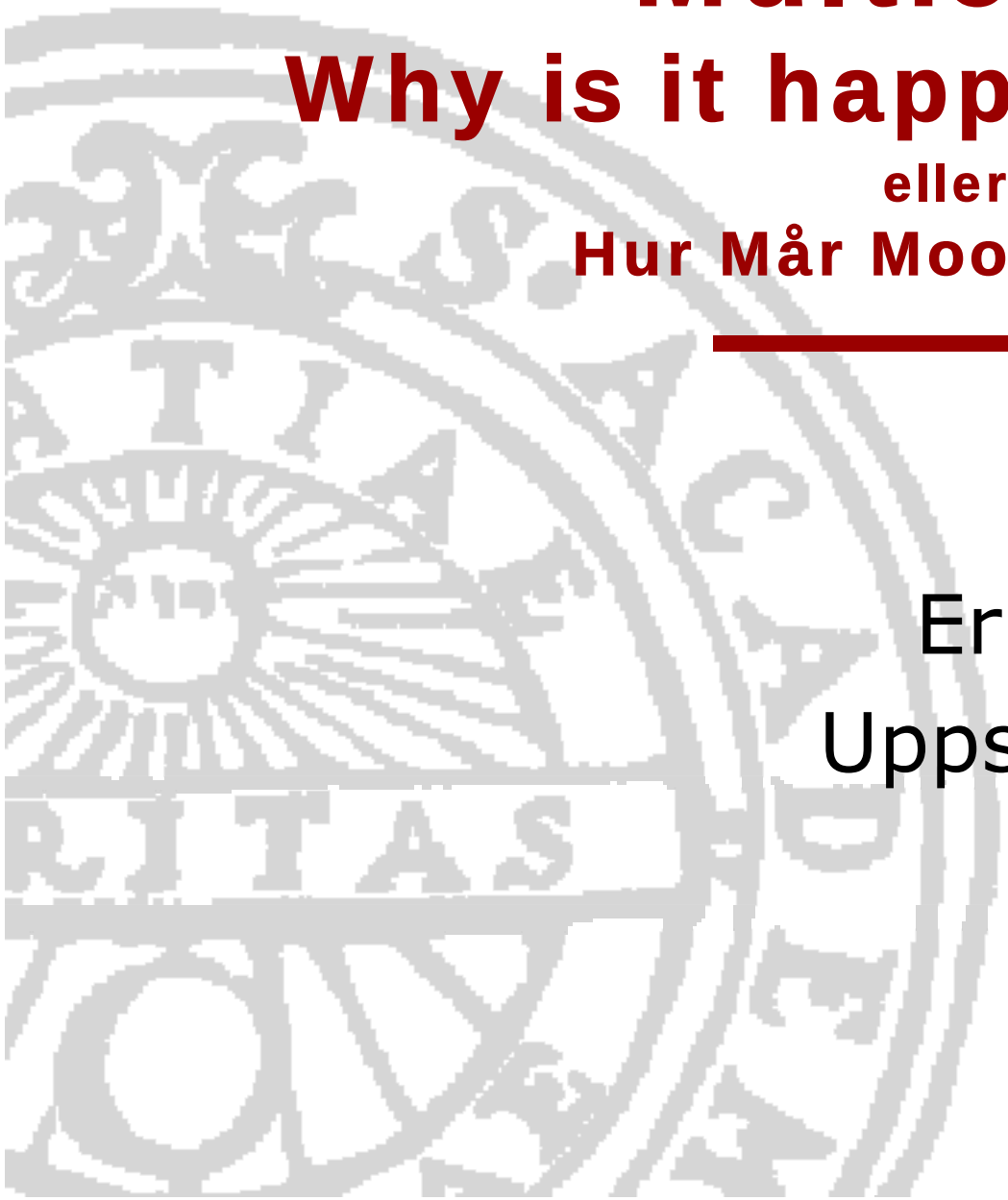
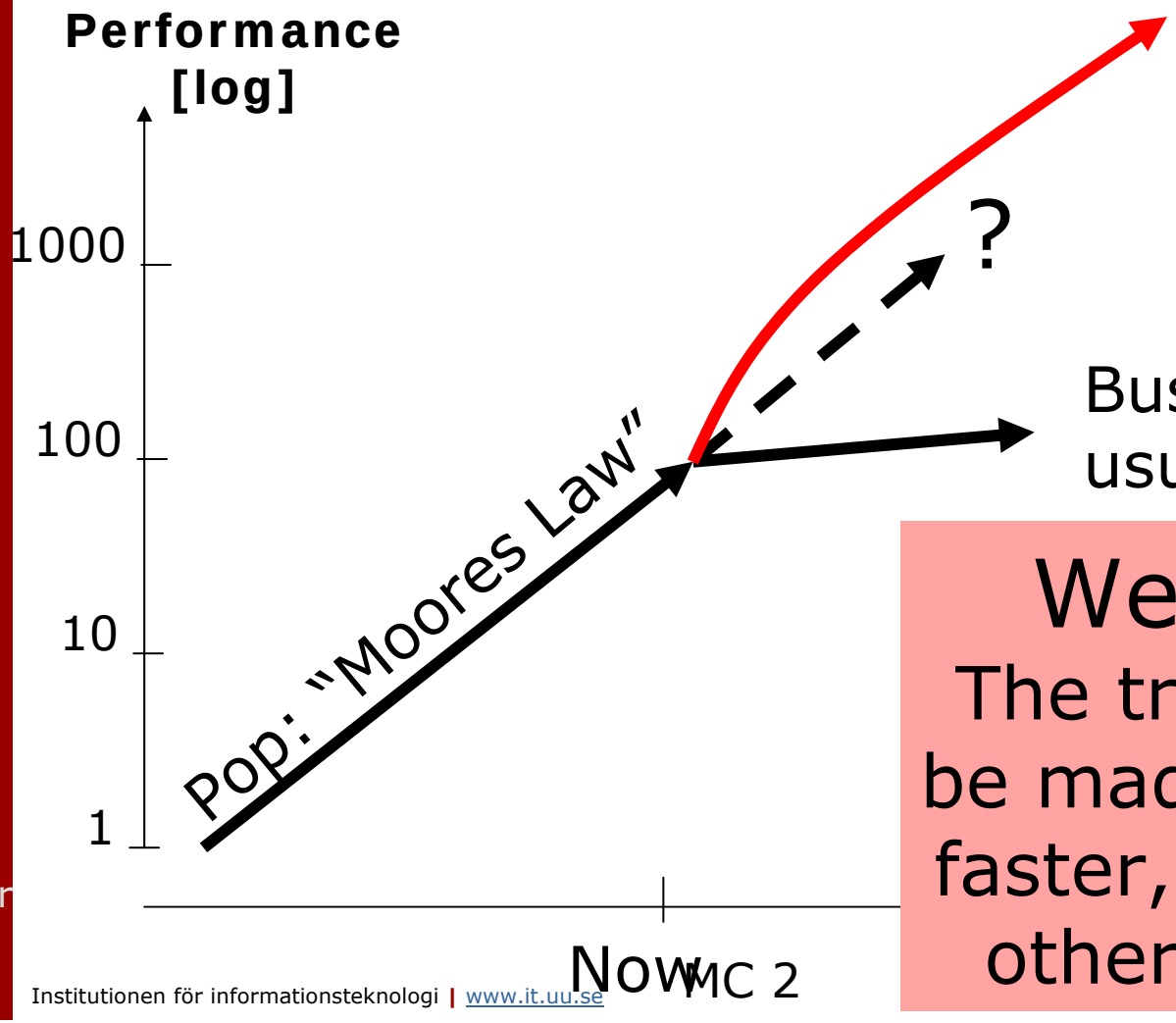


Multicore:
Why is it happening now?
eller
Hur Mår Moore's Lag?

Erik Hagersten
Uppsala Universitet

Are we hitting the wall now?

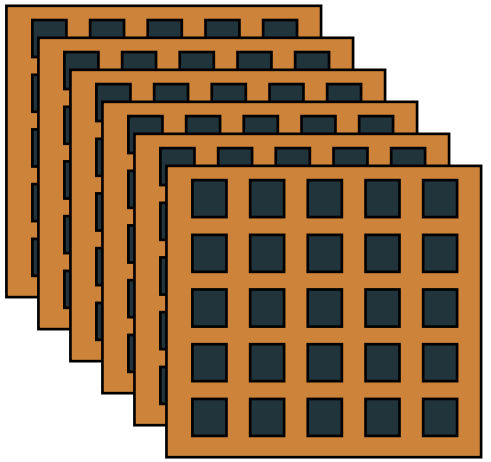
Pop: Can the transistors be made even smaller and faster?



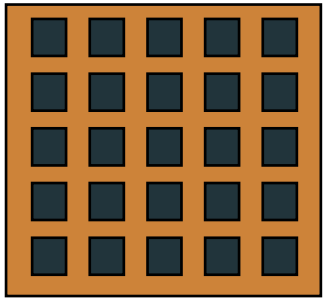
Well uhm …

The transistors can be made smaller and faster, but there are other problems ☹

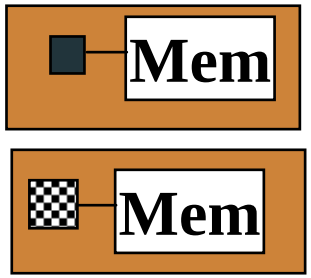
Darling, I shrunk the computer



Mainframes



Super Minis:



Microprocessor:

Multicore: Many CPUs on a chip!

UPPSALA UNIVERSITET

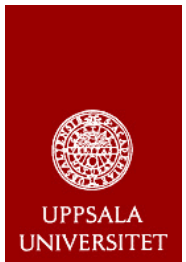
Multi-core CPUs:

- **Ageia** PhysX, a multi-core physics processing unit.
- **Ambric** Am2045, a 336-core Massively Parallel Processor Array (MPPA)
- **AMD**
  - Athlon 64, Athlon 64 FX and Athlon 64 X2 family, dual-core desktop processors.
  - Opteron, dual- and quad-core server/workstation processors.
  - Phenom, triple- and quad-core desktop processors.
  - Sempron X2, dual-core entry level processors.
  - Turion 64 X2, dual-core laptop processors.
  - Radeon and FireStream multi-core GPU/GPGPU (10 cores, 16 5-issue wide superscalar stream processors per core)
- **ARM** MPCore is a fully synthesizable multicore container for ARM9 and ARM11 processor cores, intended for high-performance embedded and entertainment **applications**.
- **Azul** Systems Vega 2, a 48-core processor.
- **Broadcom** SiByte SB1250, SB1255 and SB1455.
- **Cradle** Technologies CT3400 and CT3600, both multi-core DSPs.
- **Cavium** Networks Octeon, a 16-core MIPS MPU.
- **HP** PA-8800 and PA-8900, dual core PA-RISC processors.
- **IBM**
  - POWER4, the world's first dual-core processor, released in 2001.
  - POWER5, a dual-core processor, released in 2004.
  - POWER6, a dual-core processor, released in 2007.
  - PowerPC 970MP, a dual-core processor, used in the Apple Power Mac G5.
  - Xenon, a triple-core, SMT-capable, PowerPC microprocessor used in the Microsoft Xbox 360 game console.
- **IBM**, Sony, and Toshiba Cell processor, a nine-core processor with one general purpose PowerPC core and eight specialized SPUs (Synergystic Processing Unit) **optimized** for vector operations used in the Sony PlayStation 3.
- **Infineon** Danube, a dual-core, MIPS-based, home gateway processor.
- **Intel**
  - Celeron Dual Core, the first dual-core processor for the budget/entry-level market.
  - Core Duo, a dual-core processor.
  - Core 2 Duo, a dual-core processor.
  - Core 2 Quad, a quad-core processor.
  - Core i7, a quad-core processor, the successor of the Core 2 Duo and the Core 2 Quad.
  - Itanium 2, a dual-core processor.
  - Pentium D, a dual-core processor.
  - Teraflops Research Chip (Polaris), an 3.16 GHz, 80-core processor prototype, which the company says will be released within the next five years[6].
  - Xeon dual-, quad- and hexa-core processors.
- **IntellaSys** seaForth24, a 24-core processor.
- **Nvidia**
  - GeForce 9 multi-core GPU (8 cores, 16 scalar stream processors per core)
  - GeForce 200 multi-core GPU (10 cores, 24 scalar stream processors per core)
  - Tesla multi-core GPGPU (8 cores, 16 scalar stream processors per core)
- Parallax Propeller P8X32, an eight-core microcontroller.
- picoChip PC200 series 200-300 cores per device for DSP & wireless
- Rapport Kilocore KC256, a 257-core microcontroller with a PowerPC core and 256 8-bit "processing elements".
- Raza Microelectronics XLR, an eight-core MIPS MPU
- **Sun Microsystems**
  - UltraSPARC IV and UltraSPARC IV+, dual-core processors.
  - UltraSPARC T1, an eight-core, 32-thread proc
  - UltraSPARC T2, an eight-core, 64-concurrent
- Texas Instruments TMS320C80 MVP, a five-core mult
- Tilera TILE64, a 64-core processor
- XMOS Software Defined Silicon quad-core XS1-G4

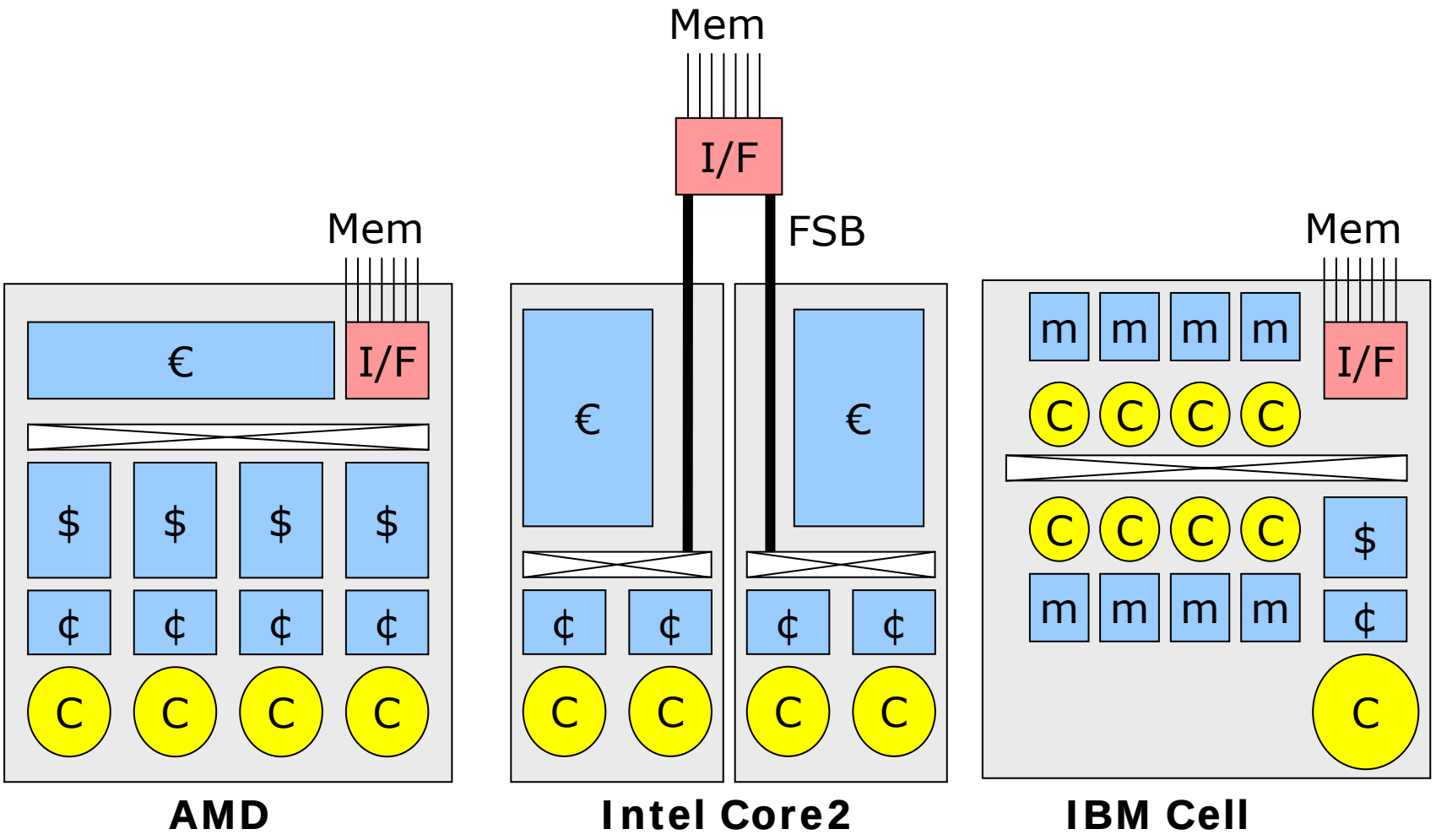
[source: Wikipedia]

UPPSALA UNIVERSITET

High-performance single-core CPUs:



Many shapes and forms…

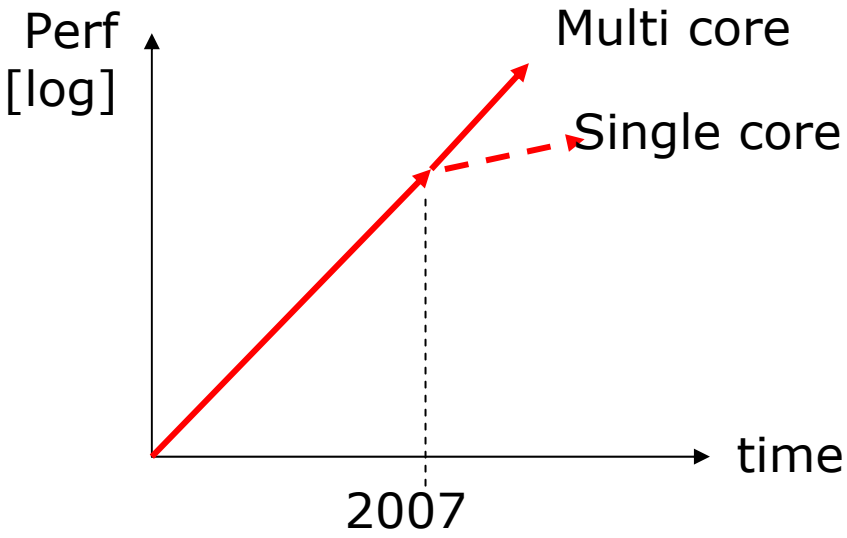


UPPSALA UNIVERSITET

Outline

- Why multicore now?
- Technology reasons
- Performance bottlenecks in MCs
- Commercial offerings
- Reflection for the future

Everybody is doing is! But, why now?



1. Not enough ILP to get payoff from using more transistors
2. Signal propagation delay » transistor delay
3. Power consumption $P_{dyn} \sim C \bullet f \bullet V^2$

UPPSALA UNIVERSITET

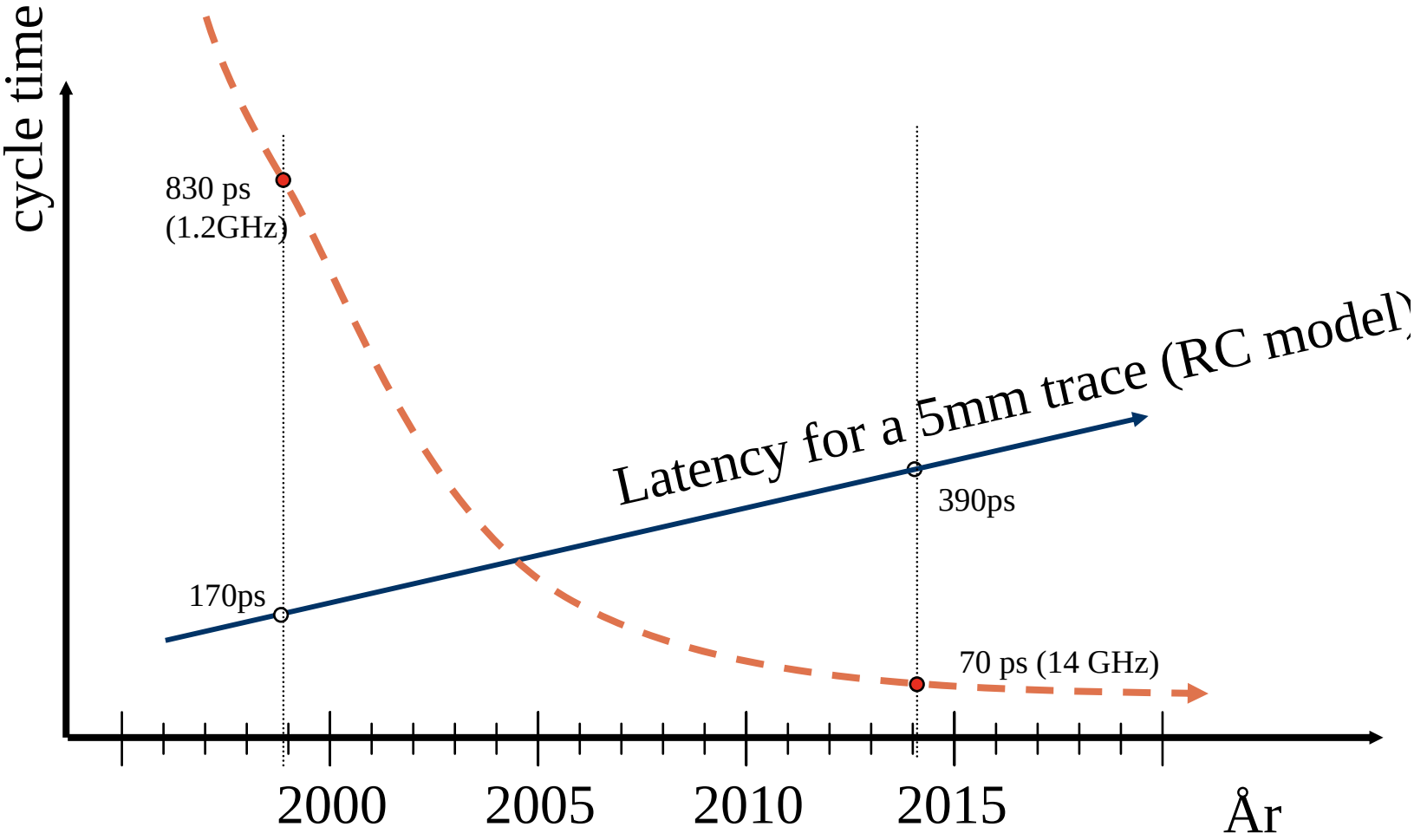
Microprocessors today: What it takes to run one progra

Exploring ILP (instruction-level

- Faster clocks ➔ Deep pip
- Superscalar Pipelines
- Branch Prediction
- Prefetching
- Out-of-Order
- Trace Cac
- Specula
- Pred
- A ddress Table
- s Stack
- .

