf network
Cheap
By Networking
Parallel/Distributed Computers
MPI, GRID, P2P
MPP, Cluster
1990s
Parallel Processing Units
Inspired by Moore's Law till around 2016
(reaches its limitation)
Vector machine
Cray series
(TCP/IP, Ethernet)
Networking
1970s
NETWORK PROGRAMMING
PIPELINED CPU
CONCURRENT
PROGRAMMING
SMP
By BUS/Crossbar Switch
Parallel CPUs
Higher Perf. CPU
ENIAC
von Neumann
1940s
1 CPU + MEMORY

Bird's Eye View of Some Distributed Systems



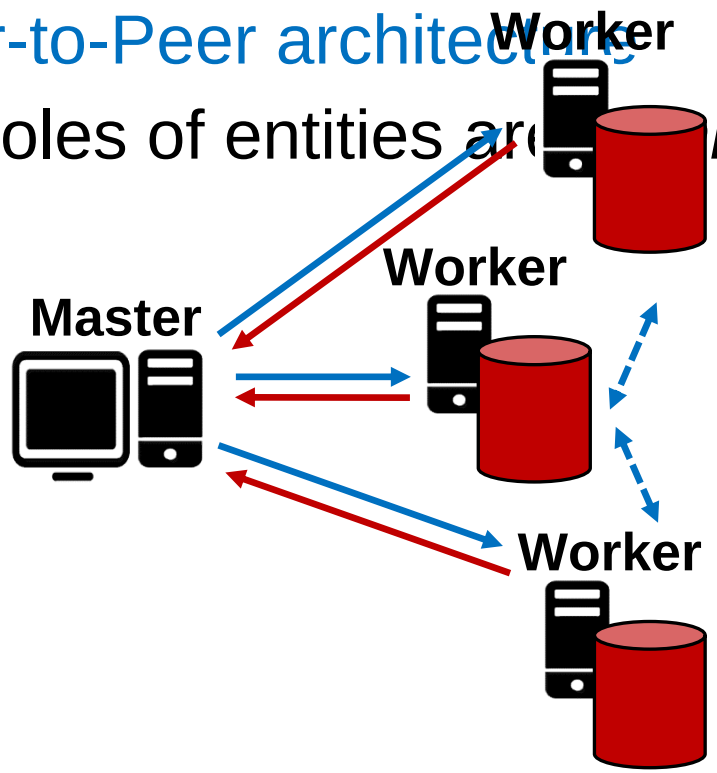
Google Search
Airline Booking

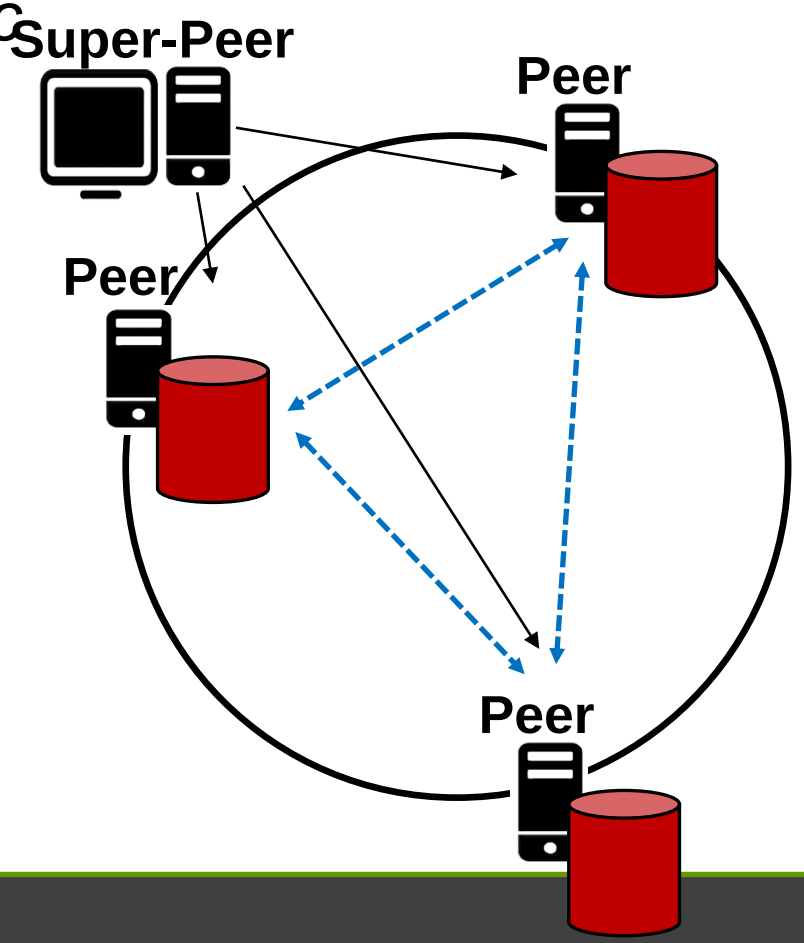


Bit-torrent
Skype

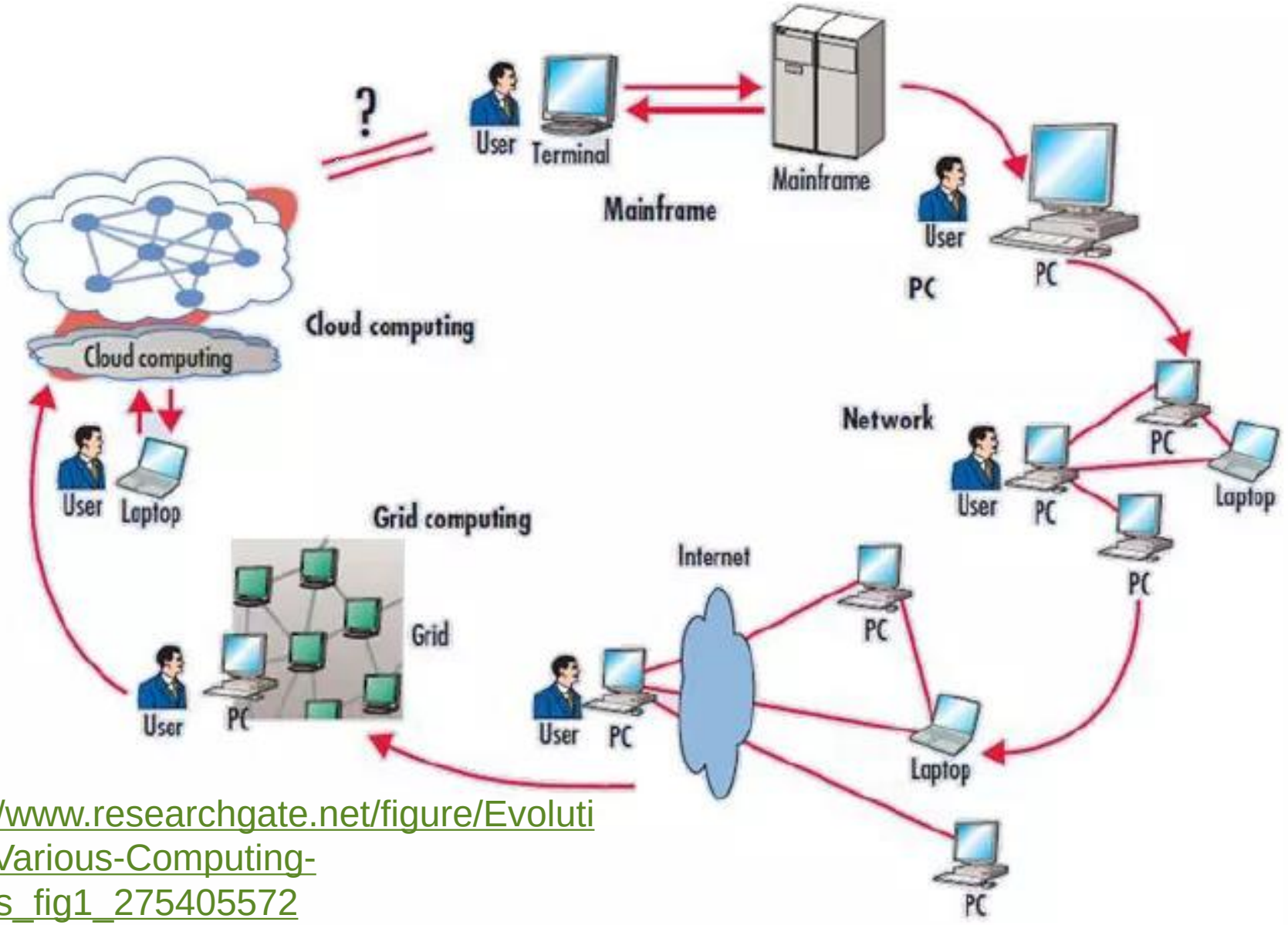
Architectures

- **Two main architectures:**
  - M
    - Roles of entities are *asymmetric*
  - Peer-to-Peer architectures
    - Roles of entities ar*metric*





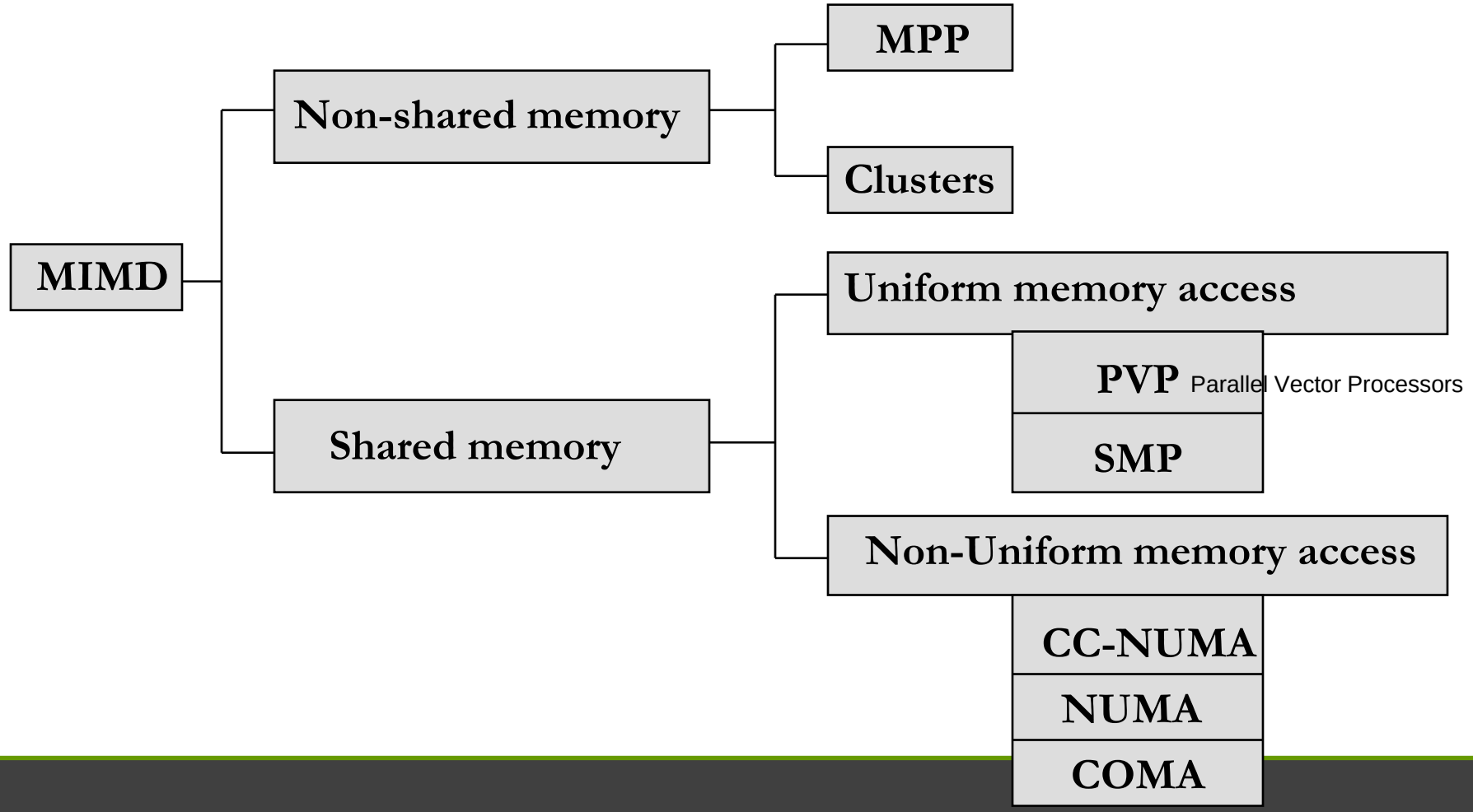
Evolution of Various Computing Models



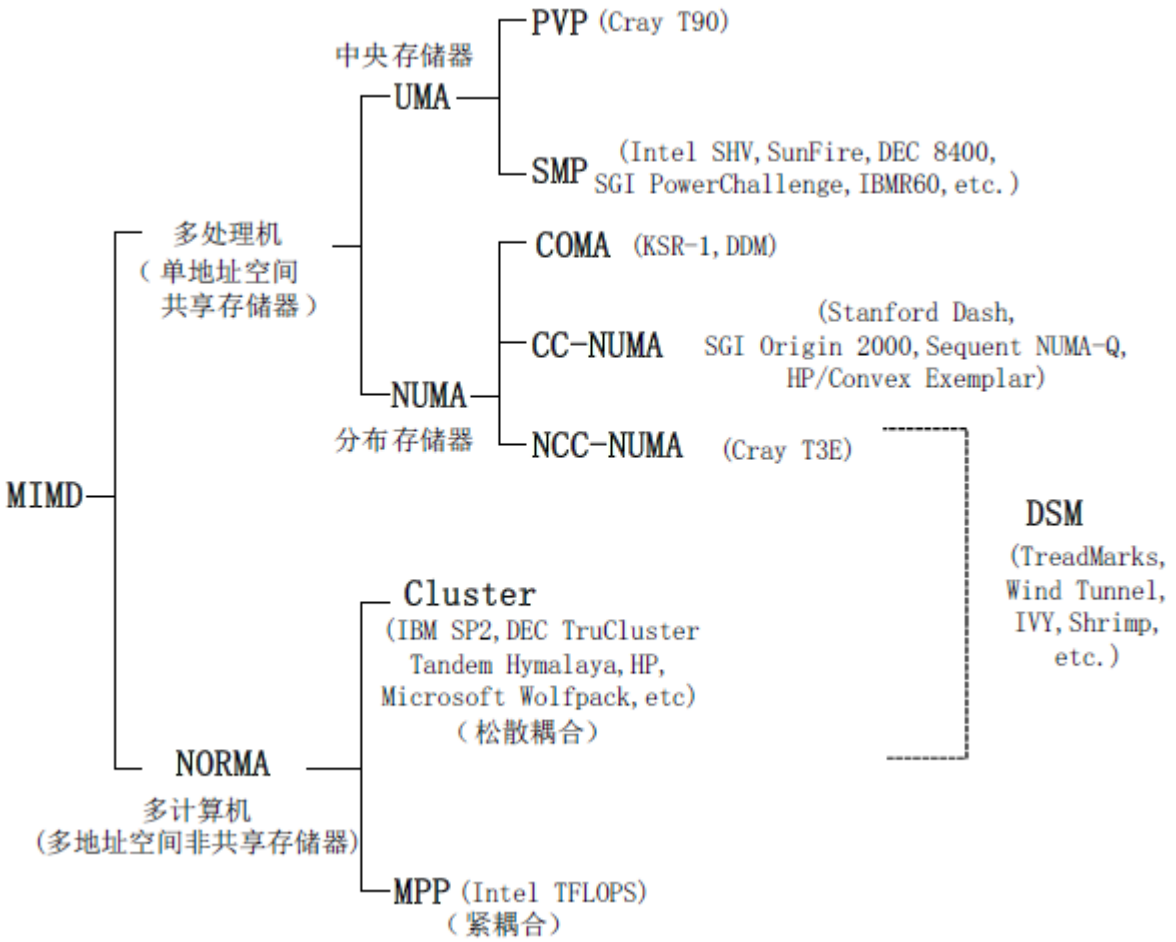
https://www.researchgate.net/figure/Evolution-of-Various-Computing-Models_fig1_275405572

HPC ↔ MIMD Architectures

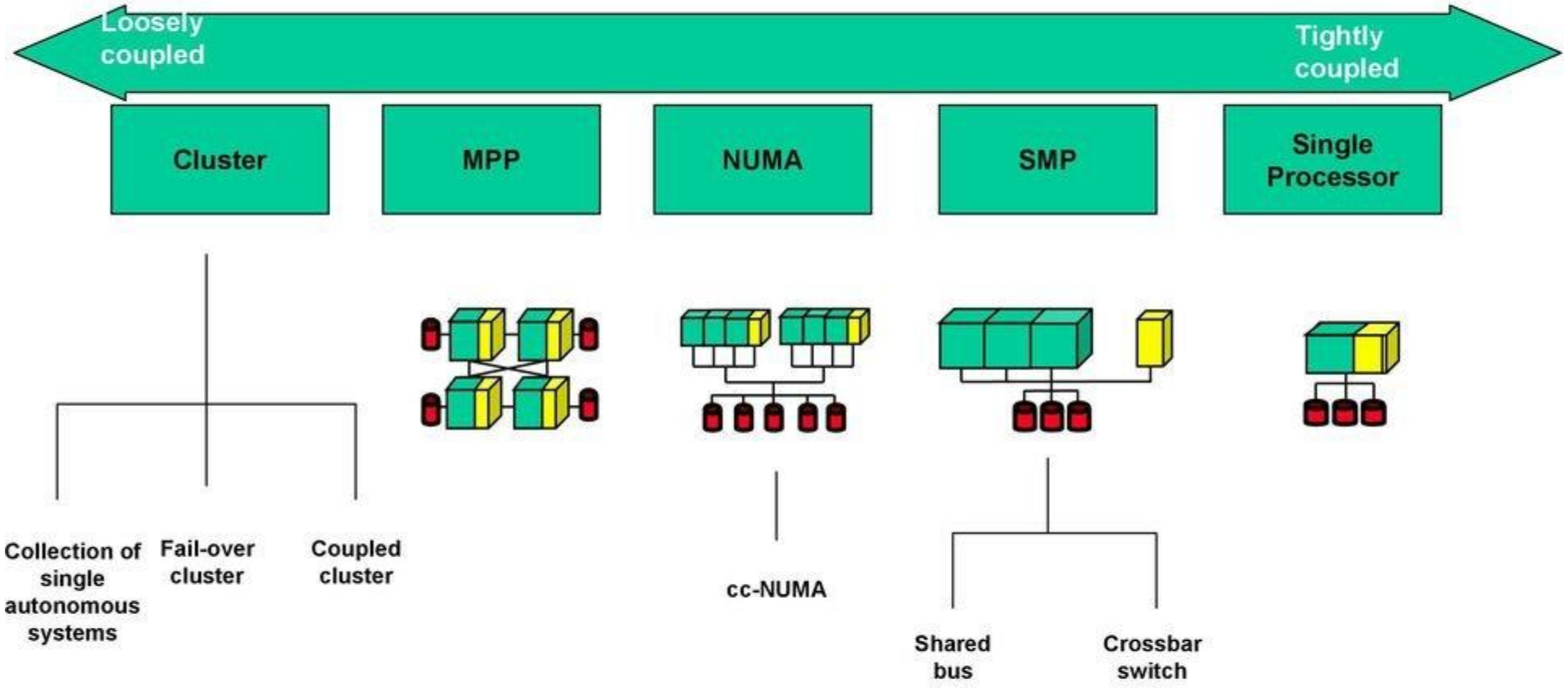
Current focus is on MIMD model, using general purpose processors or multicomputers.



