Starting with a tiny joke !

- How do you call people speak 3 languages ?
  - Trilingual people !
- How do you call people speak 2 languages ?
  - Bilingual people !
- How do you call people speak 1 language ?
  - French people !

I'm french :

I will try not to twist your eardrums...

So : slides in globish & voice in french

What's CBP ?


Trainings
Conferences
Projects
Hôtel
CBP
CENTRE BLAISE PASCAL

CBP : a small example

Nasa X-29

- Cell of F-5
- Engine of F-18
- Gear of F-16
- Studies
  - Fwd swept wing
  - Incidence >50°
  - *« Fly-By-Wire »*

Recycle, Re-use and explore new domains

CBP : Experimental platform
7 technical facilities

- **Multi-nodes** : 3 clusters from 4 to 48 nodes, Nodes/Cores : 48/384, 24/192, 4/48
- **Multi-cores** : 10 from 2 to 20 cores,
  - Nodes/Servers : from 8 to 20 cores,
  - Workstations : from 2 to 12 cores
- **GPU : 20 different models of GPU (AMD & Nvidia)**
  - GPGPU : 6 ; GPU Nvidia : 10 ; GPU AMD/ATI : 4 types
- **Integration : 14 virtual machines :** Debian Lenny, Squeeze, Wheezy, Jessie, Sid in 32 & 64 bits, ...
- **Exotic hardware : 4 machines** Sparc, PowerPC, ARMv7 under Debian Wheezy
- **3D facilities : 2 workstations, 2 video projectors, 20 monitors, 4 glasses**
  - 2 complete Workstations, Monitor & Video Projector capabilities
- **COMOD : *« Compute On My Own Device »***
  - Same ***Single Instance Distributing Universal System*** (SIDUS)

If computing was cooking...

Code ~ Recipie

Computer ~ Kitchen

Input Data ~ Ingredients

Output Data ~ Meal Dish

Process ~ Cooking process

Control Unit ~ Cooker

ALU* ~ Utensil

Me ~ Client

Batch Request ~ Order

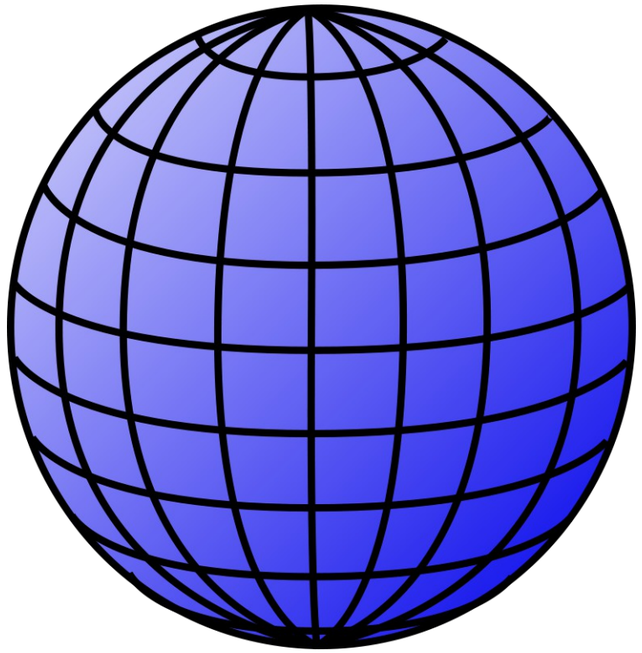
ALU : Arithmetic & Logical Unit

Parallelism in 7 questions
5 Ws & 2 Hs

- Analytical Method : answer 7 questions
  - Why ? What ? When ? Where ? Who ?
  - How much ? How ?
- In french, CQQCOQP !
- Interesting approach not to forget :
  - Problem : intrication between questions
  - Advantage : really separate ambiguities
- Try to answer without to many overlappings !

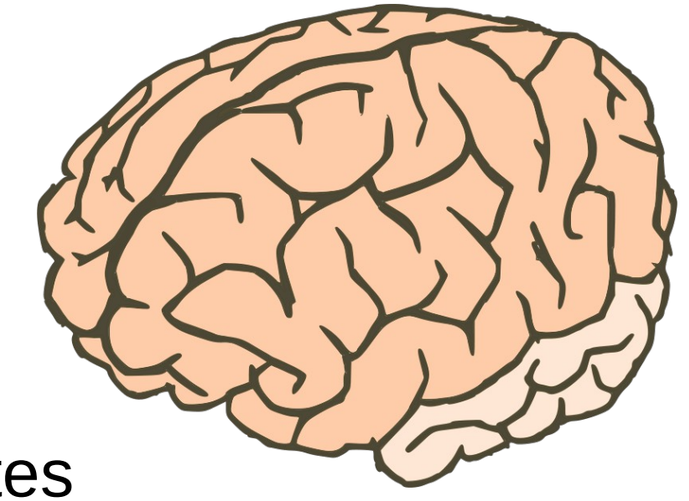
What is Parallelism ?
Let's « *Return to the Source* »

- Etymology (etymonline.com) : **beside one another**
  - From ***para***- « beside »
  - From ***allelois*** « each other », from *allos* « other »
- Parallelism : tasks to achieve, limited ressources...
  - Execute independant tasks in parallel :
    - Embarrassing Parallelism
  - Execute one task in parallel on all resources
    - Sparse communications : Coarse grain
    - Heavy communications : Fine grain
- Paradox of parallelism, meridian !



Where is Parallelism ?

- Where is the best computer ?
  - Between your ears !
  - 20 to 200 billions neurons
  - 125 to 220 trillions synapses
  - Computational capacity (IBM) : 36 Pflops, 3.2 Pbytes



- The best GPGPU processing card :
  - 2880 ALUs
  - 12 GB and 288 GB/s of Bandwidth
  - Process capability : 4 Tflops

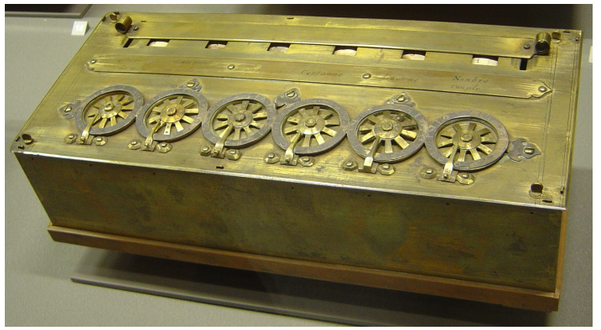
- In fact, getting 1 Tflops, it's amazing !

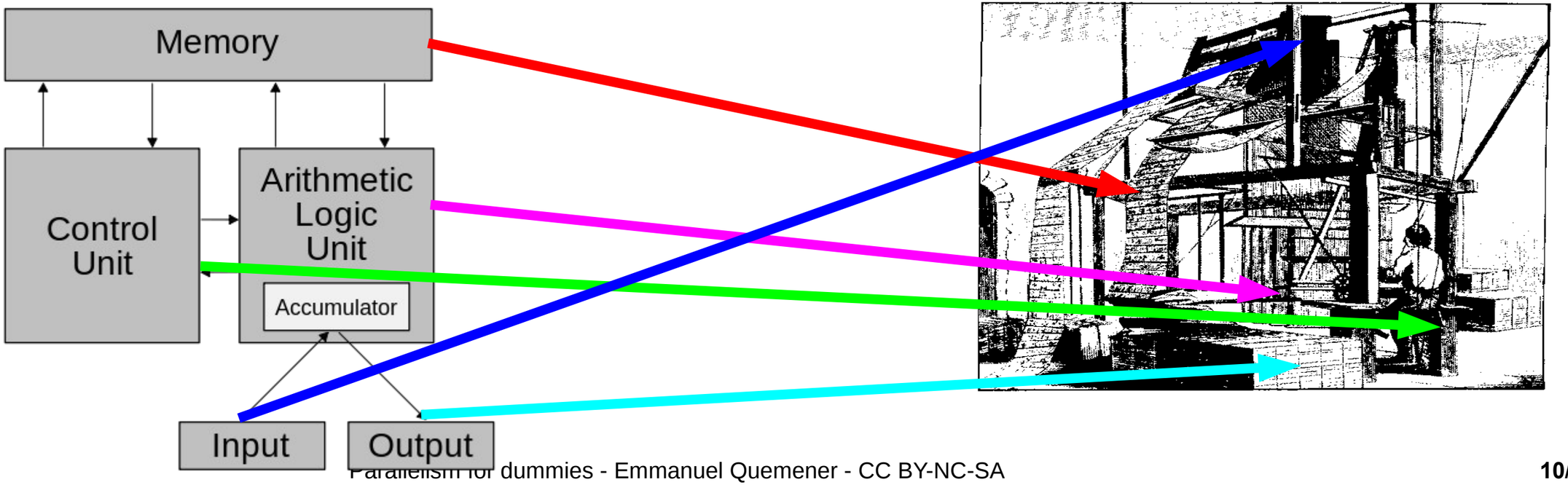
Who uses Parallelism : Are-you like Mr Jourdain

- Mister Jourdain :
  - « Le Bourgeois Gentilhomme » by Molière
  - *Speaking prose all his life without knowing it !*
- You :
  - *Have you got any smartphone, tablet, or laptop ?*
  - *Do you know how much cores in your component ?*

When appeared Parallelism ?
With « computing » machines !