Bad News #1:
We have already explored most ILP
(instruction-level parallelism)

Bad News #2
Looong wire delay →slow CPUs

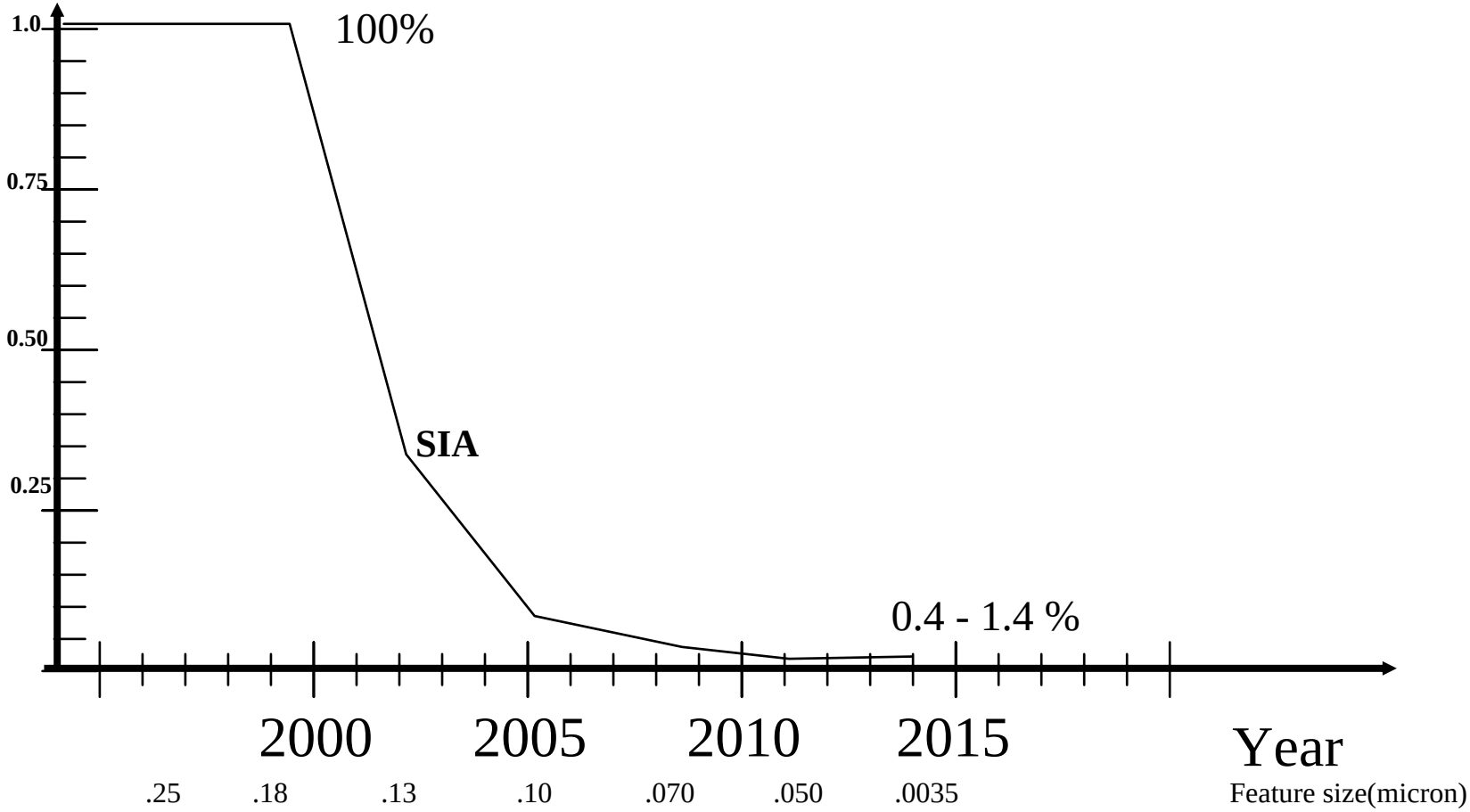


Quantitative data and trends according to V. Agarwal et al., ISCA 2000

***Based on SIA* (Semiconductor Industry Association) prediction, 1999**

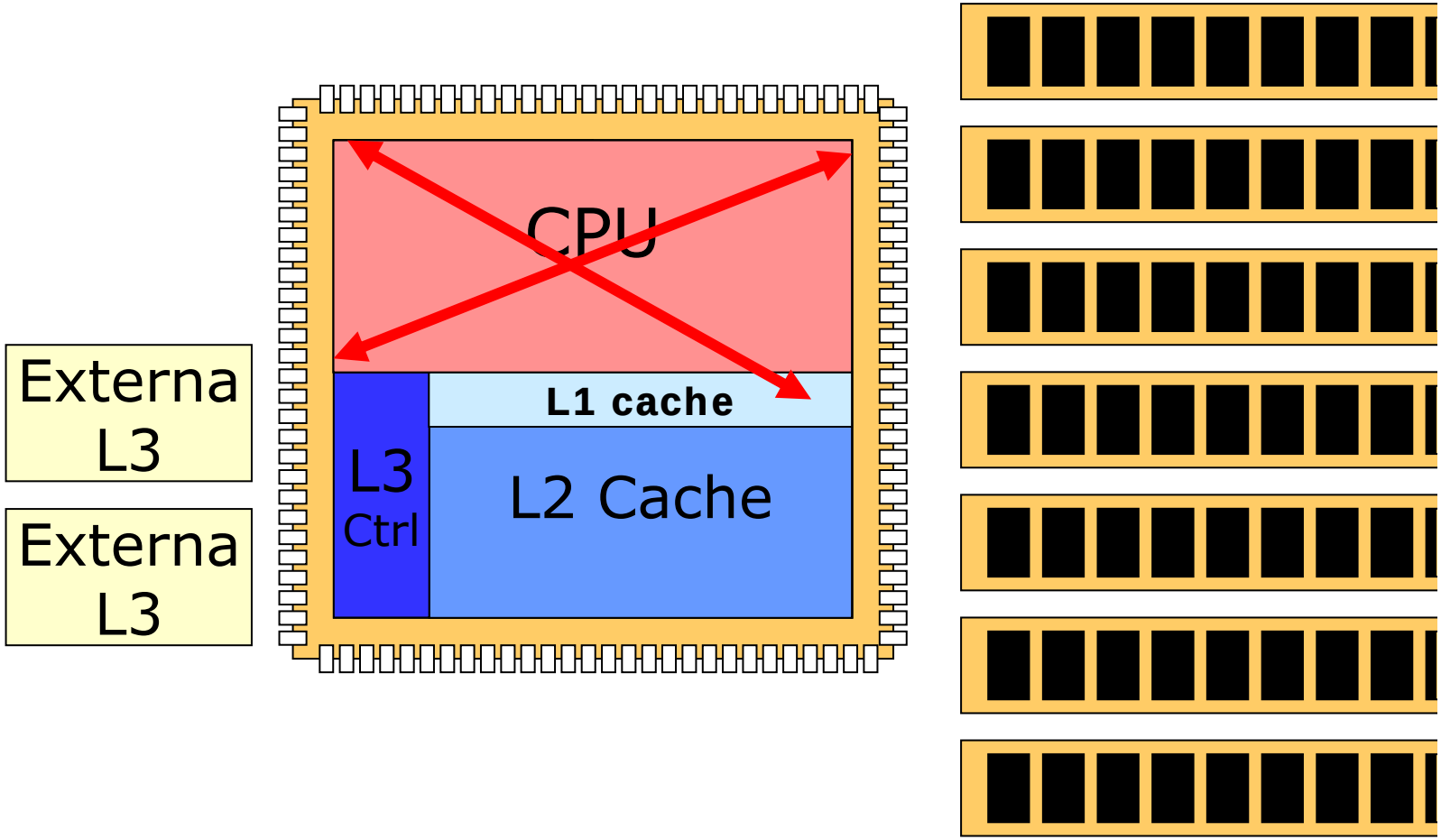
Bad News #2
Looong wire delay →slow CPUs

Span -- Fraction of chip reachable in one cycle



Bad News #2: wire delay

➔ Multiple small CPUs with private L1$

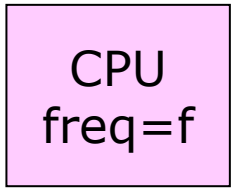


UPPSALA UNIVERSITET

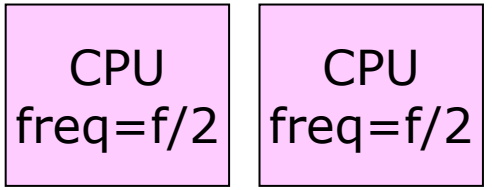
Bad News #3: Power consumption

➔ Lower frequency ➔ lower voltage

$$P_{dyn} = C * f * V^2 \approx \text{area} * \text{freq} * \text{voltage}^2$$

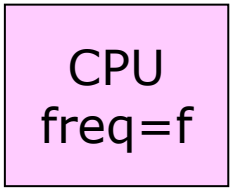


vs.

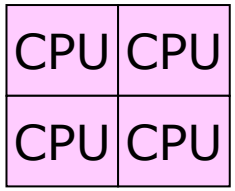


$P_{dyn}(C, f, V) = CfV^2$

$P_{dyn}(2C, f/2, <V) < CfV^2$



vs.



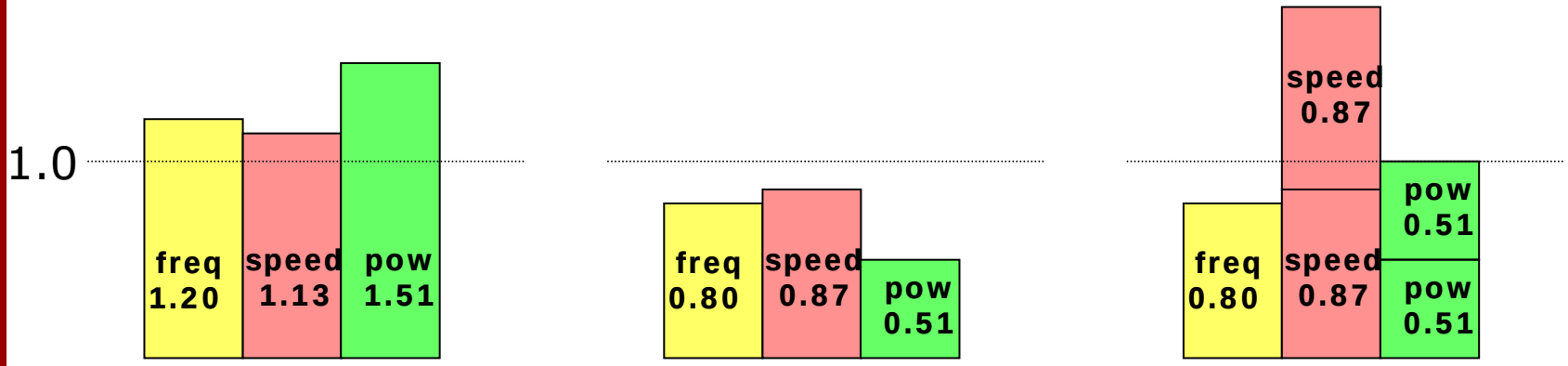
freq = f/2

$P_{dyn}(C, f, V) = CfV^2$

$P_{dyn}(C, f/2, <V) < ½\ CfV^2$

UPPSALA UNIVERSITET

Example: Freq. Scaling

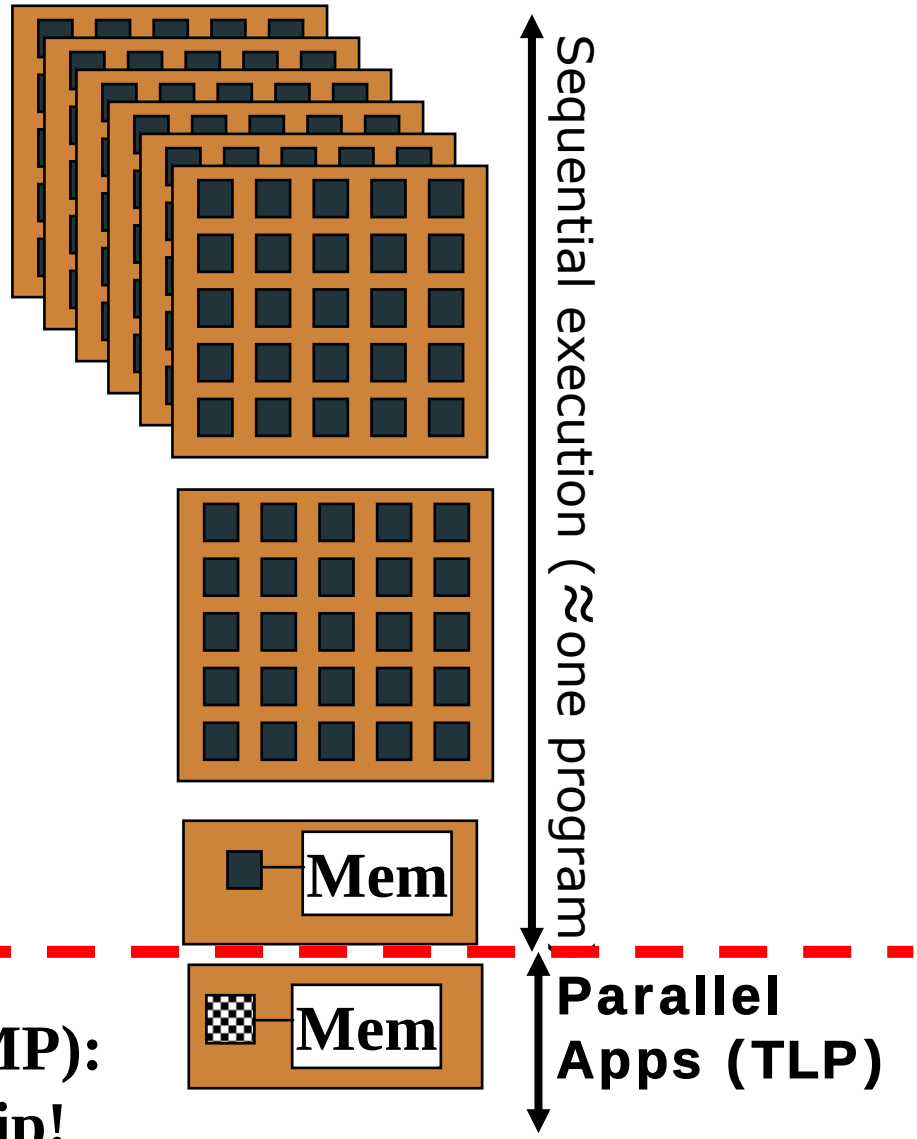


20% higher freq.

20% lower freq.

20% lower freq.
Two cores

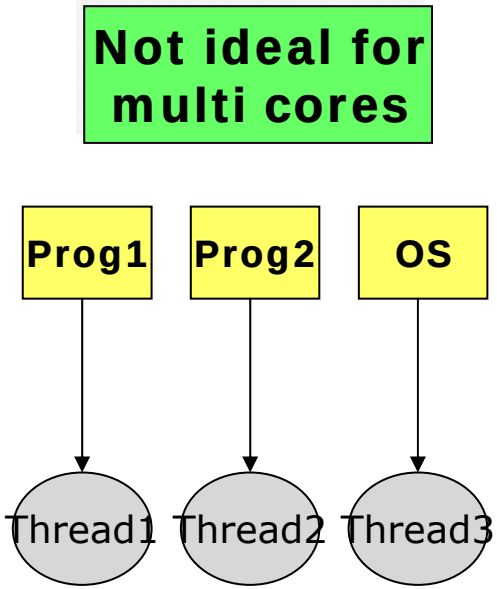
Darling, I shrunk the computer

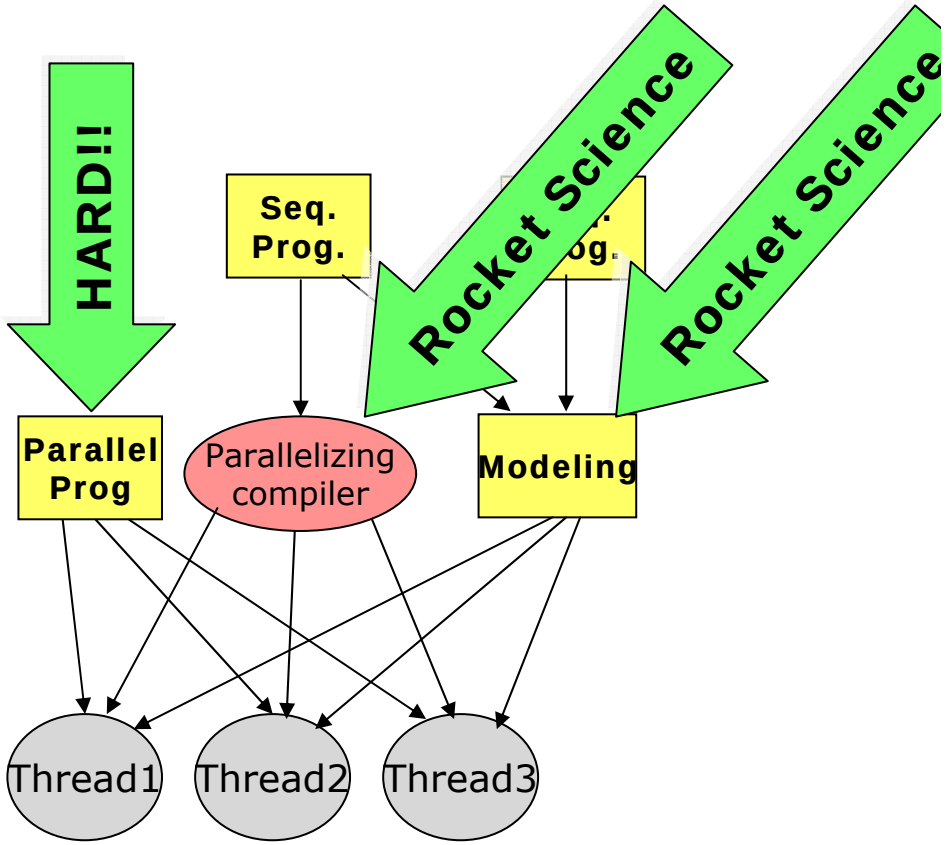


Paradigm Shift

Chip Multiprocessor (CMP):
A multiprocessor on a chip!