并行机系统的不同存储结构

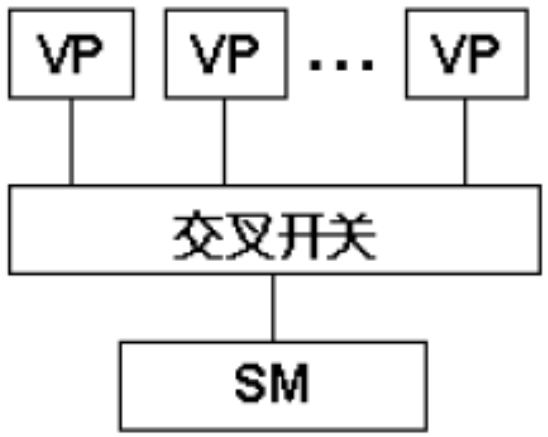


Multiprocessor system architectures

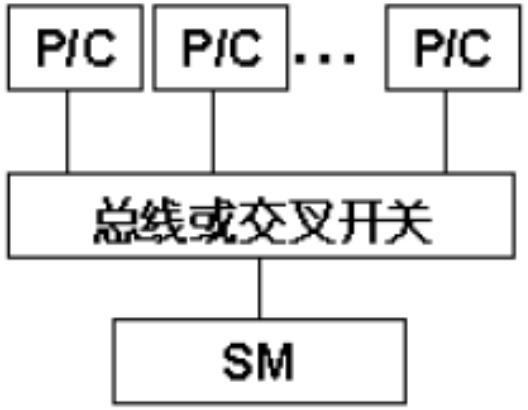


Da: M.Bozzetti/EITO

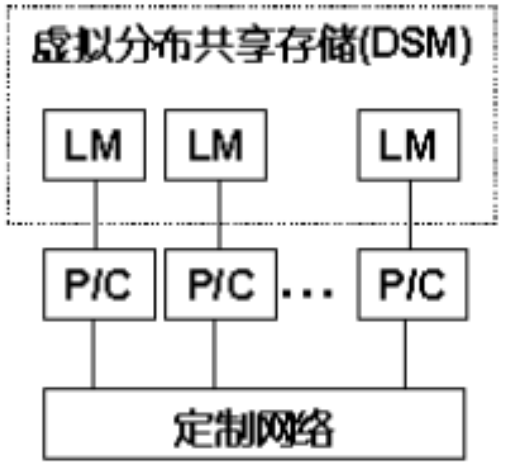
Parallel Vector Processor Symmetric Multiprocessor Distributed Shared Memory



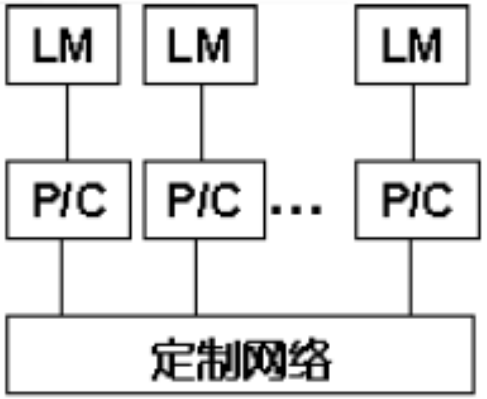
(a) PVP



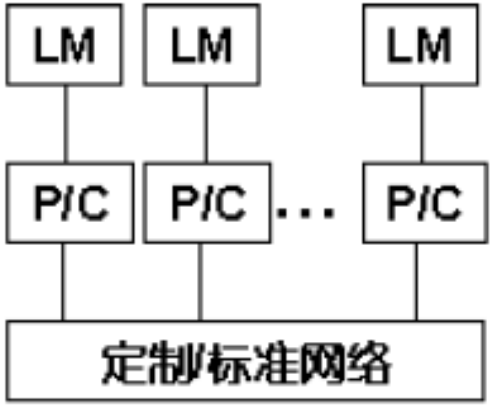
(b) SMP, 物理上单一地址空间



(d) DSM, 逻辑上单一地址空间



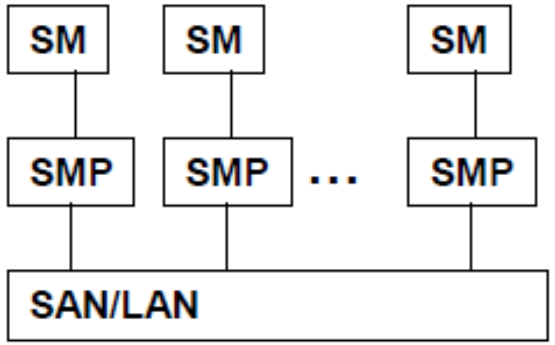
(c) MPP, 物理/逻辑上多地址空间



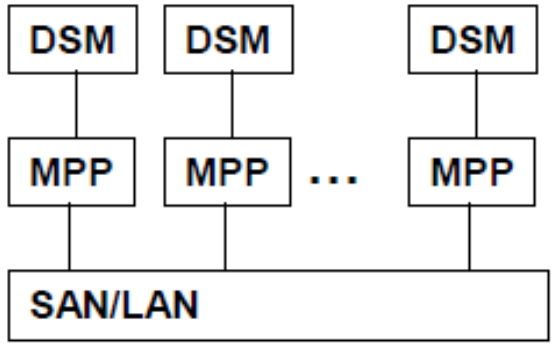
(e) Cluster/COW, 物理/逻辑上多地址空间

Massively Parallel Processor Cluster of Workstations

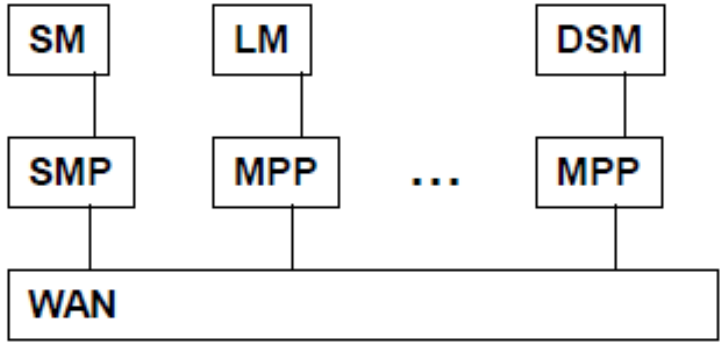
HPC following Hybrid architecture now - Data Center



(f) SMP-Cluster

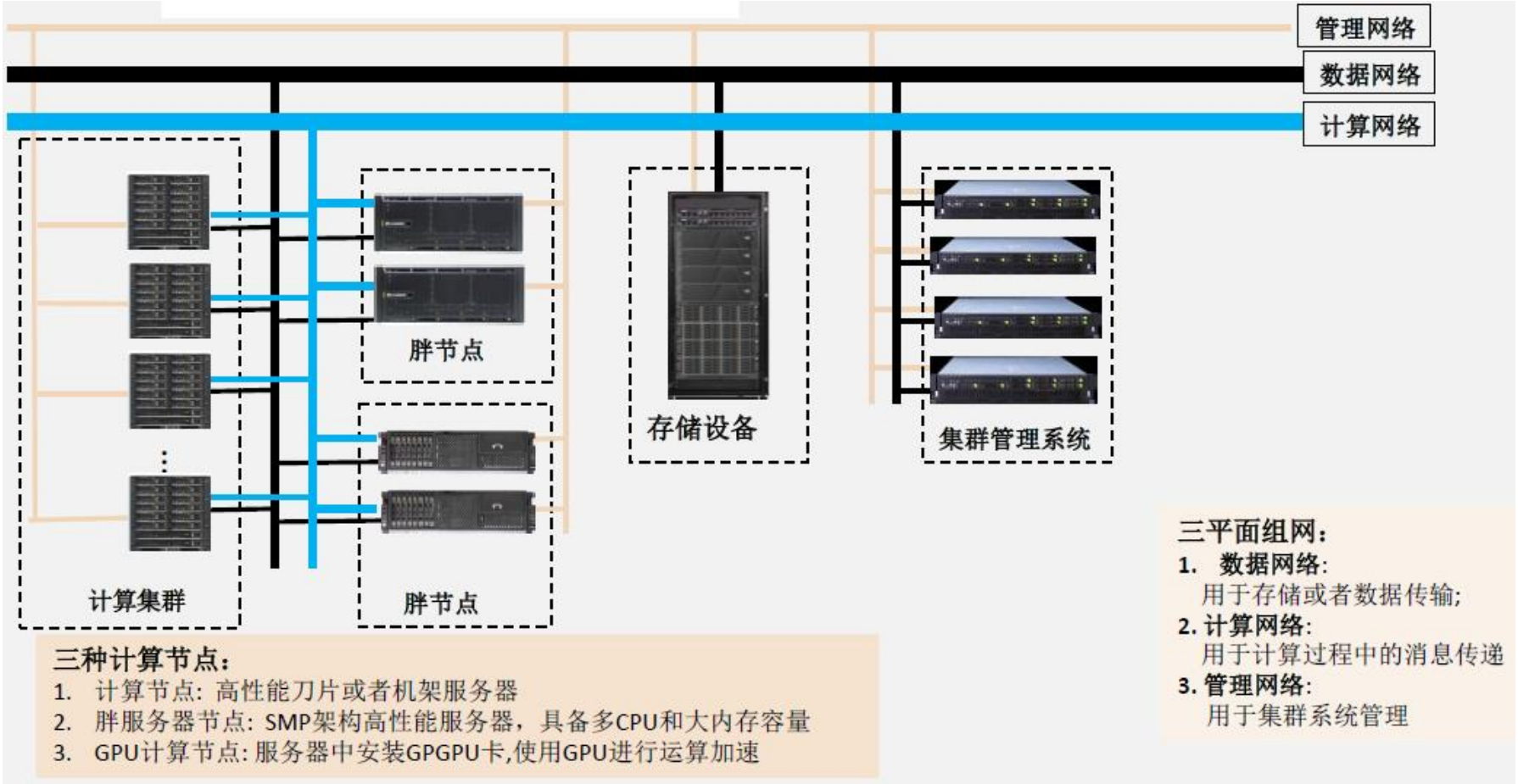


(g) DSM-Cluster



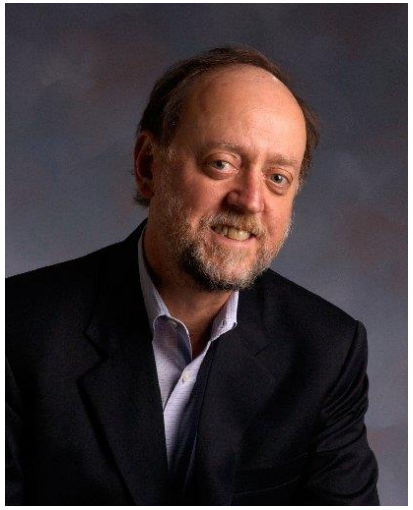
(h) Grid (Cluster of Clusters)

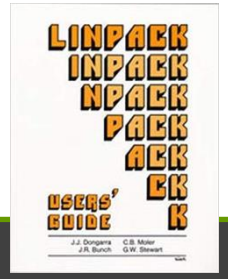
HPC集群系统架构


管理网络
数据网络
计算网络
胖节点
存储设备
集群管理系统
计算集群
胖节点
三种计算节点：
1. 计算节点：高性能刀片或者机架服务器
2. 胖服务器节点：SMP架构高性能服务器，具备多CPU和大内存容量
3. GPU计算节点：服务器中安装GPGPU卡,使用GPU进行运算加速
三平面组网：
1. 数据网络：
用于存储或者数据传输；
2. 计算网络：
用于计算过程中的消息传递
3. 管理网络：
用于集群系统管理

Evolution from Top 500

- **The main objective of the TOP500 list is to provide a ranked list of general purpose systems that are in common use for high end applications**
- **The LINPACK Benchmark**
  - As a yardstick of performance we are using the `best' performance as measured by the **LINPACK** Benchmark, which was introduced by Jack Dongarra, because it is widely used and performance numbers are available for almost all relevant systems.

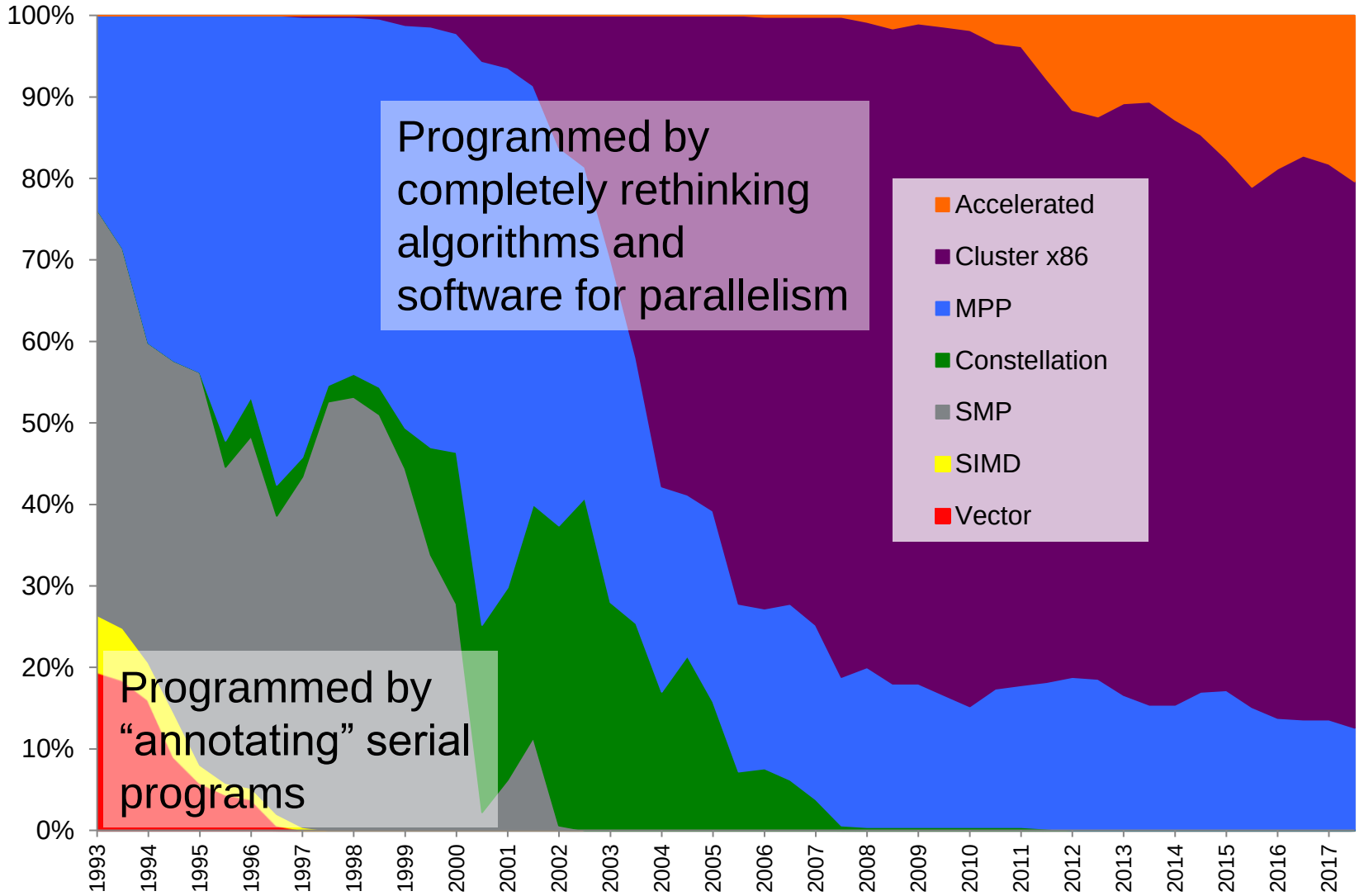




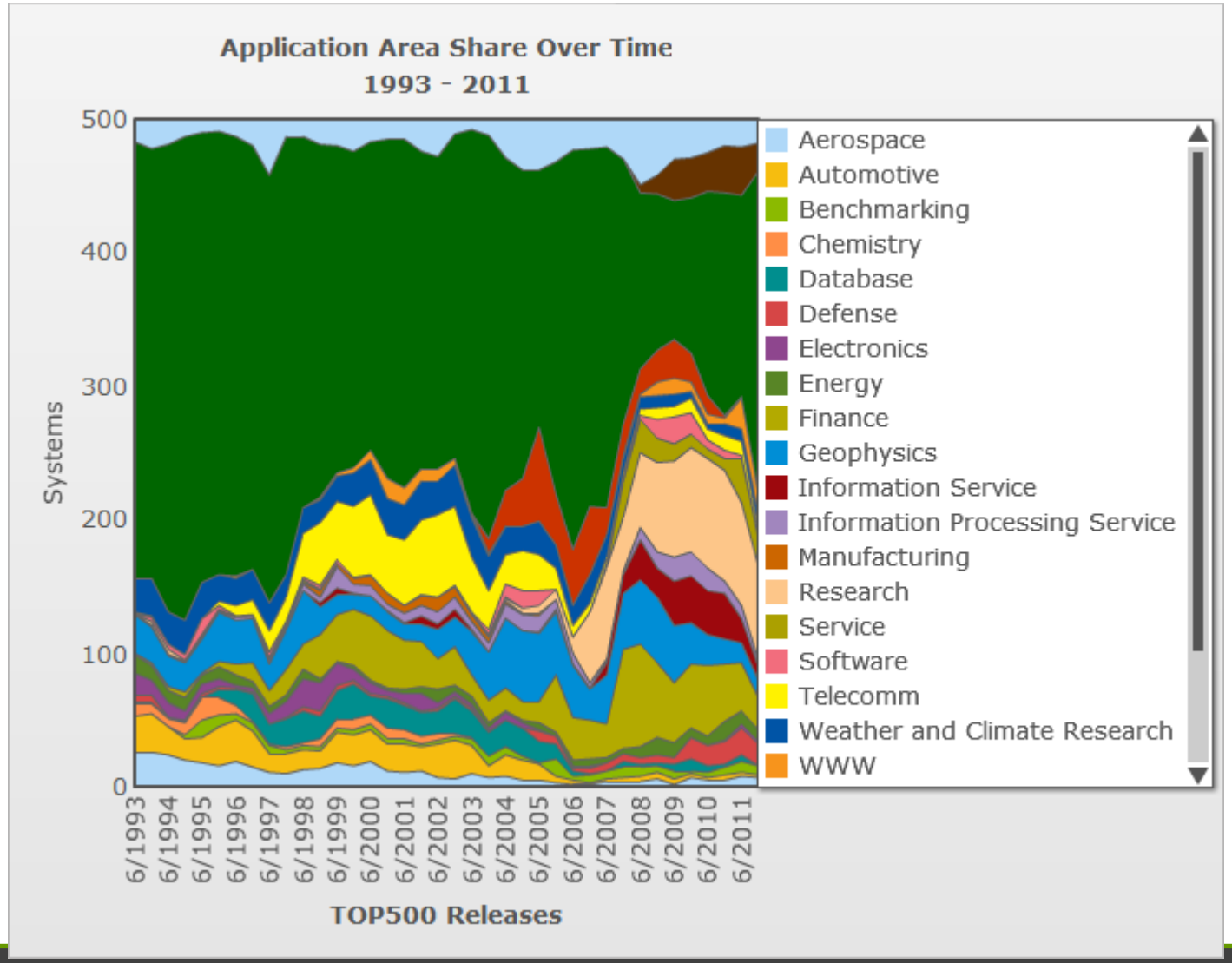
Linpack
Benchmark

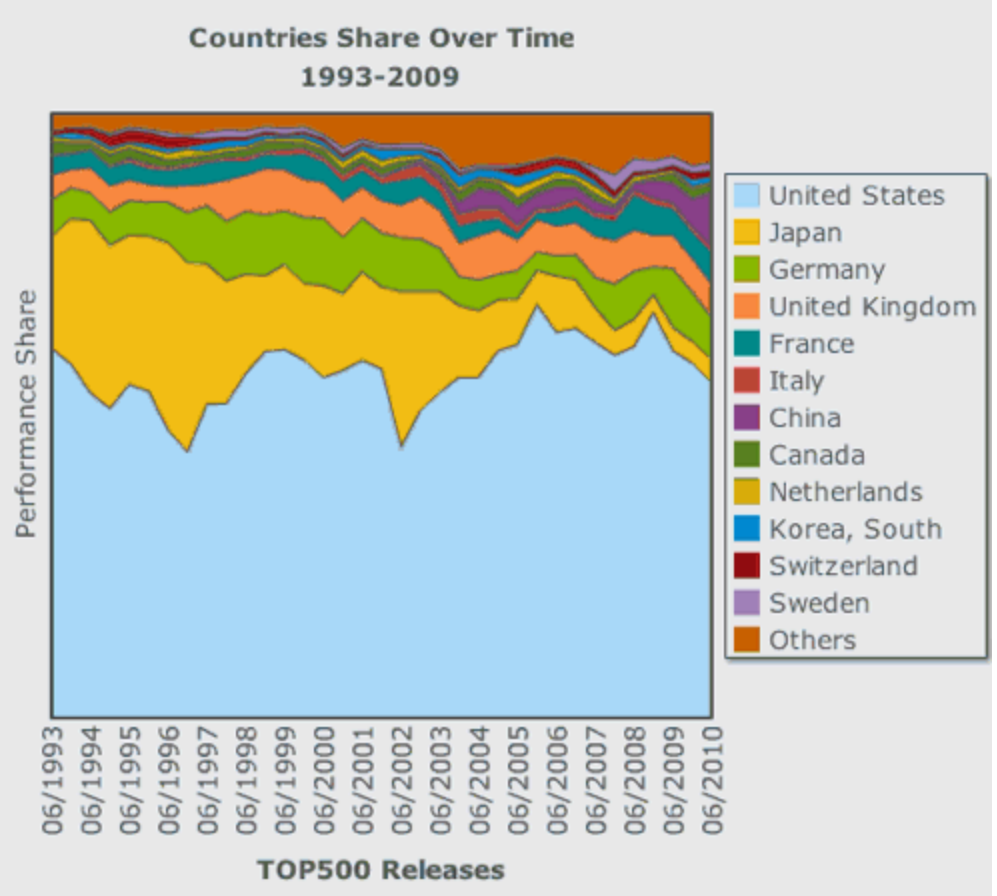
HPL
stands for
High Performance Linpack
ABBREVIATIONS
Abbreviations.com

