The Abdus Salam
International Centre for Theoretical Physics

United Nations
Educational, Scientific and
Cultural Organization

IAEA
International Atomic Energy Agency

2068-11

Advanced School in High Performance and GRID Computing - Concepts and Applications

30 November - 11 December, 2009

Modern Architectures for HPC Computation

S. Cozzini
CNR-INFM Democritos
Trieste
Italy

Strada Costiera 11, 34151 Trieste, Italy - Tel.+39 040 2240 111; Fax +39 040 224 163 - sci_info@ictp.it

smr2068

Modern architectures for HPC computation

Stefano Cozzini

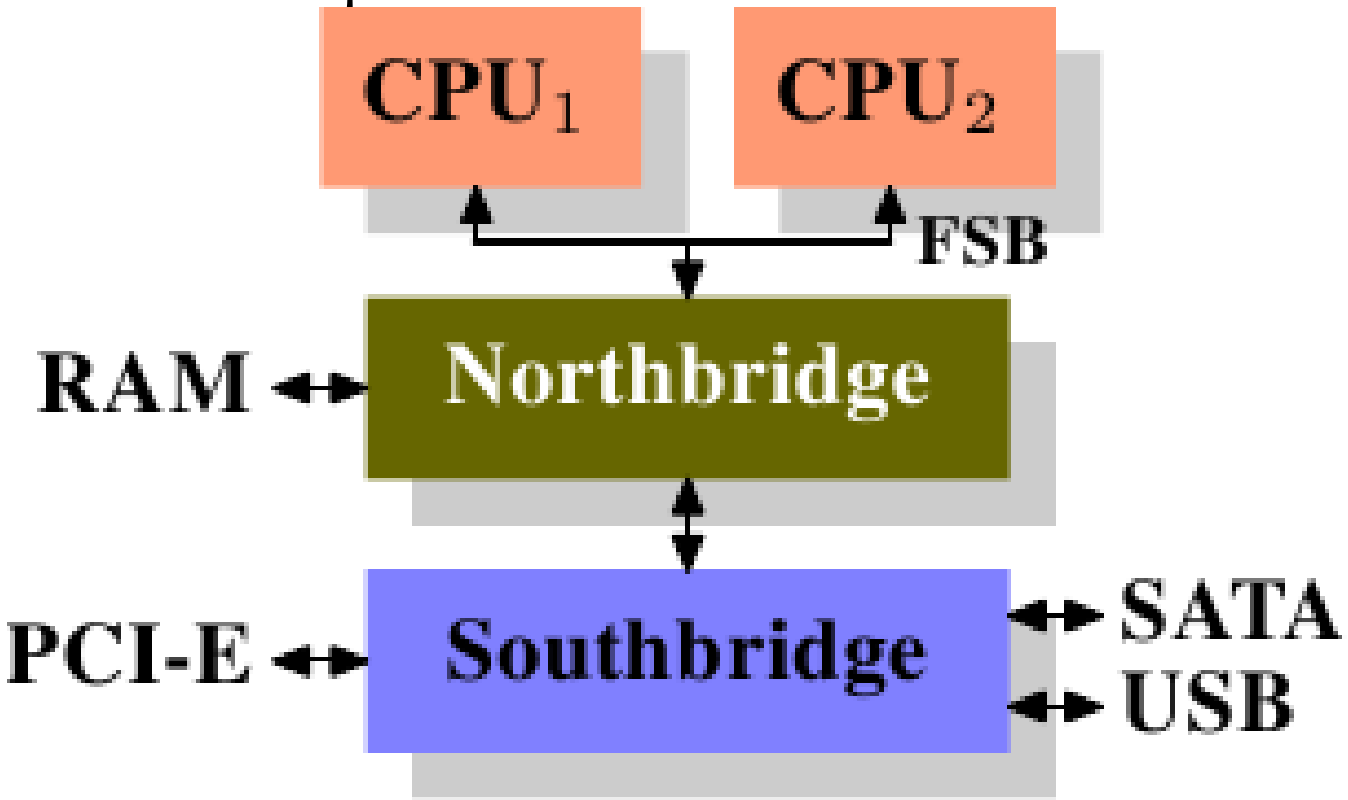
Democrito and SISSA/eLAB - Trieste

Outline

- Standard architectures of modern systems
- Multicore architecture
- Interesting hardware within the cpus
- Memory hierarchy
- Final consideration

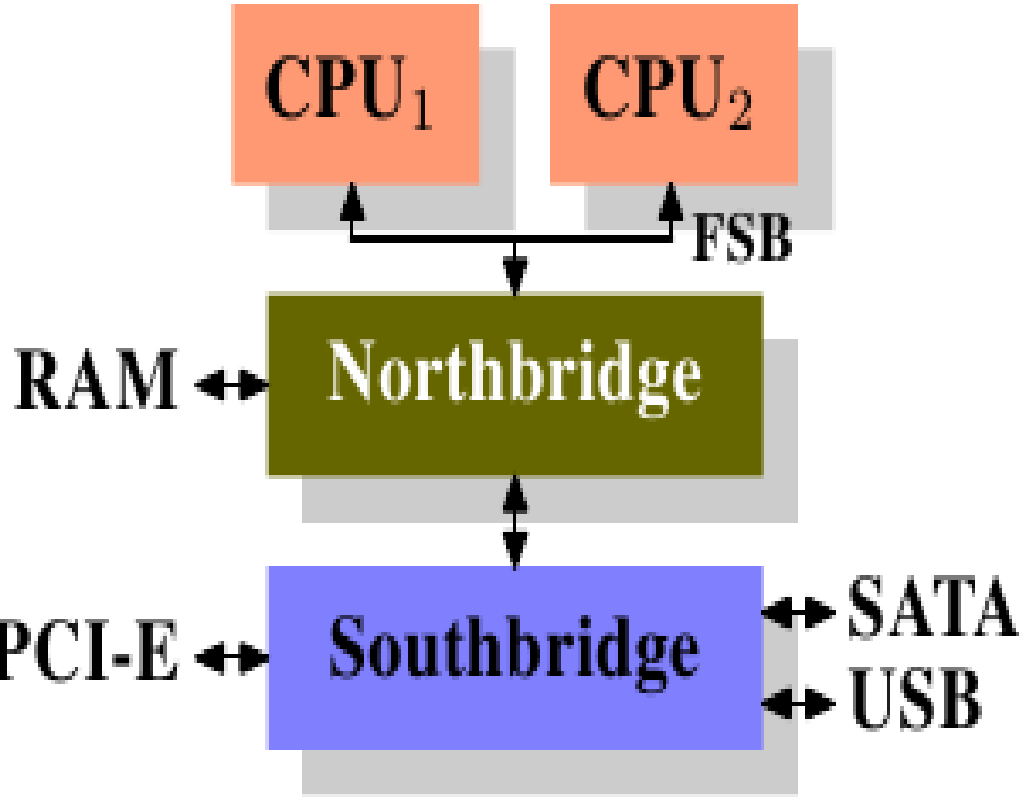
standard architecture

- Characteristics:
  - more than one CPU !
  - 64 bit address space



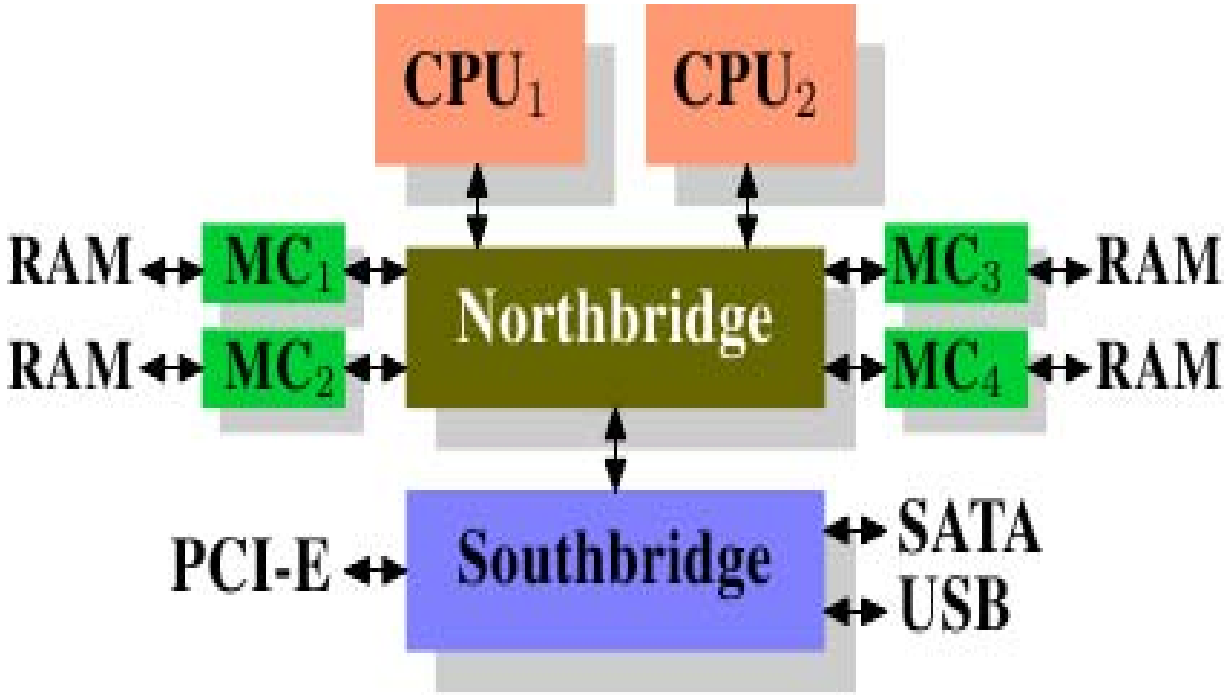
standard modern architecture

- All data communication from one CPU to another must travel over the same bus used to communicate with the Northbridge.
- All communication with RAM must pass through the Northbridge.
- Communication between a CPU and a device attached to the Southbridge is routed through the Northbridge.



more expensive architecture

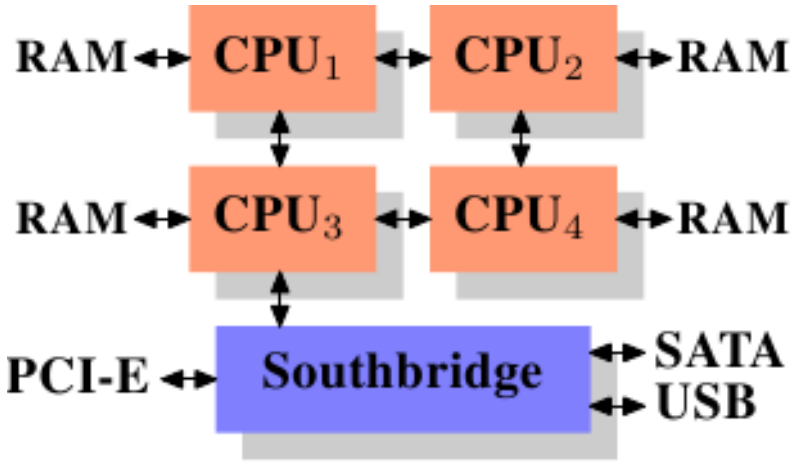
- Northbridge can be connected to a number of external memory controllers (in the following example, four of them).