- Which is the first ?
  - **Analogical** One ?
  - **Numerical** One ?
  - **Programming** One ?





Why Parallelism (is inevitable) ?

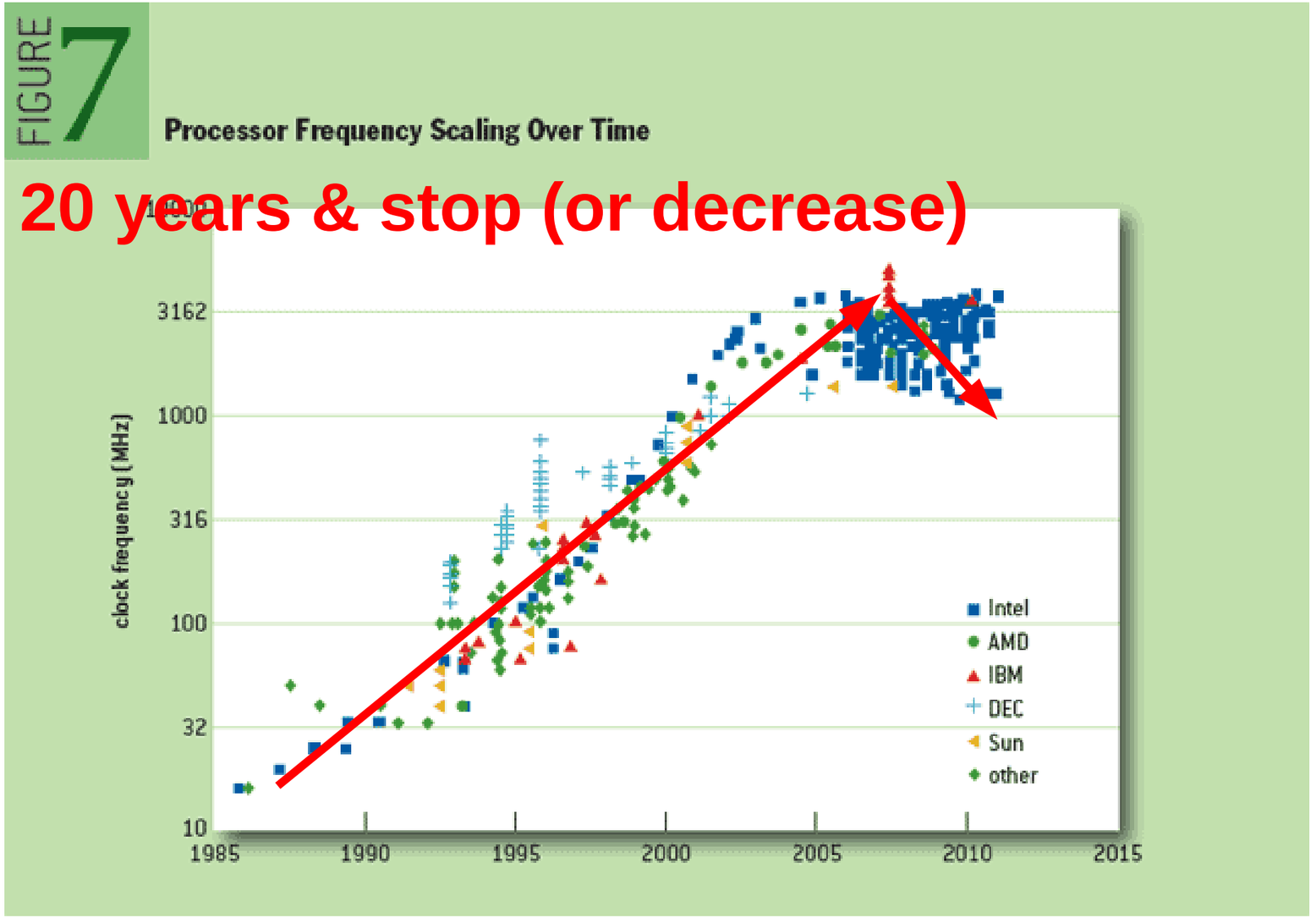
- Between 1989 and 1999 : from 4 MHz to 400 MHz
- Between 1999 and 2004 : from 400 MHz to 3 GHz
- Between 2004 and 2009 : from 3 GHz to 2 GHz
- TDP = ½ C V^2 f
  - C = Capacitance, f = frequency, V = voltage
- TDP for a processor : 115 W (on 4 cm^2)
  - Density of heat of an Induction Hob
- TDP becomes the blocking factor of a processor

Capacitance = $Schrink^2$. Nb Transistors . Mylq Constant (~ 0.015)

Why Parallelism ?

When clock speed was synonym of Velocity...

x200 in 20 years & stop (or decrease)



How much is Parallelism ? Time, Silicone, Complexity...

- The **3-Time** costs :
  - Entry cost, Operating cost , Exit cost
  - Execution time compared to adaptation time
- **Silicone** : technologies have different prices
  - SMP (Shared Memory Processors) are expensive & limited
  - MPP need very specific networks
  - Clusters are easely extensible
- **Complexity** : corollary of large amount of gates
  - A GPU core is simpler than a CPU core
  - A GPU core is about 30 times slower than CPU core

How to « think » : Parallelism : « *Grain... The problem is Grain.* »

- 1 Input / 1 Process ? Optimize process !
- 1 Input / Y Process ? Optimize each process !
- X Inputs / 1 Process ? Optimize distribution !
- X Inputs / Y Process ? Optimize both !

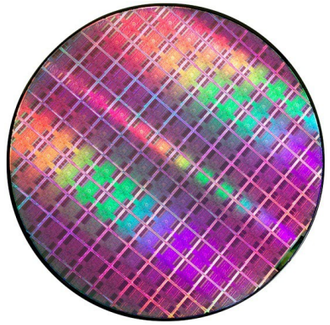
Grain is defined by communication rate !

- **Fine grain** : heavy communications (>> 1/second)
- **Coase grain** : sparse communications (< 1/second)
- **Embarrassing parallelism** : independant tasks

How to « think » ParallelisM : Flynn Taxonomy

- SISD : Simple Instruction Simple Data
- SIMD : Simple Instruction Multiple Data
  - Vectorization

- MISD : Multiple Instructions Simple Data
  - Pipelining
- MIMD : Multiple Instructions Multiple Data

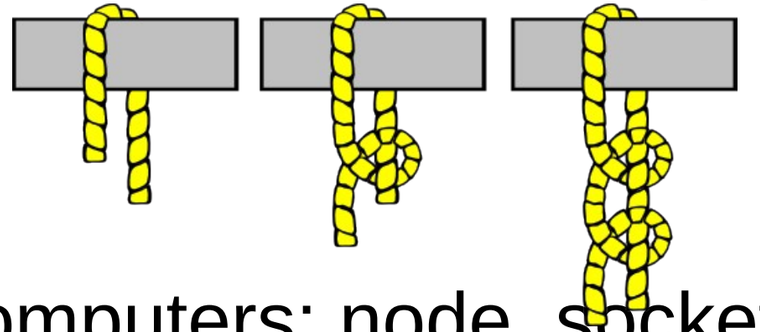


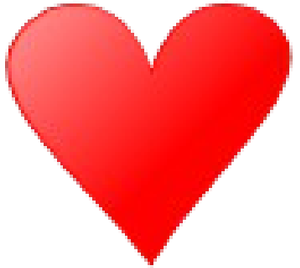
Let's have a look « Behind the Kitchen Door » (In Silicon) ?

To be (a matriochka) or not to be ?
From a processing point of view

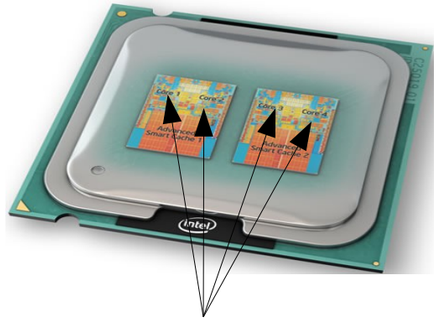
Computing System ?
A Silicium/Copper Matrioshka...