From Vector Supercomputers to Massively Parallel Accelerator Systems

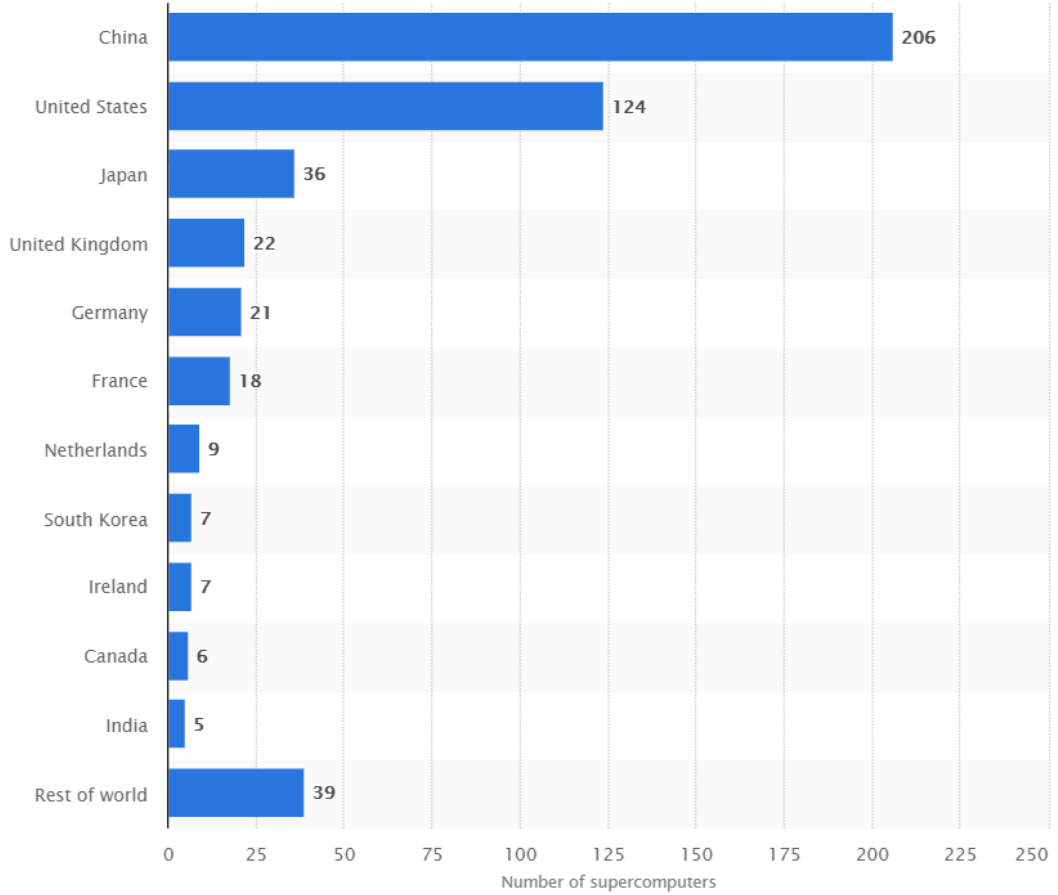


Application Categories

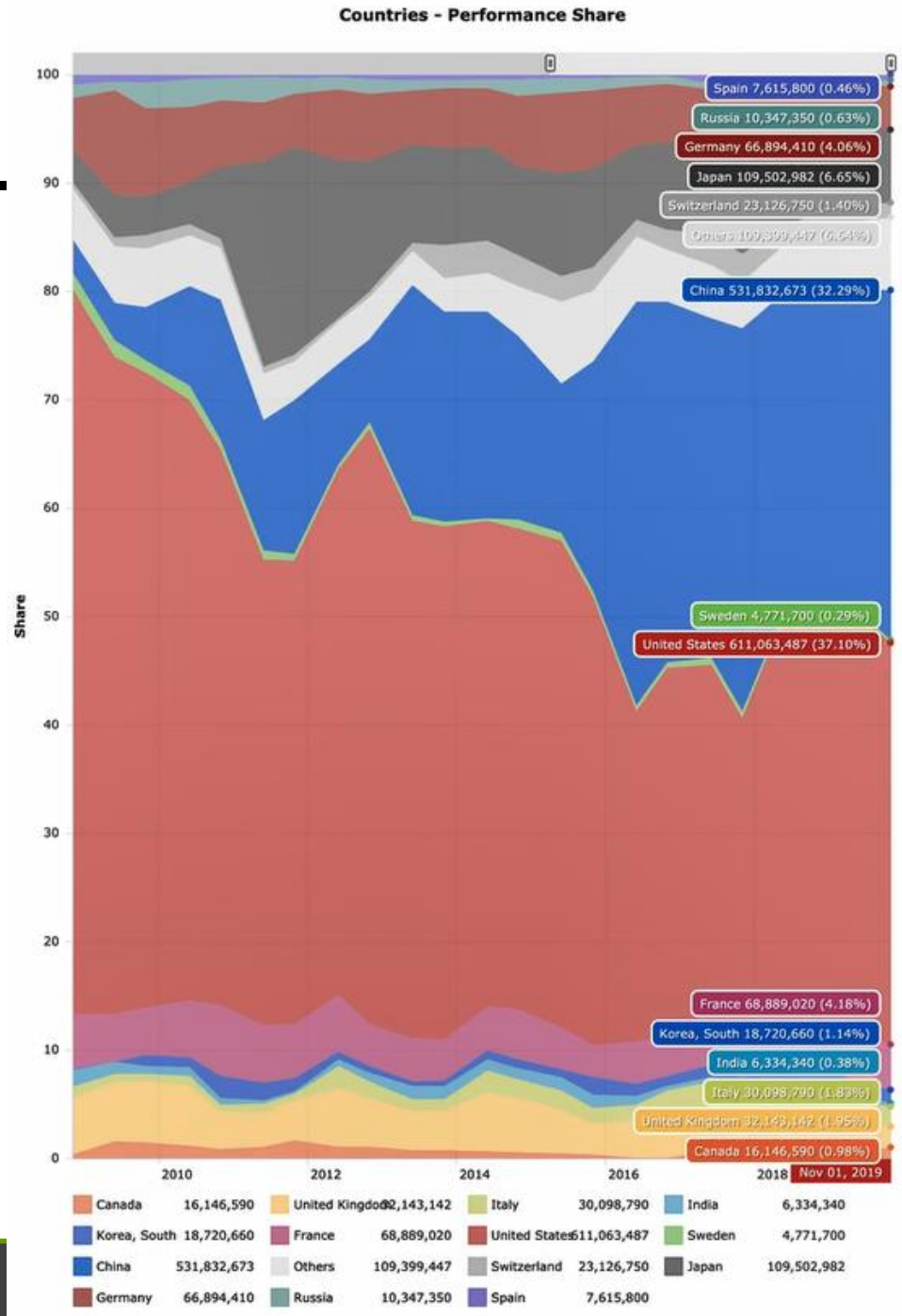




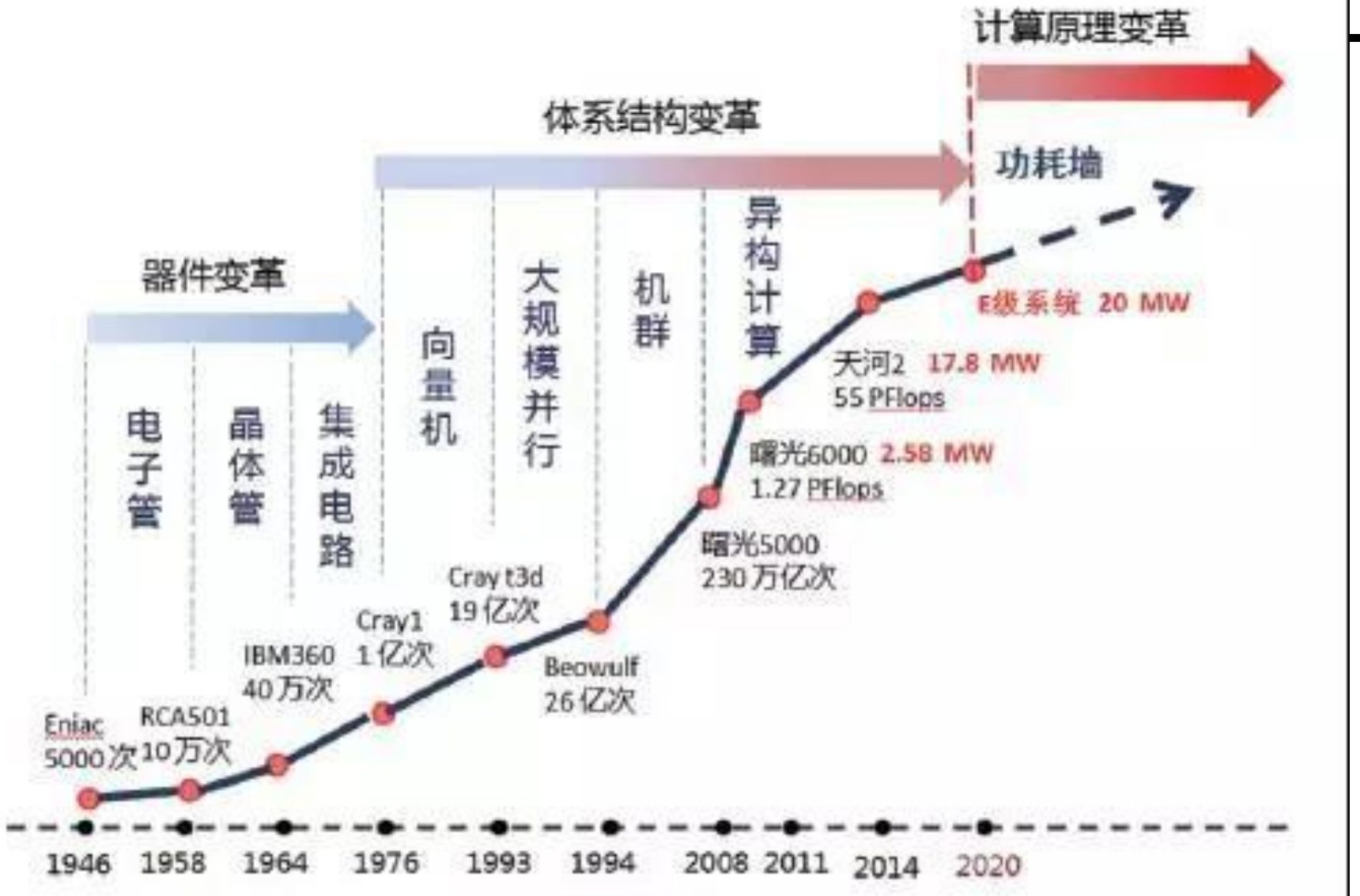
1993 - 2009



2018


Countries - Performance Share
Share
100
90
80
70
60
50
40
30
20
10
0
2010
2012
2014
2016
2018
Nov 01, 2019
Spain 7,615,800 (0.46%)
Russia 10,347,350 (0.63%)
Germany 66,894,410 (4.06%)
Japan 109,502,982 (6.65%)
Switzerland 23,126,750 (1.40%)
China 531,832,673 (32.29%)
Sweden 4,771,700 (0.29%)
United States 611,063,487 (37.10%)
France 68,889,020 (4.18%)
Korea, South 18,720,660 (1.14%)
India 6,334,340 (0.38%)
Canada 16,146,590 (0.98%)
Canada 16,146,590
Korea, South 18,720,660
China 531,832,673
Germany 66,894,410
United Kingdom 32,143,142
France 68,889,020
Others 109,399,447
Russia 10,347,350
Italy 30,098,790
United States 611,063,487
Switzerland 23,126,750
Spain 7,615,800
India 6,334,340
Sweden 4,771,700
Japan 109,502,982

高性能计算机发展历史


计算原理变革
体系结构变革
功耗墙
器件变革
电子管
晶体管
集成电路
向量机
大规模并行
机群
异构计算
E级系统 20 MW
天河2 17.8 MW
55 PFlops
曙光6000 2.58 MW
1.27 PFlops
曙光5000
230万亿次
Cray t3d
19亿次
Cray1
1亿次
IBM360
40万次
RCA501
10万次
Eniac
5000次
Beowulf
26亿次
1946
1958
1964
1976
1993
1994
2008
2011
2014
2020

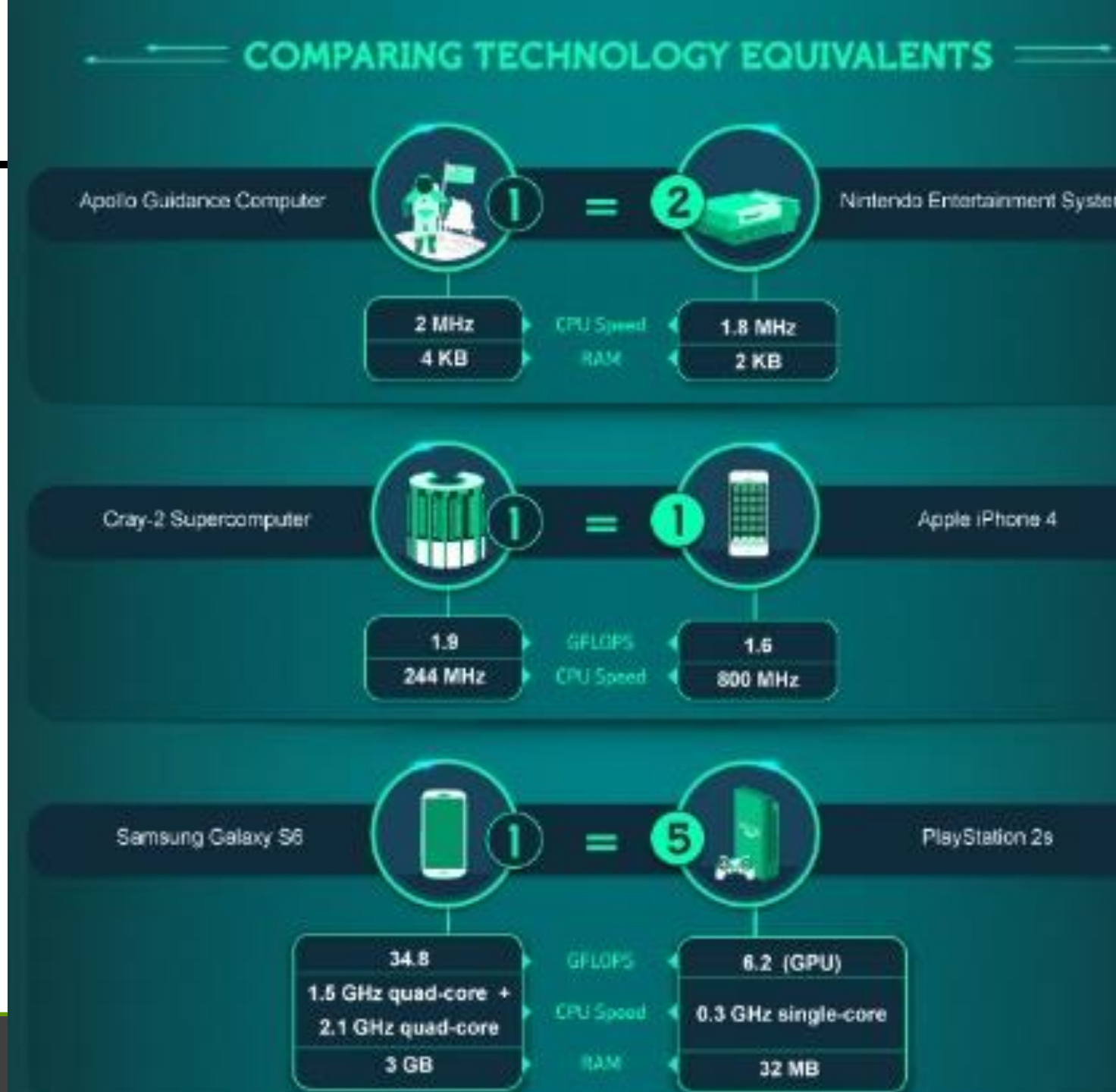
HPC的历史

作为一个很好的计算能力发展的信息回顾路线图，建议你访问以下网页：

http://pages.experts-exchange.com/processing-power-compared/。

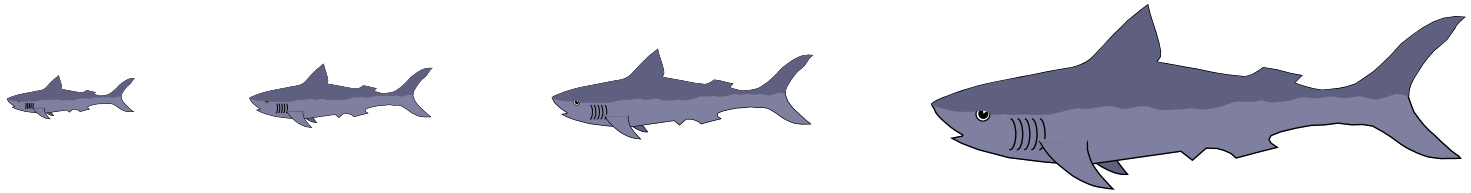
它很好地展示了这样的问题，例如2010年发布的iPhone 4是如何与1985年具有每秒10^9次浮点运算的Cray 2超级计算机表现近乎相当的；2015年发布的苹果手表如何大概具有iPhone 4和Cray 2性能的两倍！

虽然芯片制造商已经设法保护著名的摩尔定律，该定律预测的晶体管数量每两年增加一倍，但在单个芯片中有大约100个复杂的处理核心，如今在芯片制造中是14纳米（nm）。2015年7月，IBM宣布原型芯片为7nm（宽度是人头发的万分之一）。有些科学家表明量子隧穿效应将在5nm处产生影响（Intel预期在2020年走向市场），尽管有些研究人员已经说明在实验室中像石墨烯这样的材料单个晶体结构仅是1nm那么小，但相比于如今的芯片大小，在一个芯片包中放