INCREASE IN BANDWIDTH TOWARD MEMORY

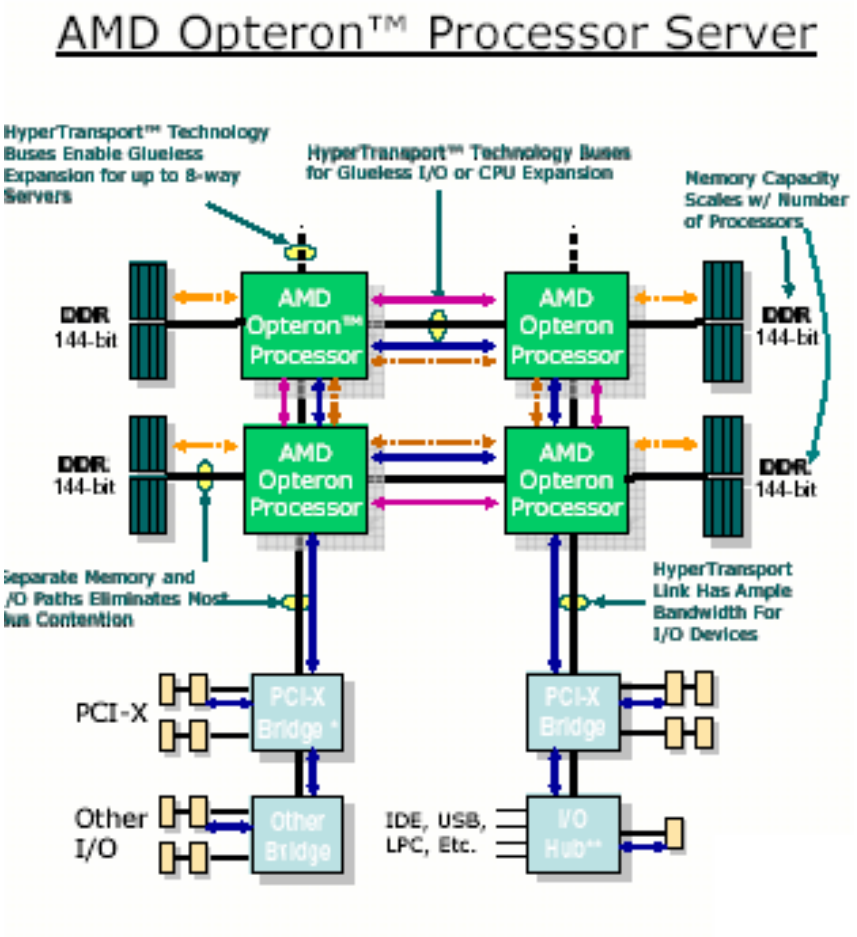
Another kind of architecture..

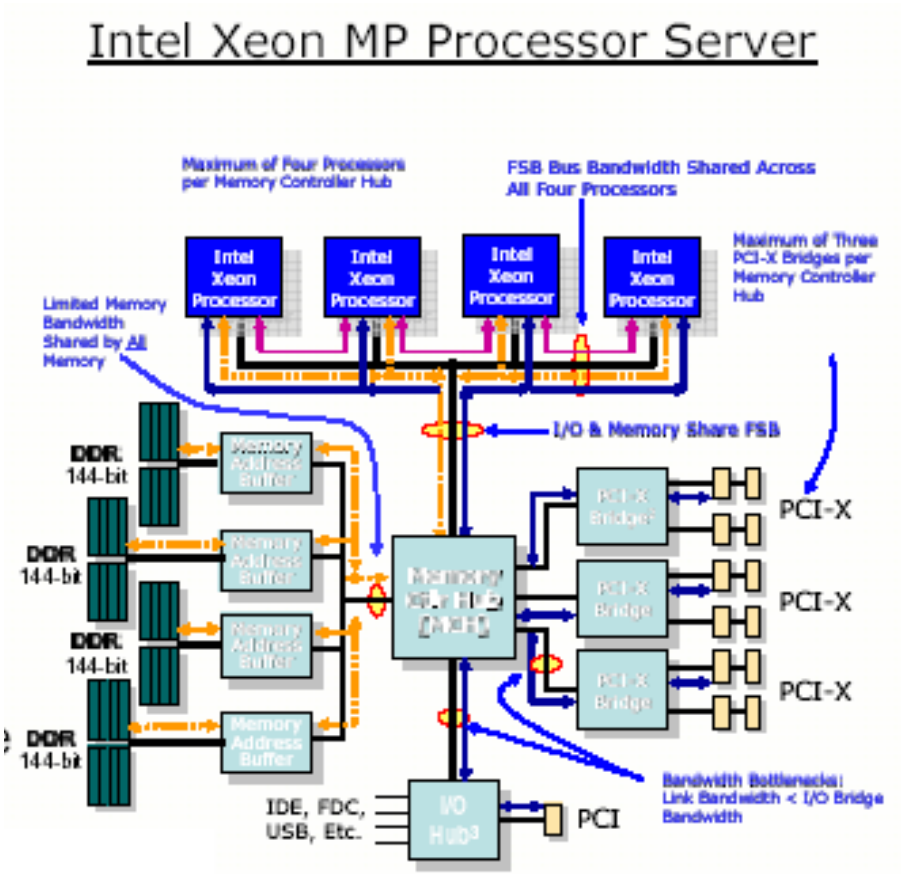
- Integrated memory controllers (AMD style)



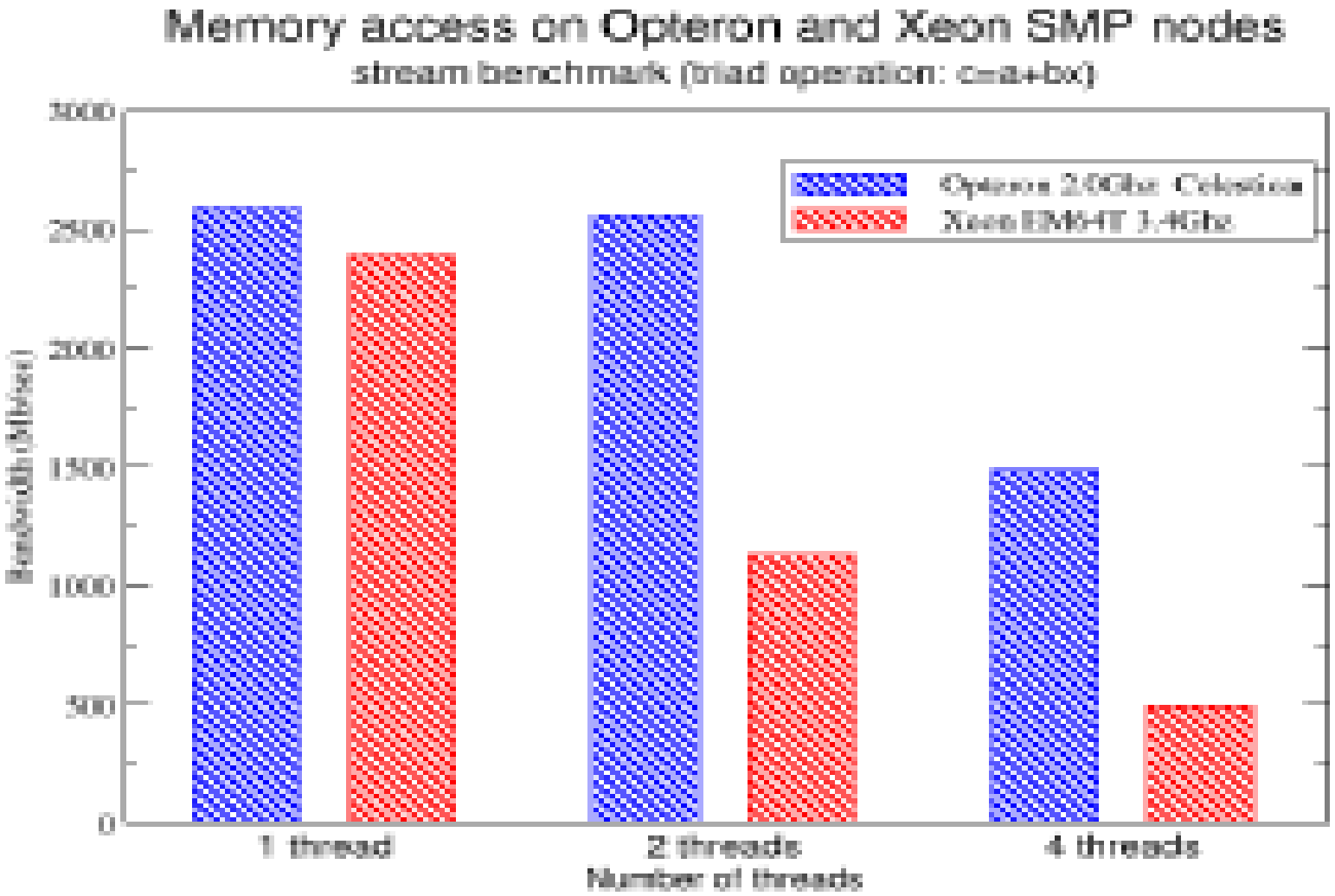
NUMA ARCHITECTURE !