- In French : *Nœud ? Socquette ? Cœur ?*

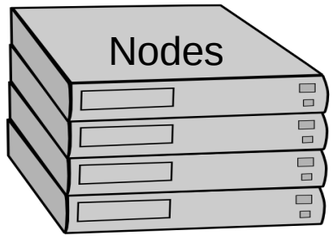


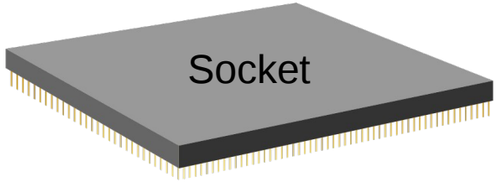


- In computers: node, socket, core :
  - In a cluster, from 2 to 400 nodes (in PSMN)
  - In a node, from 1 to 4 sockets
  - In a socket, from 1 to 16 cores
  - In a core, up to 9 ALU

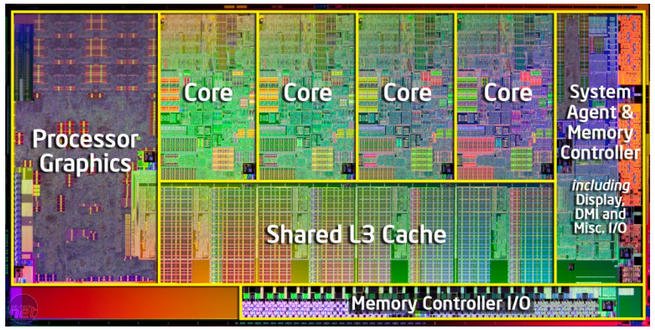


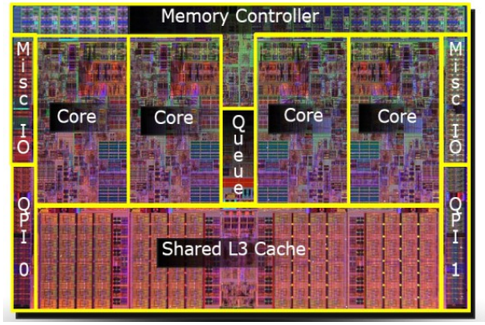
Cores

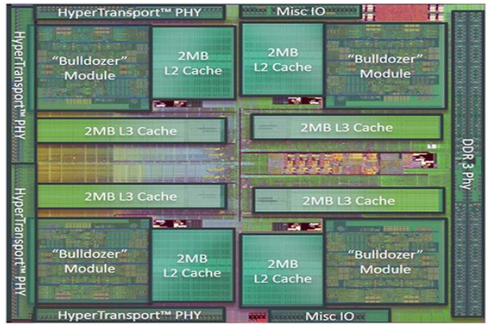


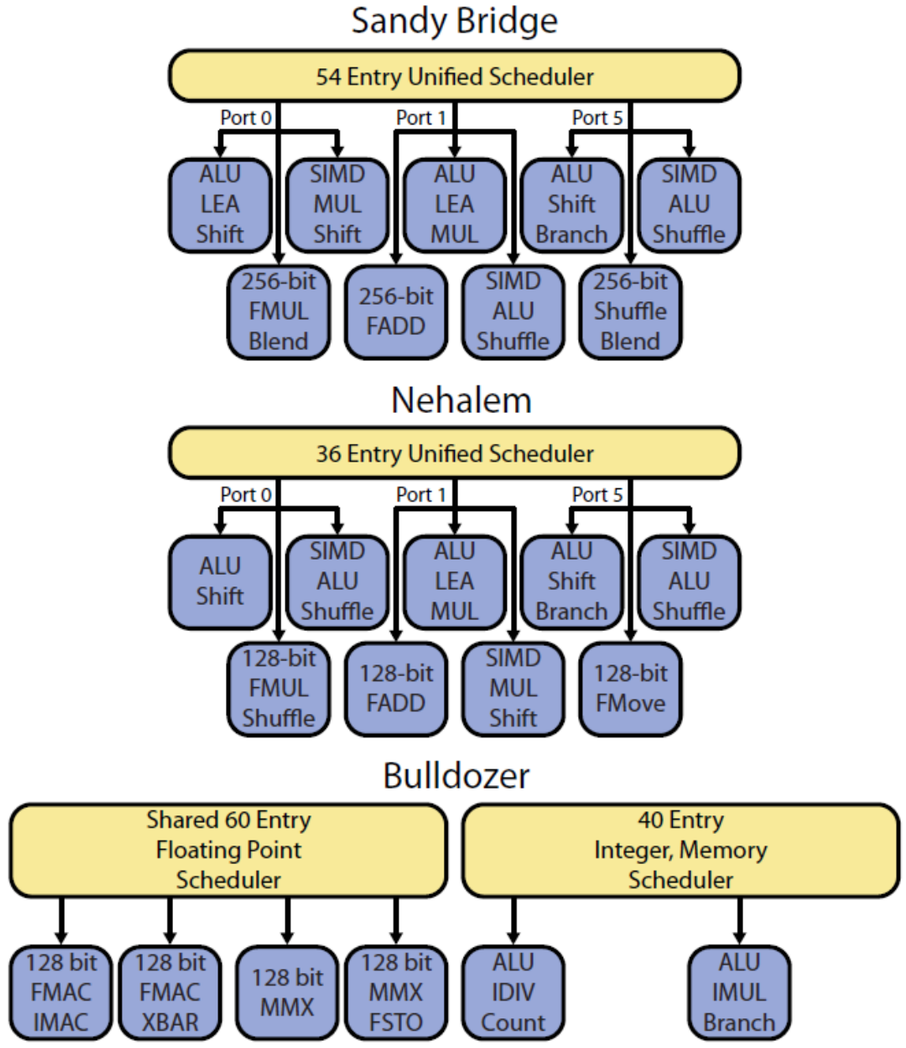


Inside a processor (on a Socket)
4-cores Processors Example

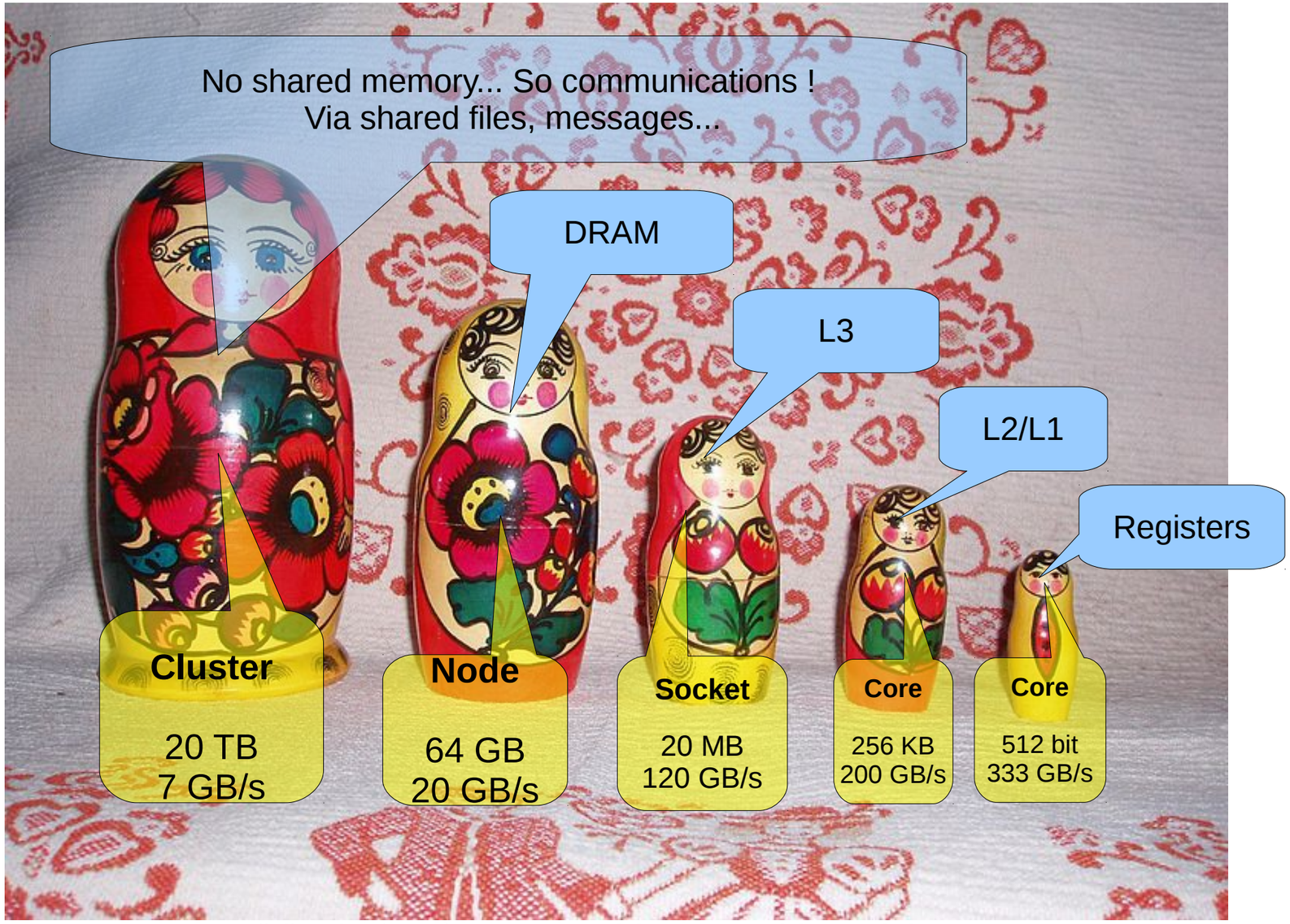








To be (a matriochka) or not to be ? Hierachical Memories !



If computing is cooking…
For Memory...