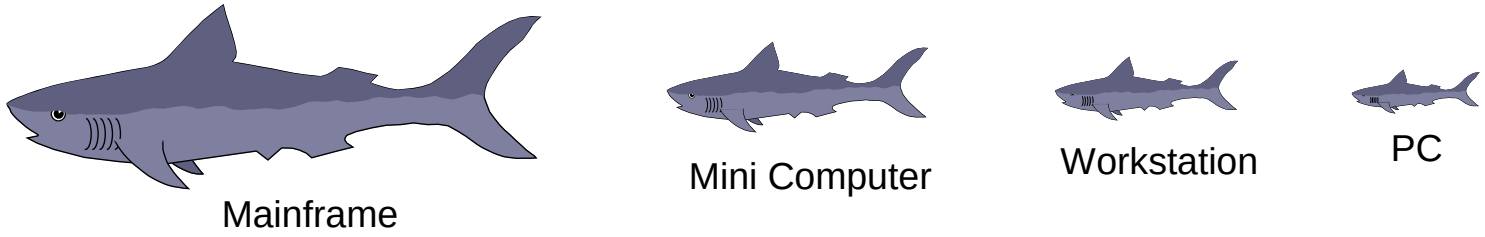


Interesting Food Chain in IT

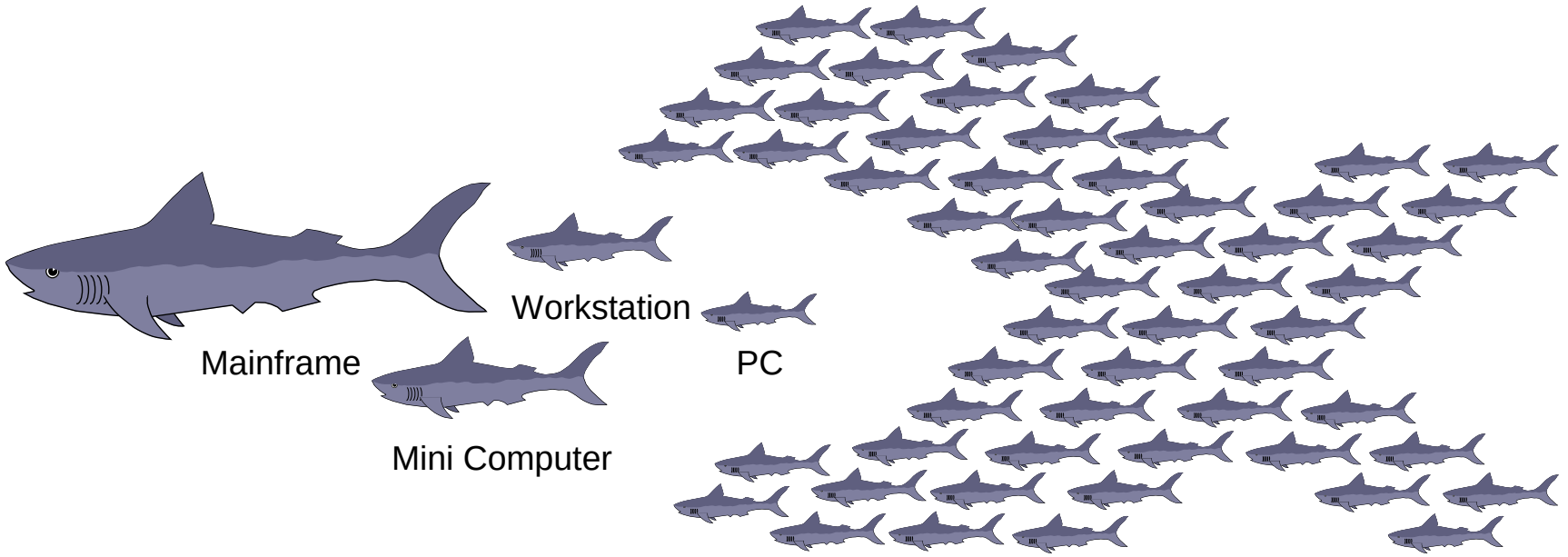
- **Traditional Food Chain**



- **Food Chain of Computer**



Food Chain of Distributed Computing



Gordon Bell Prizes: Science at Scale

Established in 1987 with a cash award of $10,000 (since 2011), funded by Gordon Bell, a pioneer in HPC.
For innovation in applying *HPC to applications in science, engineering, and data analytics*.

China 1st wined Gordon 2016

- **A Chinese team on Friday won the 2016 ACM Gordon Bell prize, a top honor in high-performance computing, for an application running on China's fastest supercomputer.**
  - It is the first time a Chinese team has won the award.

The project, named "10M-Core Scalable Fully-Implicit Solver for Nonhydrostatic Atmospheric Dynamics," presents a method for calculating atmospheric dynamics, according to the Association for Computing Machinery, which presented the award at the International Supercomputing Conference in Salt Lake City in the United States

- **2017 ACM Gordon Bell Prize awarded to Chinese team led by Tsinghua on Nonlinear Earthquake Simulation employing the world's fastest supercomputer**
  - 18.9-Pflops Nonlinear Earthquake Simulation on Sunway TaihuLight: Enabling Depiction of 18-Hz and 8-Meter Scenarios
- **2020年11月19日，全球高性能计算（HPC）最高奖——戈登贝尔奖（Gordon Bell Prize）正式揭晓。智源青年科学家、北京应用物理与计算数学研究所副研究员王涵所在团队，凭借其在“HPC+AI+第一性原理分子动力学”方向的工作：“Pushing the limit of molecular dynamics with ab initio accuracy to 100 million atoms with machine learning”，成功将本年度奖项收入囊中**
  - **这也是戈登贝尔奖历史上，中国团队第三次获奖。**
    - 智源学者中，杨超（时任中科院研究员，现为北京大学教授）团队在2016年也曾凭借“千万核可扩展大气动力学全隐式模拟”研究成果成为首支获得该奖的中国团队，一举打破美国、日本在该奖项上近30年的垄断，实现我国高性能计算应用戈登贝尔奖零突破

2021

https://www.hpcwire.com/2021/11/18/2021-gordon-bell-prize-goes-to-exascale-powered-quantum-supremacy-challenge/

2021 Gordon Bell Prize Goes to Exascale-Powered Quantum Supremacy Challenge

By Oliver Peckham

November 18, 2021

Today at the hybrid virtual/in-person SC21 conference, the organizers announced the winners of the 2021 ACM Gordon Bell Prize: a team of Chinese researchers leveraging the new exascale Sunway system to simulate quantum circuits.

Winner of the 2021 ACM Gordon Bell Prize

Closing the "Quantum Supremacy" Gap: Achieving Real-Time Simulation of a Random Quantum Circuit Using a New Sunway Supercomputer

Yong (Alexander) Liu, Xin (Lucy) Liu, Fang (Nancy) Li, Haohuan Fu, Yuling Yang, Jiawei Song, Pengpeng Zhao, Zhen Wang, Dajia Peng, Huarong Chen, Chu Guo, Heliang Huang, Wenzhao Wu and Dexun Chen

The fourteen researchers (whose affiliations span Zhejiang Lab, Tsinghua University, the National Supercomputing Center in Wuxi and the Shanghai Research Center for Quantum Sciences) leveraged the massive new Sunway exascale system that was more or less revealed during SC21 to conduct groundbreaking simulation of a quantum circuit.

"With Google's "Quantum Supremacy" declaration in 2019, stating that the Sycamore superconductive quantum computer is over a billion times faster than Summit (comparing 200 seconds against 10,000 years in the task of measuring/simulating one million samples), the dawn of the quantum age starts to unfold in a more affirmative way," the researchers wrote. "A later response from the IBM research team argues that they can accomplish the simulation on the classical Summit supercomputer … within a few days instead of 10,000 years."

https://awards.acm.org/bell/award-winners

戈登贝尔奖 [编辑]

维基百科，自由的百科全书

戈登贝尔奖（ACM Gordon Bell Prize）美国计算机协会设立于1987年，每年颁发，是一种超级电脑应用软件设计奖，奖金象征性1万美元。

戈登贝尔奖通常会由当年前500排行名列前茅的超级电脑系统之上所跑的的应用软件获得[1]，例如美国"泰坦"超级电脑、日本"京"超级电脑上的应用软件都曾得奖，从设立后30多年来都由美国和日本软件获得此奖，直到2016年中国打破此一规律[2]由神威·太湖之光超级电脑上的"全球大气非静力云分辨模拟"应用软件得奖。

奖项分为

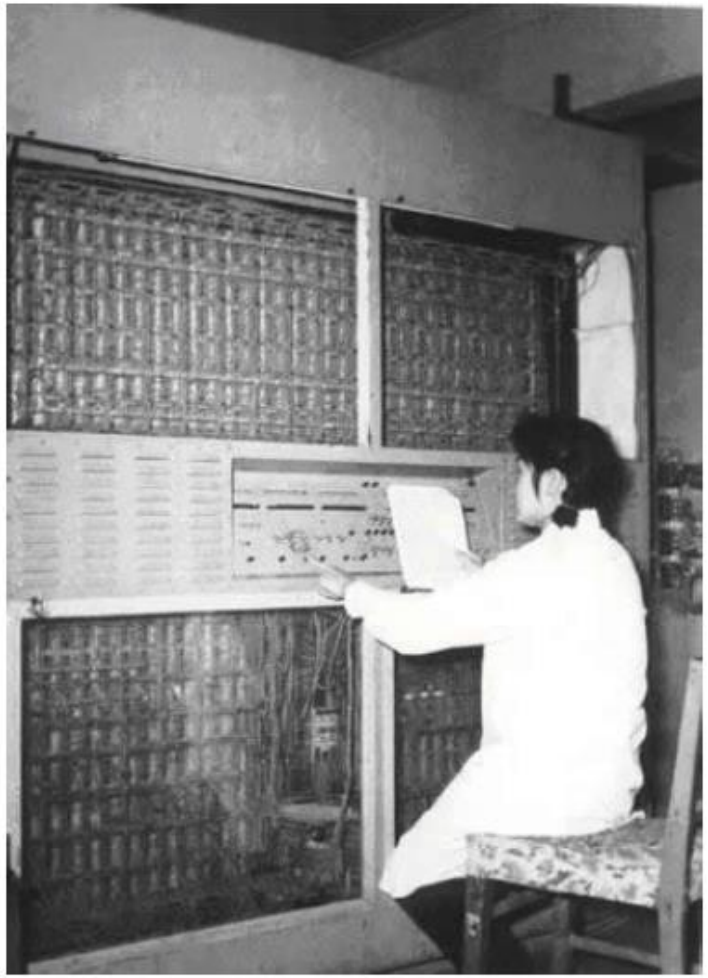
- 最高性能奖 (Peak Performance)
- 最高性价比奖(Price/Performance)
- 特别奖 (Special Achievement)

外部链接 [编辑]

- Gordon Bell Prize - Award Winners: List By Year (页面存档备份，存于互联网档案馆)
- Gordon Bell Prize description from SC13
- ACM Gordon Bell Prize Winners 2006-present
- Earlier Prize Winners 1987–1999 (页面存档备份，存于互联网档案馆)

我国第一台计算机

- 1958 年第一台国产计算机诞生 — 103型计算机



103型，运行速度 1500次每秒

109丙机（1968），两弹一星“功勋机”

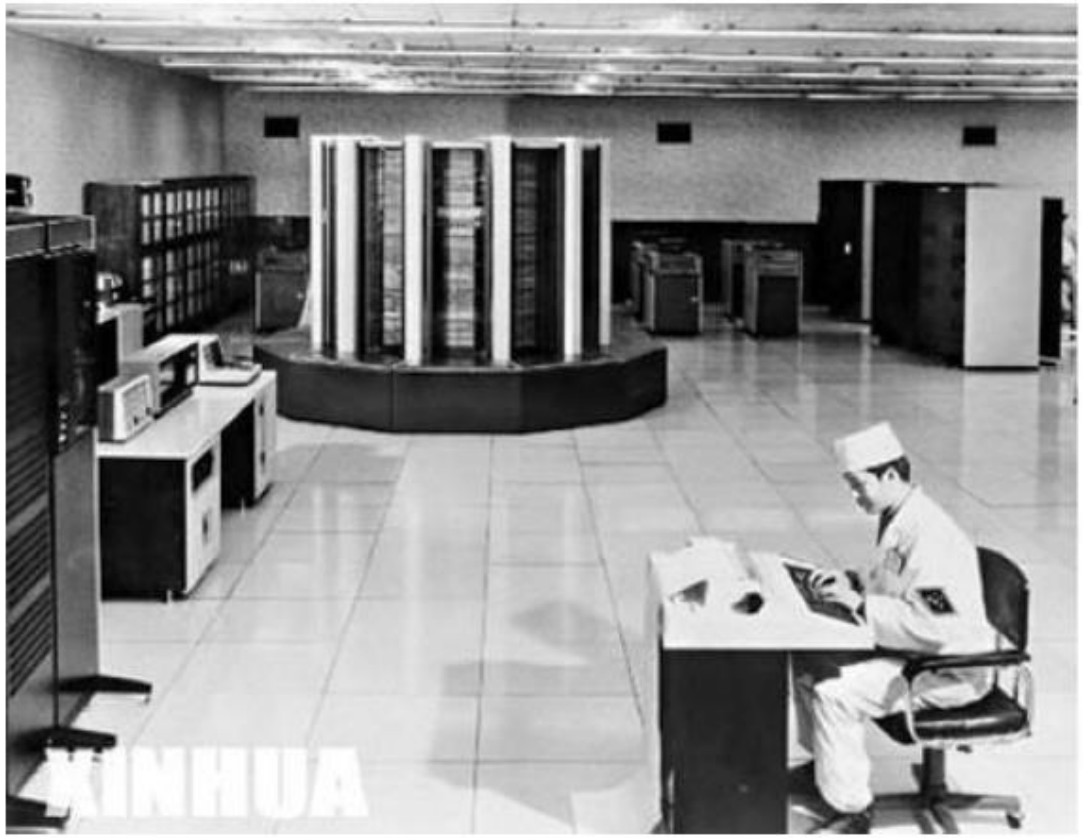
我国的并行机

- 我国的超级计算机系列
  - 银河：国防科大
  - 神威：国家并行计算机工程技术研究中心
  - 曙光：中科院计算所，曙光公司，上海超算中心
  - 深腾：联想集团
  - 天河：国防科大

银河

PVP (Parallel Vector Processor)

- 银河一号： 1983年12月，我国第一台每秒运算达1亿次以上的计算机



- 1992年， “银河-II” 问世，每秒运算达10亿次
- 1997年， “银河-III”问世，每秒运算达130亿次

Our Endeavors for HPC: 曙光系列

- 曙光一号 SMP (1993)
- 曙光1000 MPP (1995)
- 曙光2000I Cluster (1998)
- 曙光2000II Cluster (1999)
- 曙光3000 Cluster (2000)
- 曙光4000l Cluster (2002)
- 曙光4000A Cluster (2004)

曙光一号　　SMP (1993)

“曙光一号”诞生后仅3天，西方国家便宣布解除10亿次计算机对中国的禁运。

曙光

- 1993年10月，国家智能计算机研究开发中心（后成立曙光信息产业公司）研制成功“曙光一号”SMP多处理机
- 2004年6月，推出 11万亿次的曙光4000A超级计算机，Top500排名第十
- 2008年6月，曙光5000A发布，实际速度超160万亿次，Top500排名第十
- 2011年6月，曙光6000，实际速度超1200万亿次，Top500排名第四

曙光 6000

神威

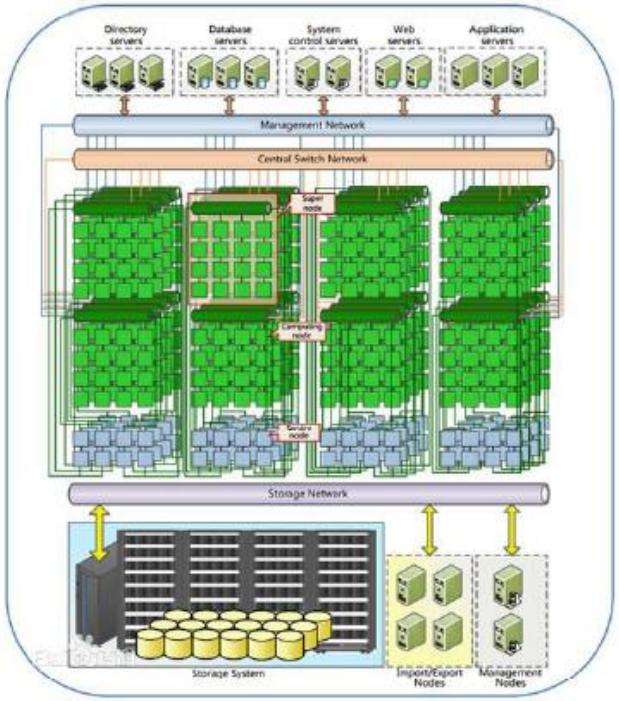
- 神威 I

  1999年，国家并行计算机工程技术研究中心牵头研制，峰值3840亿次/秒

- 神威·太湖之光

  2016年，峰值运算速度达每秒**12.5**亿亿次

  自2016年6月起，连续四次 Top500 排名第一，目前排名第三，国内第一



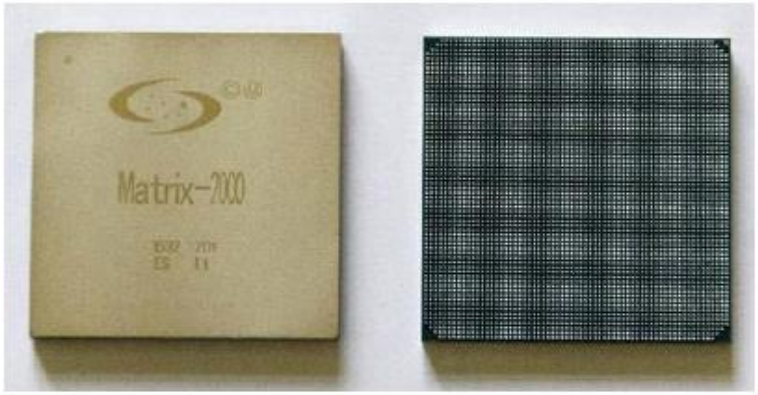
深腾

- 2002年，联想发布深腾1800，Top500排名**43**，成为首家正式进入排行榜前100的中国企业
- 2003年，深腾6800发布，Top500排名**14**，其78.5%的整机效率列世界通用高端计算机**第一名**
- 2008年12月，联想发布百万亿次超级计算机深腾7000，Top500排名**19**

深腾7000

天河

- 2009年10，中国首台千万亿次超级计算机“天河一号”诞生，使中国成为继美国之后世界上第二个能够研制千万亿次超级计算机的国家。
- 2010年10月，升级后的天河一号排名Top500第一
- 2013年6月，天河二号再次问鼎世界第一，功耗达24兆瓦，也是目前TOP500里功耗最大的
- 2017年7月，使用**国产加速器 Matrix 2000**的天河二号问世，在17年11月的Top500排名位列第2（第一是太湖之光），目前排名第4



The Modern Era – 2010
Cluster/MPP of CPU+GPU

- **Tianhe-1A**
  - First Chinese supercomputer to top 500!
  - Hybrid design – **mix of CPU and GPU**
    - Implications for usage beyond LINPACK?
    - See later lectures for GPGPU programming
  - Blade system with 14,336 Intel Xeon X5670 CPUs and 7,168 Nvidia Tesla M2050 GPGPUs connected by Infiniband
  - Cost $88m and requires 4MW to operate
  - See http://www.nscc-tj.gov.cn/en for more details

Tianhe-1A uses 7000+ NVIDIA GPUs

- Tianhe-1A uses
  - 7,168 NVIDIA Tesla M2050 GPUs
  - 14,336 Intel Westmeres
- Performance
  - 4.7 PF peak
  - 2.5 PF sustained on HPL
- 4.04 MW
  - If Tesla GPU's were not used in the system, the whole machine could have needed 12 megawatts of energy to run with the same performance, which is equivalent to 5000 homes
- Custom fat-tree interconnect
  - 2x bandwidth of Infiniband QDR

The New York Times

Business Day
Technology

WORLD U.S. N.Y. REGION BUSINESS TECHNOLOGY SCIENCE HEALTH SPORTS OPINION

Search Technology

Inside Technology
Internet Start-Ups Business Computing Companies

Bits
Blog

China Wrests Supercomputer Title From U.S.