AMD/Intel XEON comparison

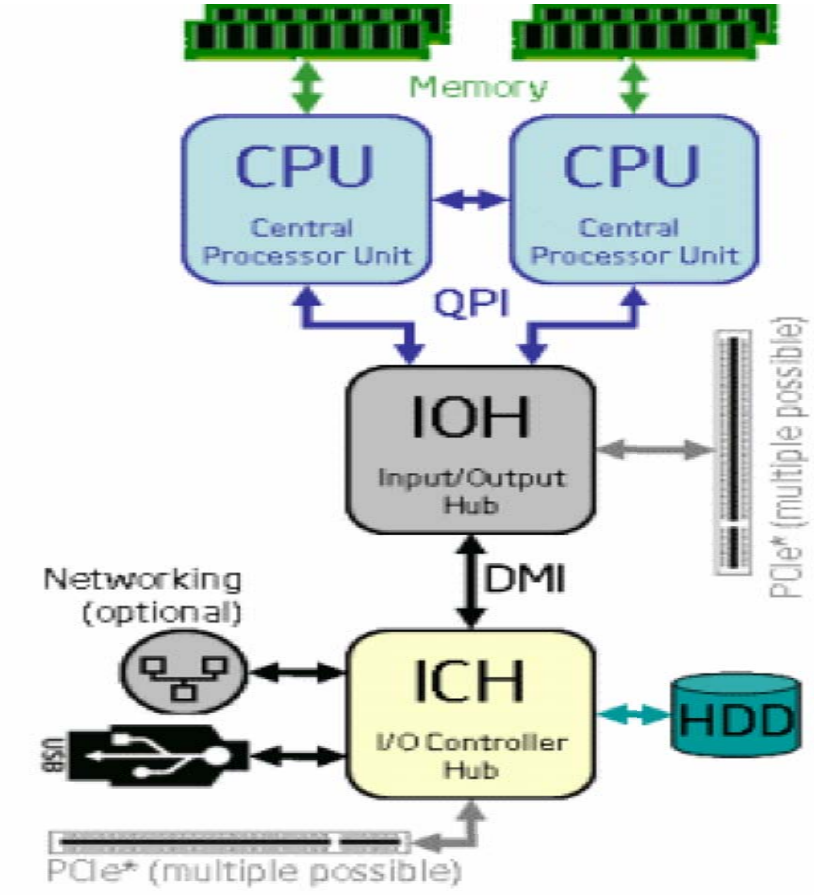




AMD/Intel XEON :



New Intel Xeon family: Nehalem

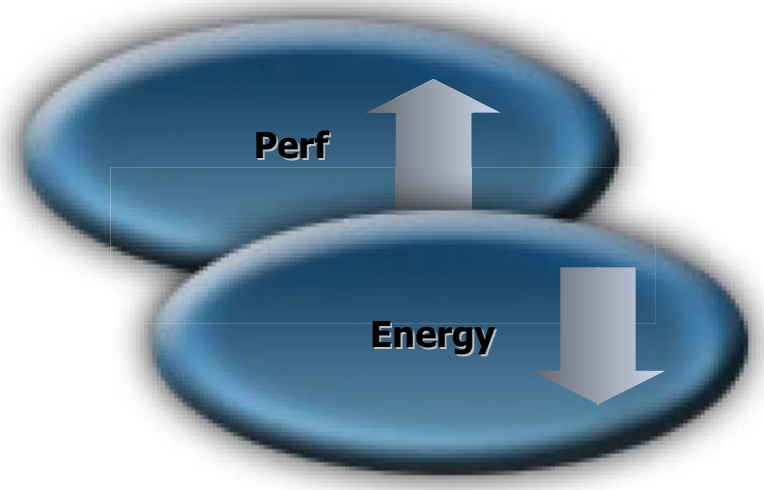


- First NUMA architecture by INTEL
- QPI among CPUs to play the role of hyper-transport in AMD
- Recently released (april 2009)

which kind of CPUS ?

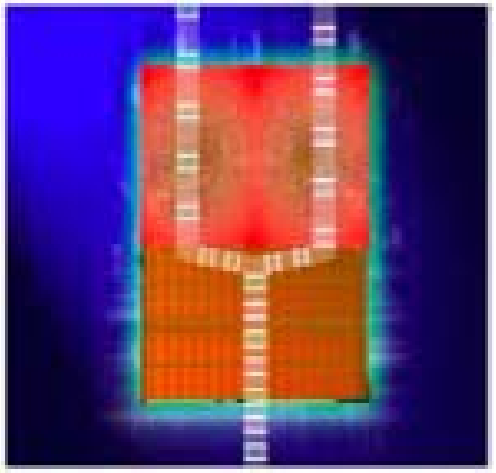
- MULTICORE !!

Multiple, externally visible processors on a single die where the processors have independent control-flow, separate internal state and no critical resource sharing



What are multi-core processors?

- Integrated circuit (IC) chips containing more than one identical physical processor (core) in the same IC package. OS perceives each core as a discrete processor.
- Each core has its own complete set of resources, and may share the on-die cache layers
- Cores may have on-die communication path to front-side bus (FSB)
- What is a multi processor?
  - a collection of multicore cpus !

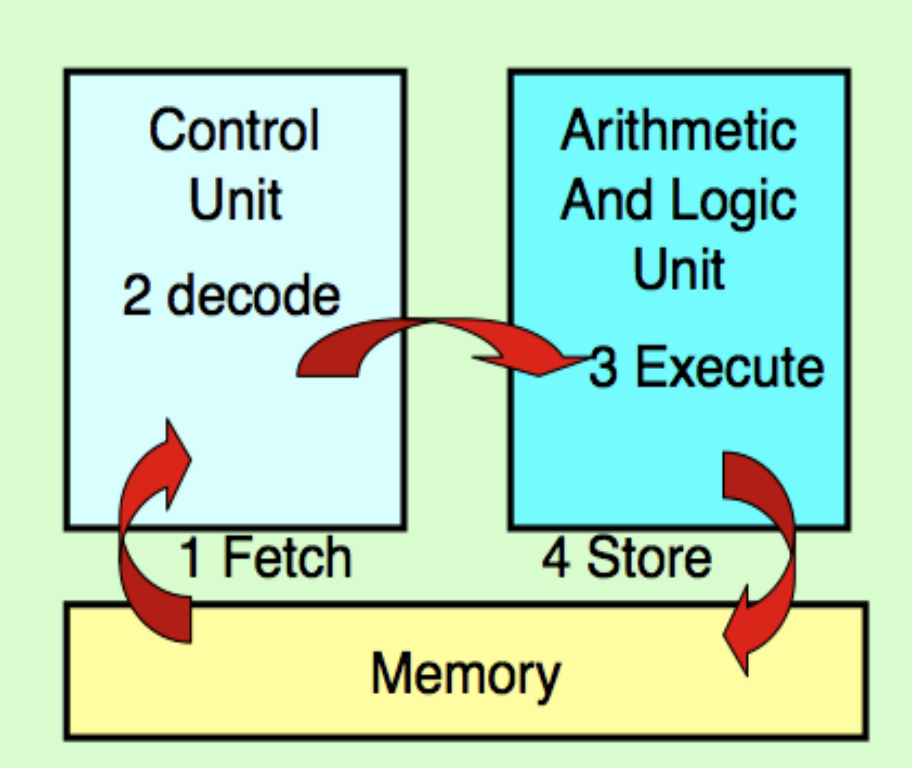


Motivation for multicores

- Exploits increased feature-size and density
- Increases functional units per chip (spatial efficiency)
- Limits energy consumption per operations
- Constrains growth in processor complexity

What within a core ?

- Control Unit: processes instructions ALU: math and logic operations
- At each cycle the CPU fetches both data and a description of what operations need to be performed and stores them in registers.



What else in the cores ?

- On modern CPU/Cores there are many other stuff:
  - Pipelined functional units
  - Superscalar execution
  - Floating point instruction set extensions
- For the processors in most modern parallel machines, the circuitry on the chip which performs a given type of operation on operands in registers is known as a functional unit.

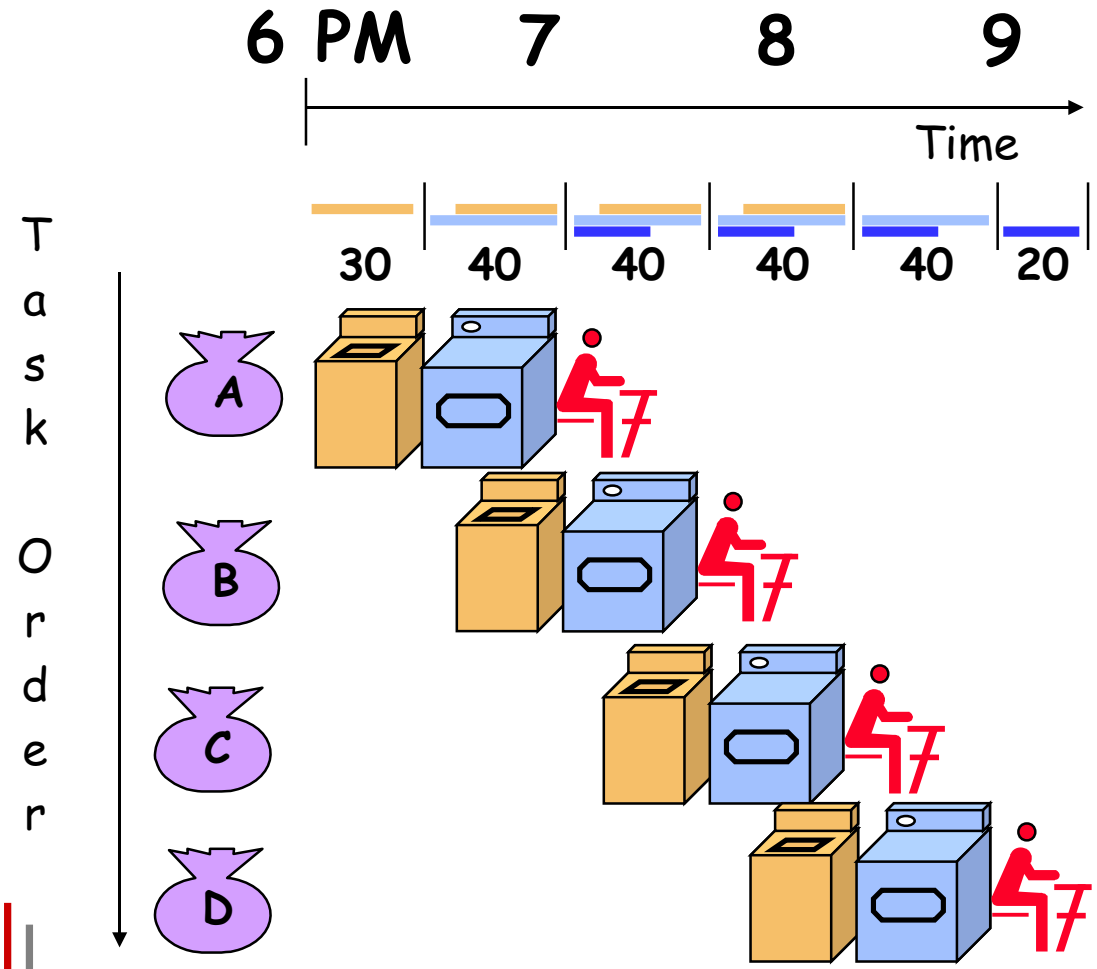
Pipelined Functional Units

- Most integer and floating point functional units are pipelined
- they can have multiple independent executions of the same instruction placed in a queue.
- The idea is that after an initial startup latency, the functional unit should be able to generate one result every clock period (CP).
- Each stage of a pipelined operation can be working simultaneously on different sets of operands.

What is Pipelining?