Code ~ Recipie

Computer ~ Kitchen

Input Data ~ Ingredients

Output Data ~ Meal Dish

Process ~ Cooking process

Control Unit ~ Cooker

ALU ~ Utensil

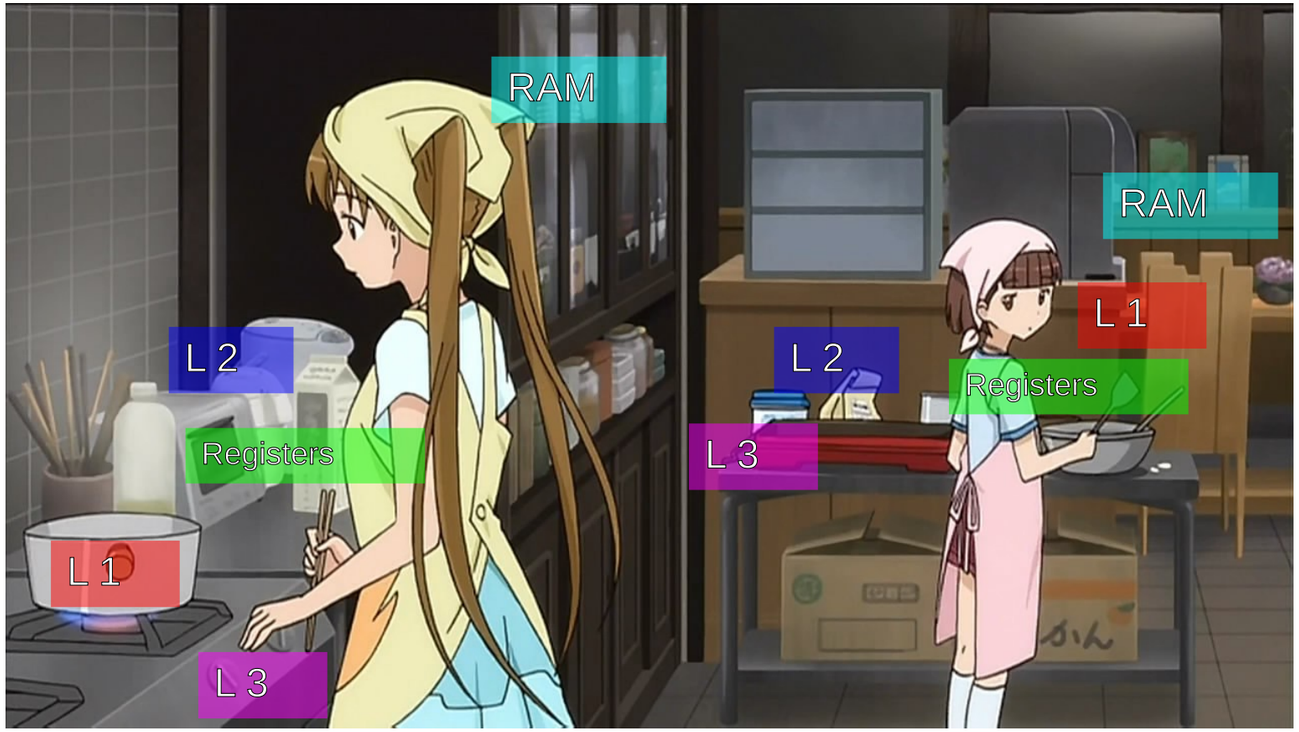
Dynamic RAM ~ Cupboards, tables, ...

L3 Cache ~ All Working planes

L2 Cache ~ Near Working plane

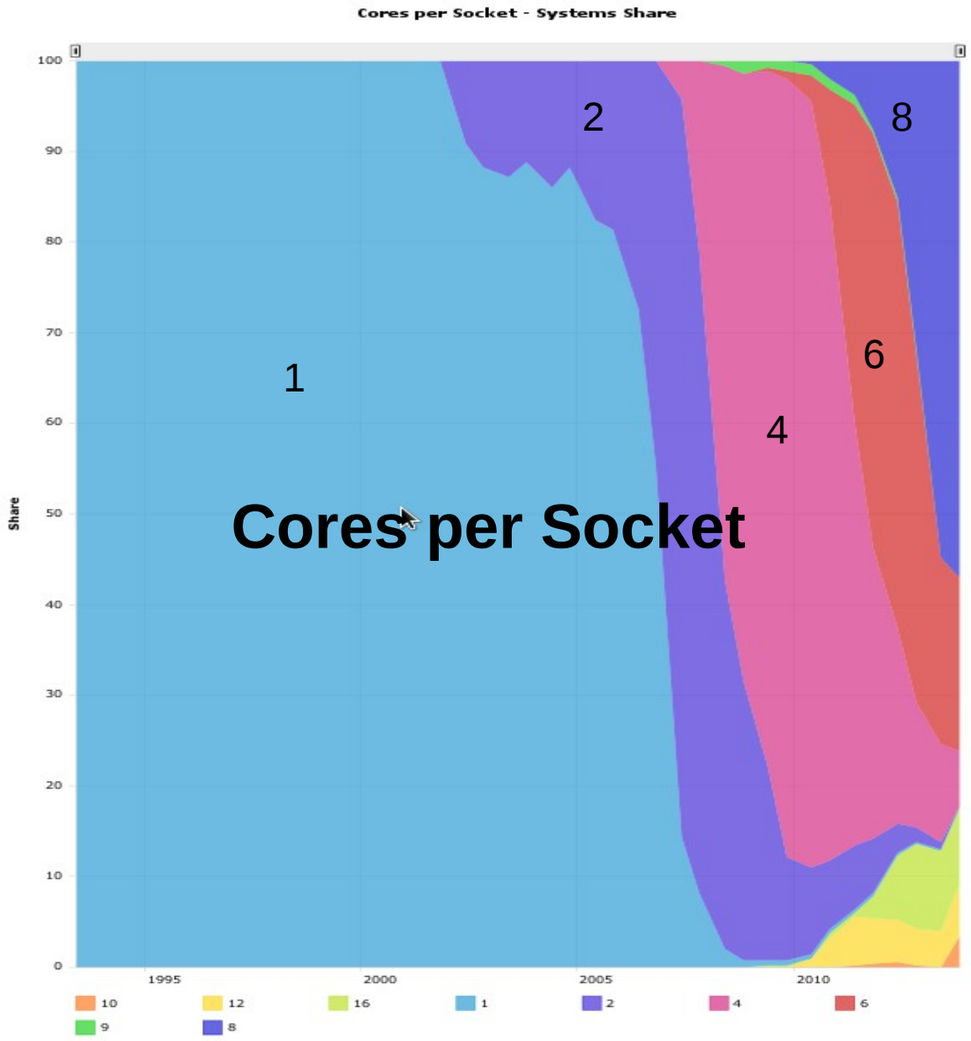
L1 Cache ~ Cutting board, container

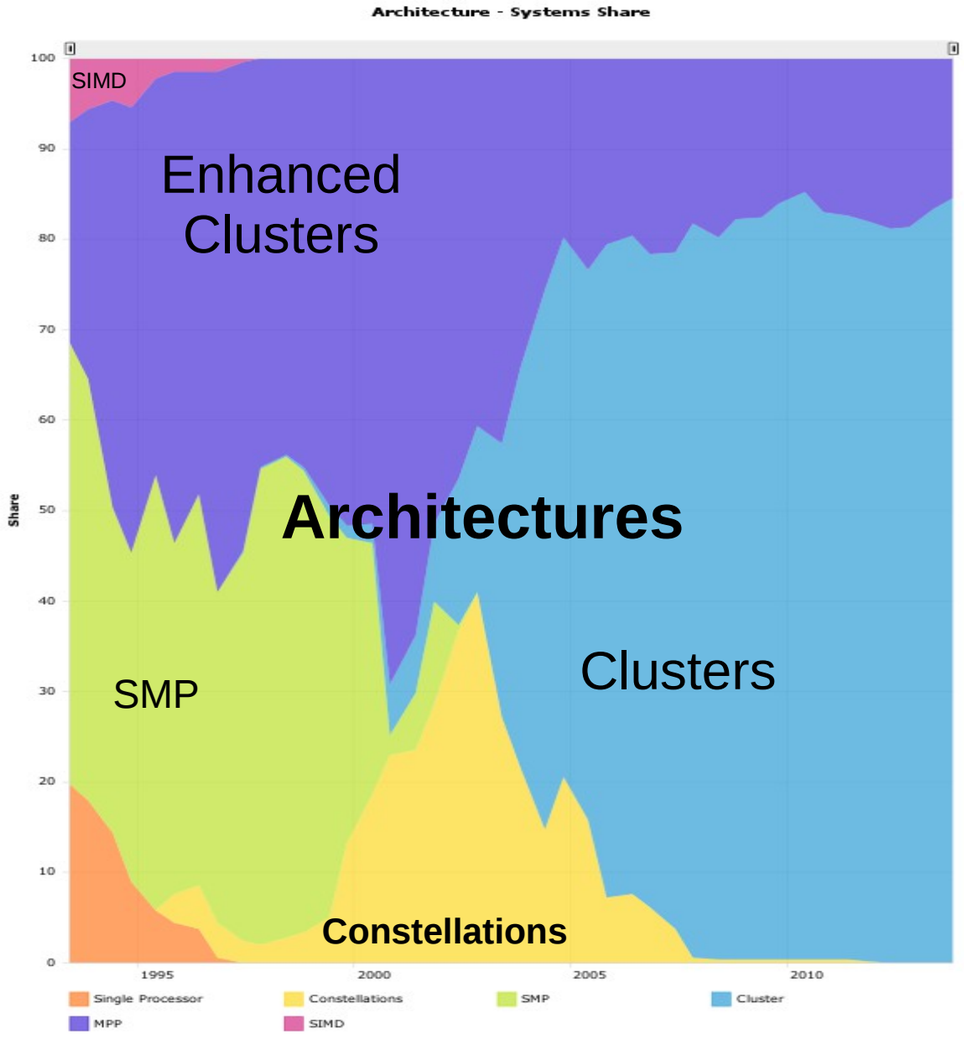
Registers ~ Hands of Cooker



Where you are ? Where you (will) go ?
Cores per Socket & Architecture...