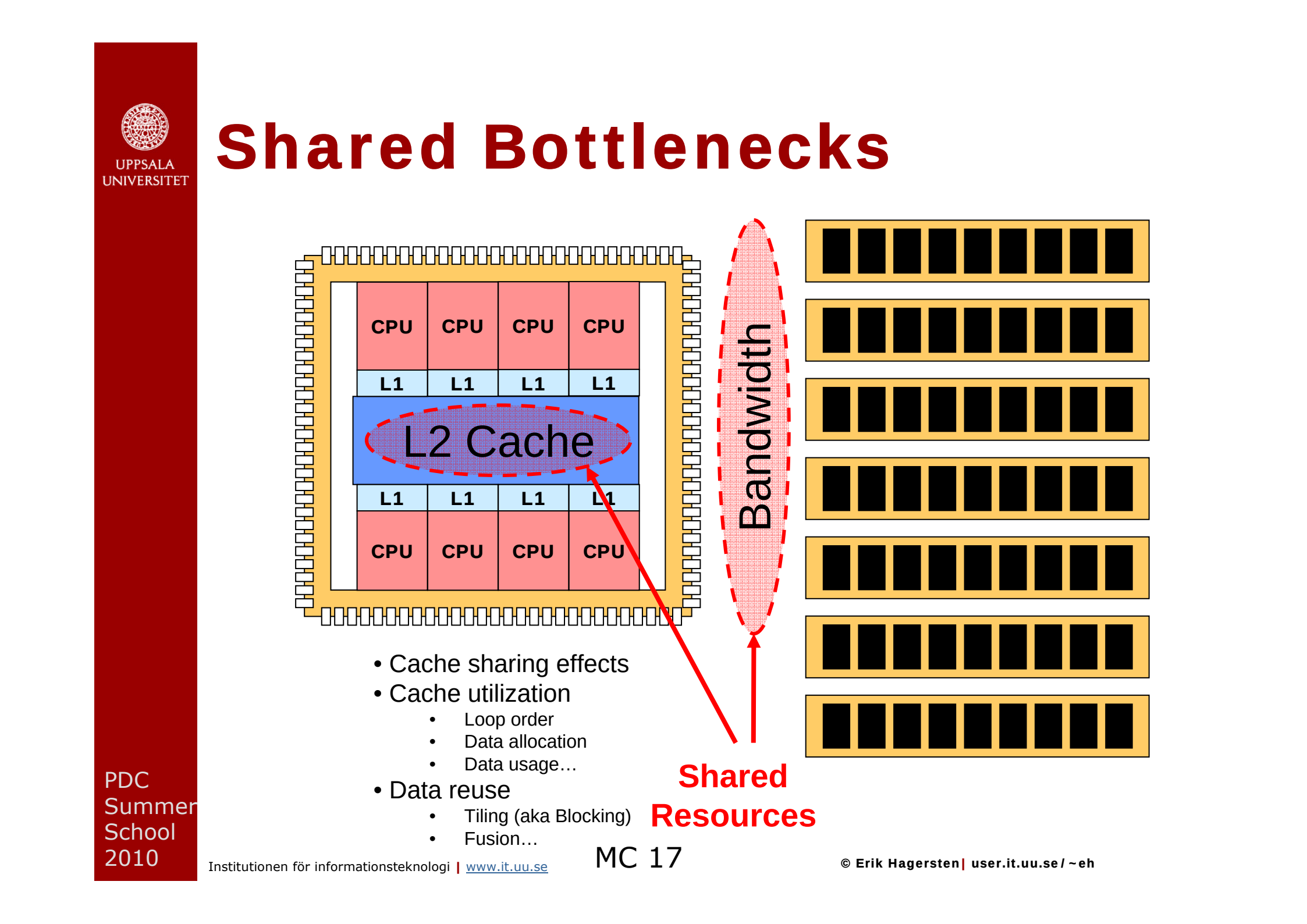
How to create threads?





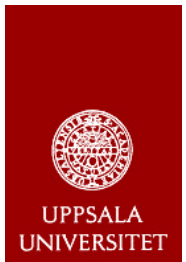
Throughput Computing
- "Multitasking"
- Virtualization
- Concurrency
- …

Capability Computing
- Weather simulation
- Computer games
- …



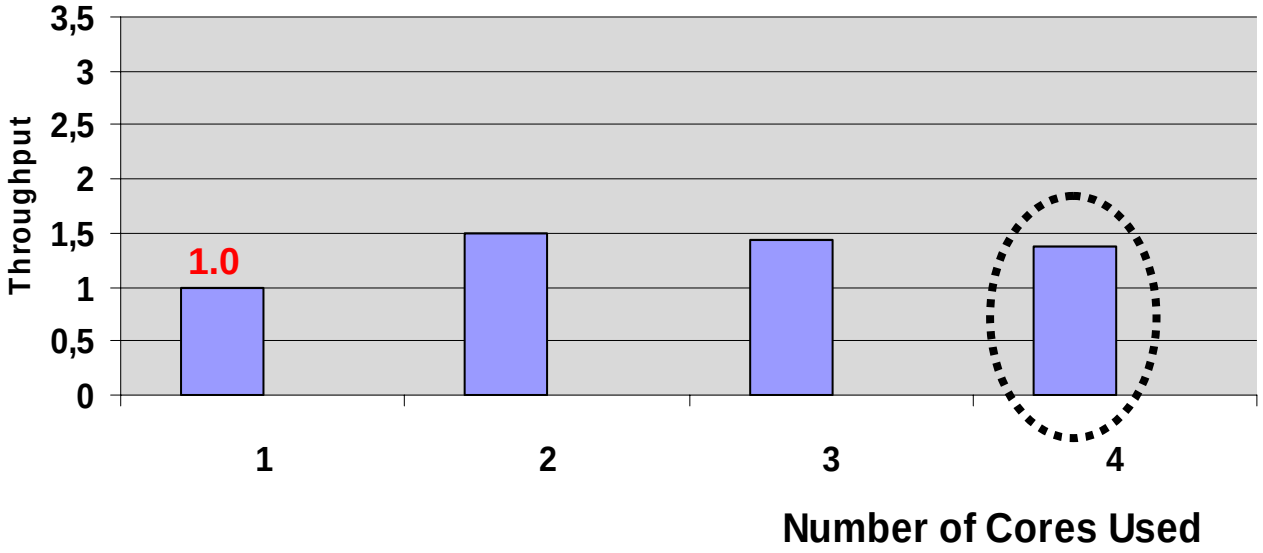
Shared Bottlenecks

- Cache sharing effects
- Cache utilization
  - Loop order
  - Data allocation
  - Data usage…
- Data reuse
  - Tiling (aka Blocking)
  - Fusion…

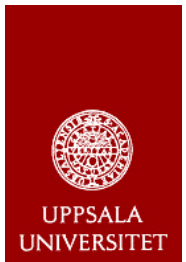


Example: Poor Throughput Scaling!

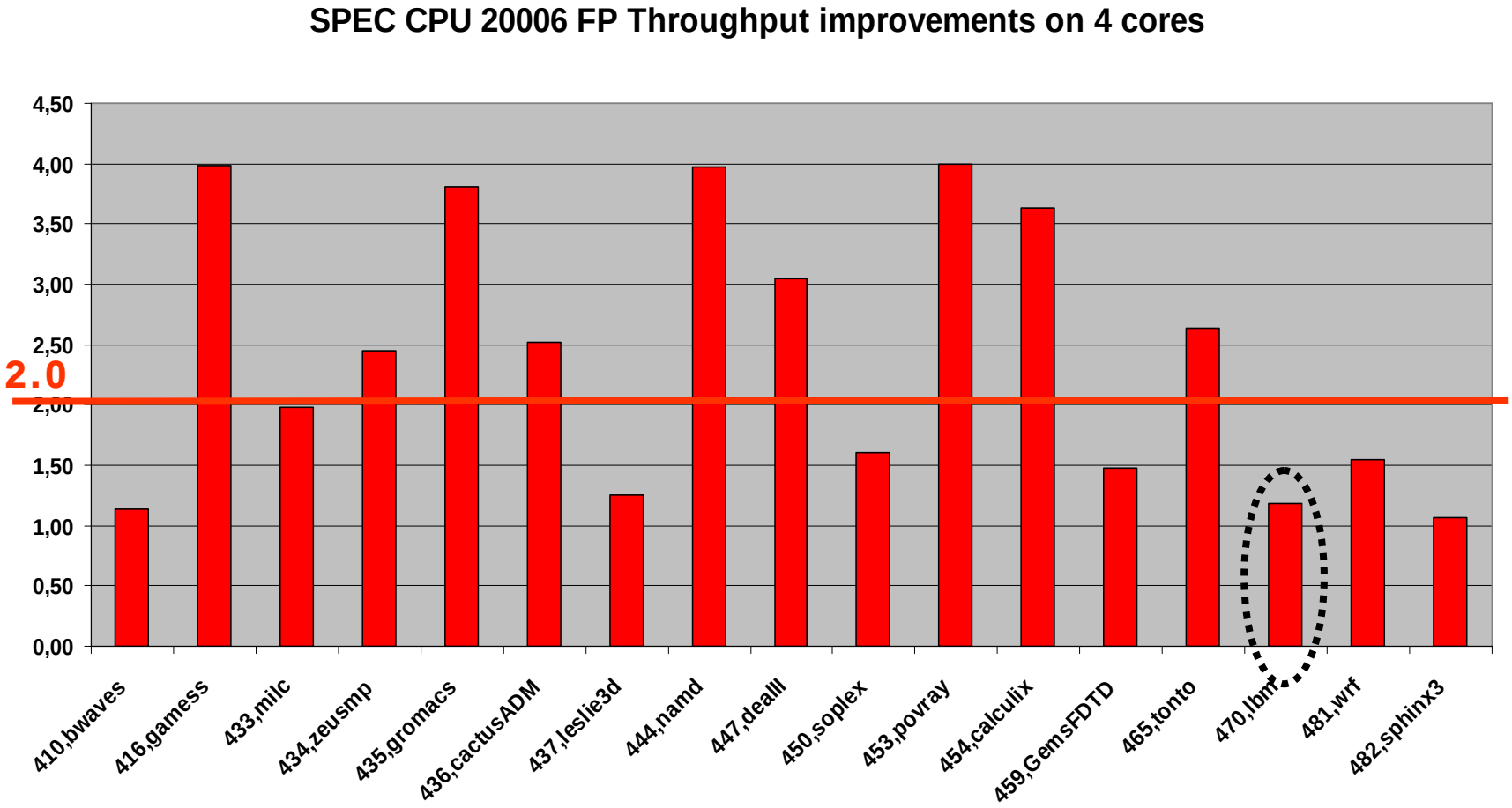
Example: 470.LBM
"Lattice Boltzmann Method" to simulate incompressible fluids in 3D



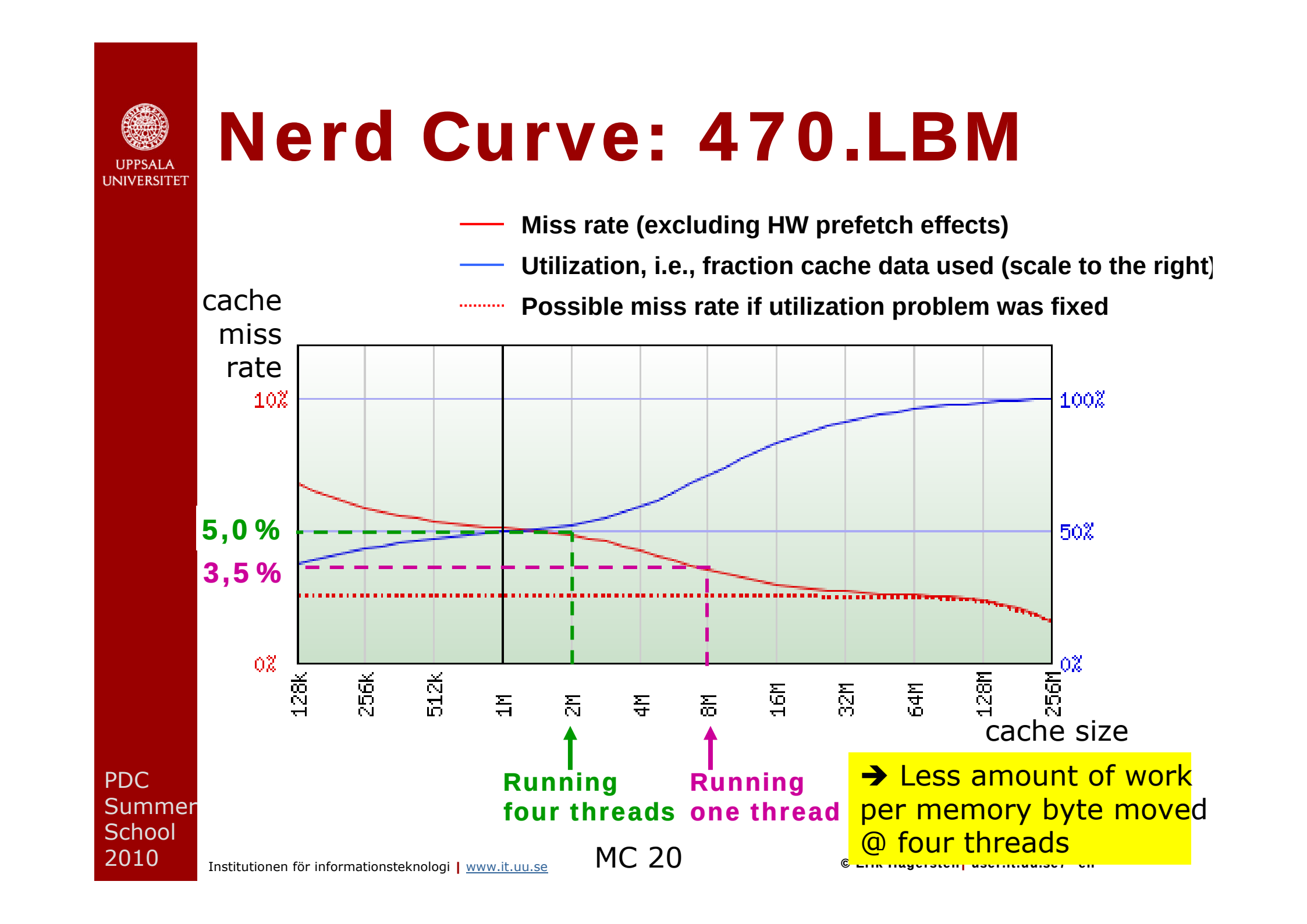
Throughput (as defined by SPEC):
Amount of work performed per time unit when several instances of the application is executed simultaneously.
Our TP study: compare TP improvement when you go from 1 core to 4 cores

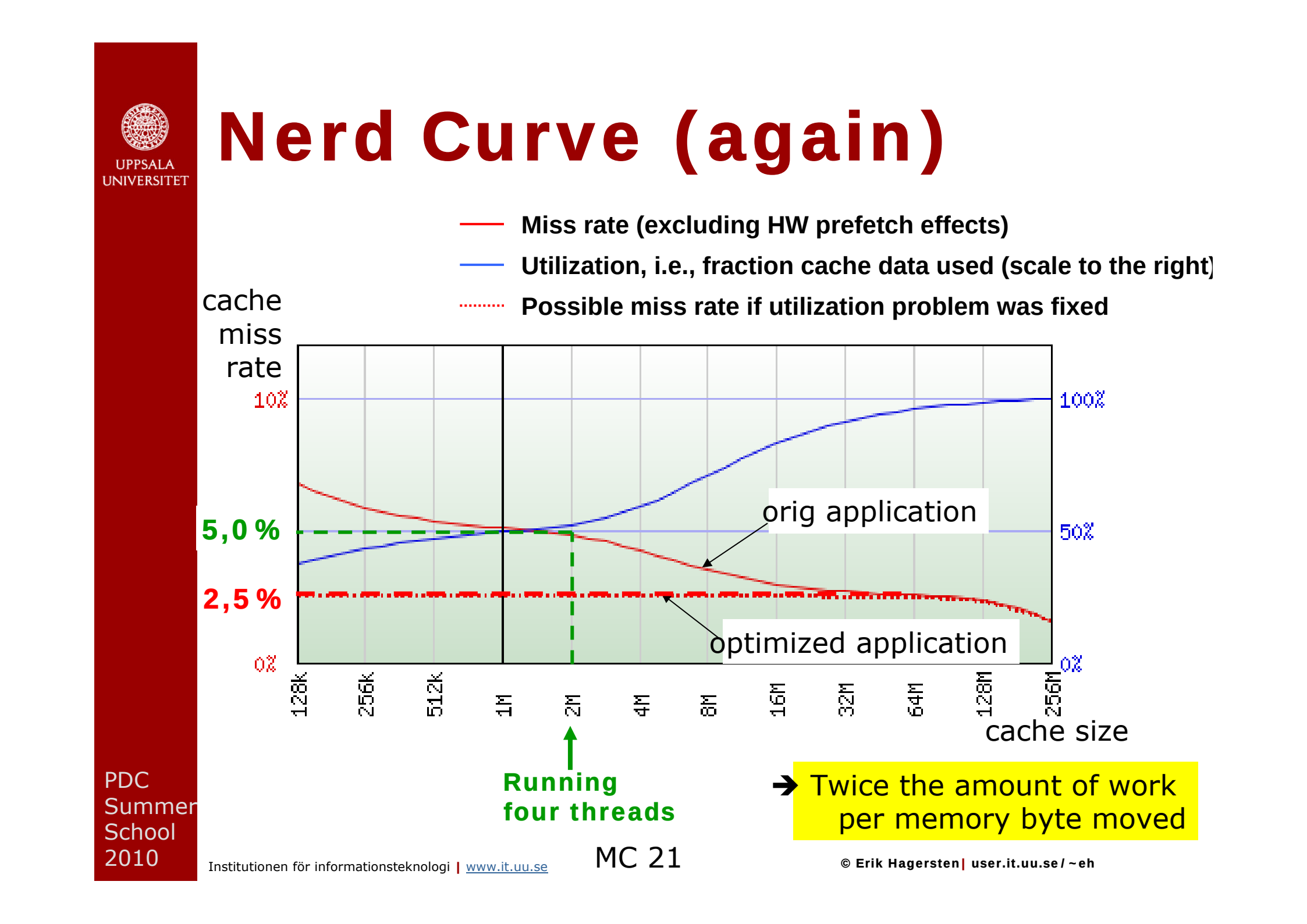


Throughput Scaling, More Apps



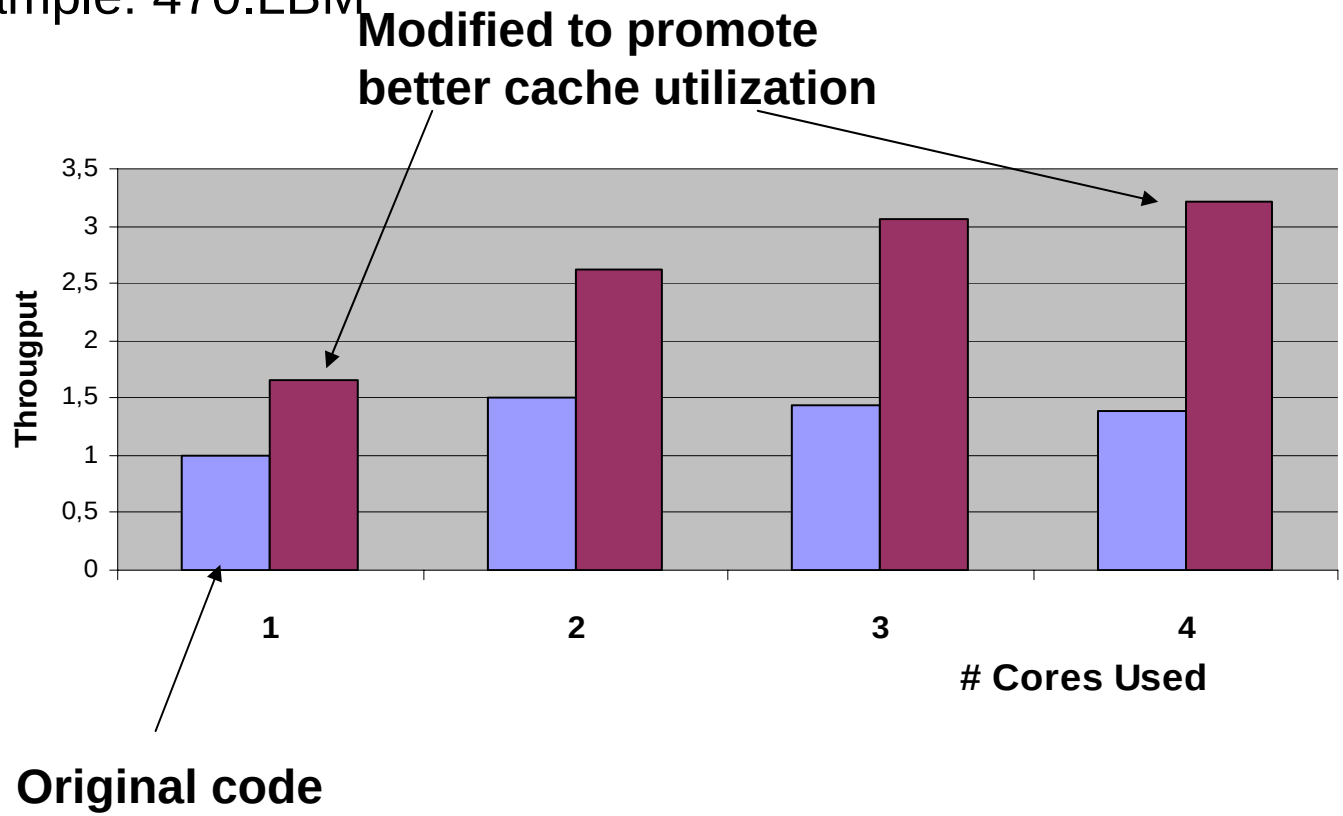
Intel X5365 3GHz "Core2", 1333 MHz FSB, 8MB L2.
(Based on data from the SPEC web)


Nerd Curve: 470.LBM
Miss rate (excluding HW prefetch effects)
Utilization, i.e., fraction cache data used (scale to the right)
Possible miss rate if utilization problem was fixed
cache miss rate
10%
5,0 %
3,5 %
0%
100%
50%
0%
128k
256k
512k
1M
2M
4M
8M
16M
32M
64M
128M
256M
cache size
Running four threads
Running one thread
➔ Less amount of work per memory byte moved @ four threads
UPPSALA UNIVERSITET
PDC Summer School 2010
Institutionen för informationsteknologi | www.it.uu.se
MC 20


Nerd Curve (again)
Miss rate (excluding HW prefetch effects)
Utilization, i.e., fraction cache data used (scale to the right)
Possible miss rate if utilization problem was fixed
cache miss rate
10%
5,0 %
2,5 %
0%
100%
50%
0%
orig application
optimized application
128k
256k
512k
1M
2M
4M
8M
16M
32M
64M
128M
256M
cache size
Running four threads
➔ Twice the amount of work per memory byte moved
UPPSALA UNIVERSITET
PDC Summer School 2010
Institutionen för informationsteknologi | www.it.uu.se
MC 21
© Erik Hagersten| user.it.uu.se/~eh

➔ Better Memory Usage!