Dave Patterson's Laundry example: 4 people doing laundry

wash (30 min) + dry (40 min) + fold (20 min)



- In this example:
  - Sequential execution takes 4 * 90min = 6 hours
  - Pipelined execution takes 30+ 4*40+ 20 = 3.3 hours
- Pipelining helps throughput, but not latency
- Pipeline rate limited by slowest pipeline stage
- Potential speedup = Number pipe stages
- Time to "fill" pipeline and time to "drain" it reduces speedup

modern processors are superscalar !

- Processors which have multiple functional units which can operate concurrently are said to be superscalar.
- Examples:
  - AMD Opteron
    - 3 Floating point/MMX/SSE units
    - 3 Integer units
    - 3 Load/store units
  - Intel Xeon
    - 2 Floating point units
    - 2 Integer units
    - 2 Load/store units

Floating Point Instruction Set Extensions

- additional floating point instructions beyond the usual floating point add and multiply instructions:
  - Square root instruction --usually not pipelined!
    - AMD Opteron / Intel Xeon
  - SIMD (a.k.a. vector) floating point instructions
    - AMD Opteron/ Intel Xeon
- Combined floating point multiply/add (MADD) instruction
  - AMD Opteron ("Barcelona" and after, using SIMD)
  - Intel Xeon ("Woodcrest" and after, using SIMD)

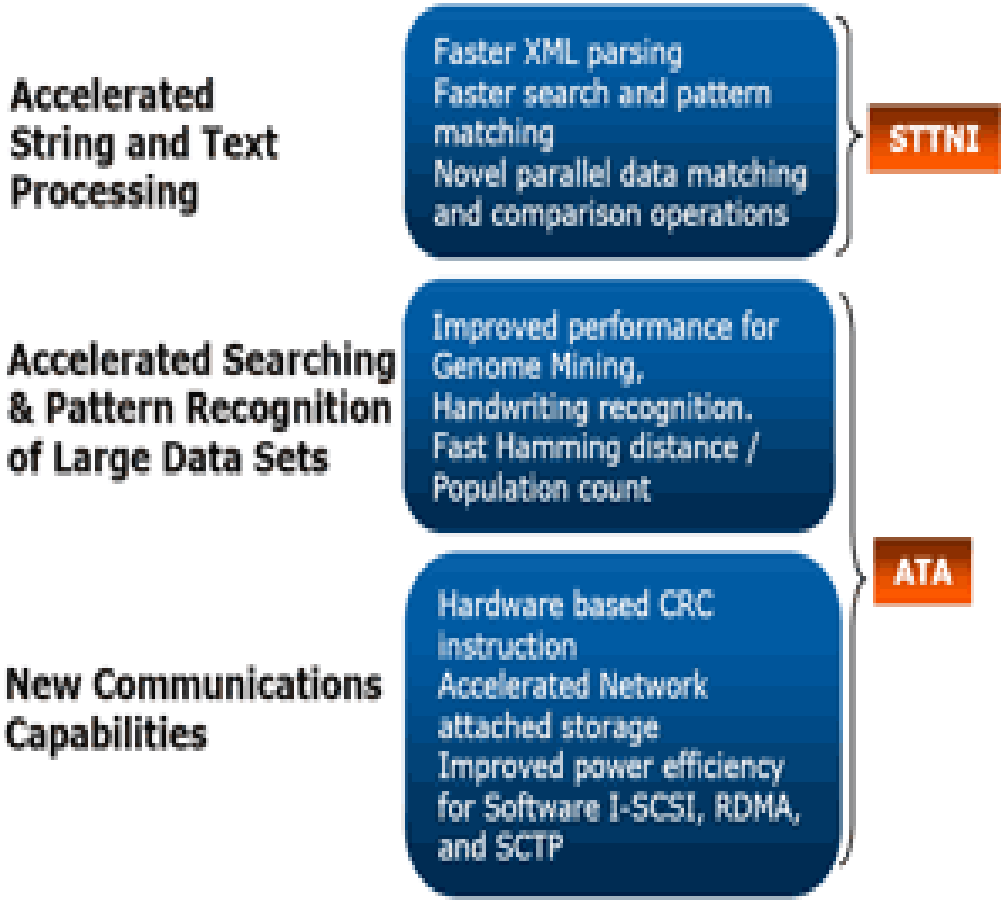
Instruction Set Extensions

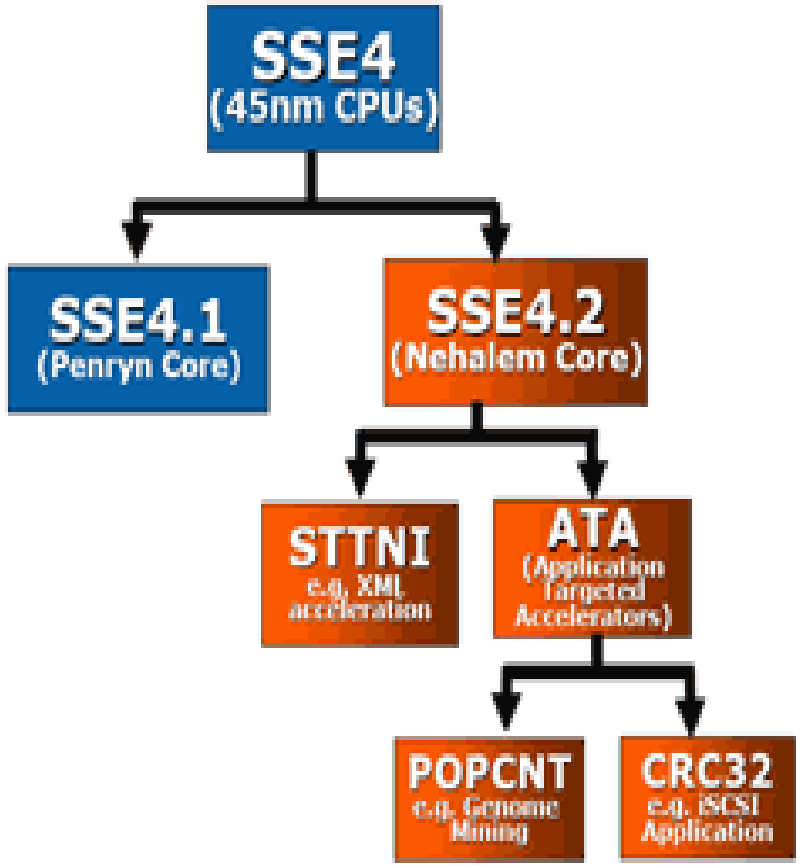
- Intel
  - MMX (Matrix Math eXtensions)
  - SSE (Streaming SIMD Extensions) /
  - SSE2 (Streaming SIMD Extensions 2)
  - SSE3 and now SSE4.2 (see next slide)
- AMD
  - 3DNow!
  - AMD 3DNow!+ (or 3DNow! Professional, or 3DNow! Athlon)

    ...
- To check what you have on your machine:
  - cat /proc/cpuinfo
- to enable them: use appropriate compiler flag..

SSE4.2

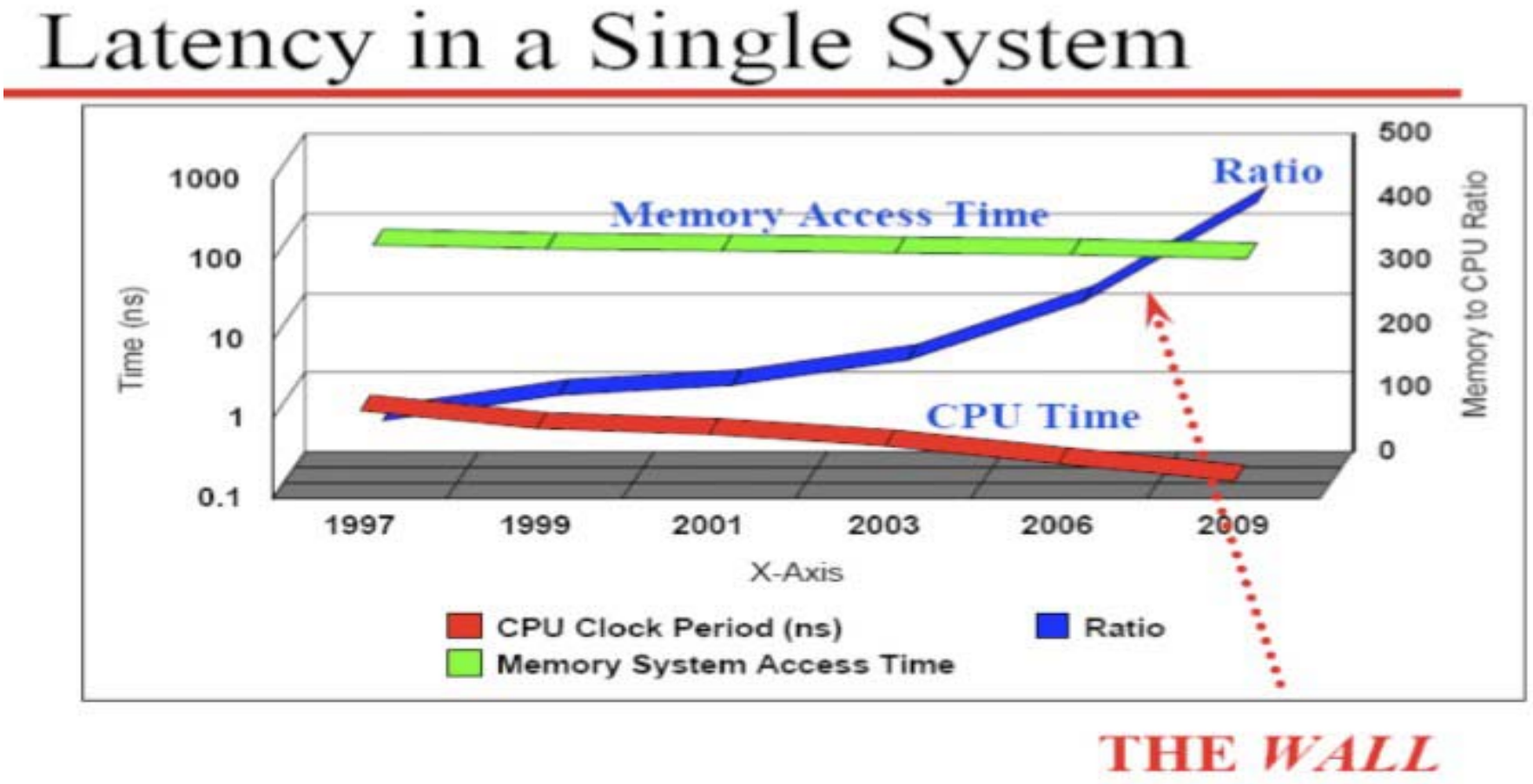
- SSE4.2 Instruction Set Architecture (ISA) Leadership in 2008