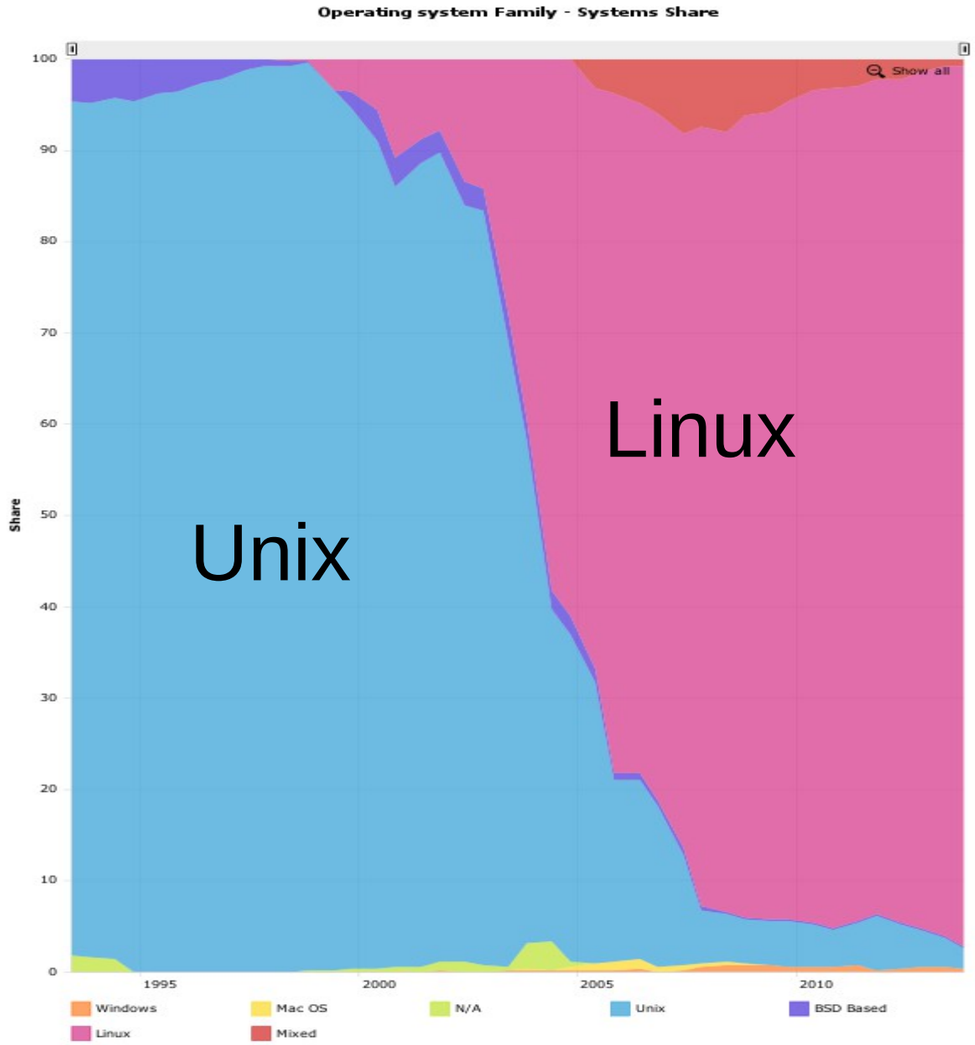
Source TOP500 : http://www.top500.org





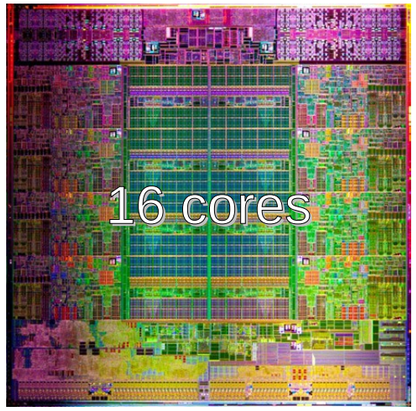
OS : *Linux for (almost) all...*
Accelerators : *work in progress...*

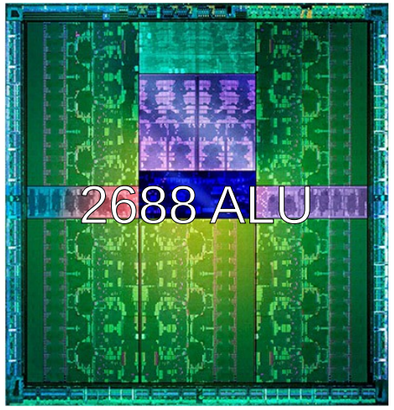
Source TOP500 : http://www.top500.org

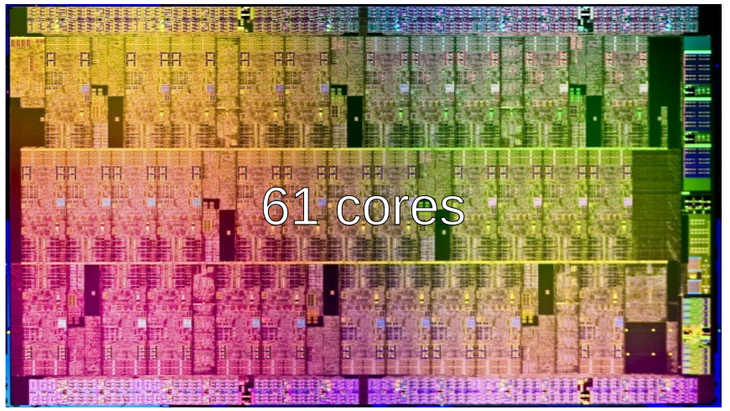


Multiplication of cores
From multi-core to myri-ALU

- CPU, 4 in laptop, 16 in workstation, 48 in node
- From GPU to GPGPU :
  - A tiny GPU card : 128 ALU, 512 MB of RAM
  - A huge GPU card : 2588 ALU, 6 GB of RAM
  - A huge GPGPU card : 2880 ALU, 12 GB of RAM
- Accelerator Xeon Phi : 61 CPU (Pentium like units)







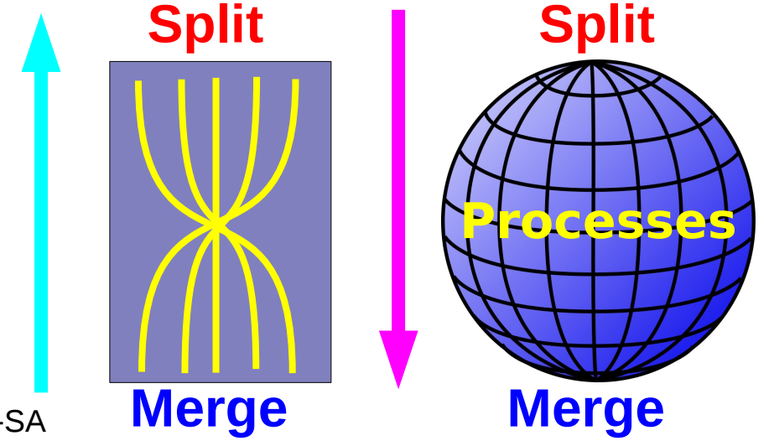
How to Parallel Programming ?
Split/Merge between process(es)

- Pipelining fine grain, a job for silicon :
  - 5 simple instructions @ a time
    - Intruction Fetch
    - Instruction Decode
    - Execute
    - (MEM)
    - Write Back
  - 2 specs of RISC : 1 instruction/cycle, using registers

Instr. No.	Pipeline Stage						
1	IF	ID	EX	MEM	WB		
2		IF	ID	EX	MEM	WB	
3			IF	ID	EX	MEM	WB
4				IF	ID	EX	MEM
5					IF	ID	EX
Clock Cycle	1	2	3	4	5	6	7

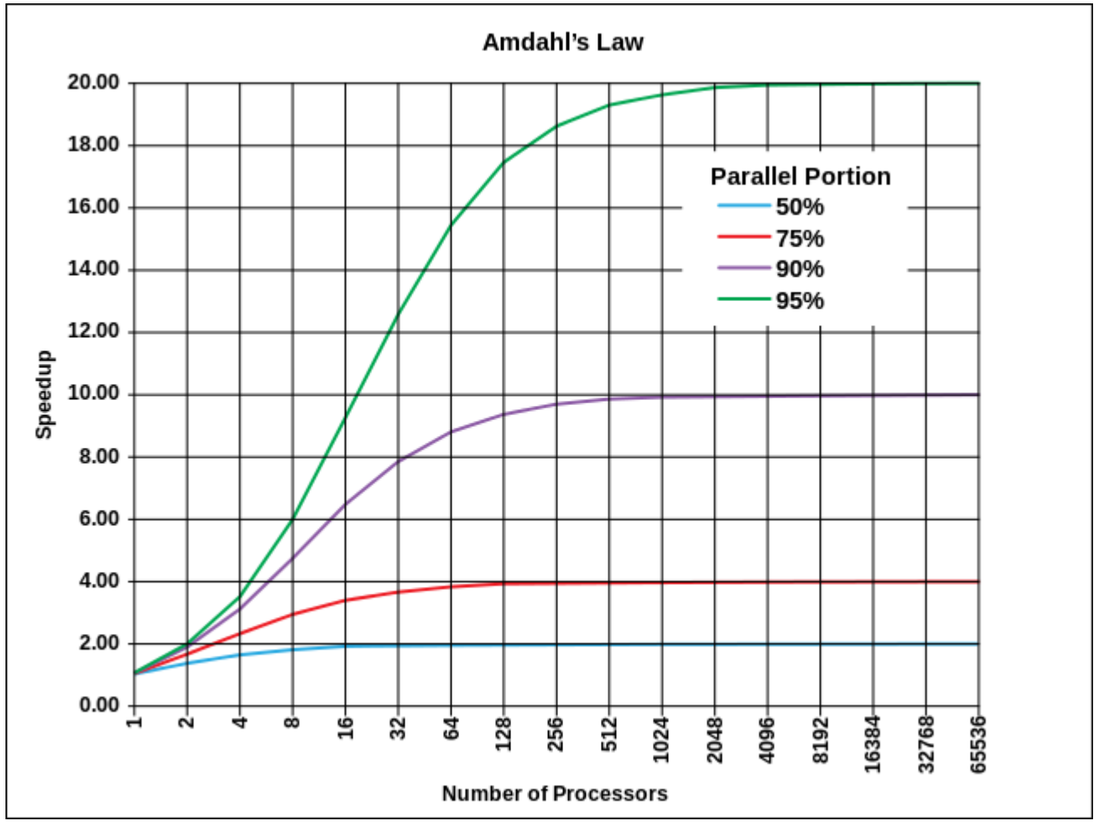
2 approaches :