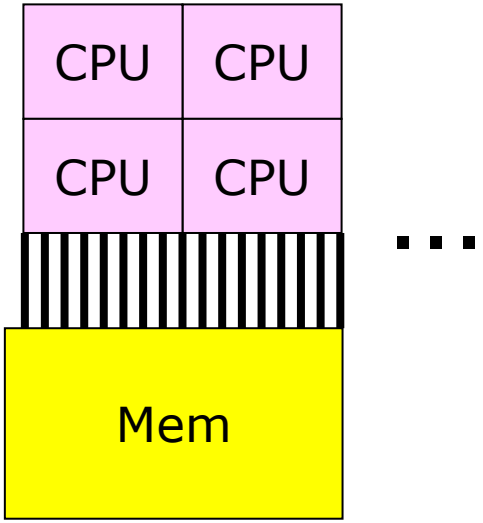
Example: 470.LBM



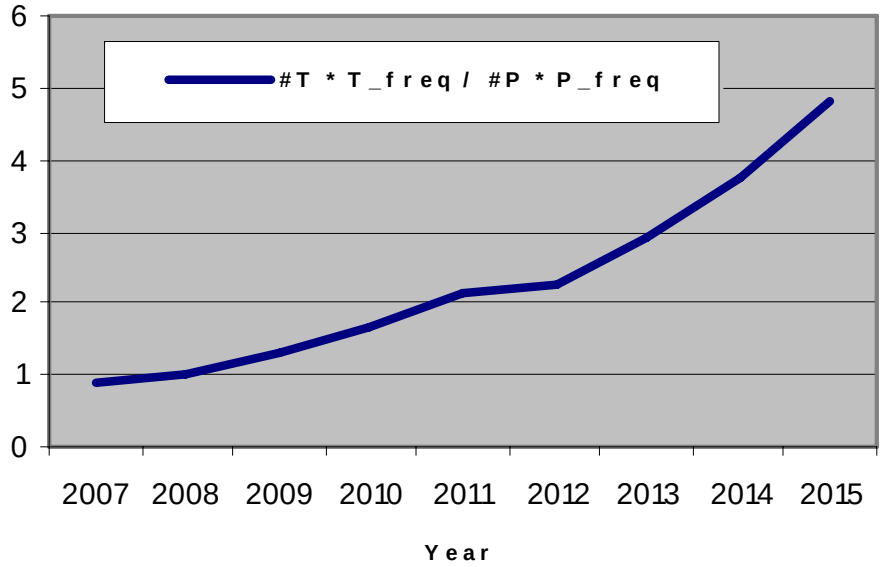
UPPSALA UNIVERSITET

BW in the Future?

#Cores ~ #Transistors



Computation vs Bandwidth



ap for Semiconductors (ITRS)

Chip Multiprocessors

HPCWire.com this morning:
Up Against the Memory Wall
”Nevermind the cores. Just hand over the cache**”**

HPCWire December 07:
More Than 16 Cores May Well Be Pointless
[by Sandia Labs]

Throughput Computing:

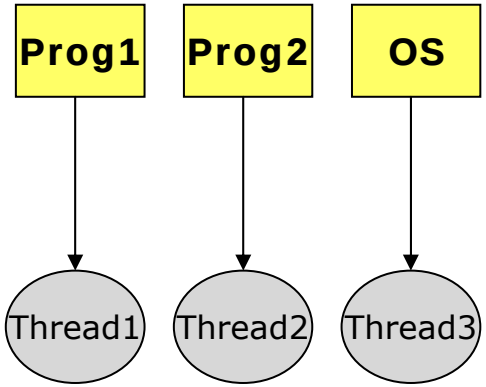
- Does not explore the shared caches well

(cache capacity/thread: ~100-500 kB)

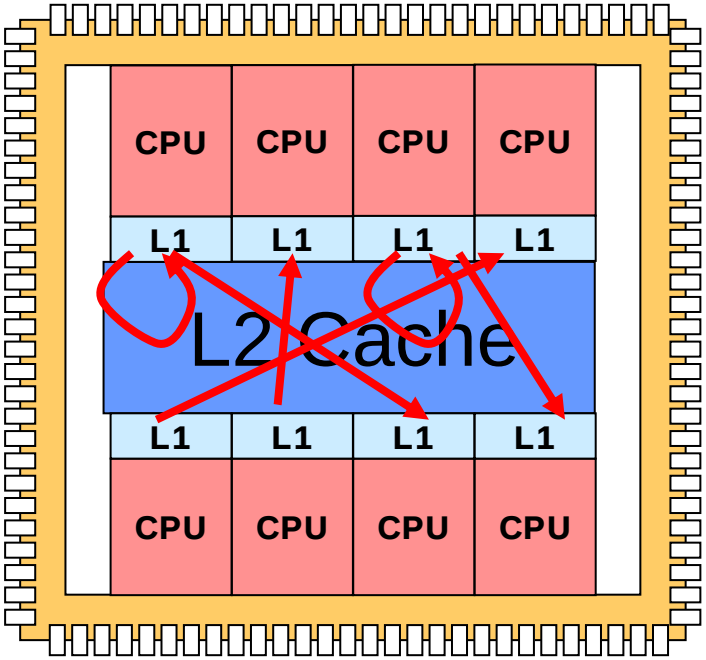
- Requires N times more memory

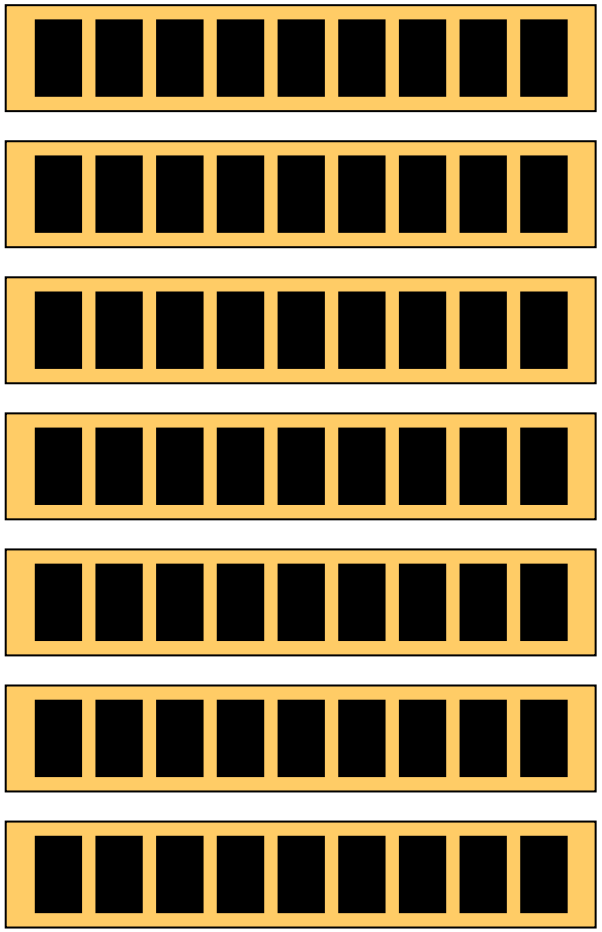
- If limited by memory capacity or memory bandwidth

  ➔ go parallel



Thread Interaction

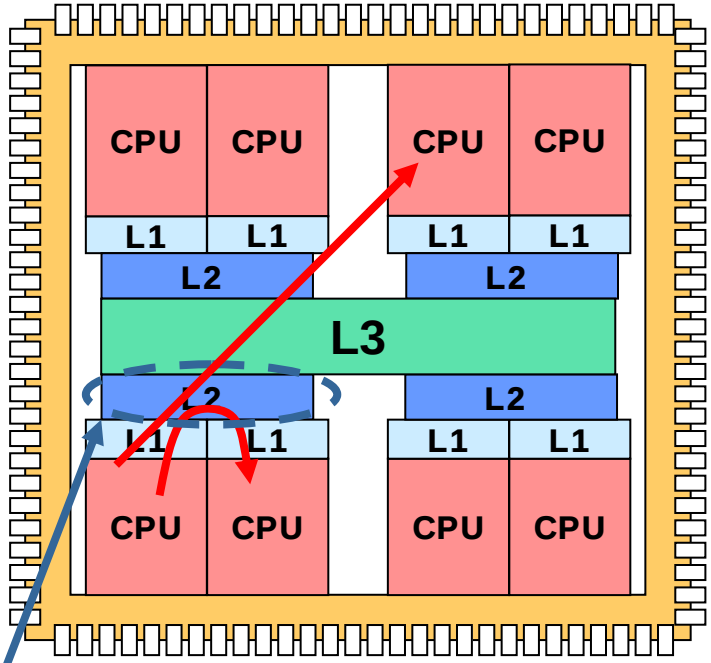


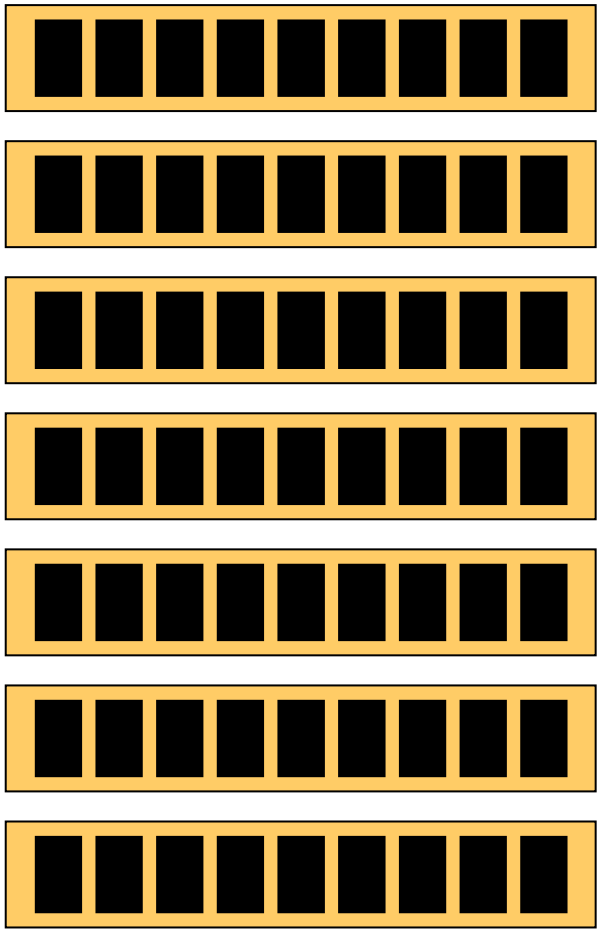


- Coherence traffic
- Communication utilization
- Load imbalance
- Synchronization
- False sharing
- ...

UPPSALA UNIVERSITET

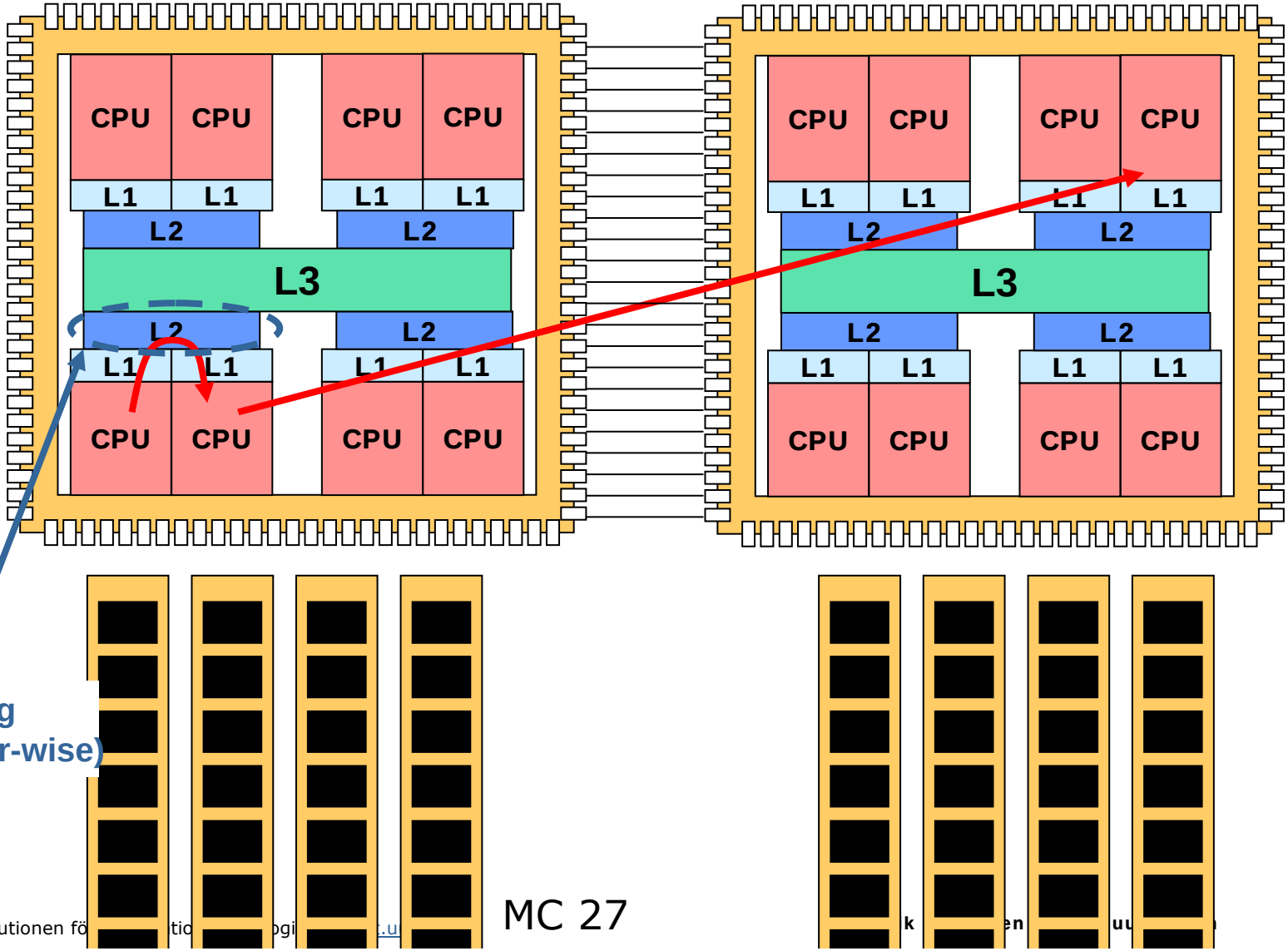
Multicore Challenges: Non-uniformity Communication



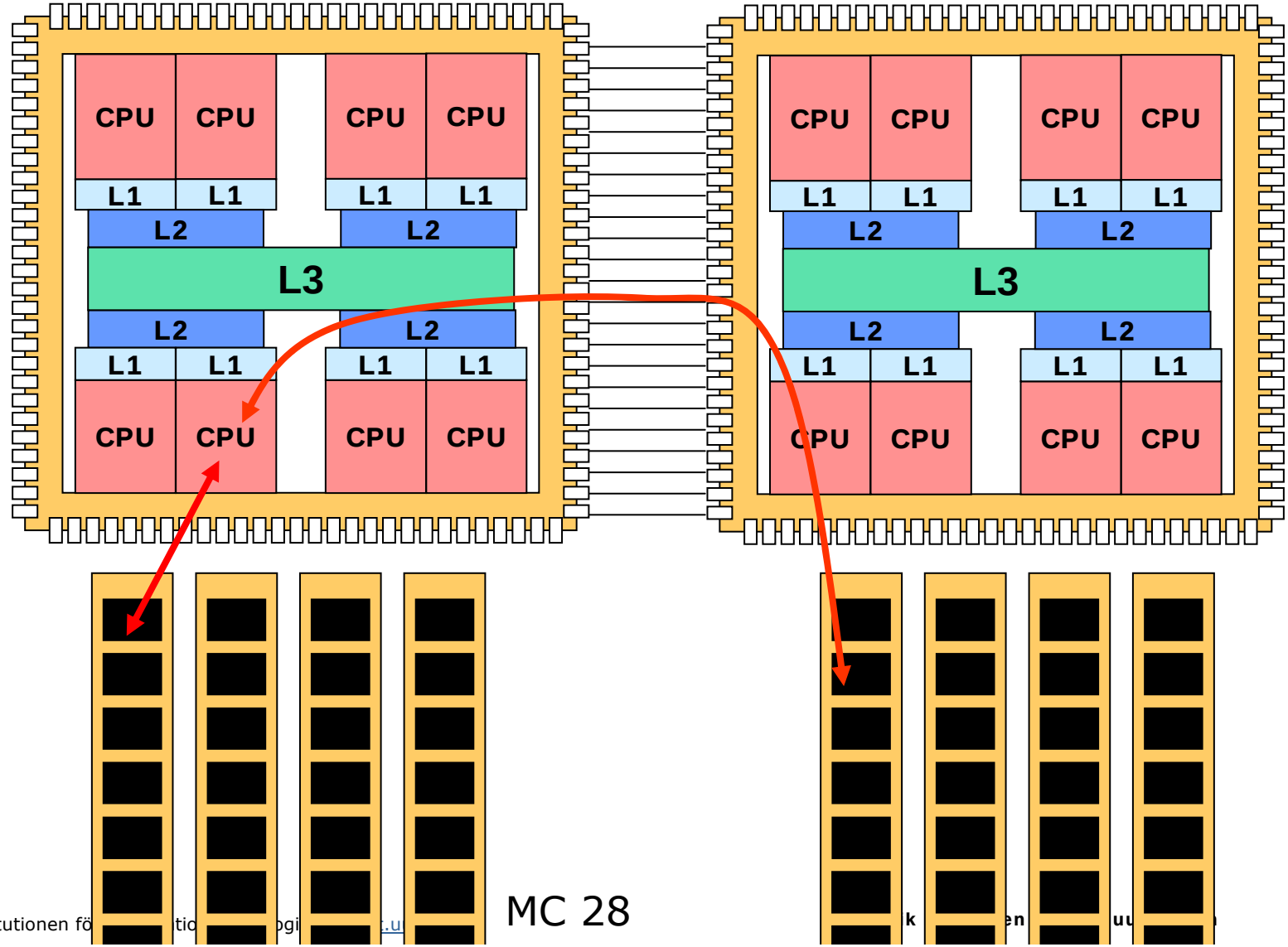


L2 sharing (here: pair-wise)

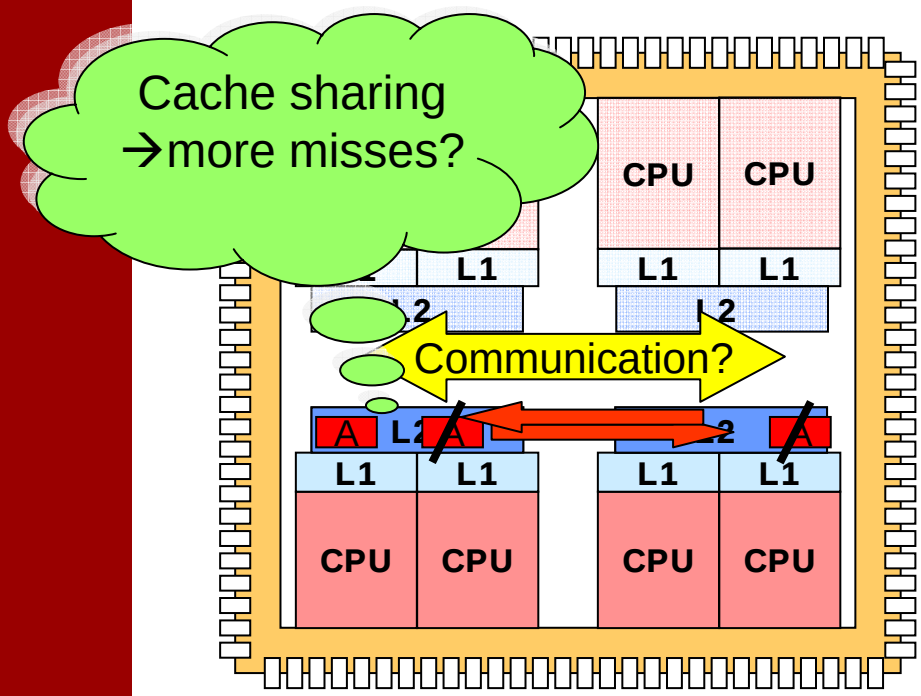
Multicore Challenges (Multisocket) Non-uniformity Communication

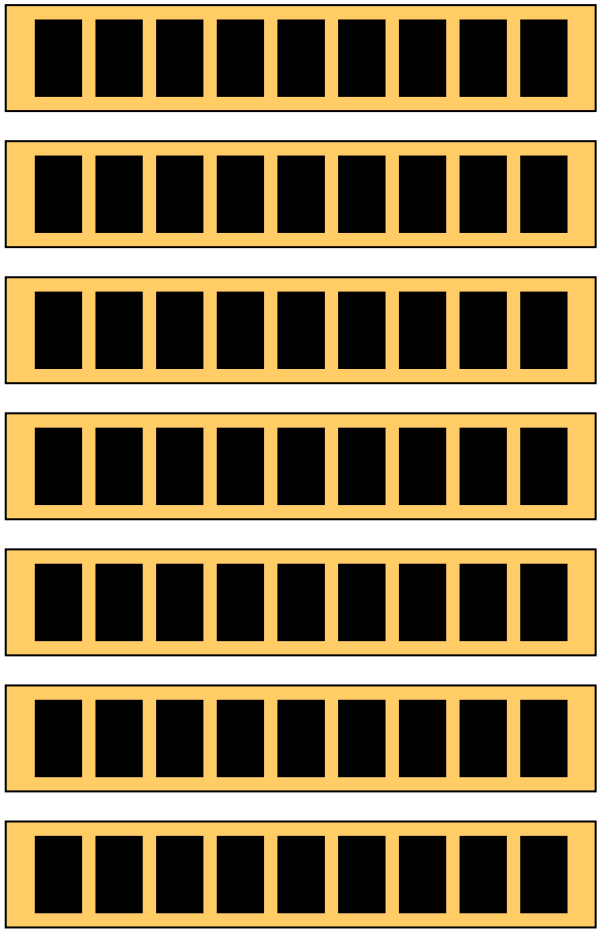


Multicore Challenges (Multisocket) Non-uniformity Memory



Example: Thread Interaction (True Communication)