By ASHLEE VANCE

A Chinese scientific research center has built the fastest supercomputer ever made, replacing the United States as maker of the swiftest machine, and giving China bragging rights as a technology superpower.

RECOMMEND
TWITTER
SIGN IN TO E-MAIL
PRINT
REPRINTS
SHARE

Enlarge This Image

The Tianhe-1A computer in Tianjin, China, links thousands upon thousands of chips.

The computer, known as Tianhe-1A, has 1.4 times the horsepower of the current top computer, which is at a national laboratory in Tennessee, as measured by the standard test used to gauge how well the systems handle mathematical calculations, said Jack Dongarra, a University of Tennessee computer scientist who maintains the official supercomputer rankings.

Although the official list of the top 500 fastest machines, which comes out every six months, is not due to be completed by Mr. Dongarra until next week, he said the Chinese computer "blows away the existing No. 1 machine." He added, "We don't close the books until Nov. 1, but I would say it is unlikely we will see a system that is faster."

Tianhe-1A

2013 TianHe-2/Milkway-2

天河二号（Tianhe-2/MilkyWay-2）·建造情况

网站	http://www.nscc-gz.cn/
制造商	国防科大（NUDT） & Inspur
位置	国家超算中心（广州）NSCC-GZ
占地面积	720 m^2
研发人员	约1300人
耗资	$390 million
首次进入Top500	2013.6
概况	170个机柜，包括125个计算机柜、8个服务机柜、13个通信机柜、24个存储机柜

天河二号（Tianhe-2/MilkyWay-2）·机器情况

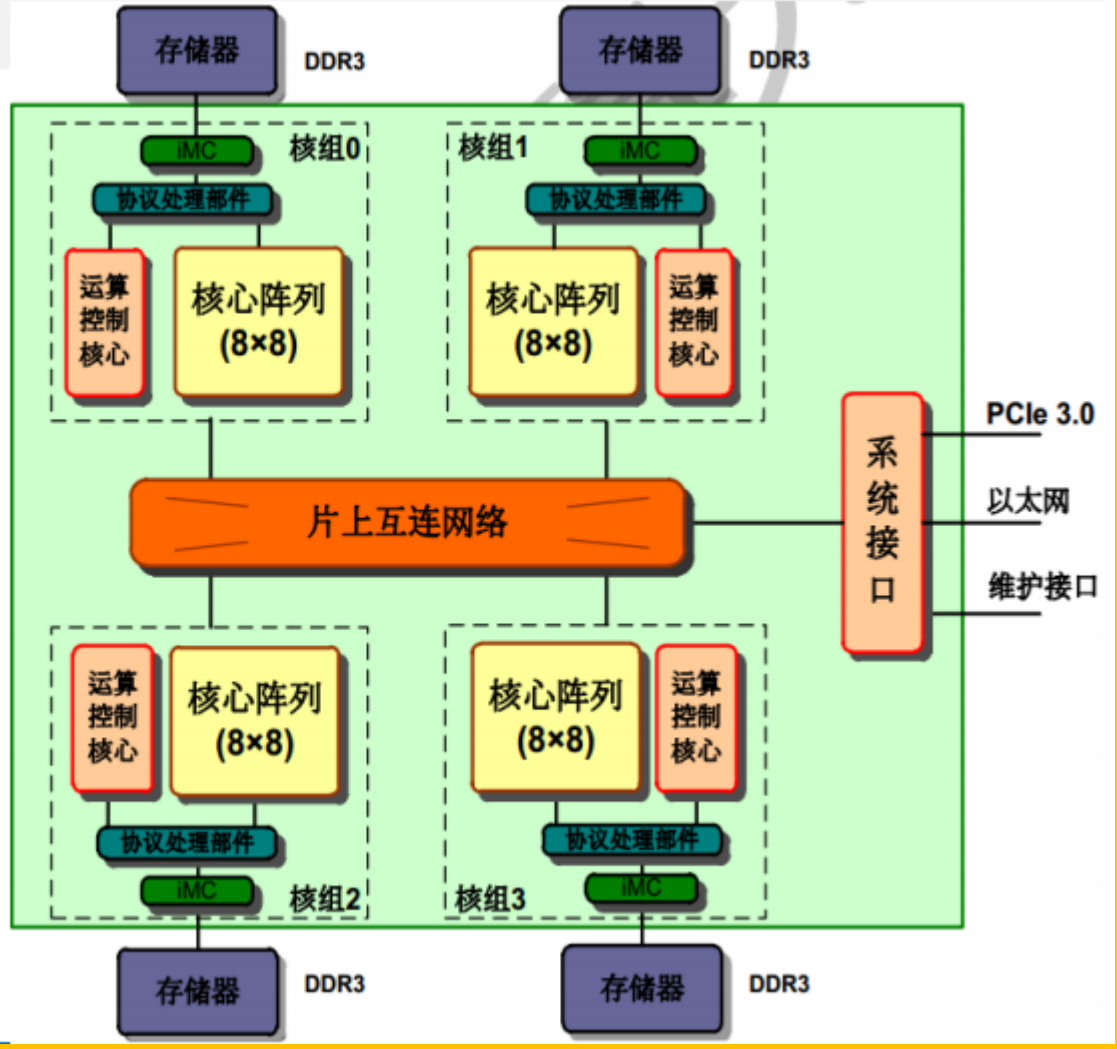
Hardware	
Architecture	TH-IVB-FEP Cluster
Processor	Intel Xeon E5-2692v2 12C 2.2GHz
Accelerator	Intel Xeon Phi 31S1P
Cores	3,120,000
Memory	1,024,000 GB
Interconnect	TH Express-2

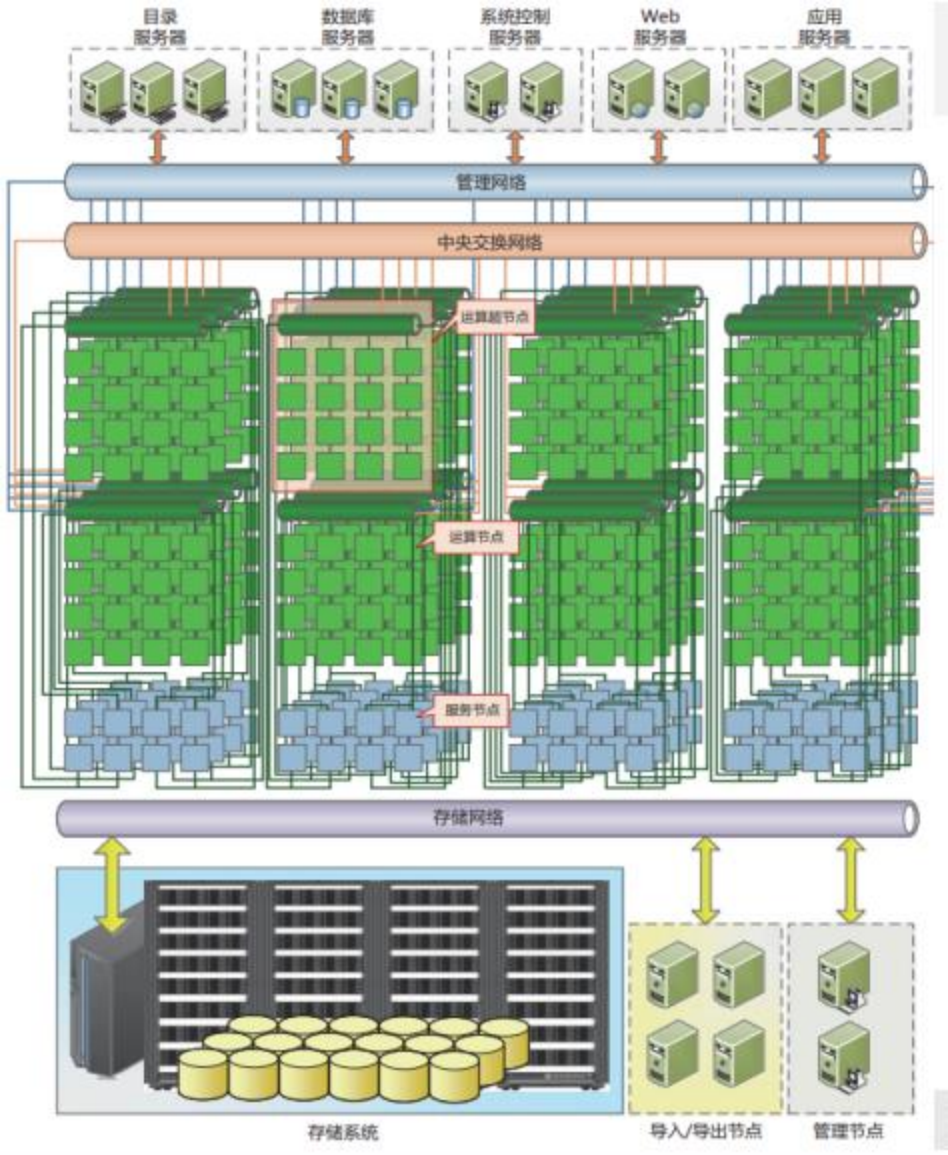
Software	
Operating System	Kylin Linux
Compiler	icc
Math Library	Intel MKL-11.0.0

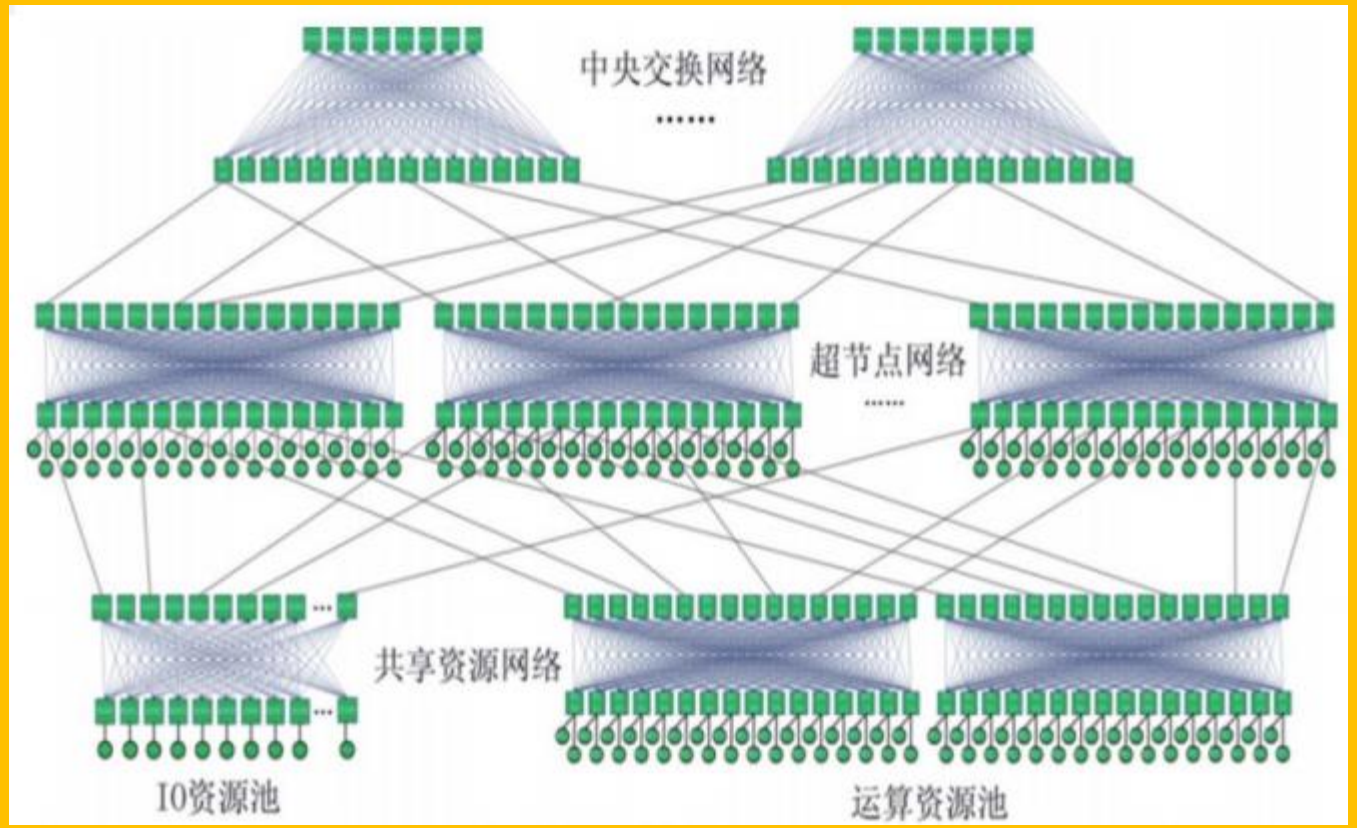
Power Consumption	
Power	17,808.00 kW

Performance	
Linpack Performance(Rmax)	33,862.7 TFlop/s
Theoretical Peak (Rpeak)	54,902.4 TFlop/s
Nmax	9,960,000

“申威 26010”
众核处理器




目录
服务器
数据库
服务器
系统控制
服务器
Web
服务器
应用
服务器
管理网络
中央交换网络
运算加节点
运算节点
服务节点
存储网络
存储系统
导入/导出节点
管理节点


中央交换网络
超节点网络
共享资源网络
IO资源池
运算资源池


并行应用
并行开发环境
· 集成开发环境
· 并行调试
· 性能监测
并行语言及编译环境
· OpenACC
· MPI
· OpenMP
国产众核CPU基础软件
基础编译系统
· C / C++, Fortran
· SIMD 扩展接口
· 异构代码生成
基础函数库
· C 库
· 加速线程库
· 数学库
自动向量化系统
· C / C++, Fortran
· 循环级向量化
· 向量代码优化
并行操作系统环境
· 作业管理
· 容错管理
· 资源管理
· 系统开工
· 功耗管理
· 安全管理
· 网络管理
高性能存储管理系统
· SWGFS并行文件系统
· LWFS 轻量级文件系统
· 存储管理平台
“神威·太湖之光”计算机系统

The Modern Era – 2010
Cluster/MPP of CPU+GPU

- **Tianhe-1A**
  - First Chinese supercomputer to top 500!
  - Hybrid design – **mix of CPU and GPU**
    - Implications for usage beyond LINPACK?
    - See later lectures for GPGPU programming
  - Blade system with 14,336 Intel Xeon X5670 CPUs and 7,168 Nvidia Tesla M2050 GPGPUs connected by **Infiniband**
  - Cost $88m and requires 4MW to operate
  - See http://www.nscc-tj.gov.cn/en for more details

Tianhe-1A uses 7000+ NVIDIA GPUs

- Tianhe-1A uses
  - 7,168 NVIDIA Tesla M2050 GPUs
  - 14,336 Intel Westmeres
- Performance
  - 4.7 PF peak
  - 2.5 PF sustained on HPL
- 4.04 MW
  - If Tesla GPU's were not used in the system, the whole machine could have needed 12 megawatts of energy to run with the same performance, which is equivalent to 5000 homes
- Custom fat-tree interconnect
  - 2x bandwidth of Infiniband QDR

The New York Times

Business Day
Technology

WORLD U.S. N.Y. REGION BUSINESS TECHNOLOGY SCIENCE HEALTH SPORTS OPINION

Search Technology

Inside Technology
Internet Start-Ups Business Computing Companies

Bits Blog

China Wrests Supercomputer Title From U.S.

By ASHLEE VANCE

A Chinese scientific research center has built the fastest supercomputer ever made, replacing the United States as maker of the swiftest machine, and giving China bragging rights as a technology superpower.

RECOMMEND
TWITTER
SIGN IN TO E-MAIL
PRINT
REPRINTS
SHARE

Enlarge This Image

The Tianhe-1A computer in Tianjin, China, links thousands upon thousands of chips.

The computer, known as Tianhe-1A, has 1.4 times the horsepower of the current top computer, which is at a national laboratory in Tennessee, as measured by the standard test used to gauge how well the systems handle mathematical calculations, said Jack Dongarra, a University of Tennessee computer scientist who maintains the official supercomputer rankings.

Although the official list of the top 500 fastest machines, which comes out every six months, is not due to be completed by Mr. Dongarra until next week, he said the Chinese computer "blows away the existing No. 1 machine." He added, "We don't close the books until Nov. 1, but I would say it is unlikely we will see a system that is faster."

2013 TianHe-2/Milkway-2

天河二号（Tianhe-2/MilkyWay-2）·建造情况

网站	http://www.nscc-gz.cn/
制造商	国防科大（NUDT） & Inspur
位置	国家超算中心（广州）NSCC-GZ
占地面积	720 m^2
研发人员	约1300人
耗资	$390 million
首次进入Top500	2013.6
概况	170个机柜，包括125个计算机柜、8个服务机柜、13个通信机柜、24个存储机柜

天河二号（Tianhe-2/MilkyWay-2）·机器情况

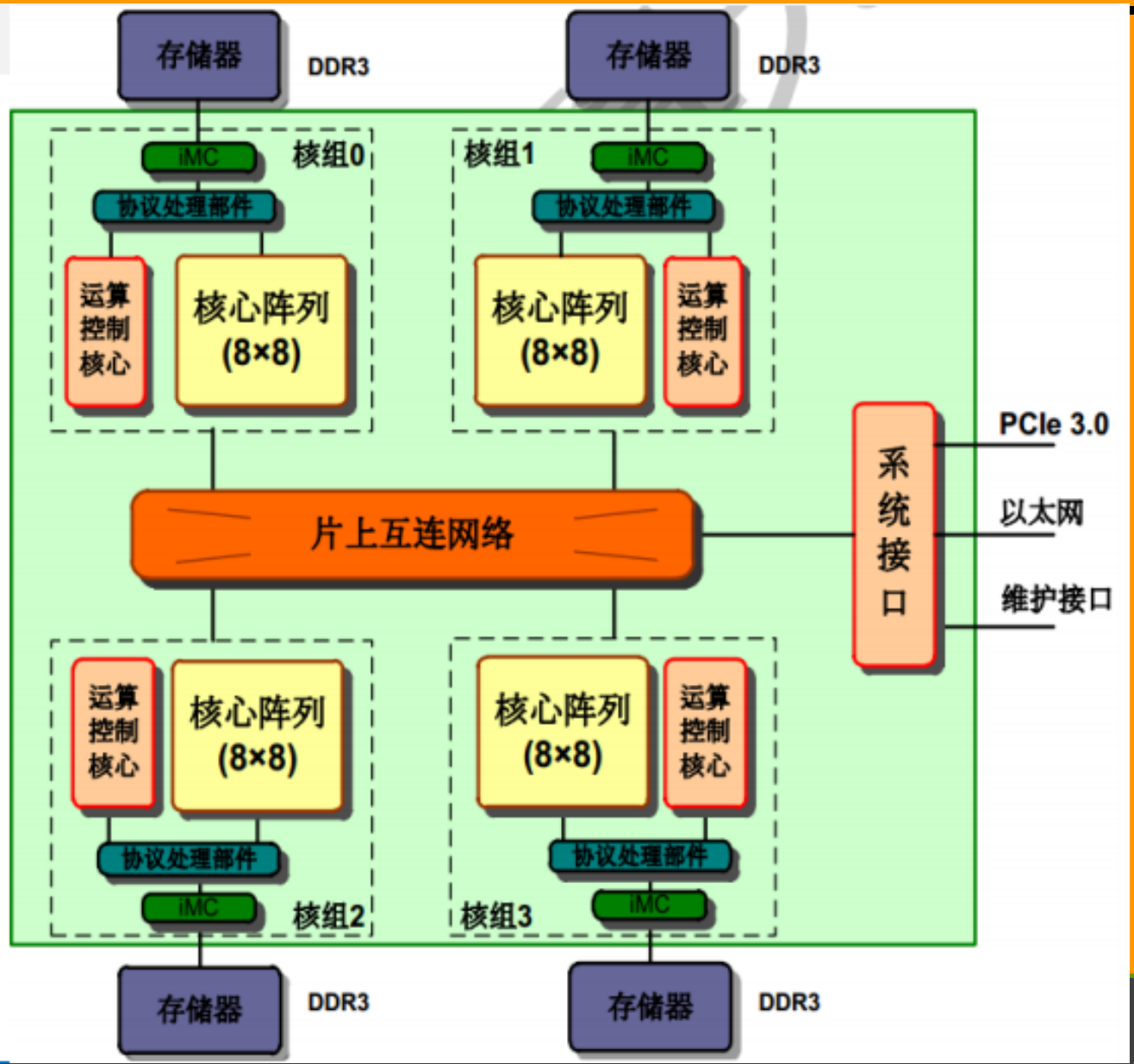
Hardware	
Architecture	TH-IVB-FEP Cluster
Processor	Intel Xeon E5-2692v2 12C 2.2GHz
Accelerator	Intel Xeon Phi 31S1P
Cores	3,120,000
Memory	1,024,000 GB
Interconnect	TH Express-2