- Vectorization : Merge/Process/Split
- Distribution : Split/Process/Merge
- In fact, not parallize but meridianize

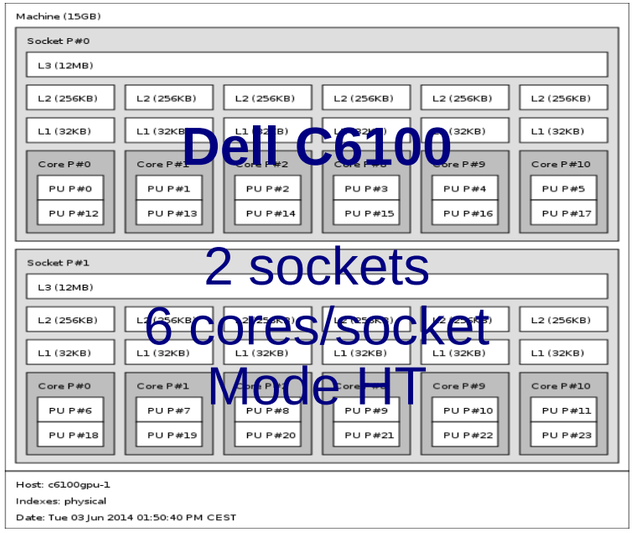


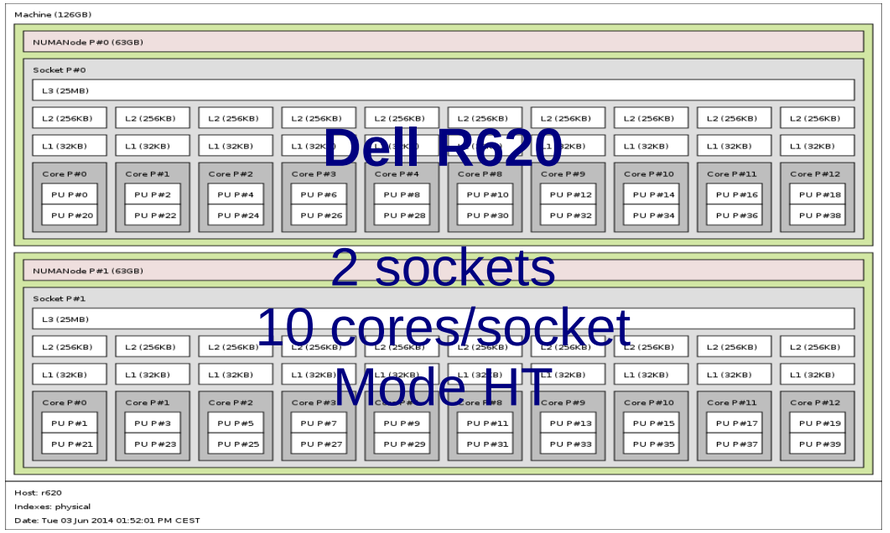
How to Parallel Programming ? Amdahl Law, order (and decay)

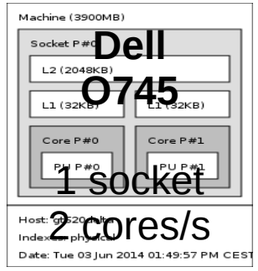
- In the process, 2 parts
  - Sequential part, in fraction *s*
  - Parallel part, in fraction *p*
  - Acceleration = *1/(1-p+p/N)*
- Acceleration (& efficiency) : N=1000
  - 90 %, acceleration 10, efficiency 1 %
  - 99 %, acceleration 91, efficiency 10 %
  - 99.9 %, acceleration 500, efficiency 50 %
- **Questions** :
  - What's about scalability of my code ?
  - Is Amdahl law representative of « real » applications ?

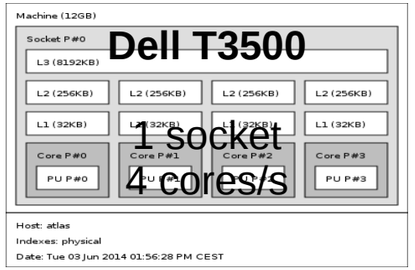


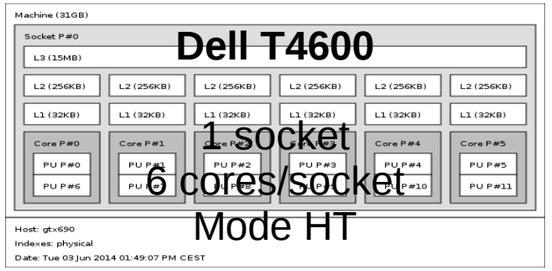
Welcome, *Amdahl*, to the Real World !
10 machines from 2 to 40 cores...