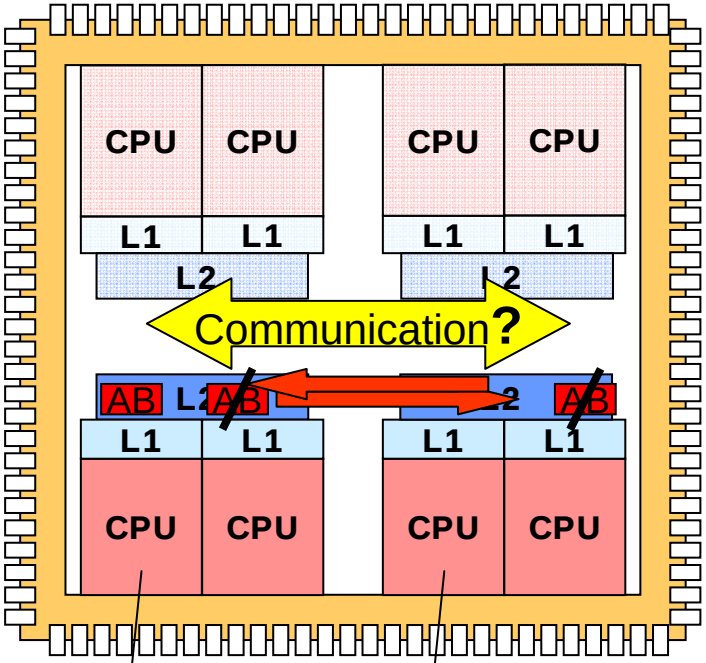


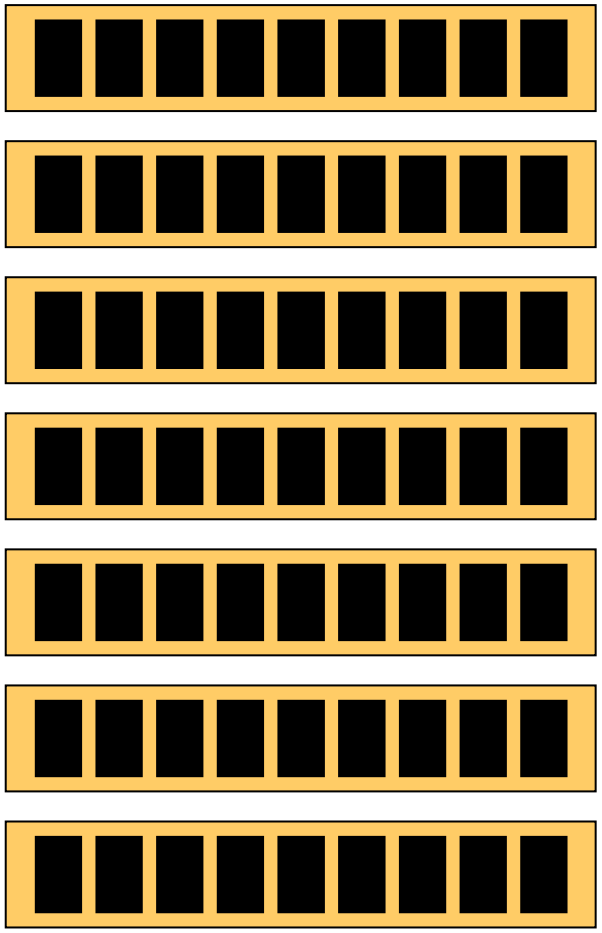
A:= 1

A := 2

A:= 3

Example: Thread Interaction (False sharing)





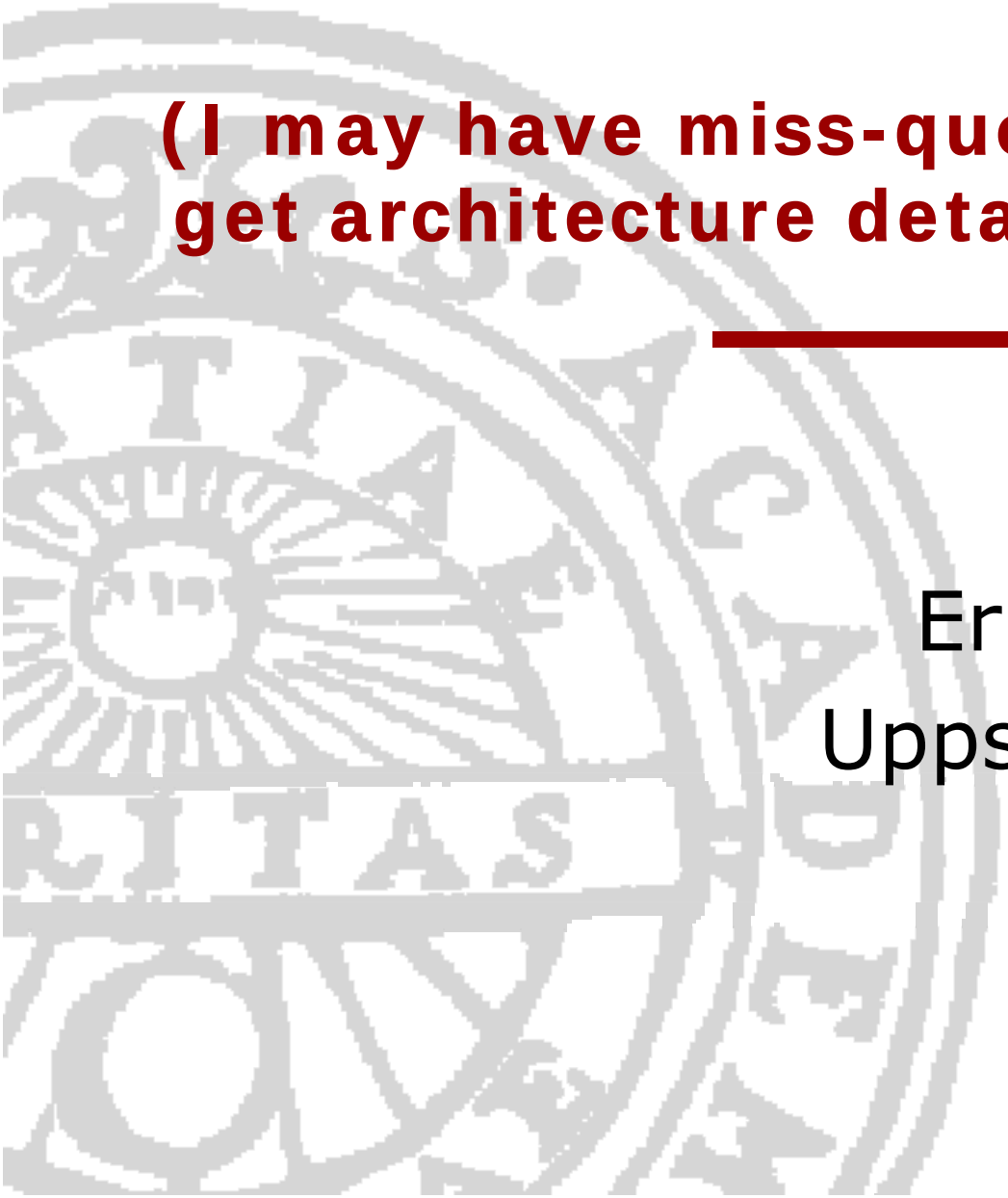
A := 1
…
…
A := 2

…
B := 1
…

Ex: Code change ➔Performance difference on a Quad: 5x

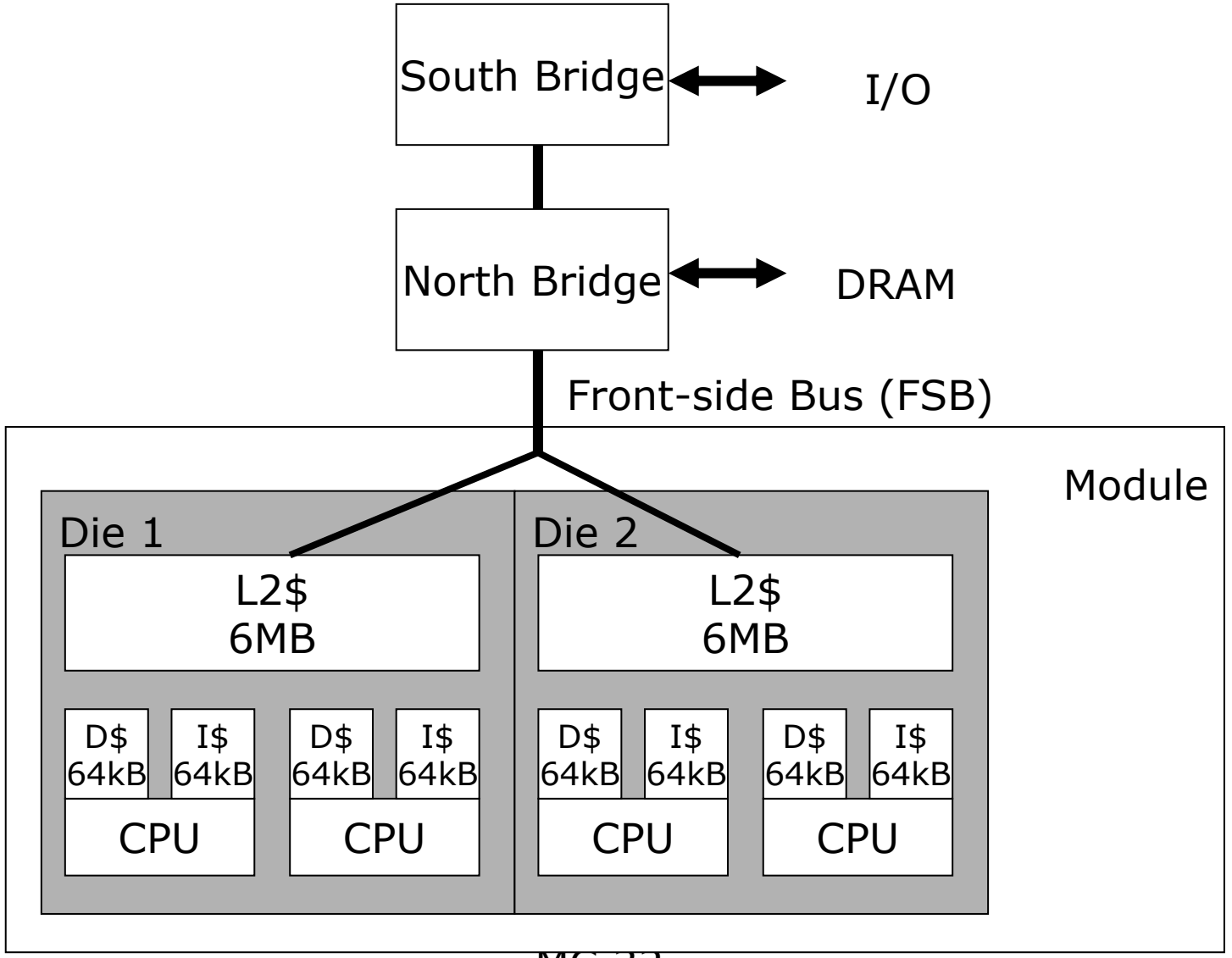
Commercial x86 snapshot

(I may have miss-quoted some details, get architecture details from vendors)