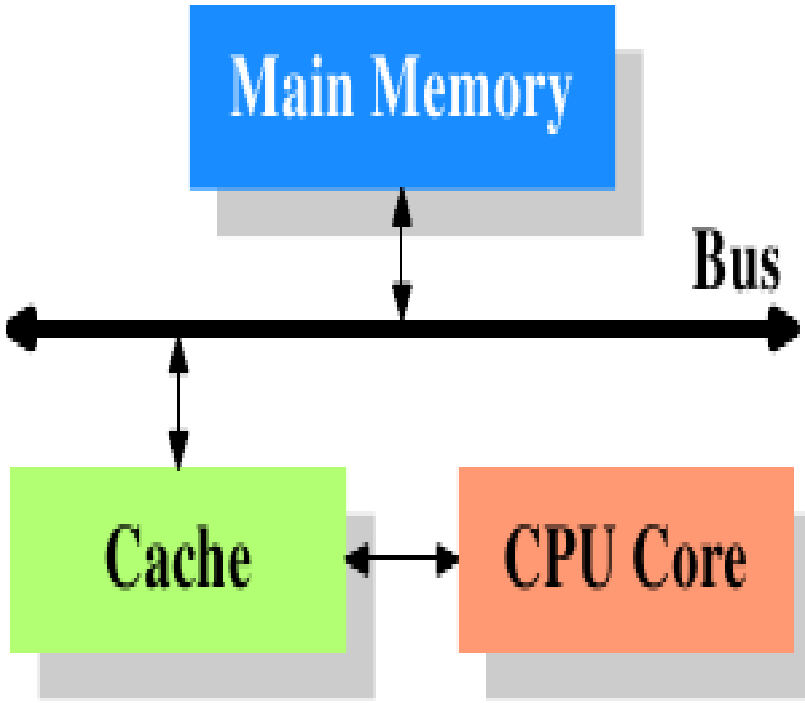
Memory wall problem

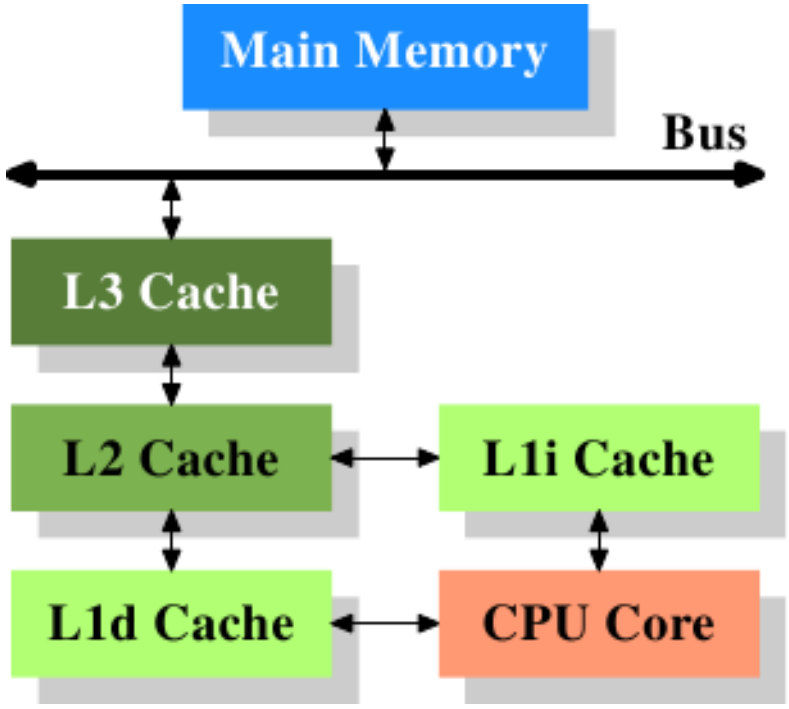
- The problem is not new referring to the growing gap between how fast a CPU can operate on data and how fast it can get the data it needs.



CACHE and MEMORY

- CACHE:A store of things that will be required in future, and can be retrieved rapidly. A cache may, or may not, be hidden.
[wikidictionary]

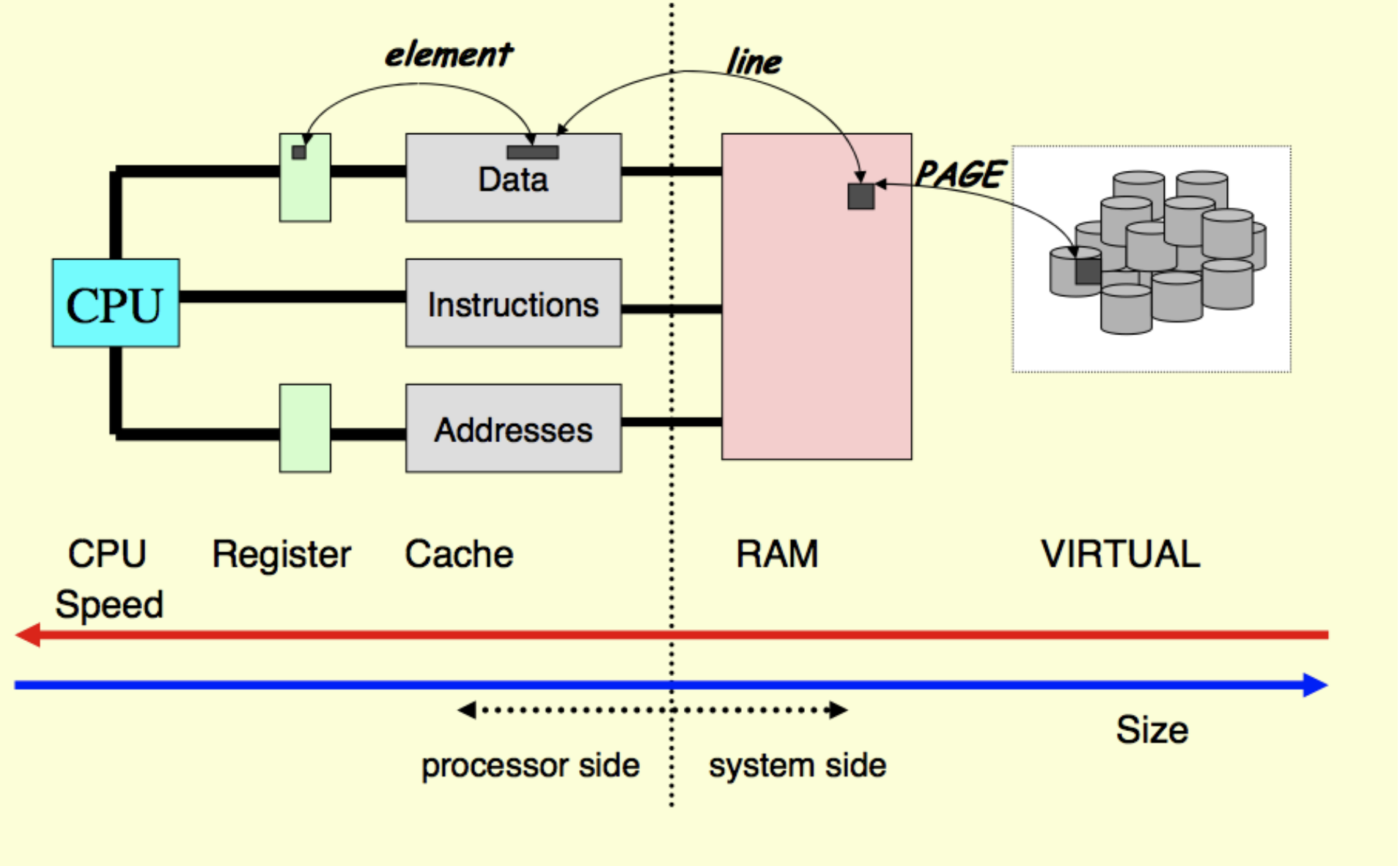




Hierarchy of memory..

- In modern computer system same data is stored in several storage devices during processing
- The storage devices can be described & ranked by their speed and "distance" from the CPU
- There is thus a hierarchy of memory objects
- Programming for a machine with memory hierarchy requires optimization for that memory structure.

Memory hierarchy



Components:

- **Registers:** On-chip circuitry used to hold operands and results of functional unit calculations.
- **L1 (Primary) Data Cache**: Small (on-chip) cache used to hold data about to operated on by processor.
- **L2 (Secondary) Cache**: Larger (on-or off-chip) cache used to hold data and instructions retrieved from local memory. Some systems also have L3 and even L4 caches.
- **Local Memory**: Memory on the same node as the processor.
- **Remote Memory**: Memory on another node but accessible to all processors in the network.
- **Disks**: Storage space where to save read large amount of data
- **Tapes/SAN:** space where to store data rarely needed.