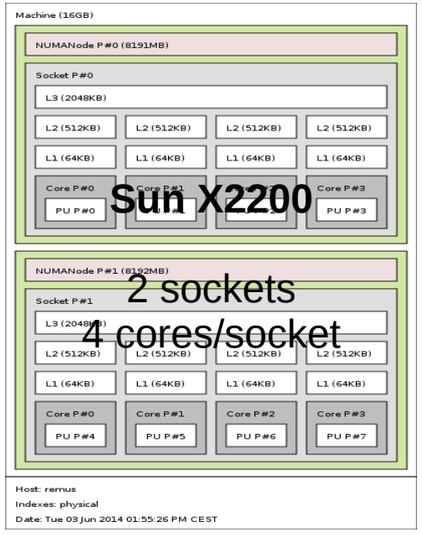


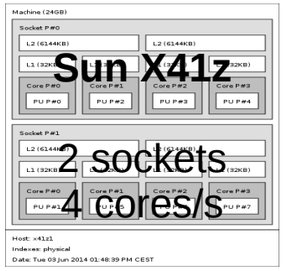




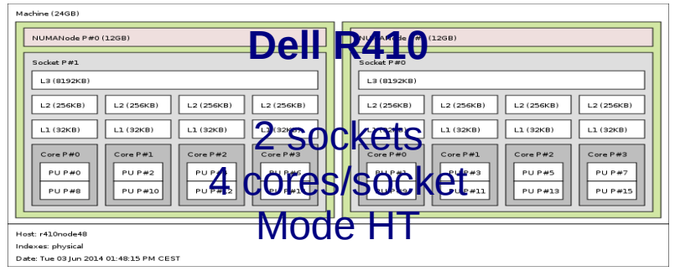


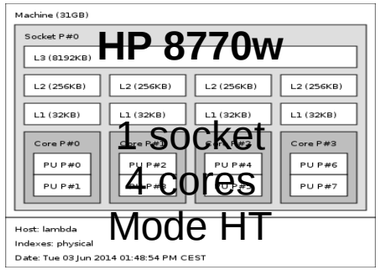


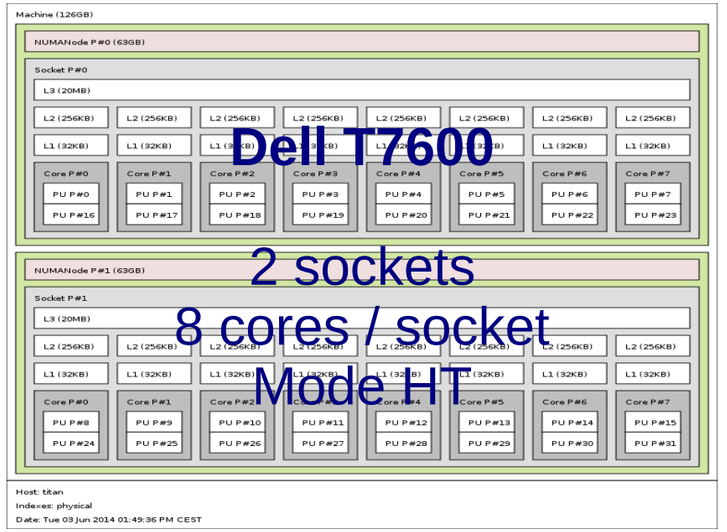




hwloc-ls as command

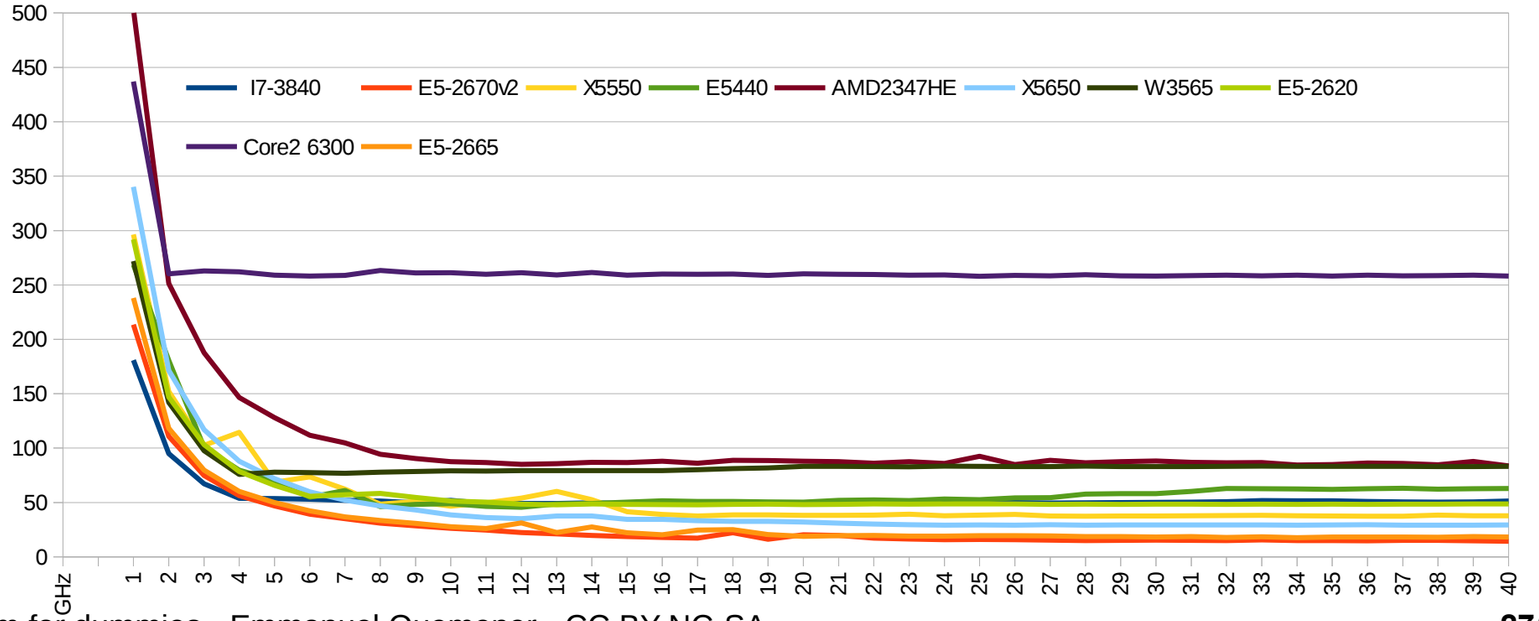






Amdahl Law in the real world
A test bench : 10 CPUs, 1 code

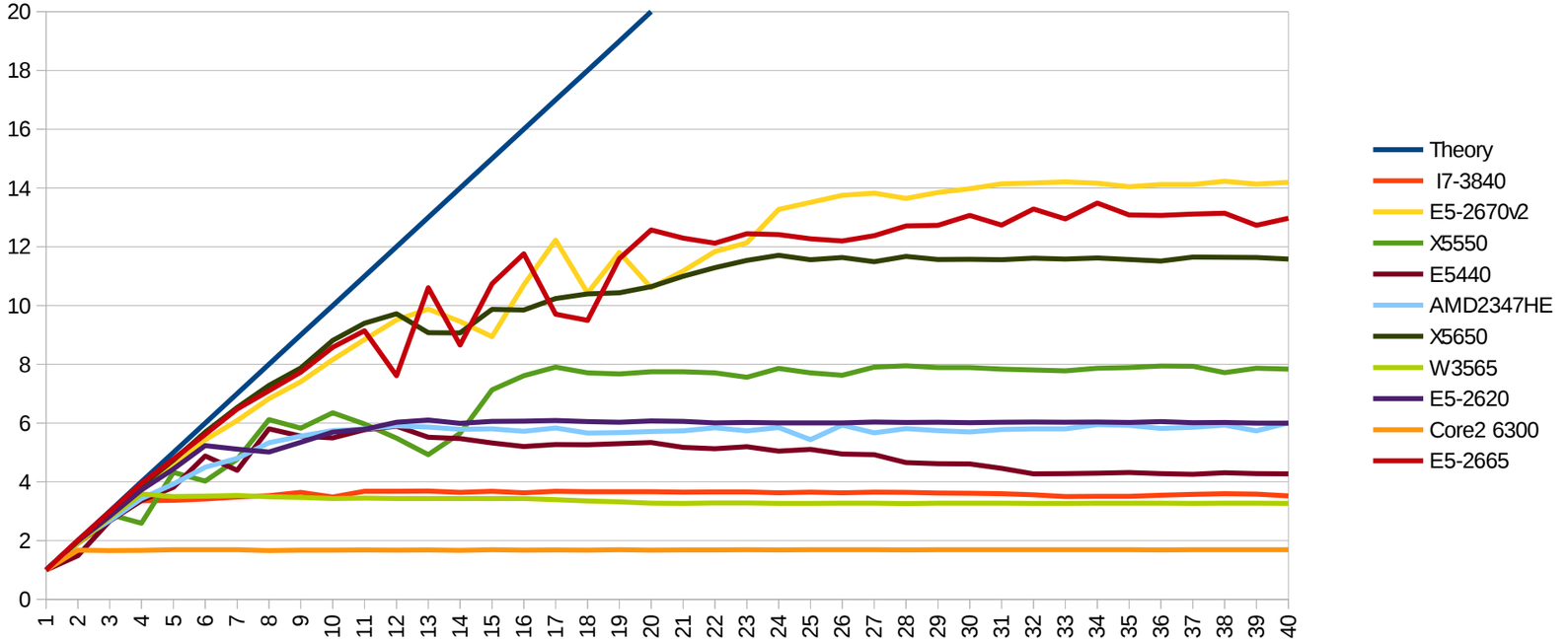
- 10 different CPUs, from 2 to 20 cores (40 in HT)
- 1 application : pbzip2 (parallel bzip2)
- 1 data set : encoded film (1.4 GB) (worst scenario)
- From 1 process to 80 process
- Metrology Tool : time
- Observable : elapsed



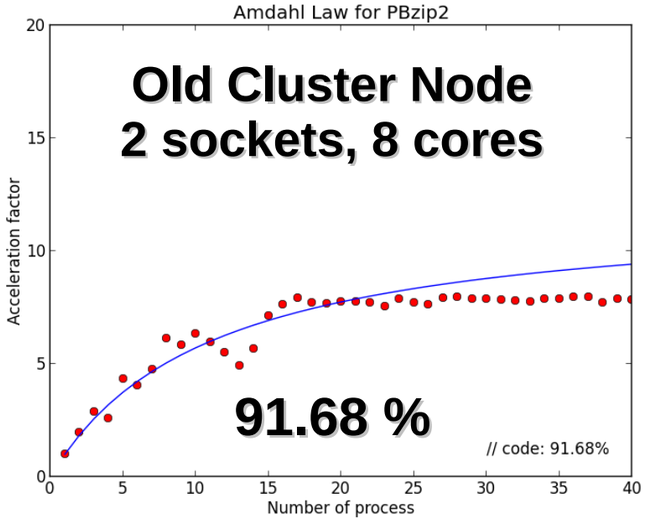
Amdahl Law in real world Acceleration & Variations

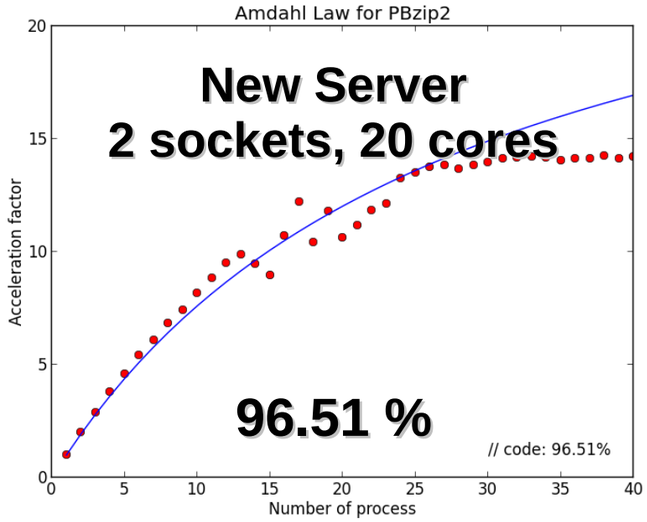
- Symptoms :
  - On large number of cores, 70 % to 80 % efficiency
  - Great variations on recent twice-sockets machines
  - Decrease for heavy charges on old processors

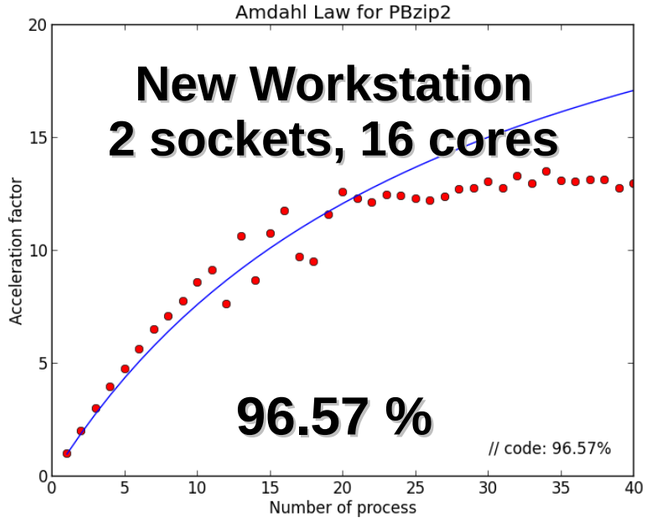
Why ?