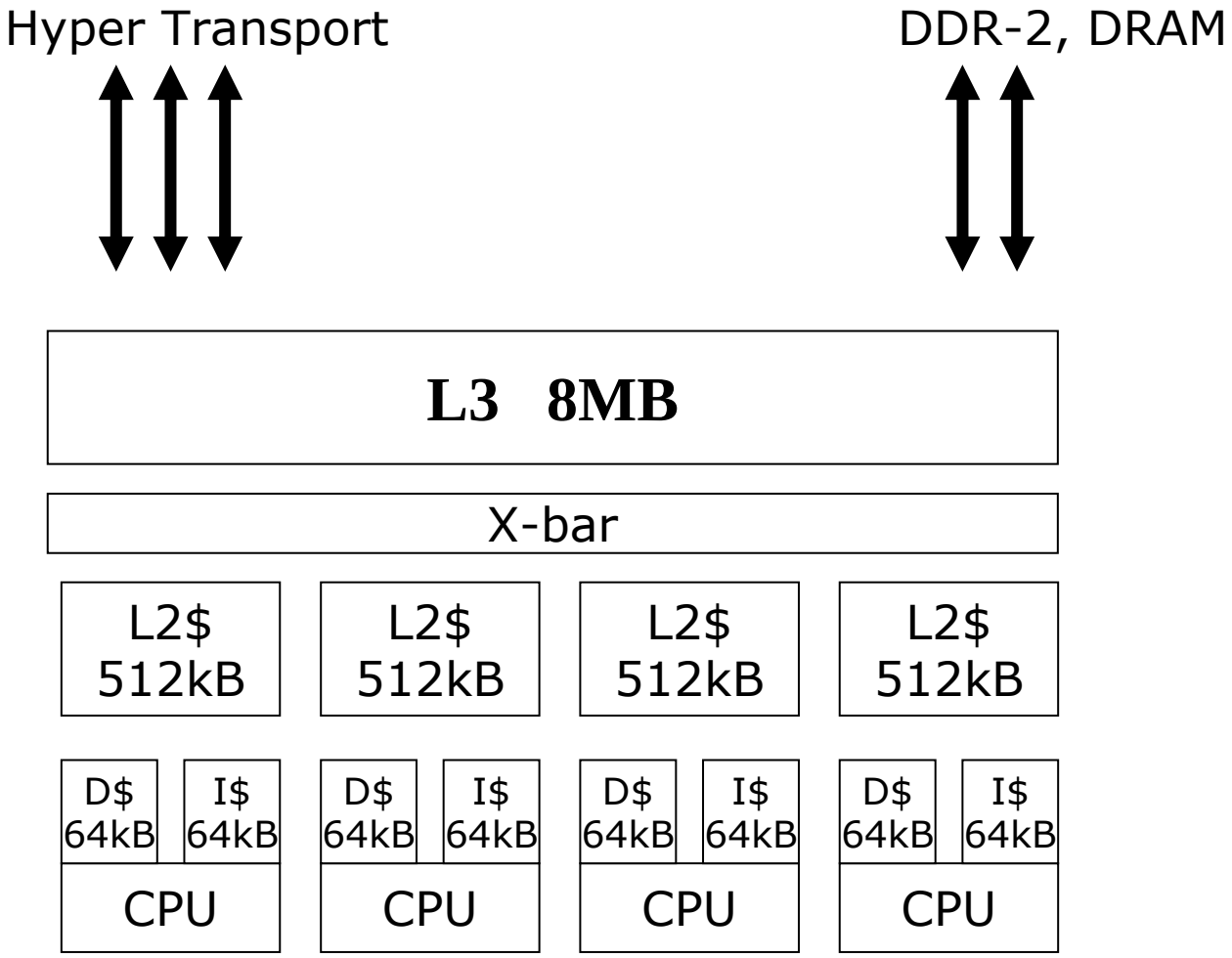


Erik Hagersten
Uppsala Universitet

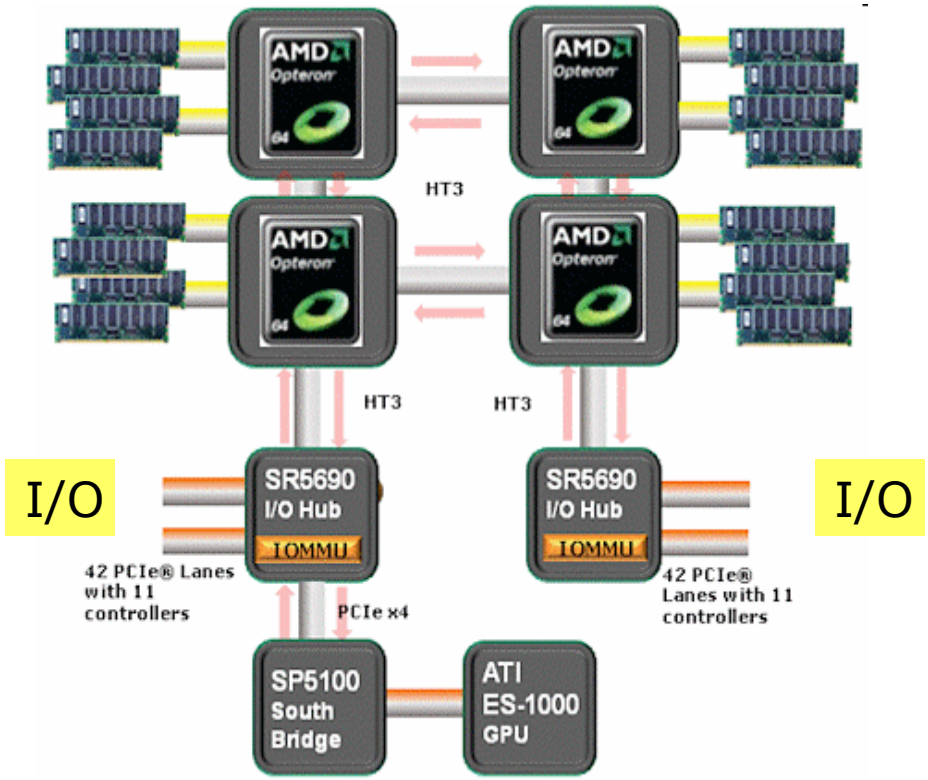
Intel Core2 Quad



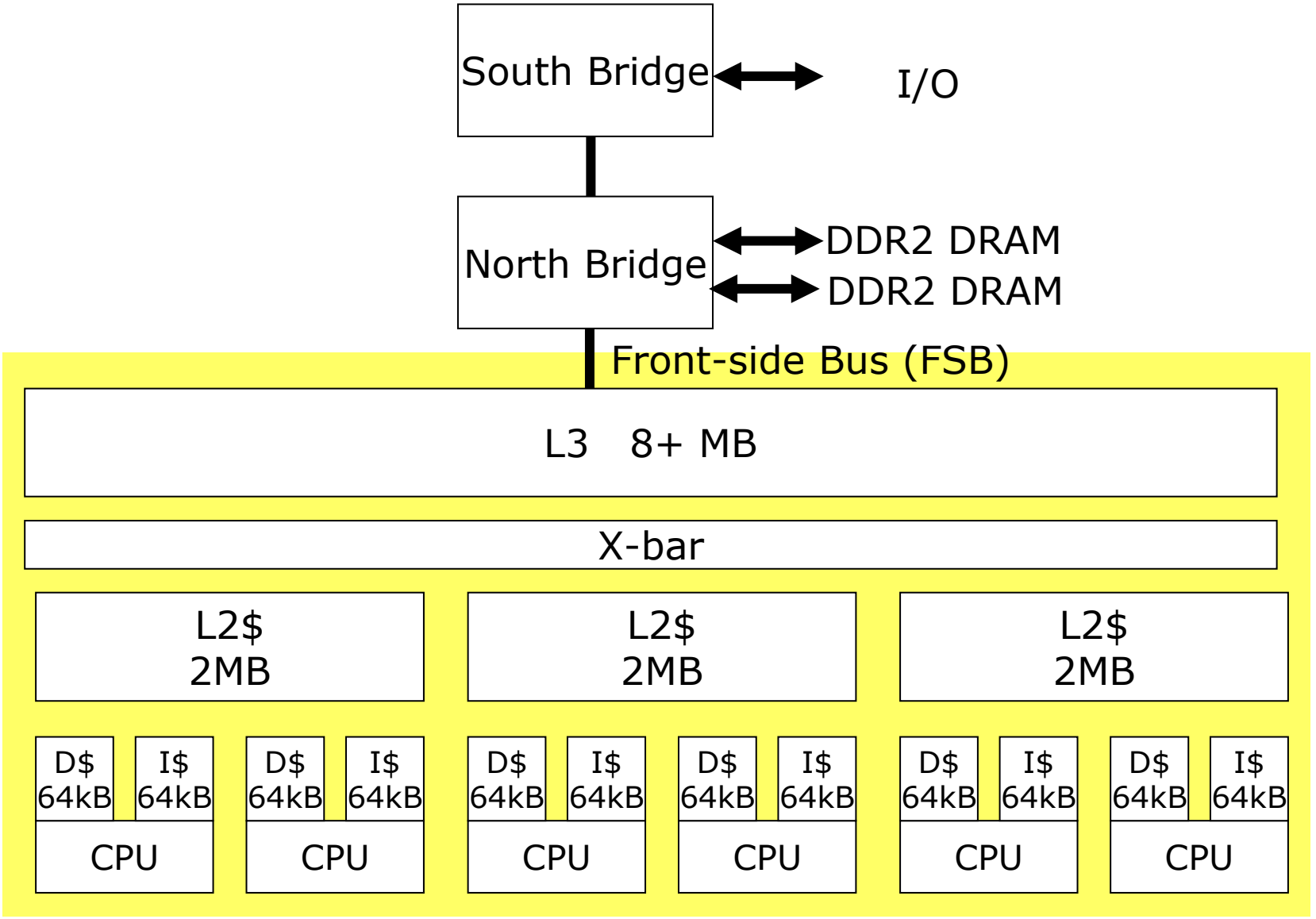
AMD Shanghai



AMD MC System Architecture

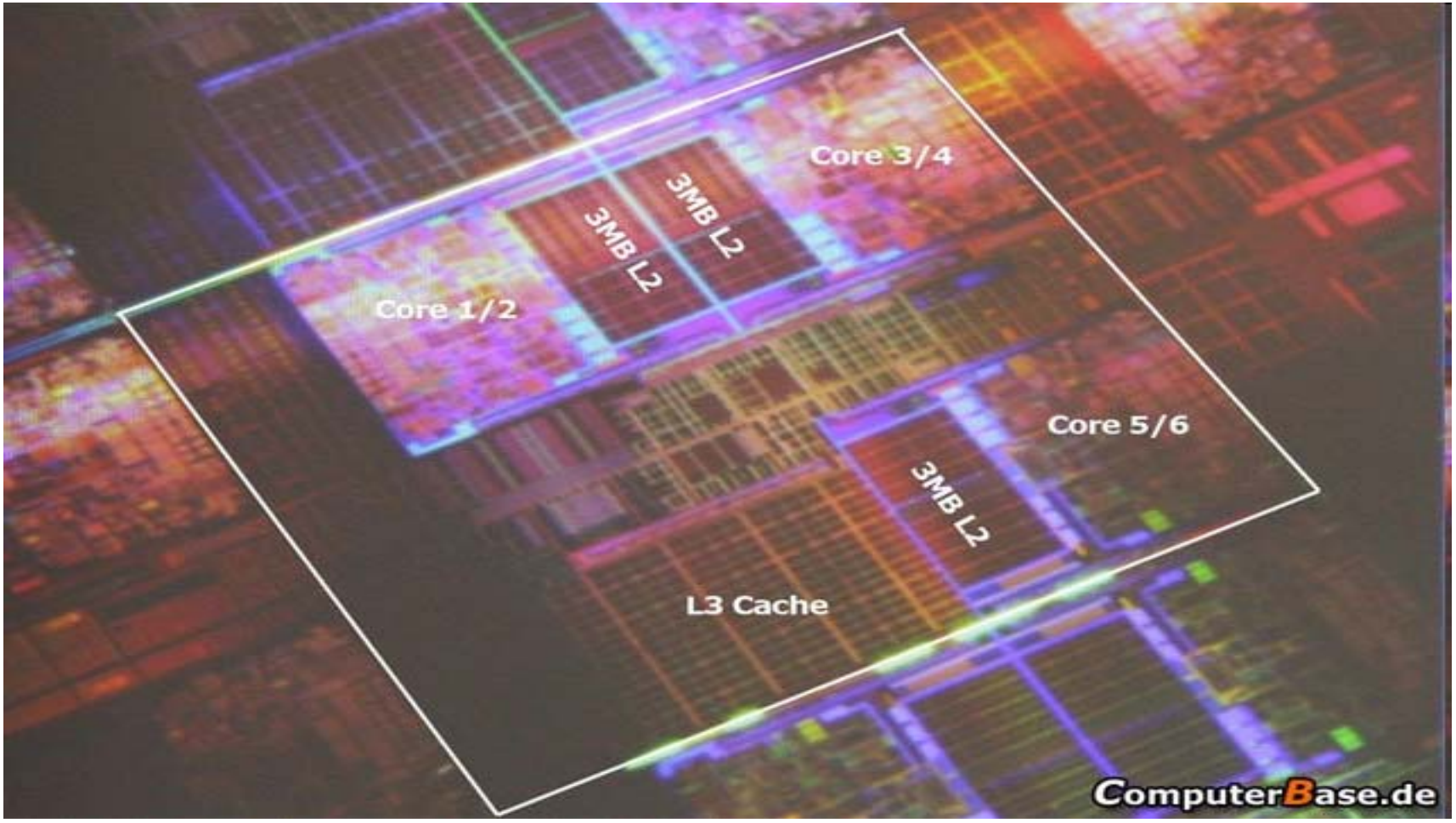


Intel: Dunnington

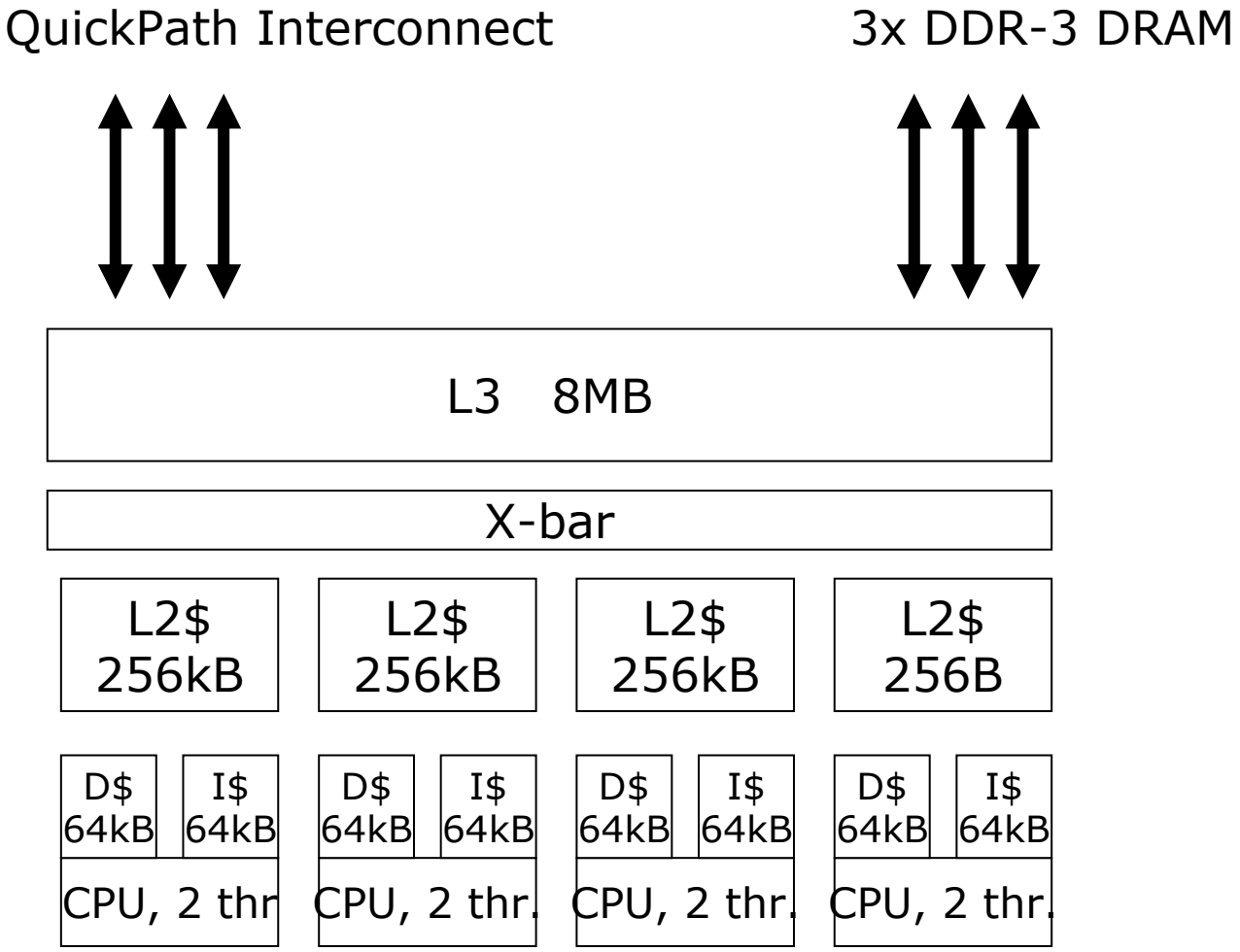


UPPSALA UNIVERSITET

Intel Dunnington

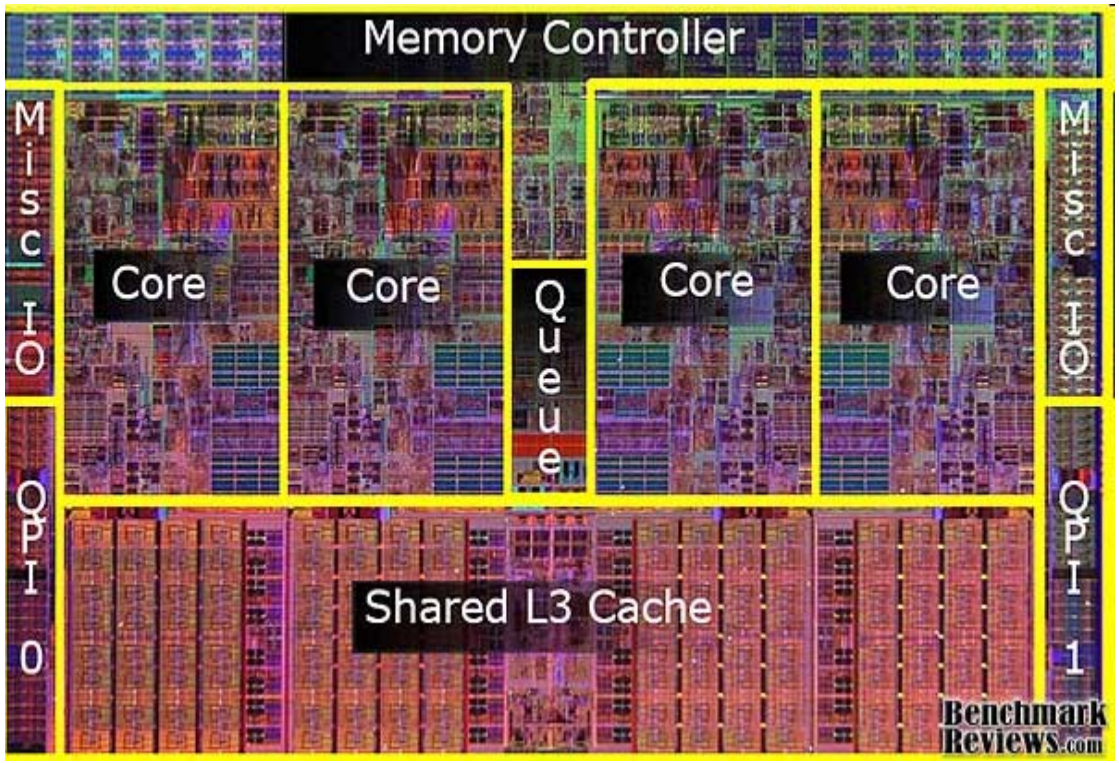


Intel: Nehalem, Core i7 Q1 2009 (4 cores)



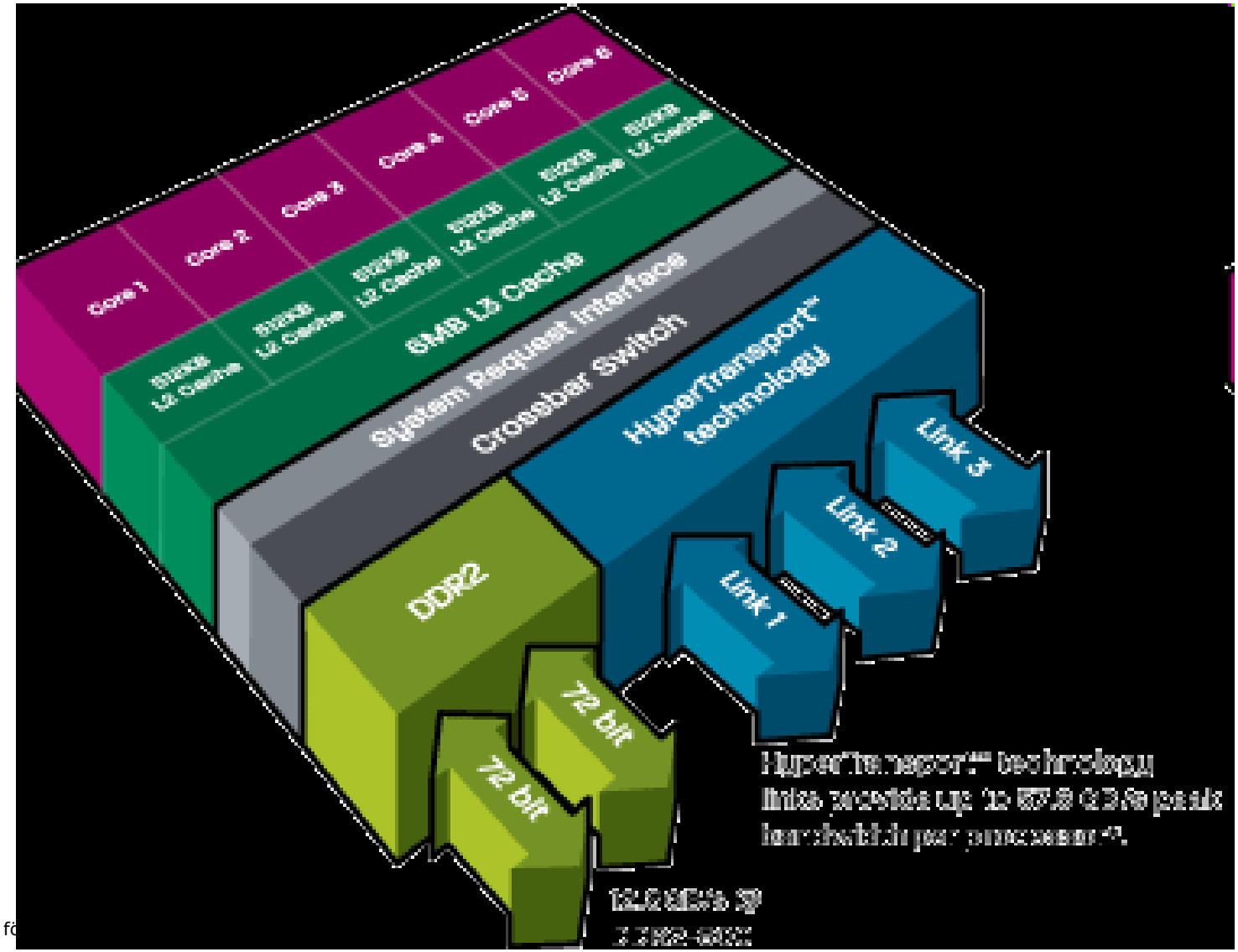
Up to 4 cores x 2 threads

Nehalem ”Core i7”