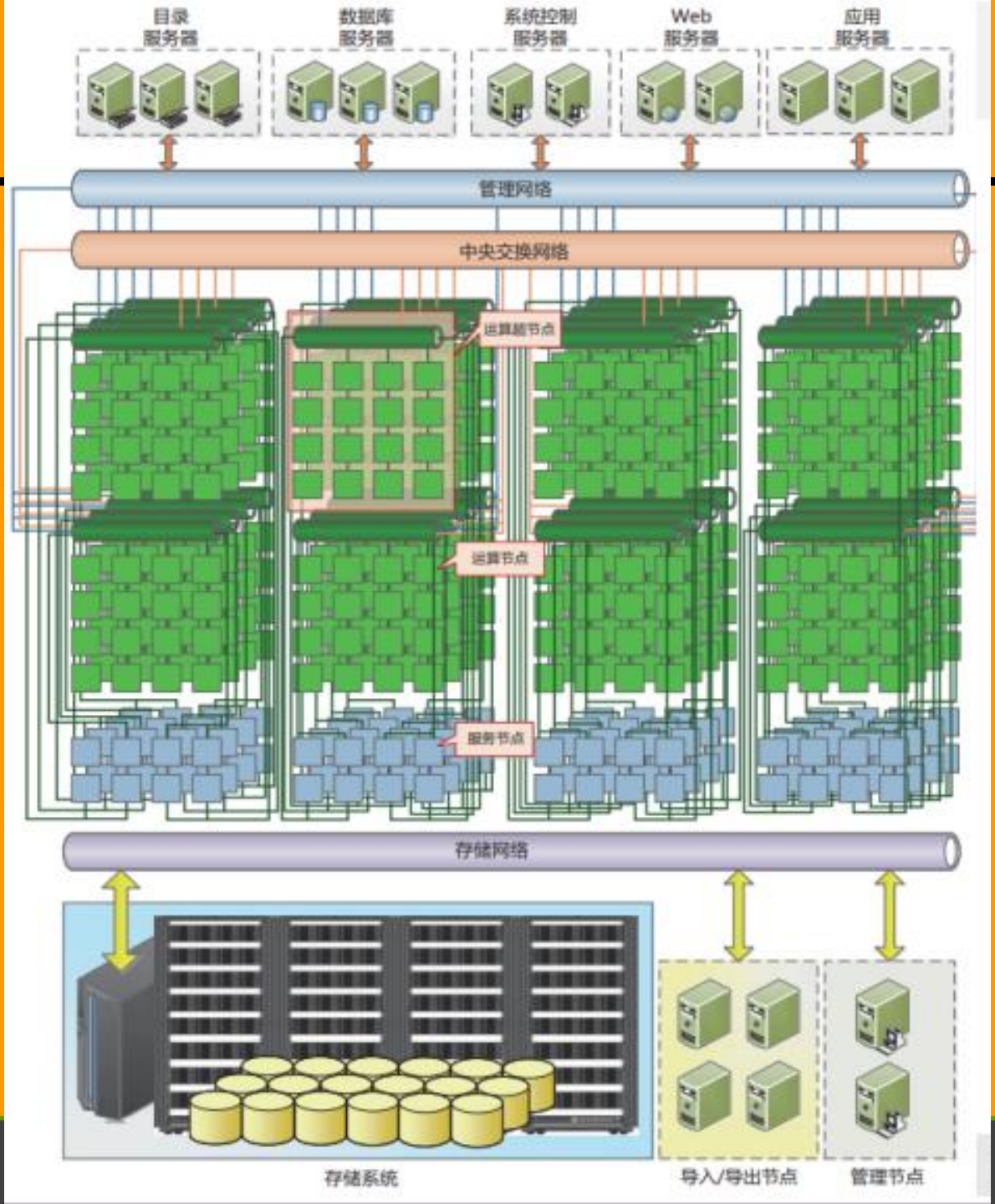
Software	
Operating System	Kylin Linux
Compiler	icc
Math Library	Intel MKL-11.0.0

Power Consumption	
Power	17,808.00 kW

Performance	
Linpack Performance(Rmax)	33,862.7 TFlop/s
Theoretical Peak (Rpeak)	54,902.4 TFlop/s
Nmax	9,960,000

“申威 26010”
众核处理器





并行应用

并行开发环境

· 集成开发环境 · 并行调试 · 性能监测

并行语言及编译环境

· OpenACC · MPI · OpenMP

国产众核CPU基础软件

基础编译系统	基础函数库	自动向量化系统
· C / C++, Fortran · SIMD 扩展接口 · 异构代码生成	· C 库 · 加速线程库 · 数学库	· C / C++, Fortran · 循环级向量化 · 向量代码优化

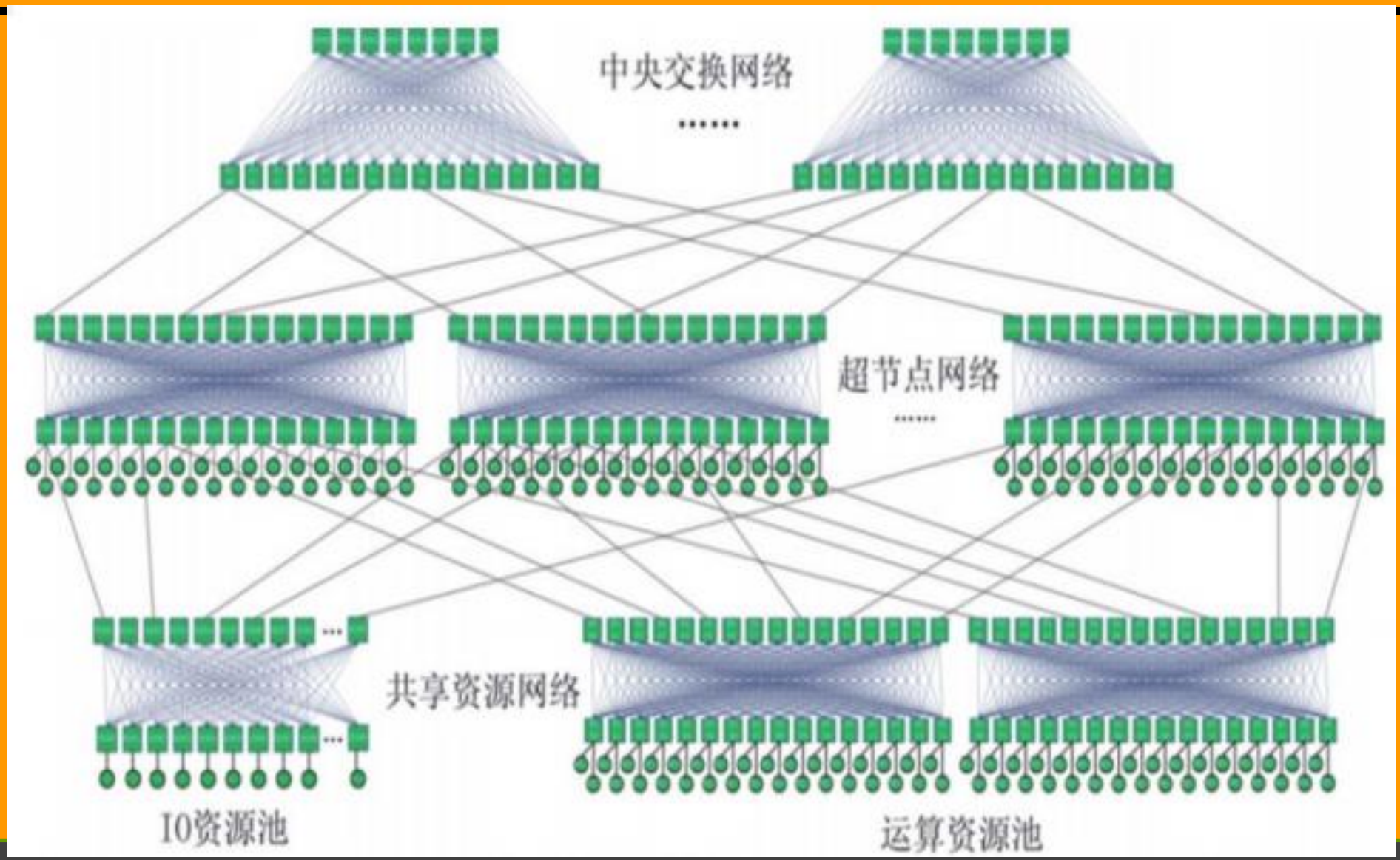
并行操作系统环境

· 作业管理 · 容错管理
· 资源管理 · 系统开工
· 功耗管理 · 安全管理
· 网络管理

高性能存储管理系统

· SWGFS并行文件系统
· LWFS 轻量级文件系统
· 存储管理平台

“神威·太湖之光”计算机系统


中央交换网络
.....
超节点网络
.....
共享资源网络
IO资源池
运算资源池

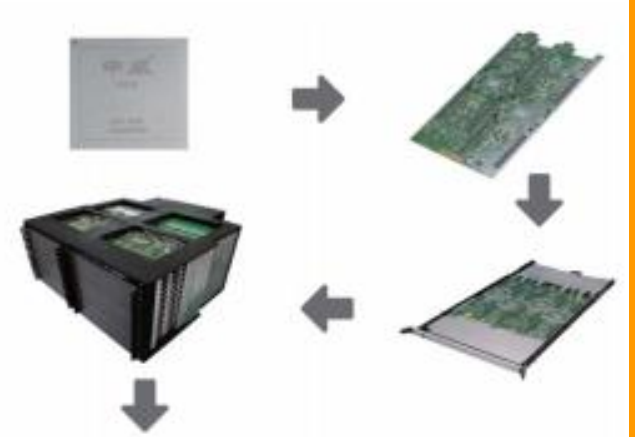
The Modern Era – 2015

- **Sunway TaihuLight (神威·太湖之光)**
  - Made entirely out of Chinese chips!
- **40,960 nodes with 10,649,600 cores**
- **Twice as fast and three times as efficient as Tianhe-2**
- **LINPACK 93 TFLOPs vs Peak=125 TFLOPs ie 75% peak**
- **Peak power consumption = 15.37 MW**
  - 6060 Gflops/kW
  - Also very high in Green500 table

Sunway TaihuLight

神威·太湖之光（Sunway TaihuLight）·建造情况

网站	http://www.nsccwx.cn/
制造商	国家并行计算机工程技术研究中心（NRCPC）
位置	国家超算中心（无锡，江苏）NSCC-Wuxi
耗资	$273 million
首次进入Top500	2016.6



神威·太湖之光（Sunway TaihuLight）·机器情况

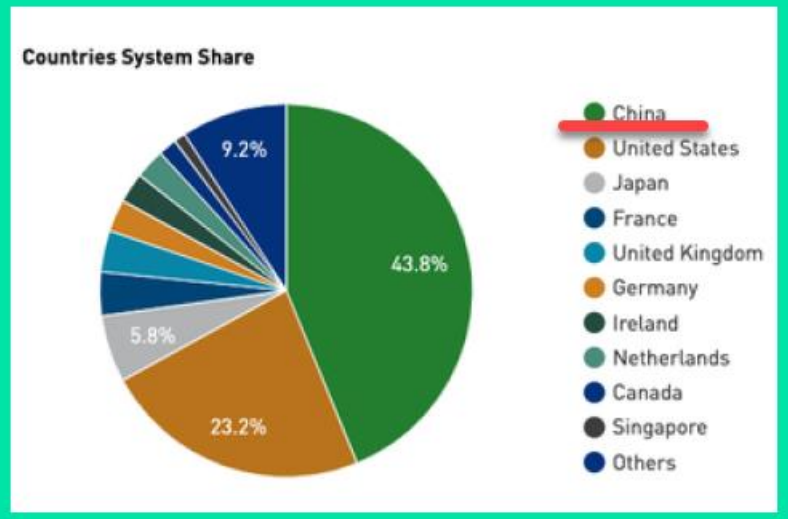
Hardware	
Architecture	Sunway MPP
Processor	Sunway SW26010 260C 1.45GHz
Cores	10,649,600
Memory	1,310,720 GB
Interconnect	Sunway

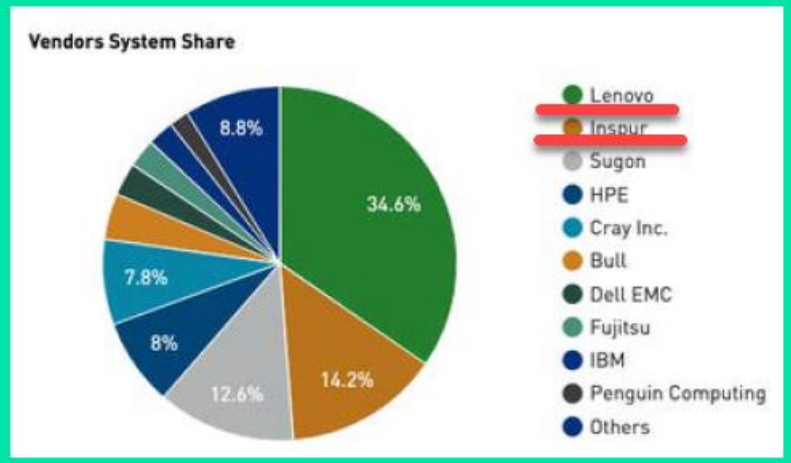
Software	
Operating System	Sunway RaiseOS
Parallel Programming	MPI, OpenMP, OpenACC

Power Consumption	
Power	15,371.00 kW

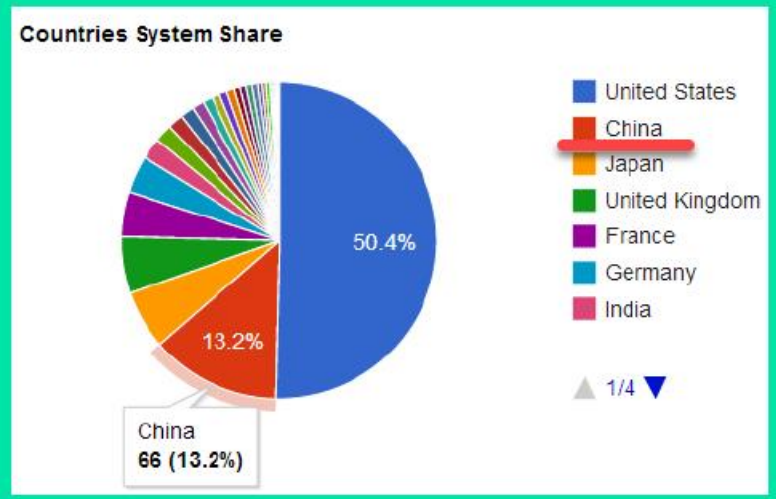
Performance	
Linpack Performance(Rmax)	93,014.6 TFlop/s
Theoretical Peak (Rpeak)	125,436 TFlop/s
Nmax	12,288,000