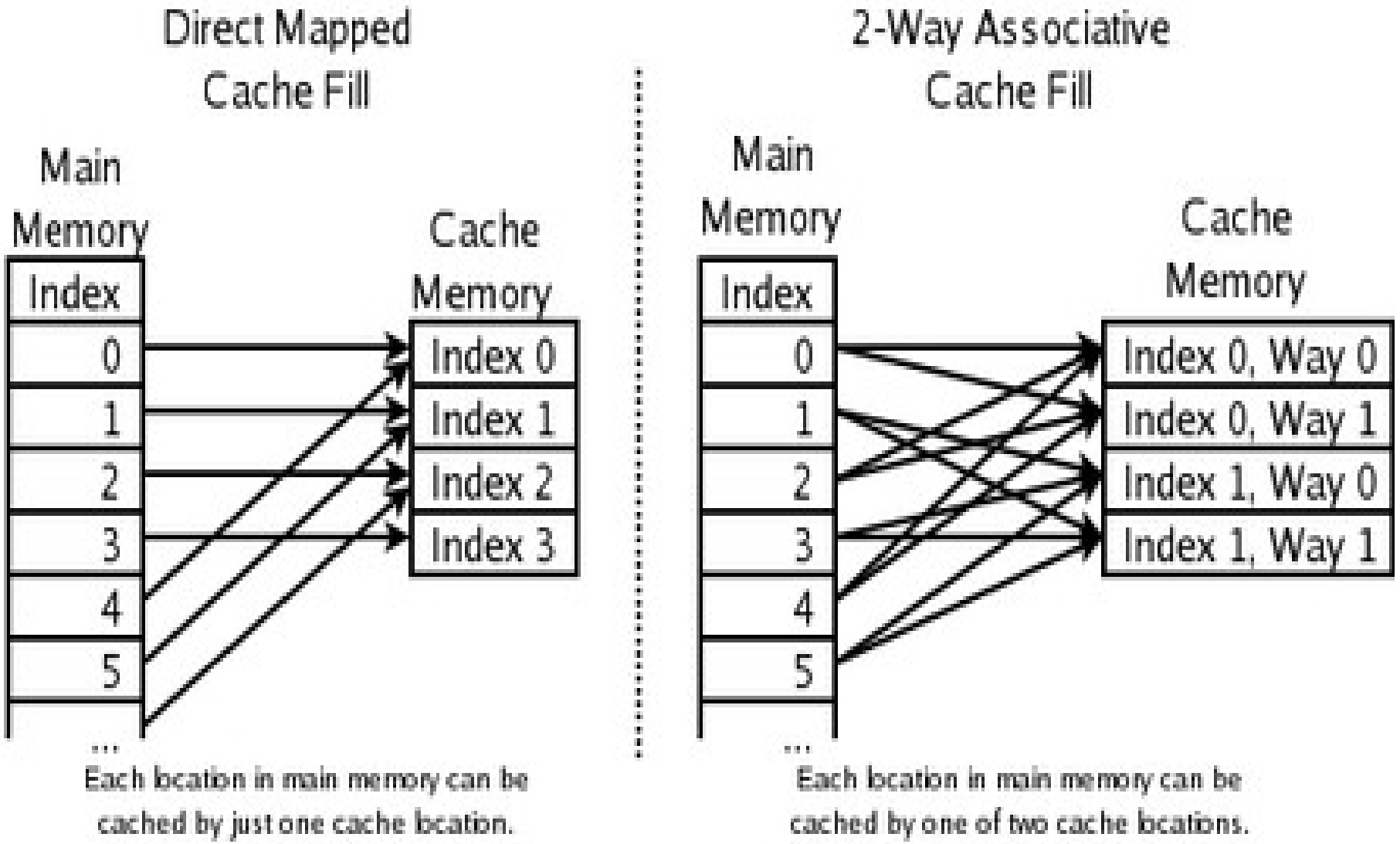
Important concepts about cache

- CACHE LINES
  - Caches are split into segments called cache lines, which are typically 4 or 8 words long. When a piece of data is fetched from either a higher level cache or local memory, an entire cache line is loaded.
- CACHE SIZE:
  - the overall dimension of the cache
- CACHE HIT:
  - It happens when the cpu is asking for a data and this data is found in the cache
- CACHE MISS:
  - It happens when the cpu is asking for a data and this data is NOT found in the cache

Cache layout

- Direct mapped:
- consecutive locations in memory are mapped to consecutive cache lines in the cache
- N-way set associative (a location in memory can map to any of N different cache lines).
- Direct mapped caches are easier to implement, but set associative caches are generally considered to be superior because they have less potential for cache thrashing.

direct mapped vs 2-way associative caches

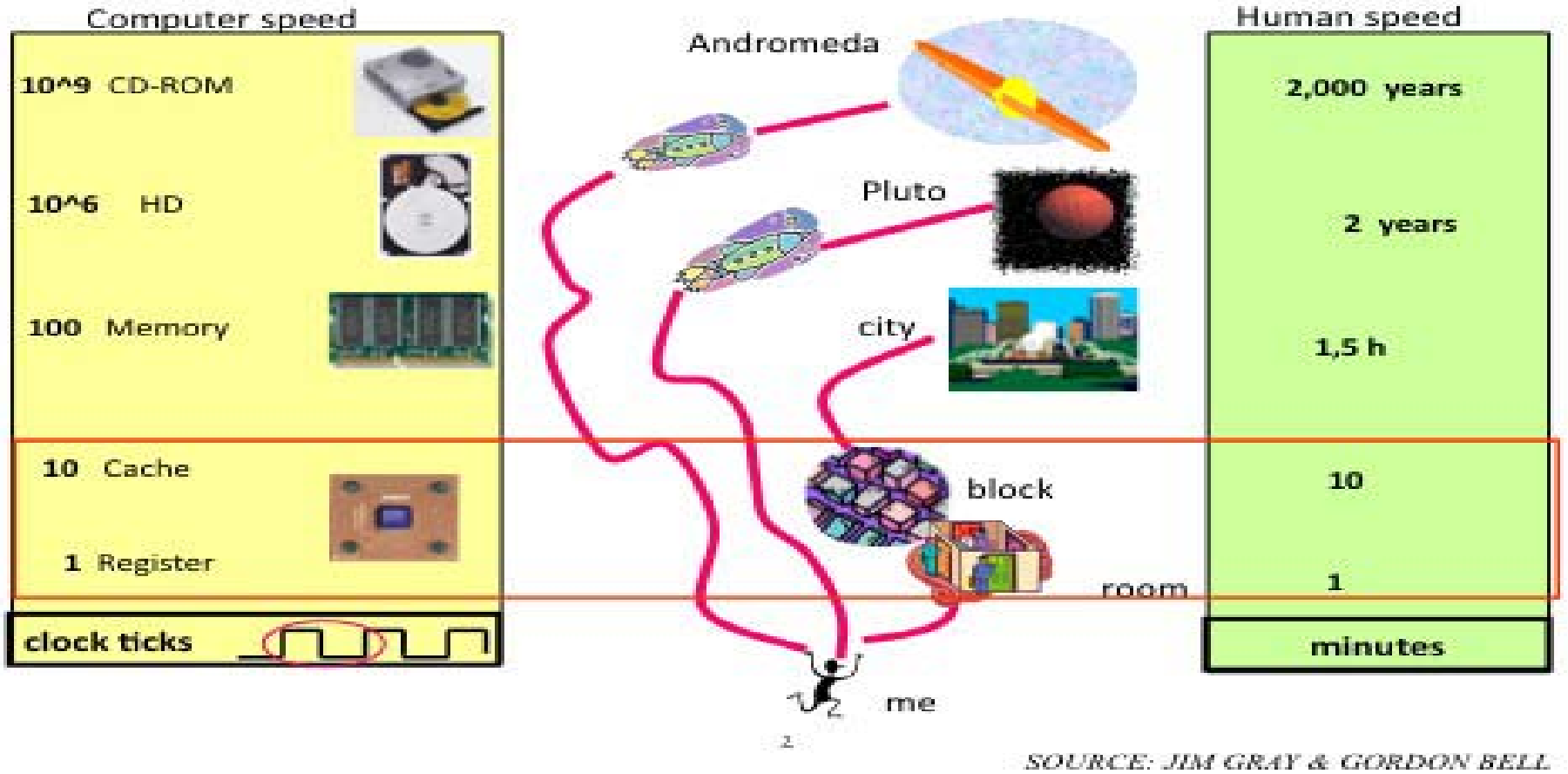


Hierarchical Memory and Latency

- The key to hierarchical memory is that going down each level of the hierarchy introduces approximately an order of magnitude more latency than the previous level.
- Actual latencies for an Nehalem (2.93GHz):
  - L1 data cache: 4 CPs
  - L2 cache: 10 CPs
  - L3 cache: 40 Cps
  - Local memory: ~ 200Cps

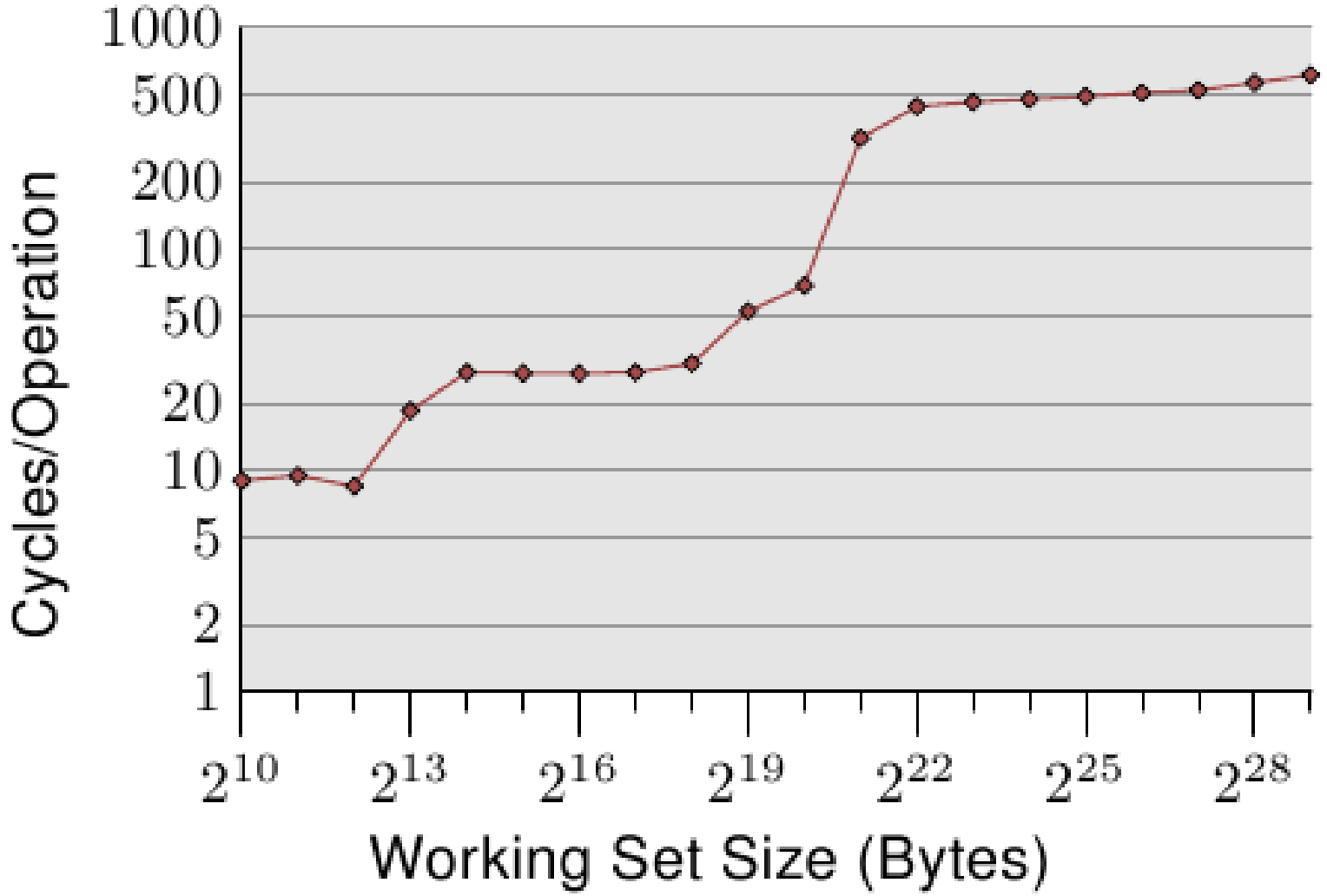
  (number from http://www.behardware.com/articles/733-4/report-intel-nehalem-architecture.html)

let's do some analogy...



SOURCE: JIM GRAY & GORDON BELL

how fast/large are the caches ?