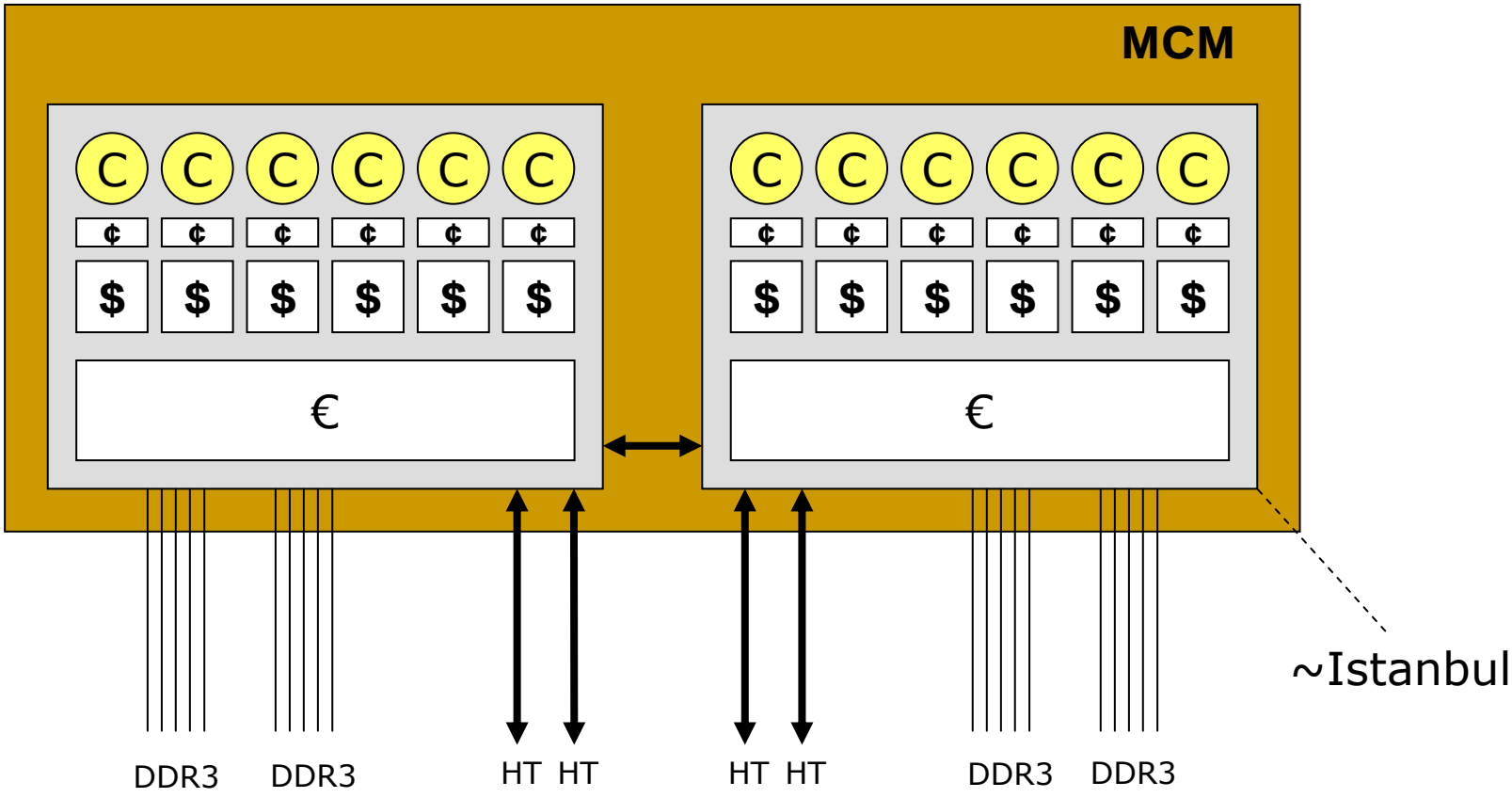
UPPSALA UNIVERSITET

AMD Istanbul, 6 cores



UPPSALA UNIVERSITET

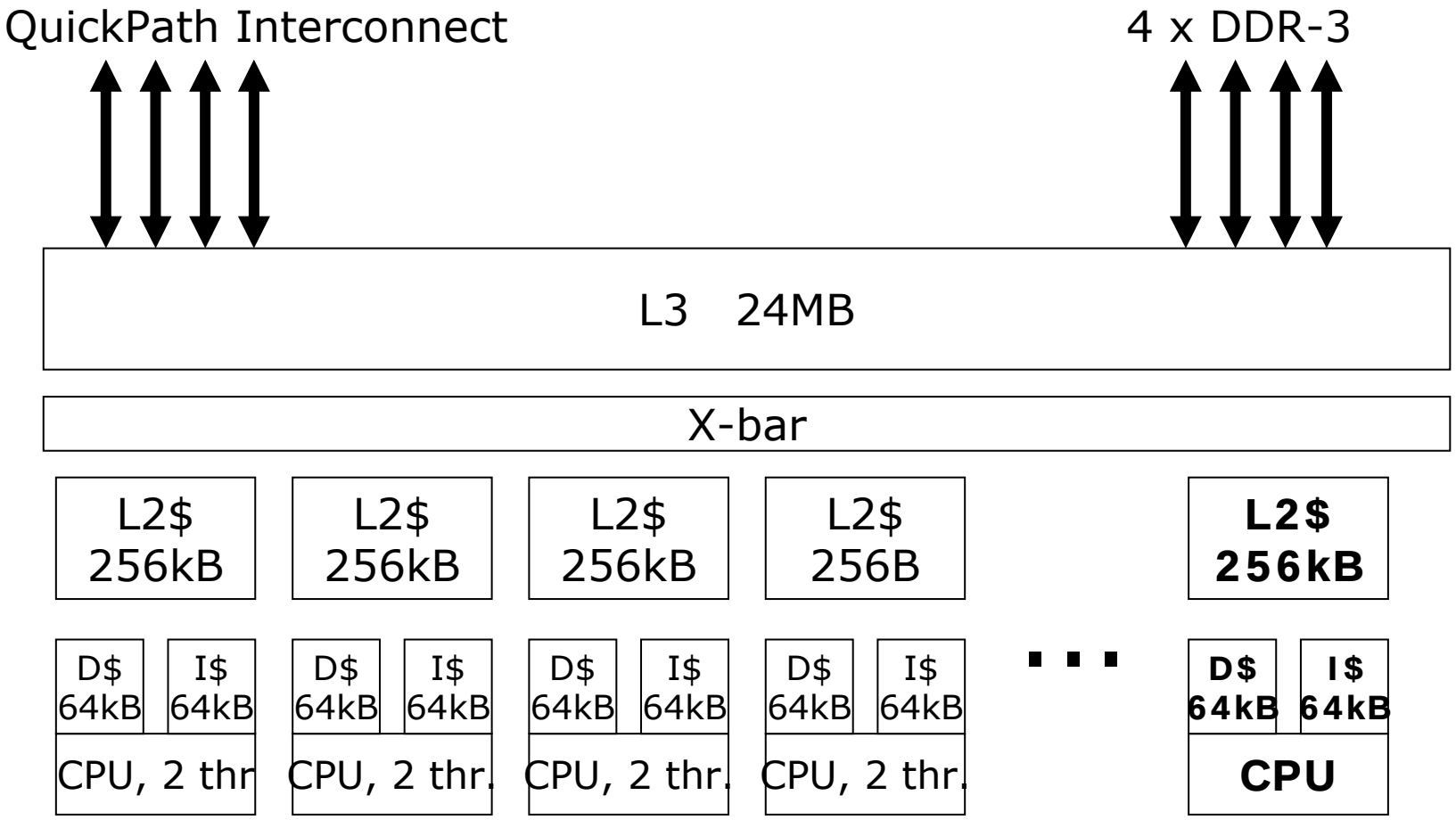
AMD Magny-Cours



MC 40

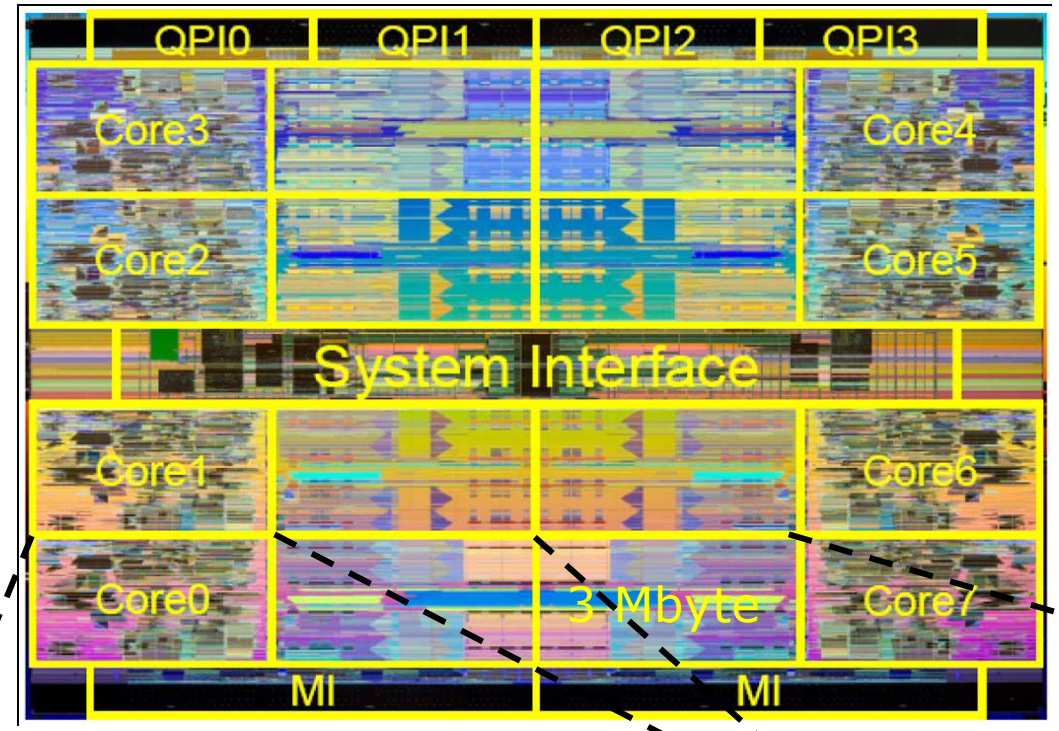
UPPSALA UNIVERSITET

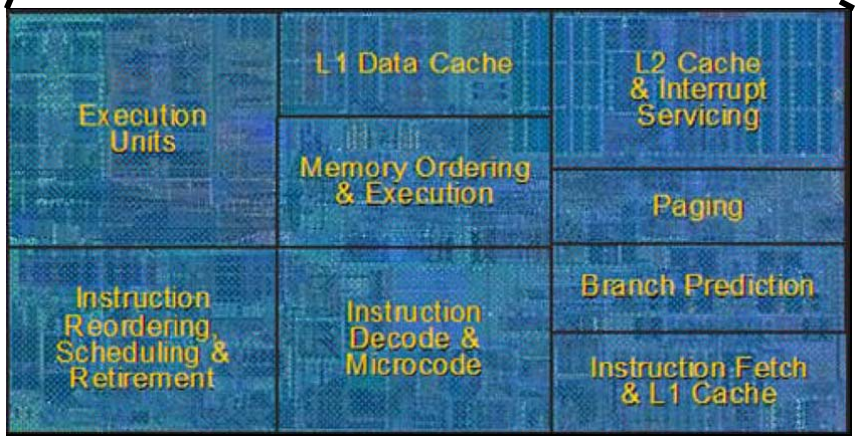
Intel: ”Nehalem-Ex” (i7)

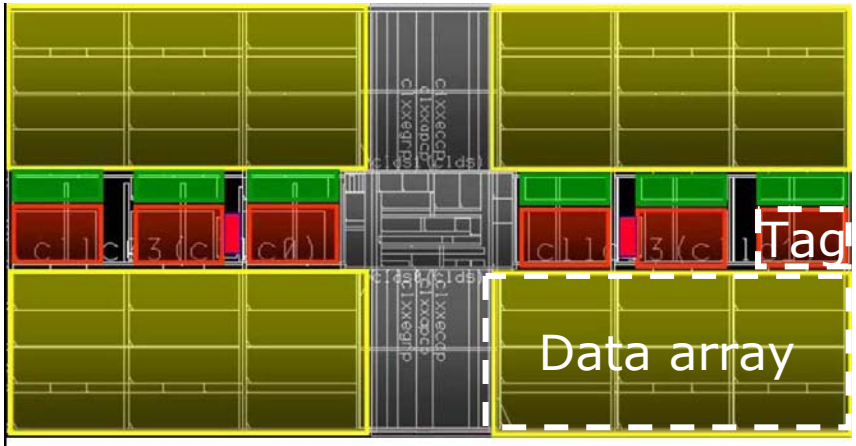


8 cores x 2 threads

How is the silicon used (i7-Ex)?



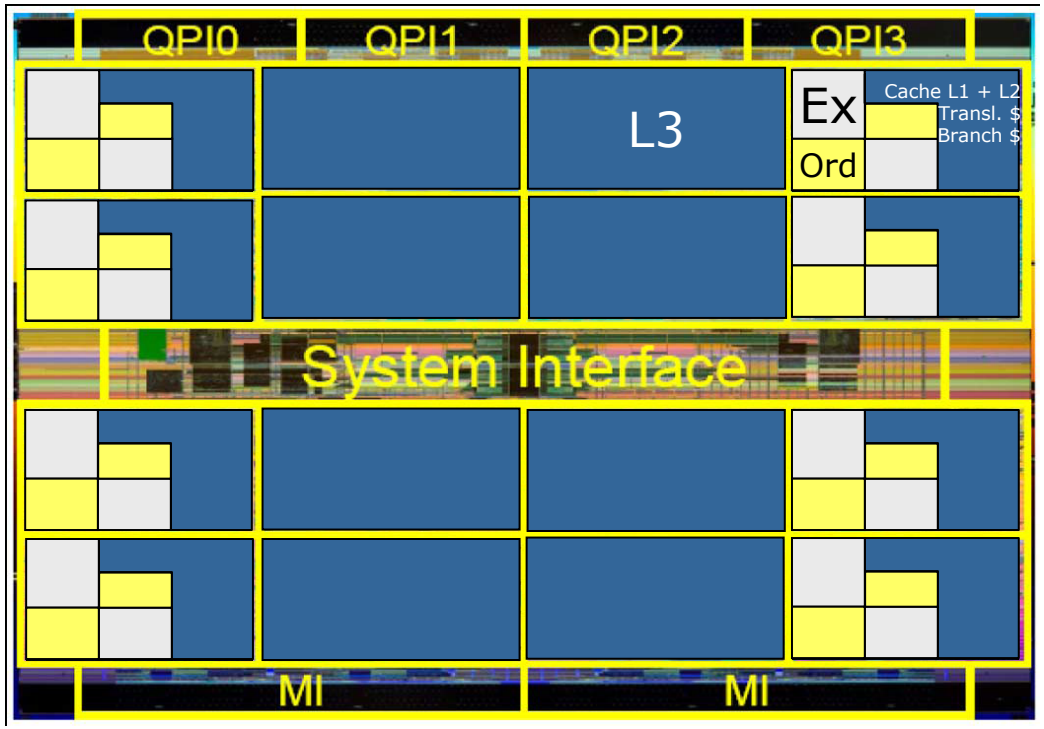


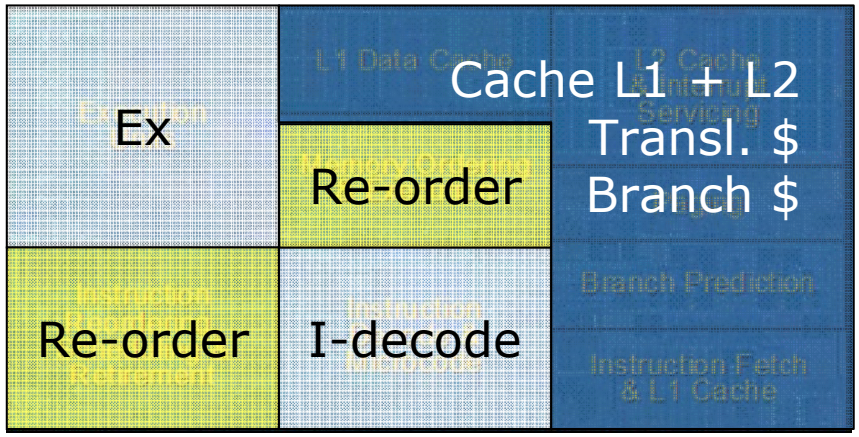


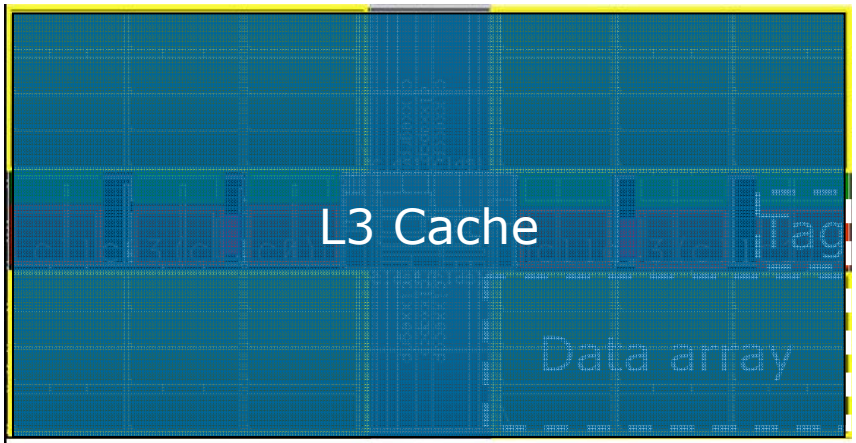
Source: JSSC Jan 2010, Rusu et. al

UPPSALA UNIVERSITET

How is the silicon used?

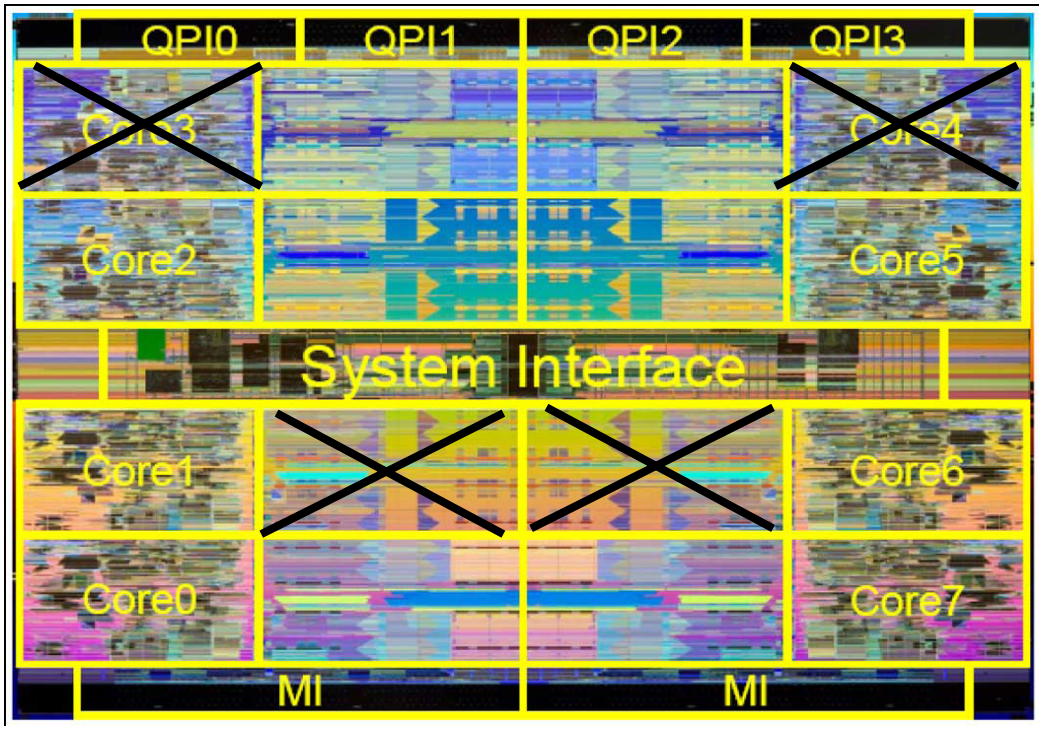






Source: JSSC Jan 2010, Rusu et. al

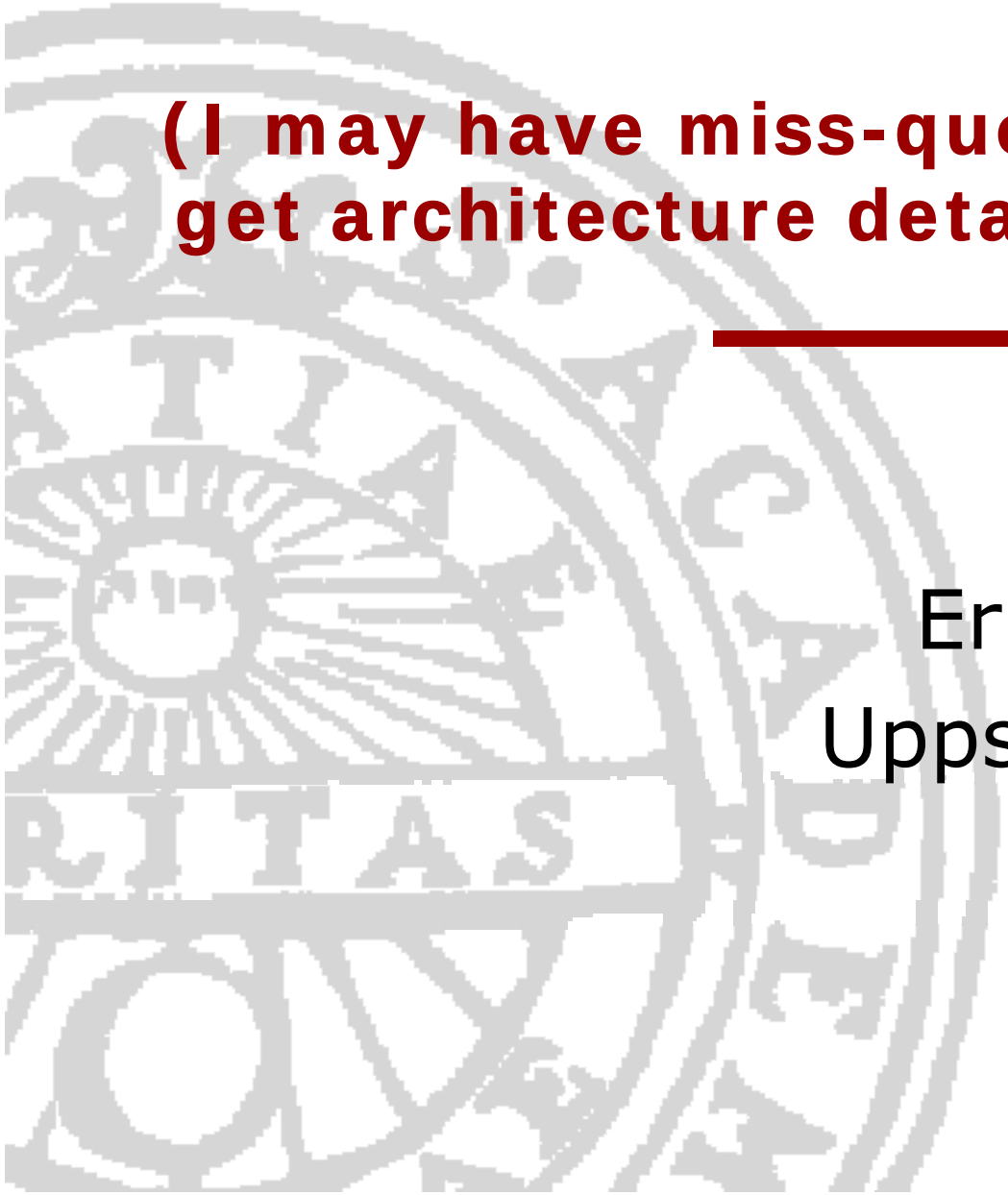
Handling silicon defects?



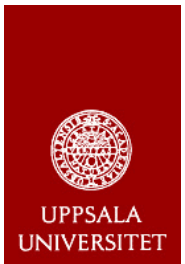
Source: JSSC Jan 2010, Rusu et. al

Some other multicores

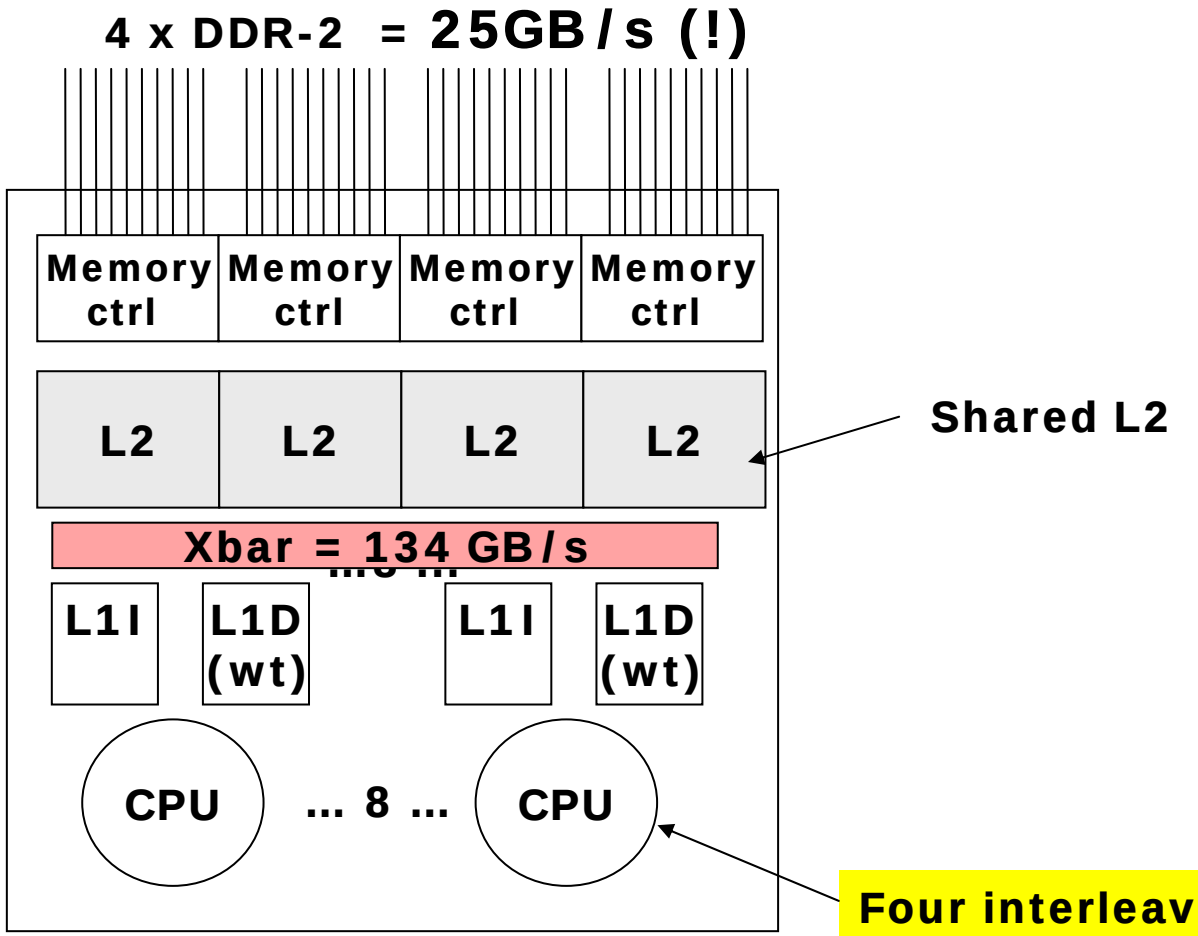
(I may have miss-quoted some details, get architecture details from vendors)



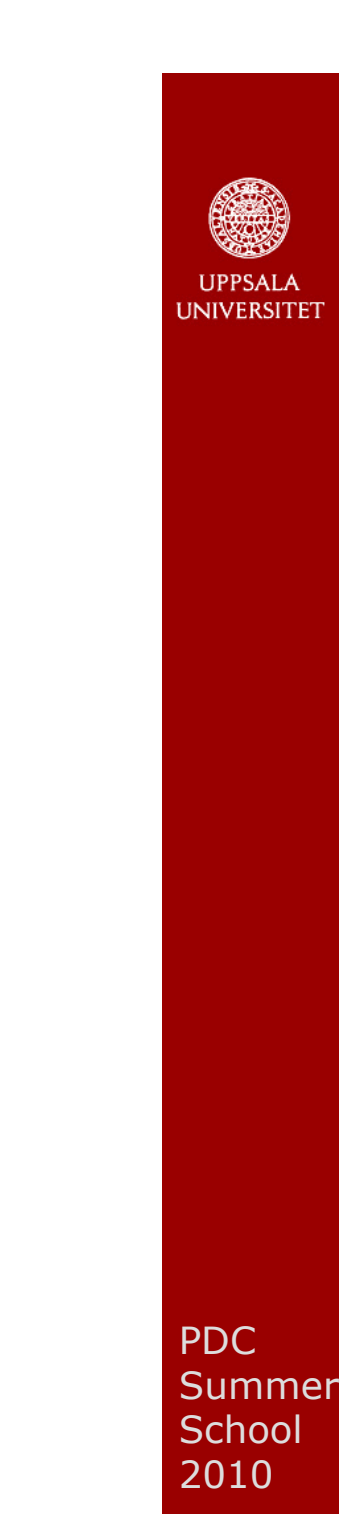
Erik Hagersten
Uppsala Universitet



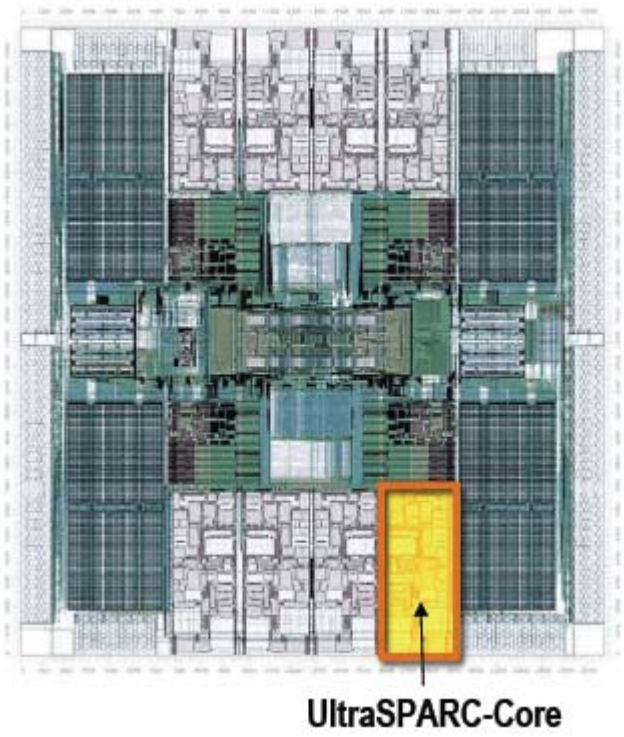
Sun Niagara, 2005!!