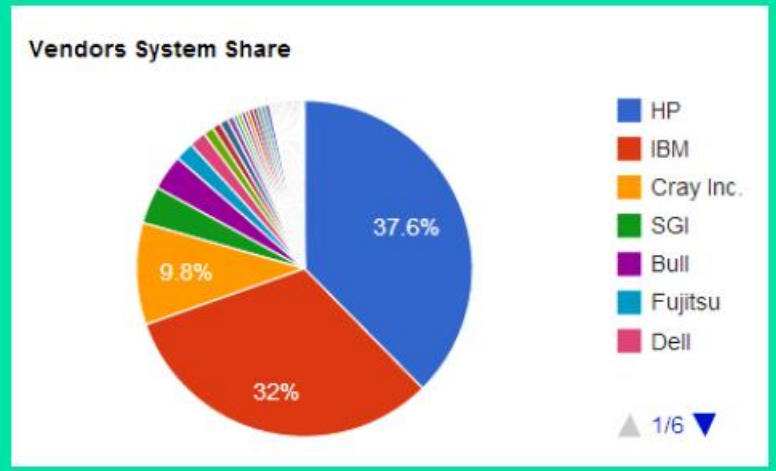
2019 年的数据





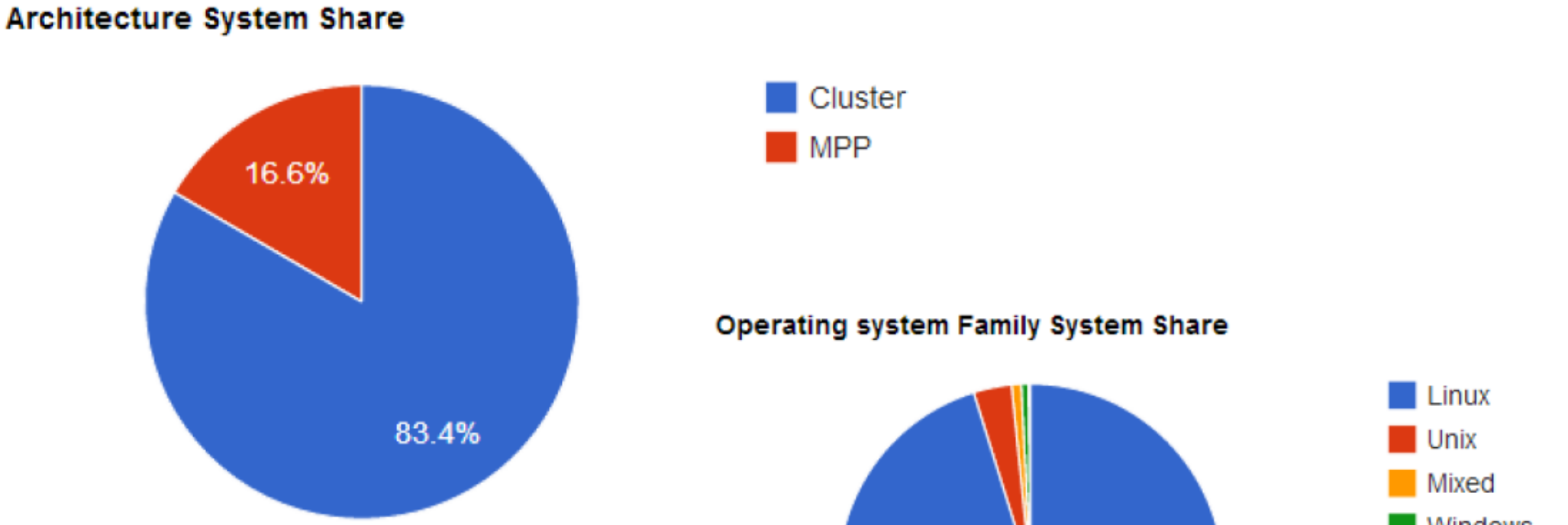
2013 年的数据

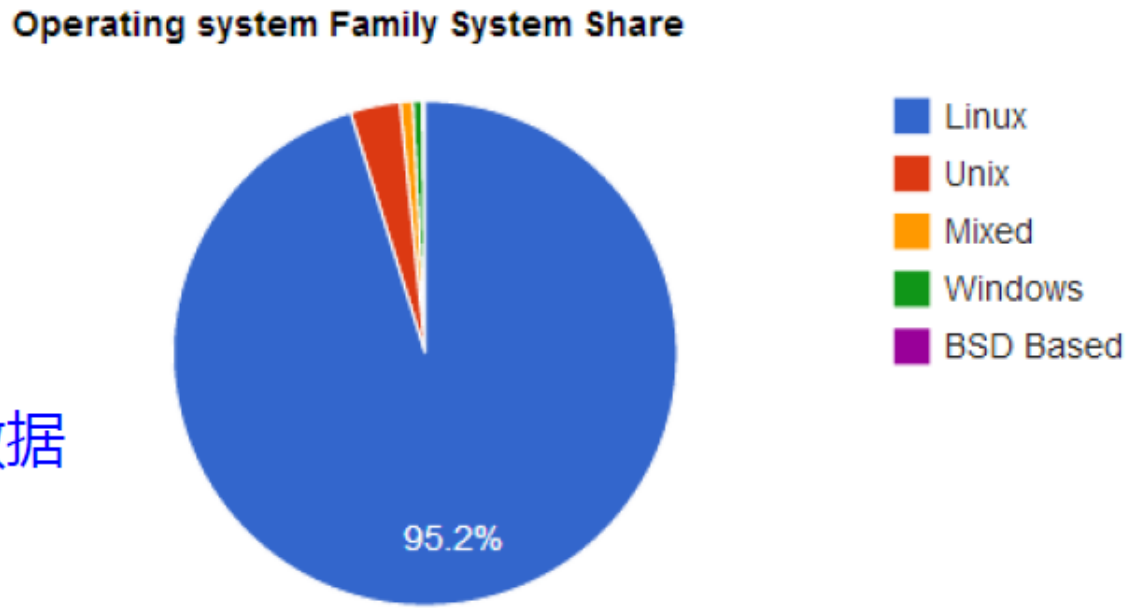




TOP500

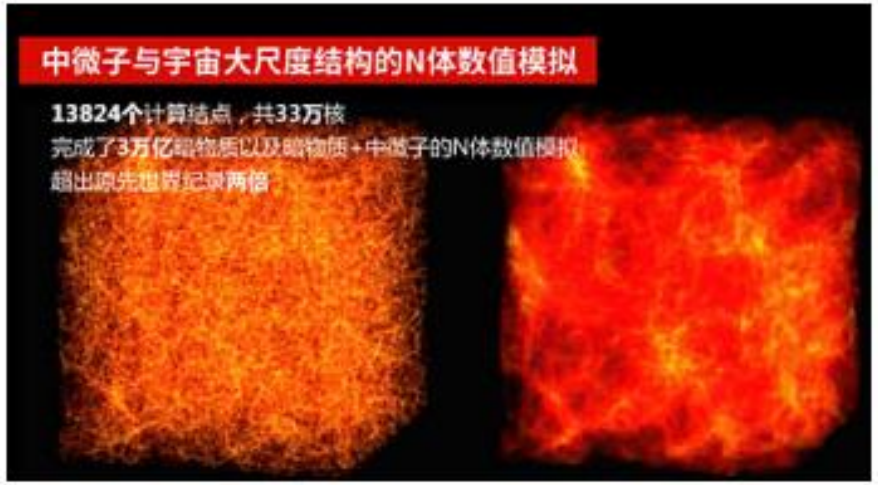
- 计算机类型与操作系统

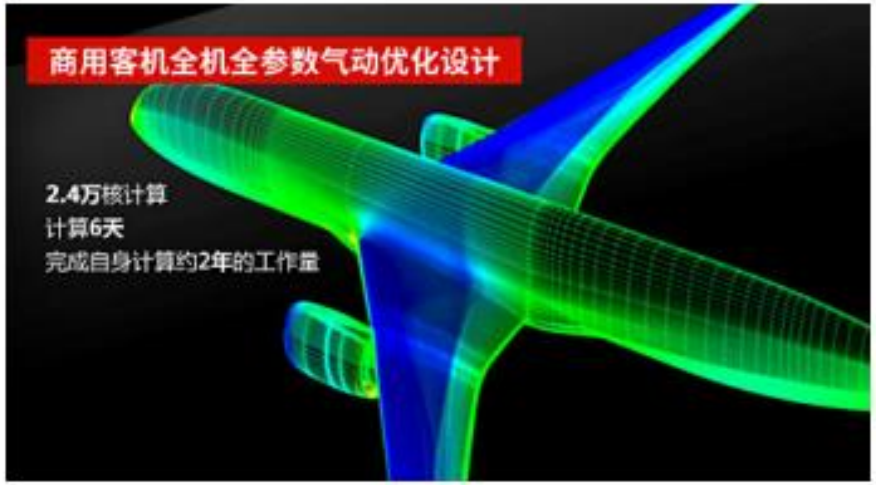


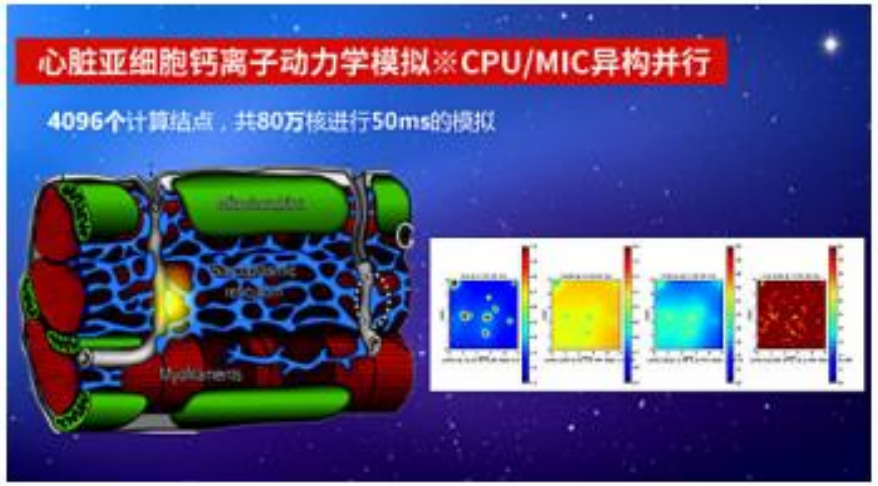


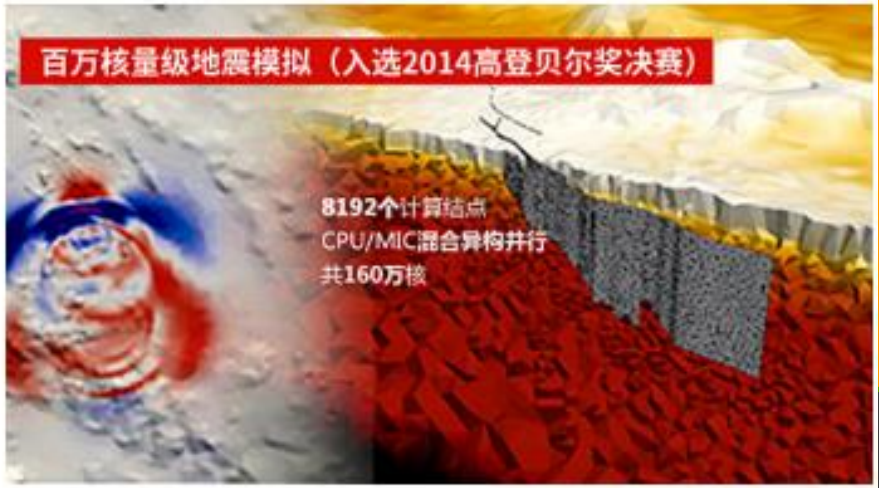
2013 年的数据

天河二号（Tianhe-2/MilkyWay-2）·应用

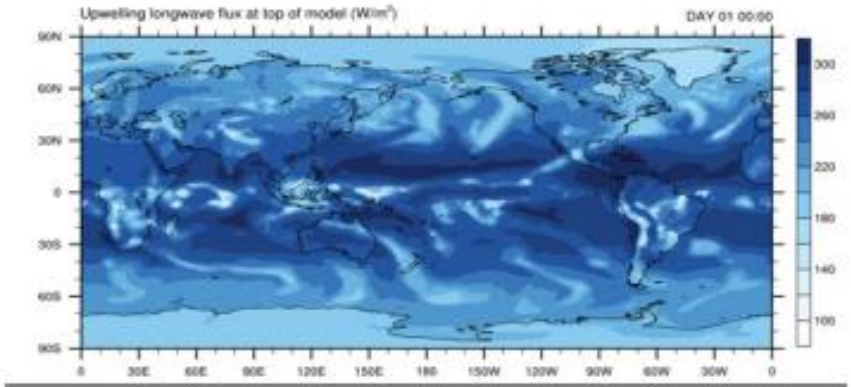






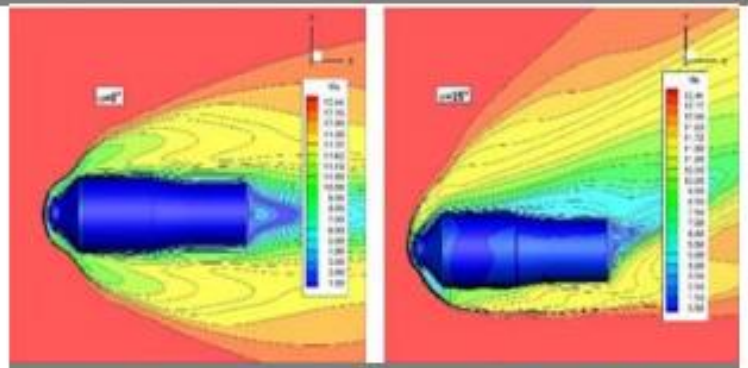


神威·太湖之光（Sunway TaihuLight）·应用



基于国产平台的国产地球系统模式

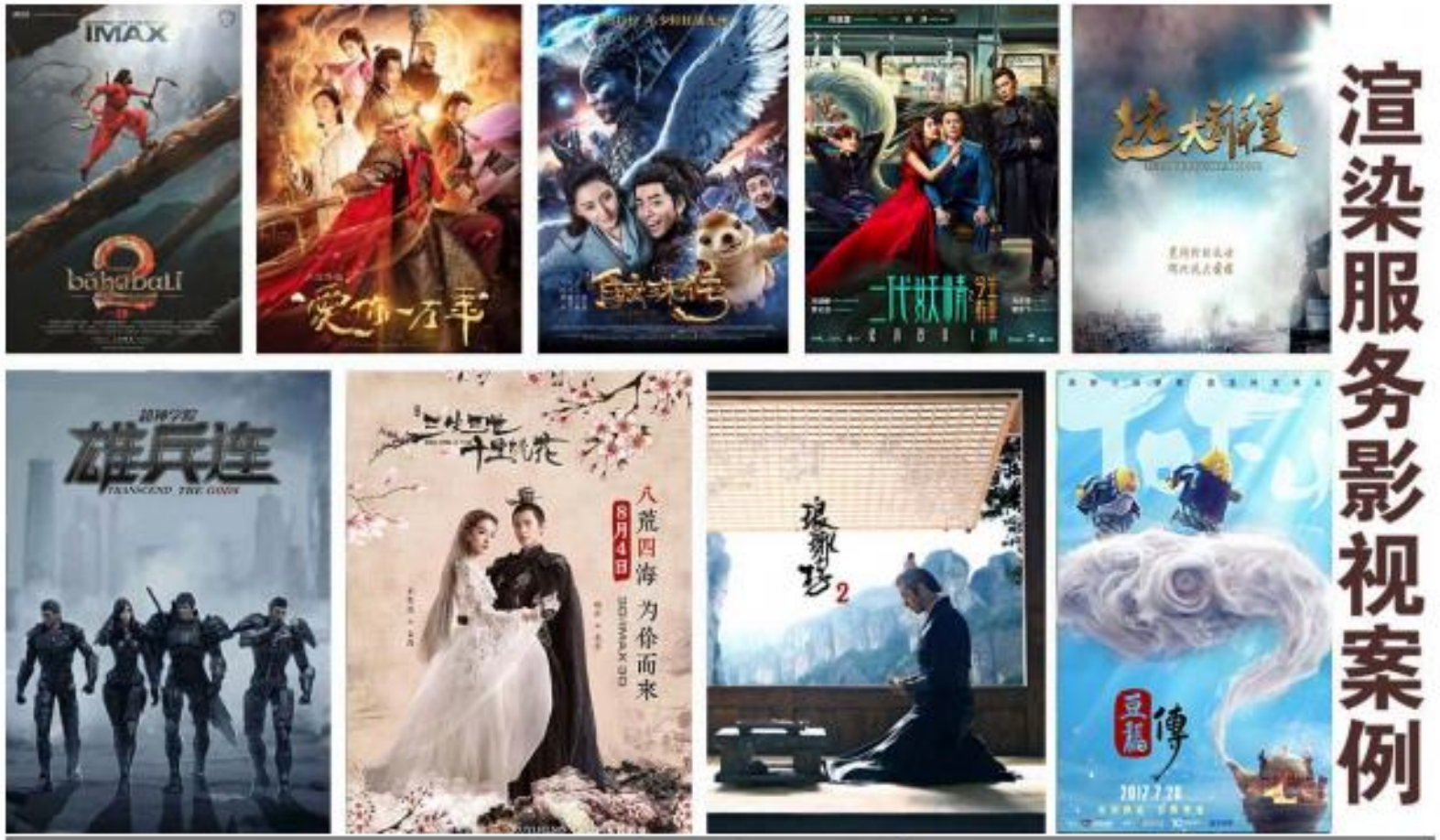
真实感动漫渲染系统



航天飞行器统一算法数值

岛礁建设浮式平台

神威·太湖之光（Sunway TaihuLight）·应用

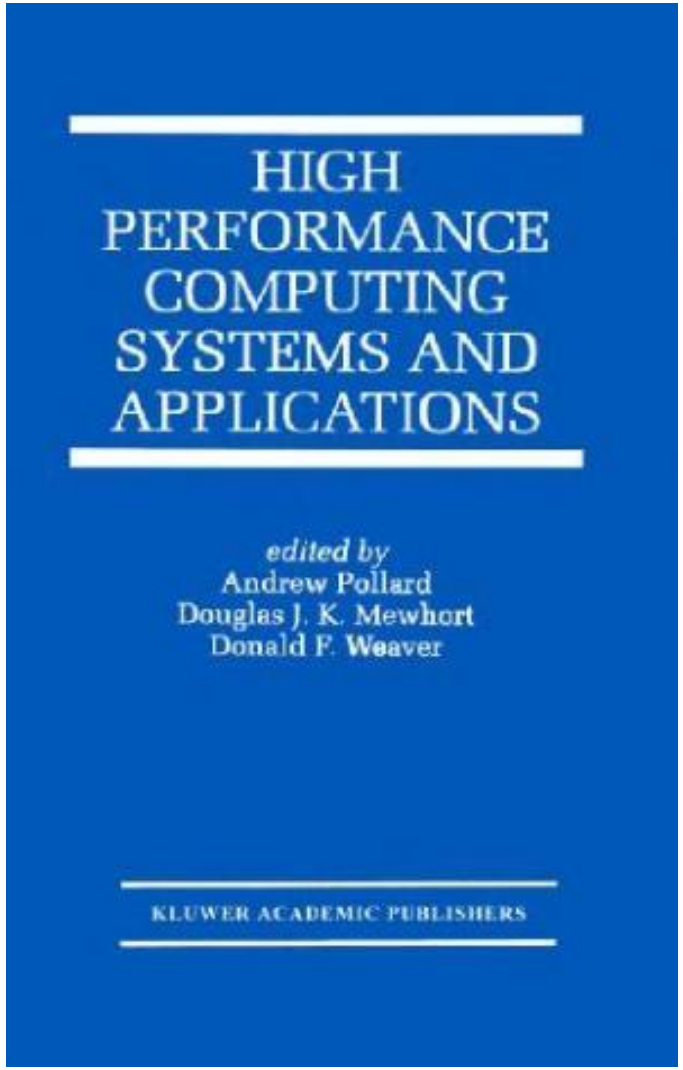


渲染服务影视案例

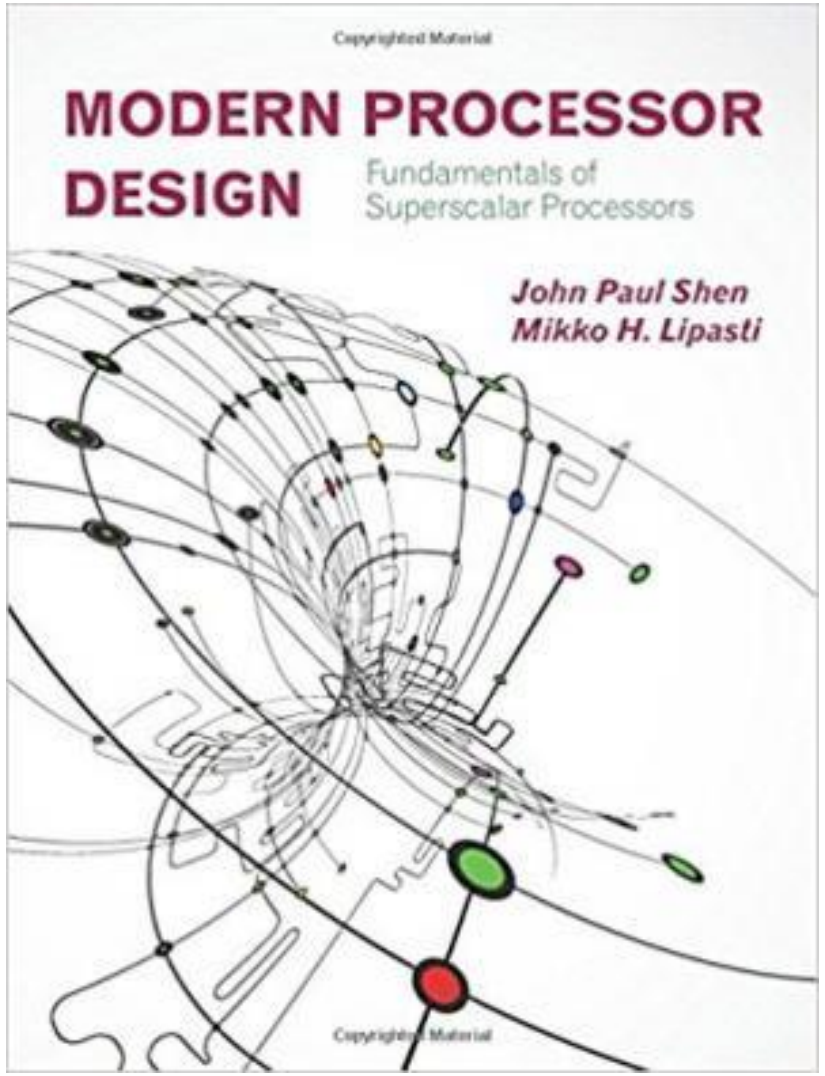
超大规模云渲染

华为 鲲鹏 HPC

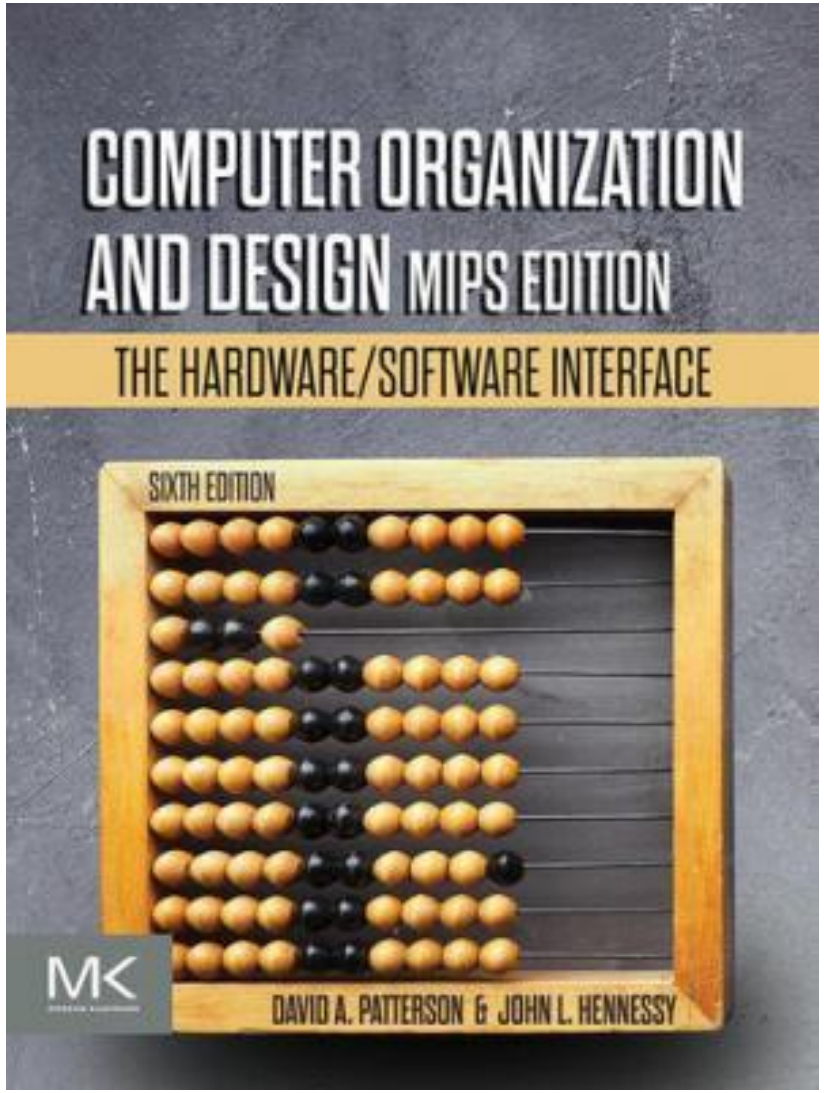

政府
金融
大企业
互联网
运营商
千亿产业
鲲鹏生态
软件市场
培训认证
合作计划
产业联盟
合作共赢
鲲鹏解决方案
鲲鹏混合云
鲲鹏云手机
鲲鹏高性能计算
鲲鹏大数据
在线按需
鲲鹏云服务
鲲鹏裸金属服务器
鲲鹏弹性云服务器
鲲鹏Kubernetes容器
鲲鹏Serverless容器
X86
鲲鹏
昇腾
GPU
FPGA
加速芯片