Single core vs dual core and memory hierarchy:

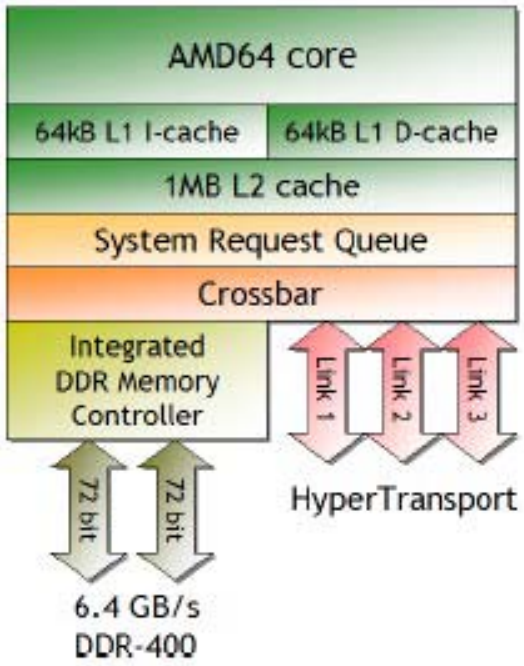


Figure 1: Single core AMD64 block diagram

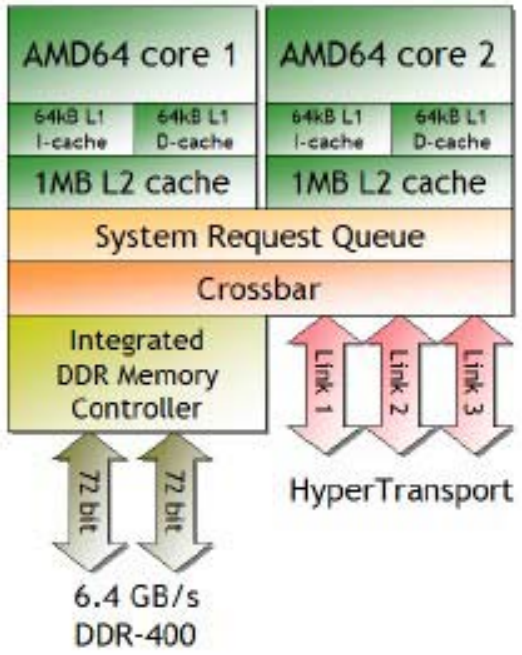
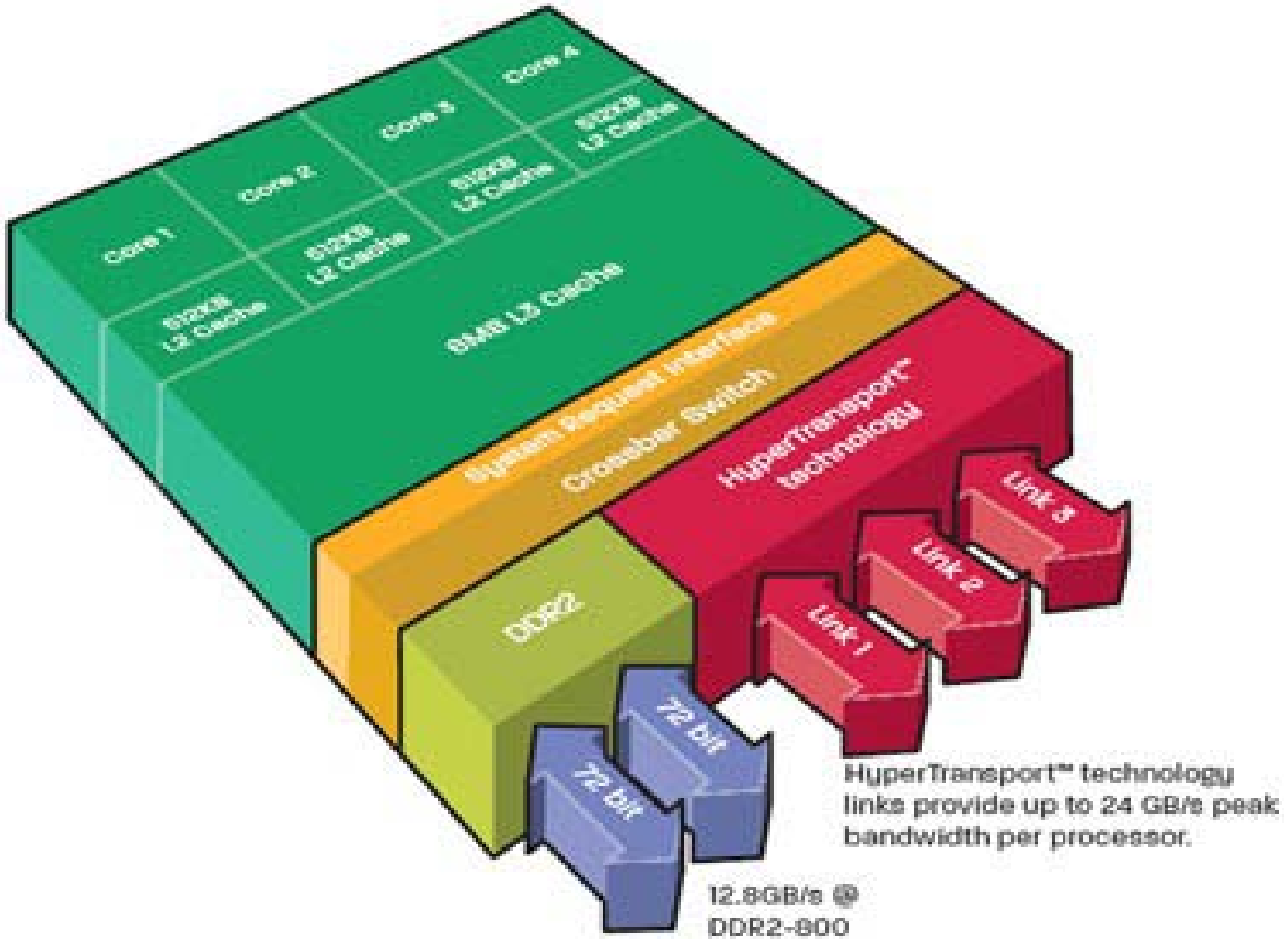


Figure 2: Dual core AMD64 block diagram

BANDWIDTH TOWARD LOCAL MEMORY IS SHARED AMONG CORES !

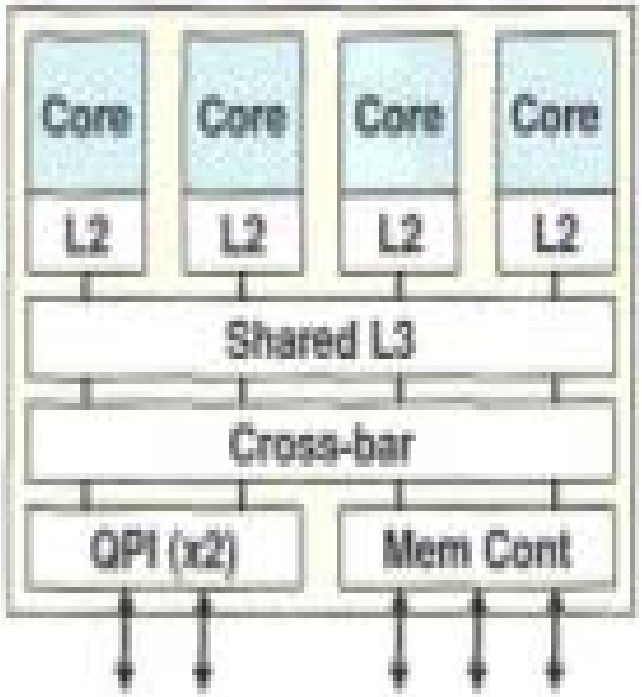
Barcelona (shangai) quad core architecture



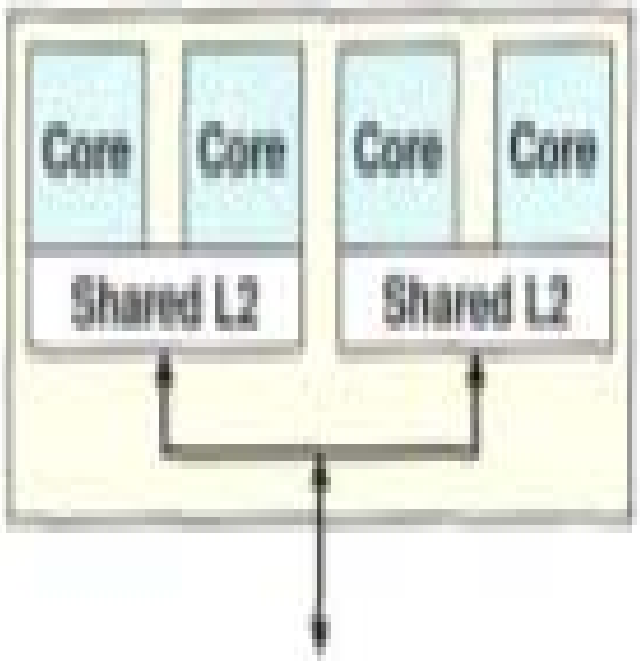
L3 CACHE IS SHARED AMONG CORES !