Now: Victoria's falls: 16 core with 16 threads each



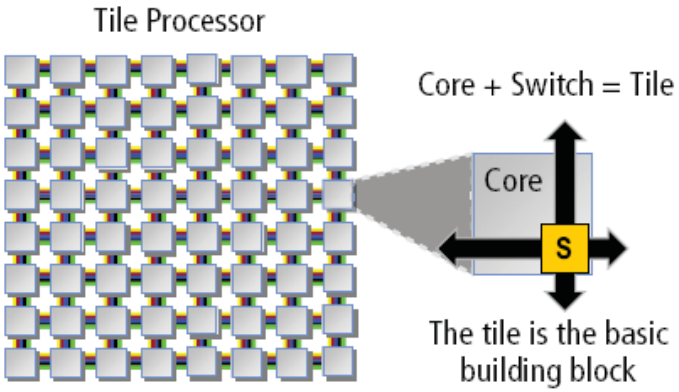
Niagara Chip



Sun Microsystems

UPPSALA UNIVERSITET

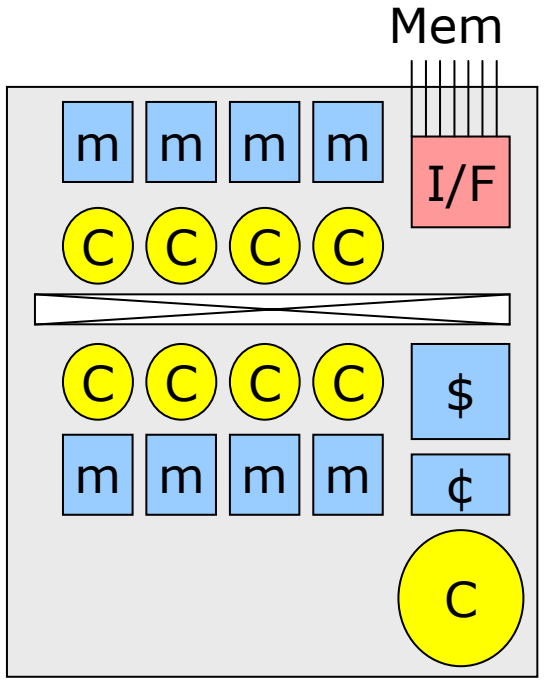
TILERA Architecture



64 cores connected in a mesh
Local L1 + L2 caches
Shared distributed L3 cache
Linux + ANSI C
New Libraries
New IDE
Stream computing
…

UPPSALA UNIVERSITET

IBM Cell Processor



IBM Cell

UPPSALA UNIVERSITET

So-called accelerators

- Sits on the IO bus (!!)
- GP Graphics processors, aka GPGPU
  [e.g. NVIDIA, AMD/ATI]
- Specialized accelerators
  [e.g., FPGA/Mitrionics, ASIC/ClearSpeed]
- Specialized languages for the above
  [CUDA, Ct, Rapid Mind, Open-CL, …]

UPPSALA UNIVERSITET

So-called accelerators

- Sits on the IO bus (!!)
- GP Graphics processors, aka GPGPU?
  [e.
- Spe
  [e.
- Spe
  [CUDA, Ct, Rapid Mind, Open-CL, …]

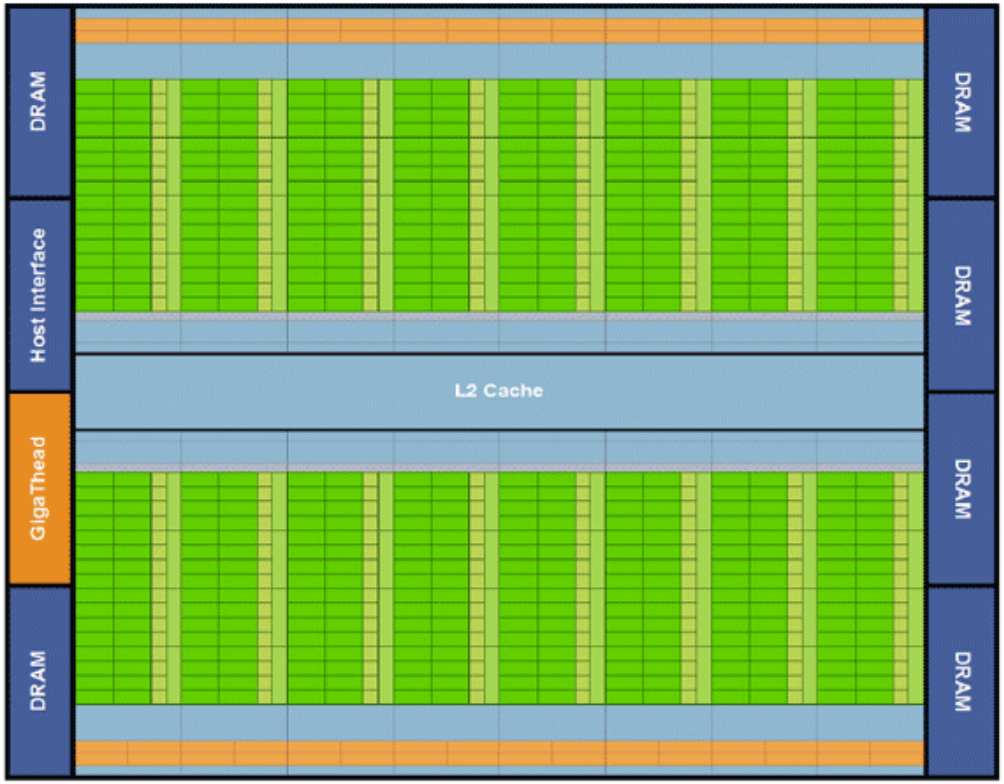
My view: Not very general purpose yet!

- **Fits well for a few VERY IMPORTANT app domains!**
- **Limited applicability?**
- **Programmer productivity?**
- **Application life time?**
- **New generation devices will be more useful…**

UPPSALA UNIVERSITET

Fermi from nVIDIA

a huge step in the right direction

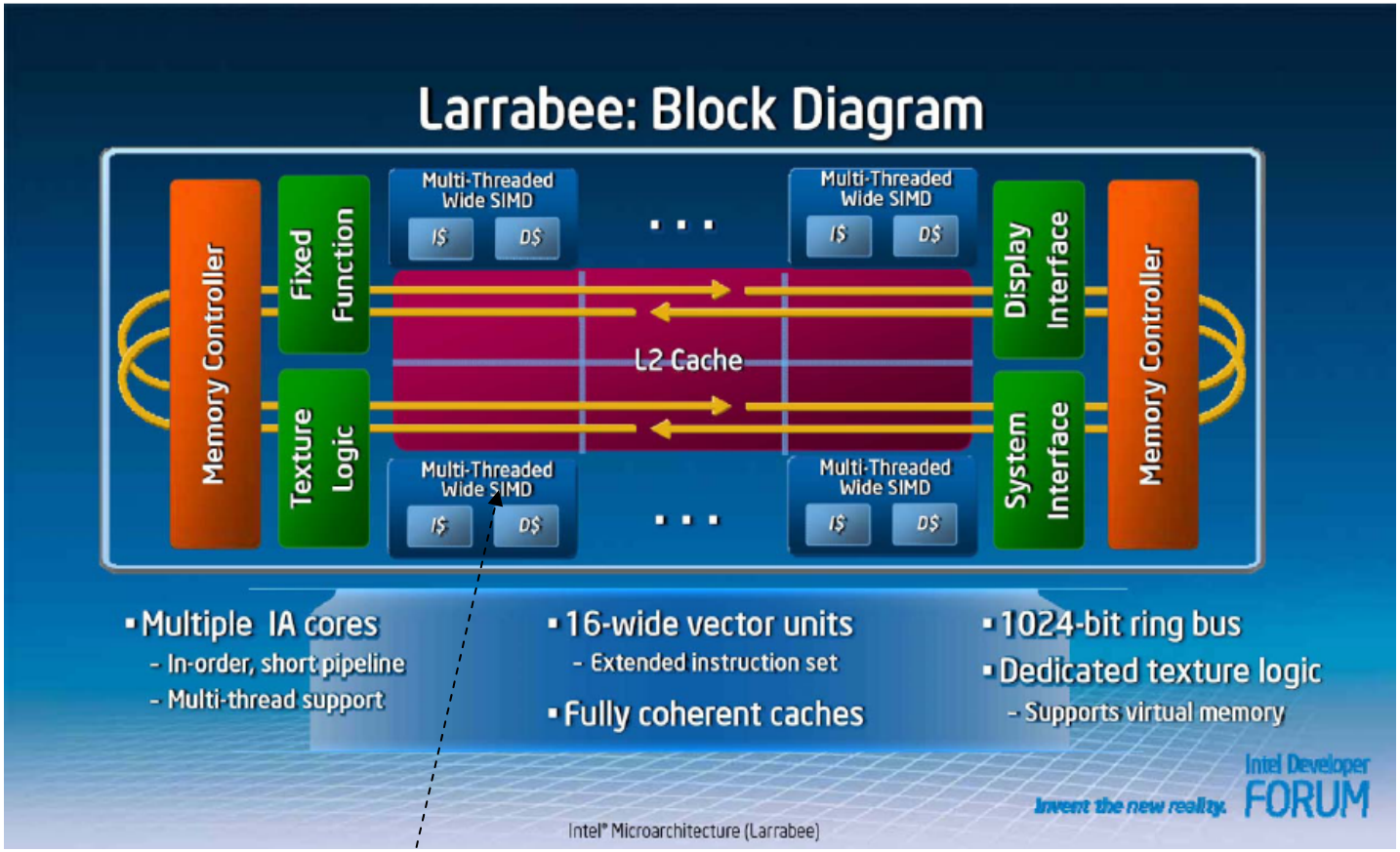


- 512 ”processors” (***P***)
- 16 ***P***/StreamProcessor (***SP***)
- Full DP-FP IEEE support
- 64kB L1 cache /***SP***
- 768kB global shared cache
- Atomic instructions
- ECC correction
- Debugging support
- …

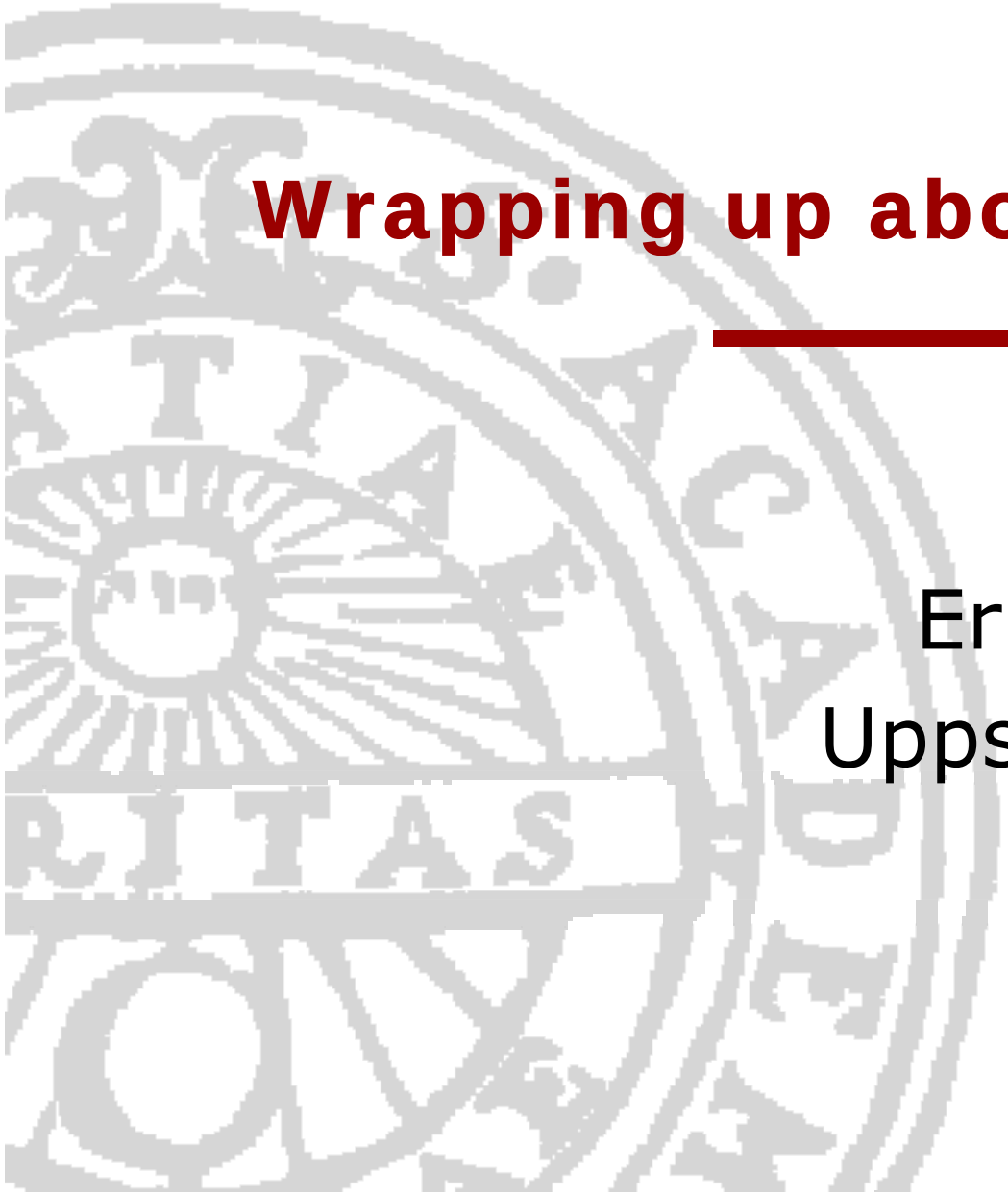
David Black-Schaffer to give you the full story

Scaling the x86 Manycore GP computing Larrabee (now called MIC) from Intel 2011??

”more than 50 cores”



SIMD instructions (16-way??)

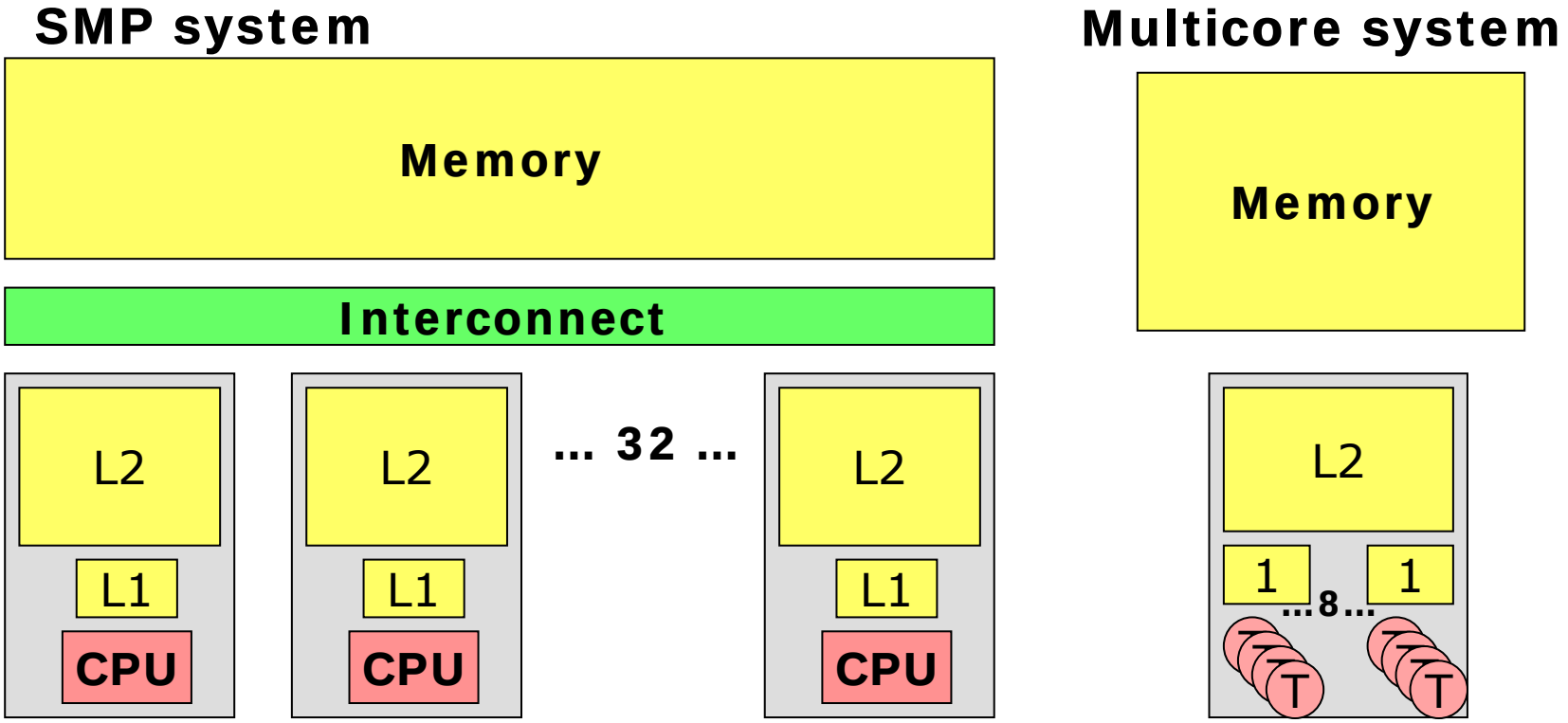


Wrapping up about multicores

Erik Hagersten
Uppsala Universitet

UPPSALA UNIVERSITET

Looks and Smells Like an SMP?



Well, how about:

- Cost of parallelism?
- Cache capacity per thread?
- Memory bandwidth per thread?
- Cost of thread communication? …

UPPSALA UNIVERSITET

What matters for multicore performance?

- Are we buying…
  - CPU frequency?
  - Number of cores?
  - MIPS and FLOPS?
  - Memory bandwidth?
  - Cache capacity?
  - Memory capacity?
  - Performance/Watt?

UPPSALA UNIVERSITET

MC Questions for the Future

- How to get parallelism?
- How to get performance/data locality?
- How to debug?
- A case for new funky languages?
- A case for automatic parallelization?
- Are we buying:
  - compute power,
  - memory capacity, or
  - memory bandwidth?
- Will 128 cores be mainstream in 5 years?
- Will the CPU market diverge into desktop/capacity/capability/special-purpose CPUs again?
- **A non-question: will it happen?**