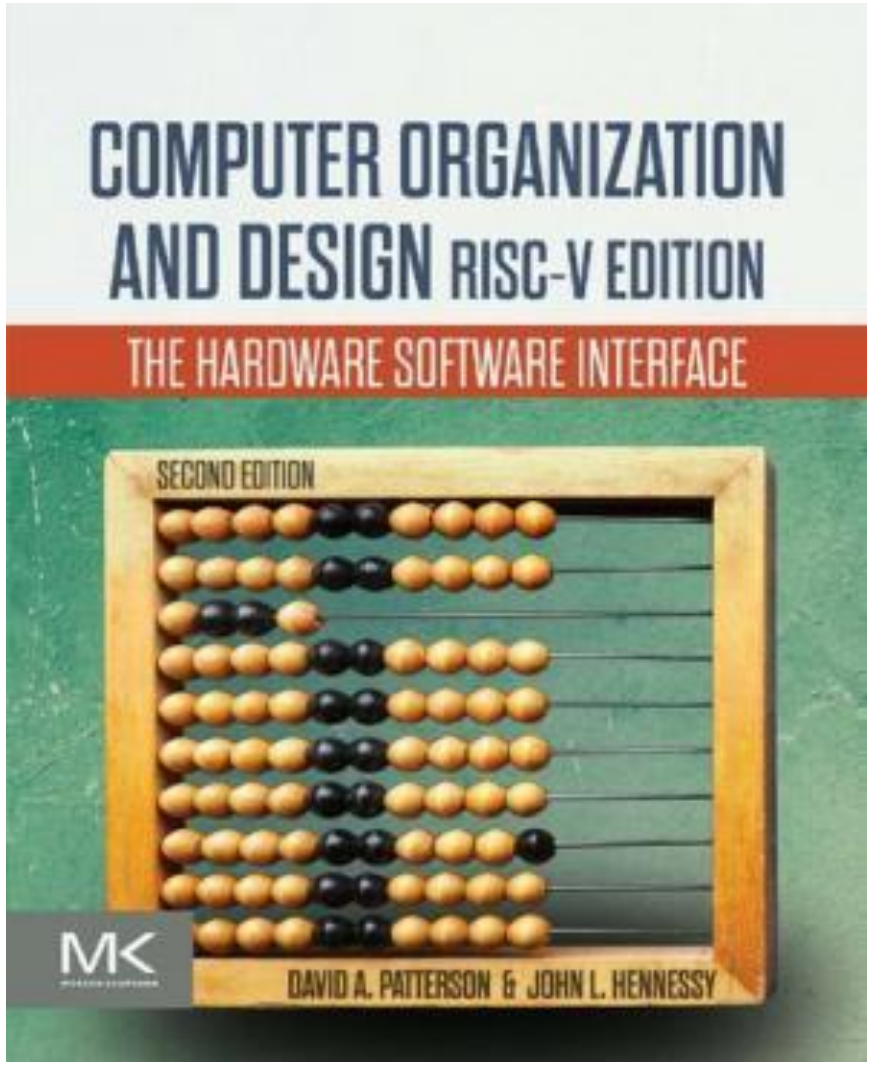
- High Performance Computing Systems and Applications
- *Andrew Pollard*, *Douglas J.K. Mewhort*, *Donald F. Weaver*
- **2000**



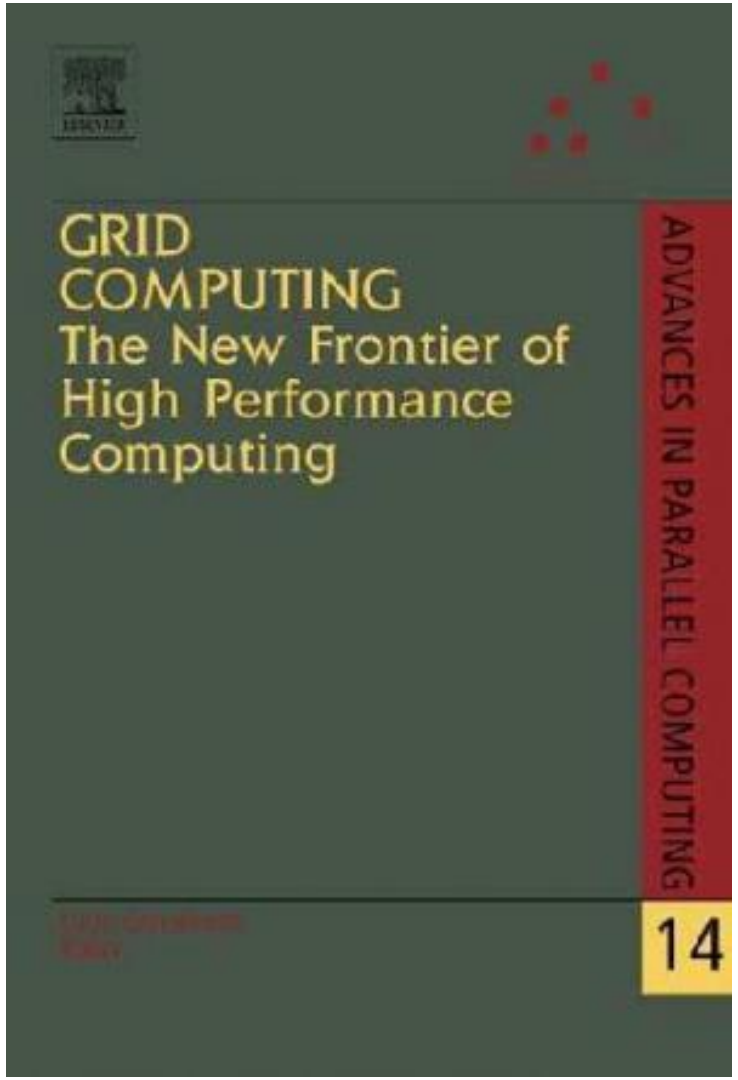
- **Modern Processor Design: Fundamentals of Superscalar Processors**
- ***John Paul Shen*, *Mikko H. Lipasti***
- **2013**



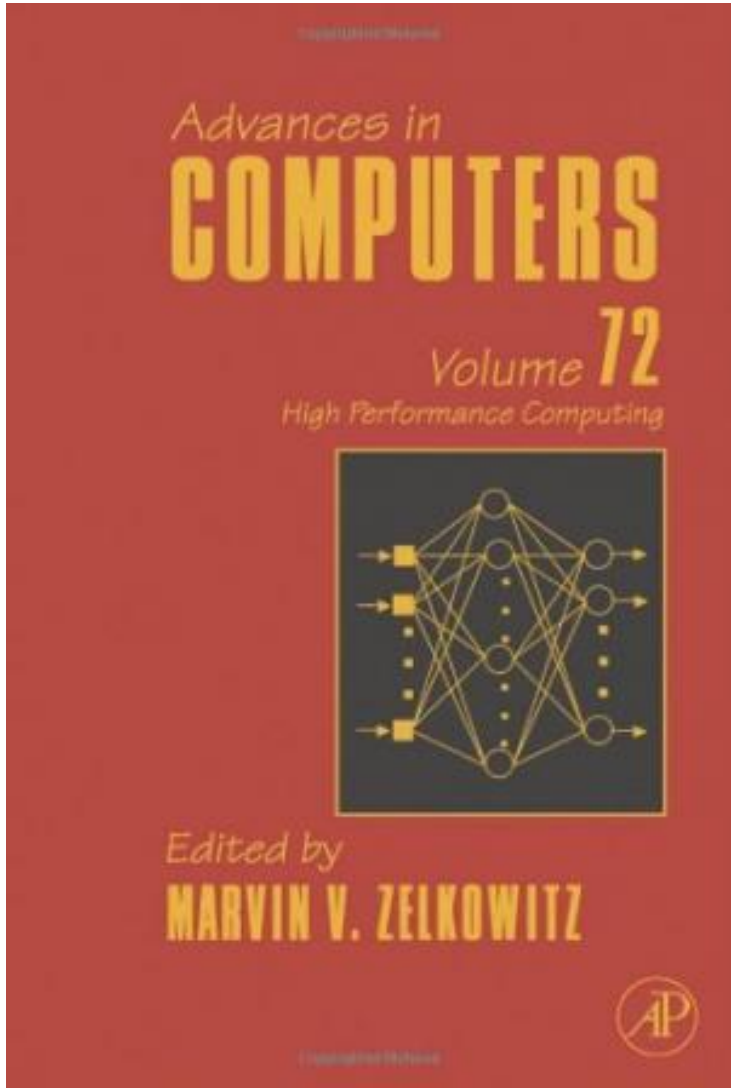
- Computer Organization and Design MIPS Edition: The Hardware/Software Interface (The Morgan Kaufmann Series in Computer Architecture and Design)
- *Patterson, David A., Hennessy, John L.*
- **2020**



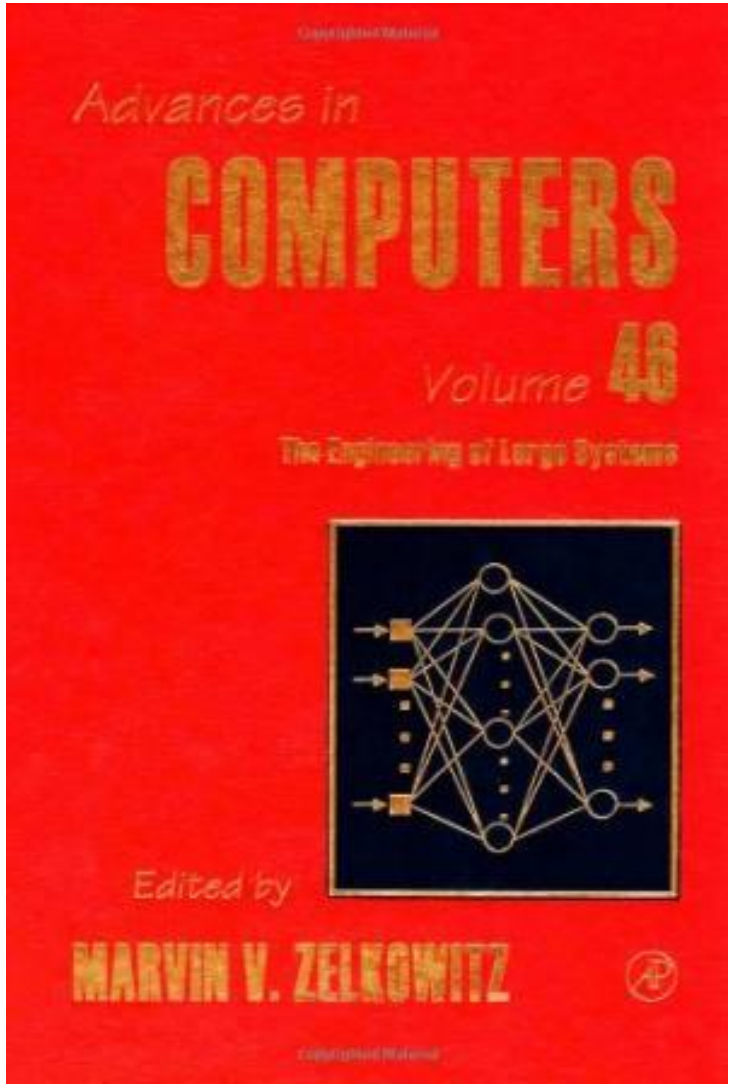
- Computer Organization and Design RISC-V Edition: The Hardware Software Interface
- *David A. Patterson*, *John L. Hennessy*
- **2020**



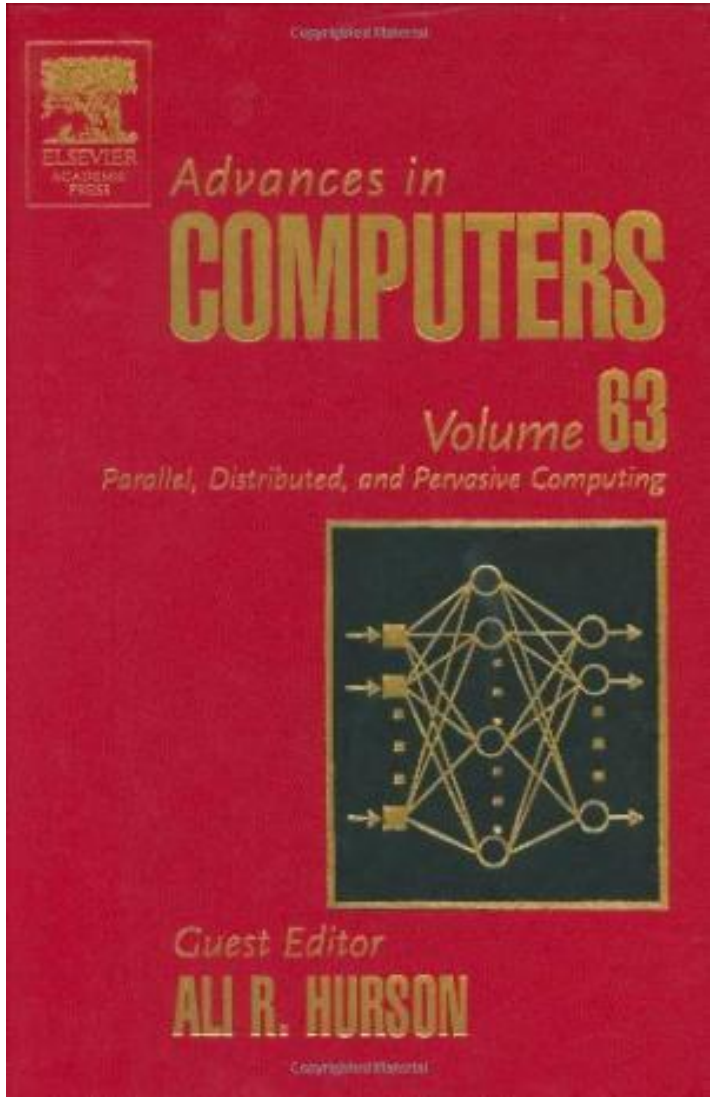
- Grid Computing: The New Frontier of High Performance Computing
- *Grandinetti L. (Ed)*
- **2005**



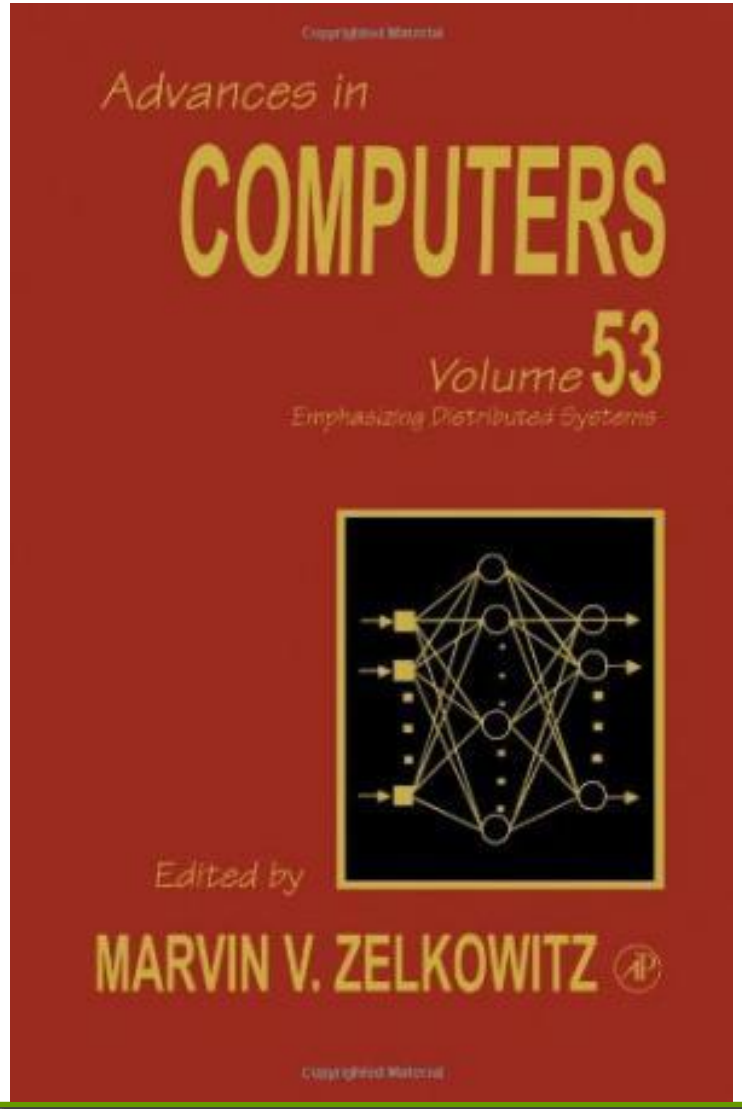
- High performance computing
- *Marvin Zelkowitz Ph.D. MS BS.*
- **2008**



- The engineering of large systems
- *Marvin Zelkowitz Ph.D. MS BS.*
- **1998**



- Parallel, Distributed, and Pervasive Computing
- *Marvin Zelkowitz Ph.D. MS BS.*
- **2005**



- Emphasizing Distributed Systems
- *Marvin Zelkowitz Ph.D. MS BS.*
- **2000**