Intel platforms

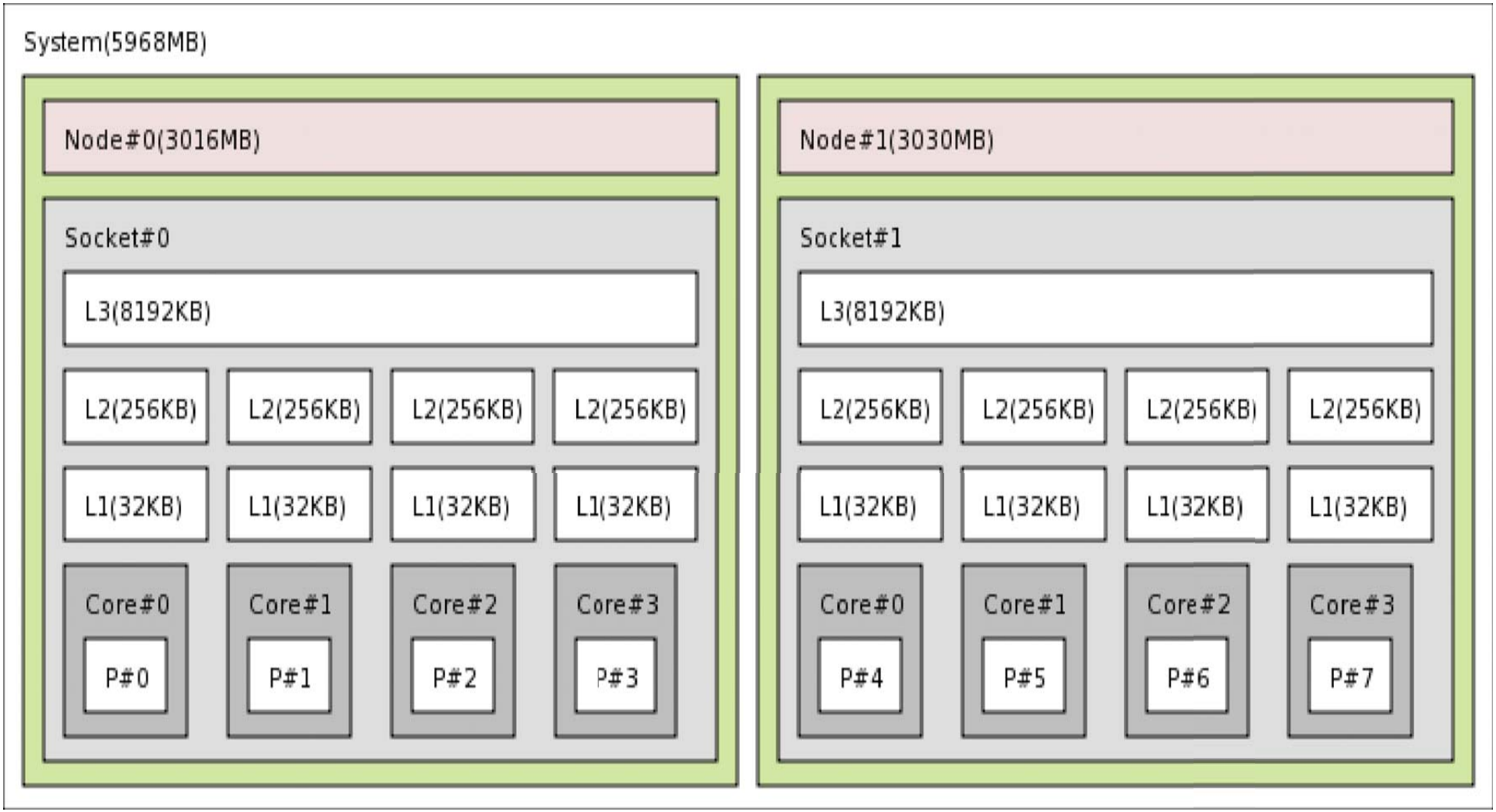
- Nehalem



- Xeon 54xx



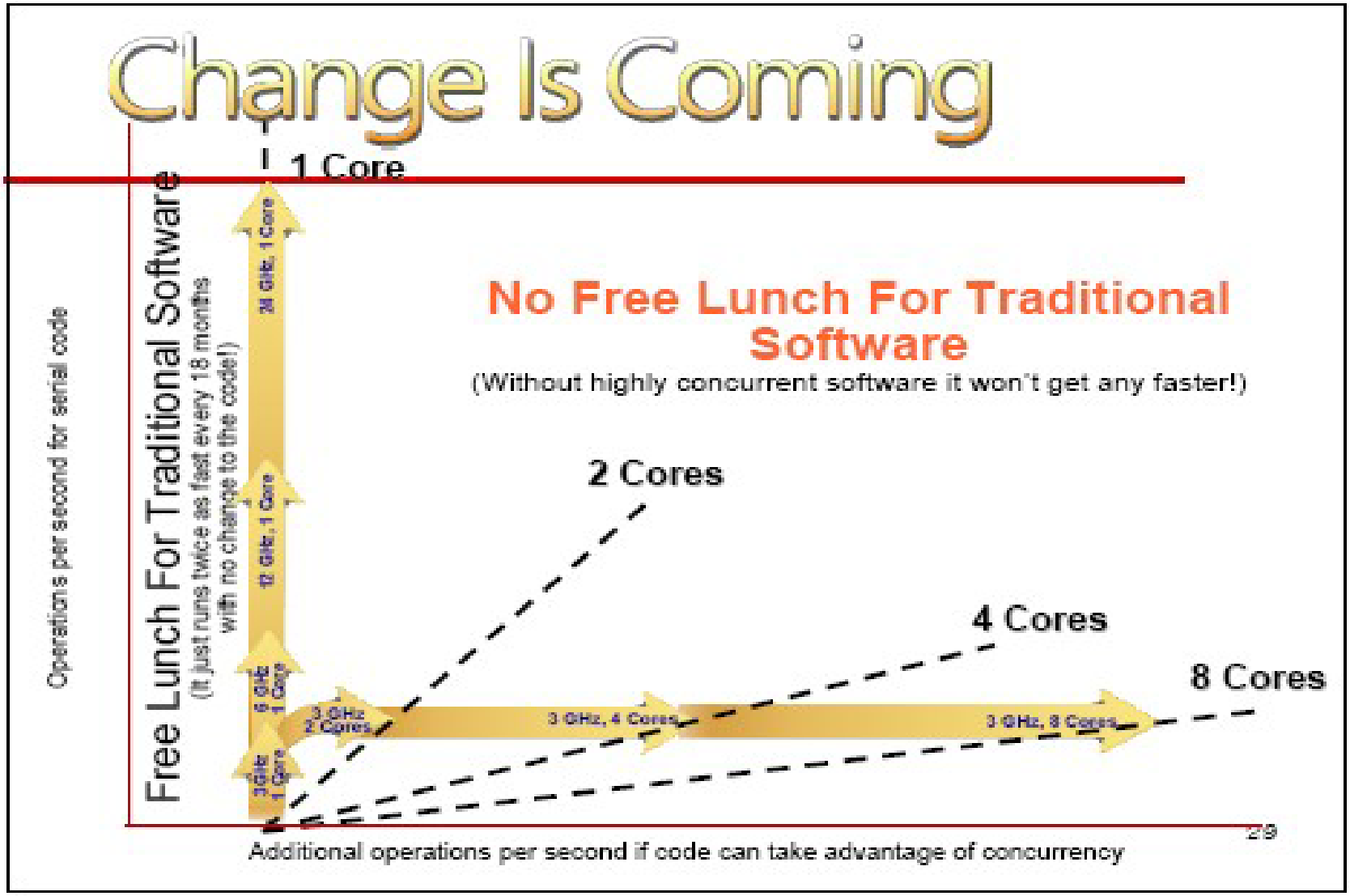
Nehalem layout



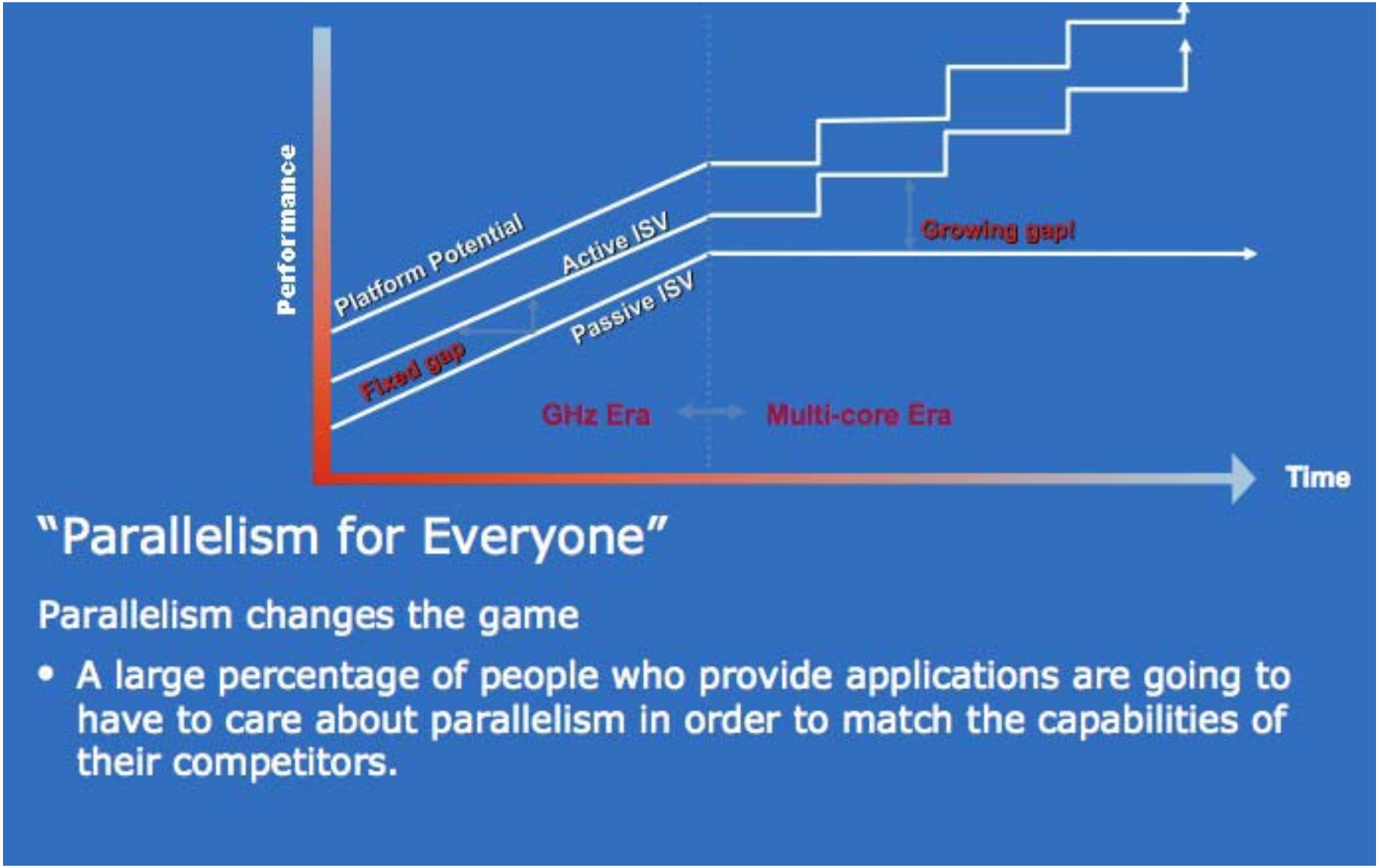
Challenges for multicore

- Relies on effective exploitation of multiple-thread parallelism
  - Need for parallel computing model and parallel programming model
- Aggravates memory wall problem
  - Memory bandwidth
    - Way to get data out of memory banks
    - Way to get data into multi-core processor array
    - Memory latency
  - Cache sharing

single Core VS Multiple core (from J.Dongarra talk)



a picture from Intel..



Conclusions

- Modern architectures have a high degree of parallelism some time hidden to the user
- In order to optimize on them you should be aware of this.
- Wall in memory should be taken into account if you are looking for performance
  - SMP is not always valid: NUMA
  - not only RAM is shared but also L2/L3 Caches