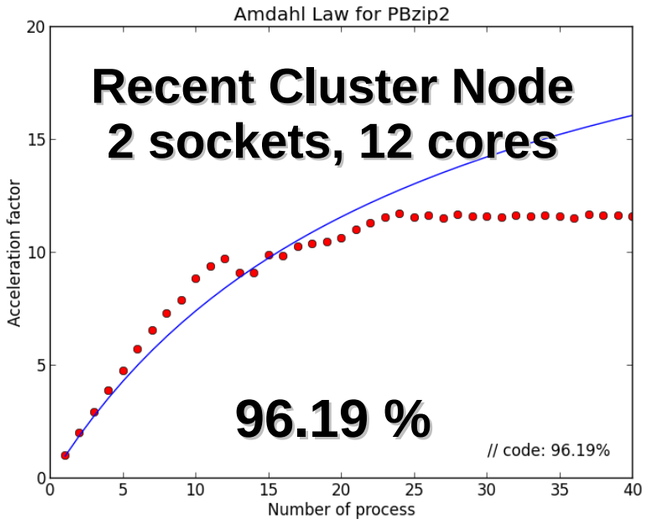


Amdahl Law : Fitting images !

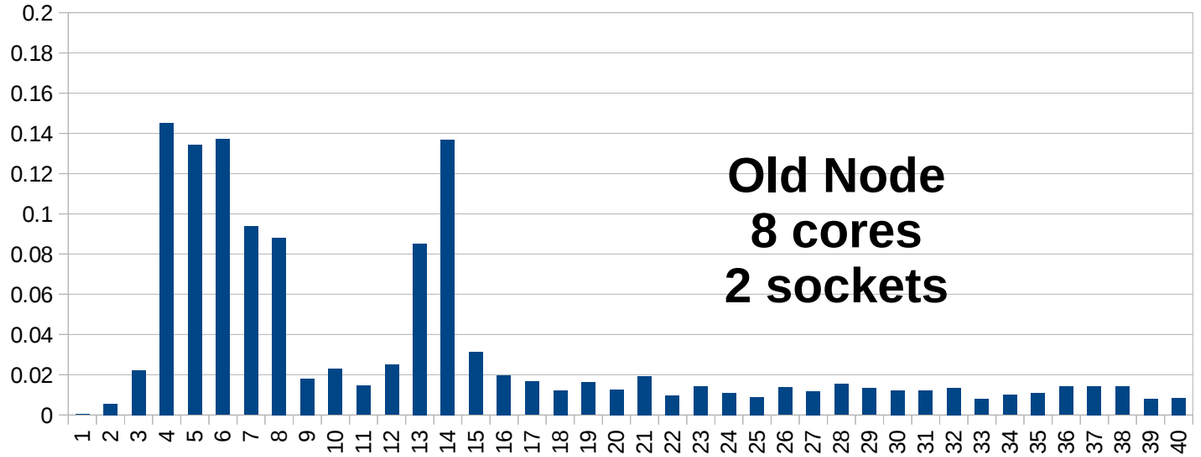


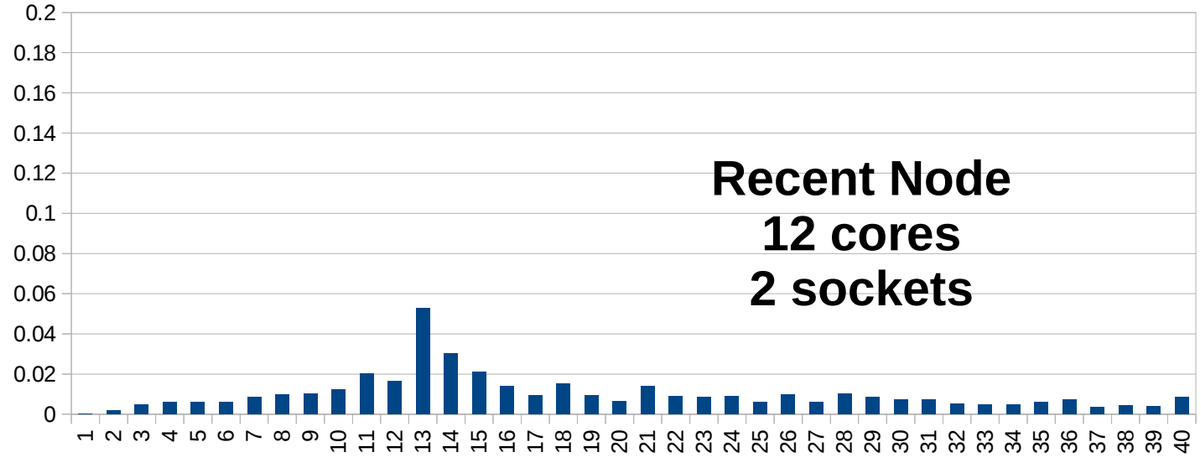


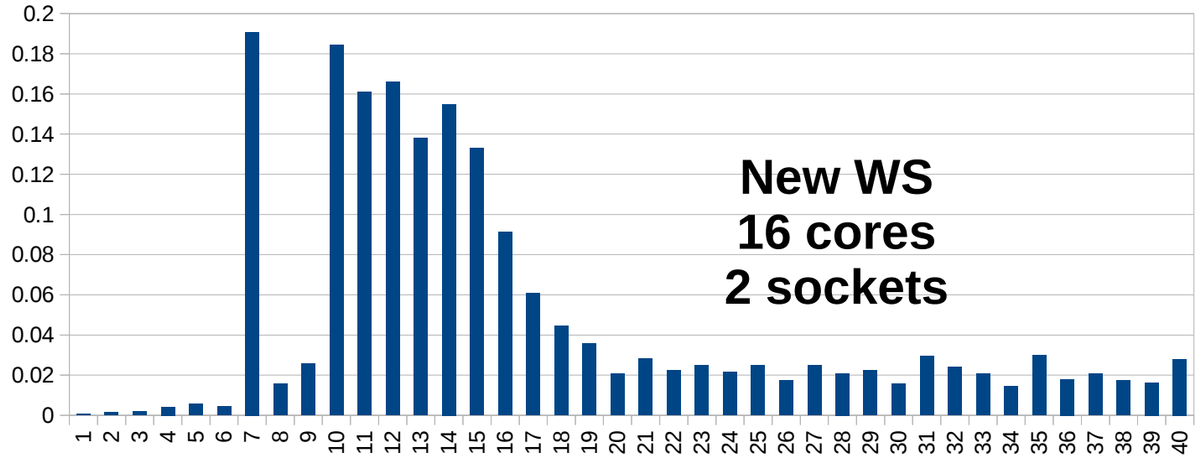


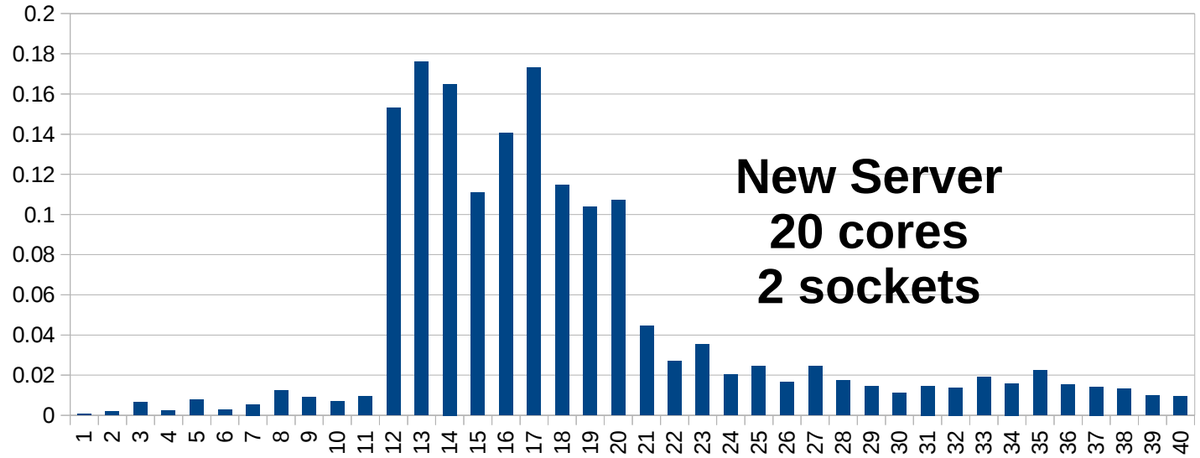


Amdahl Law in real world
Memory & Variability



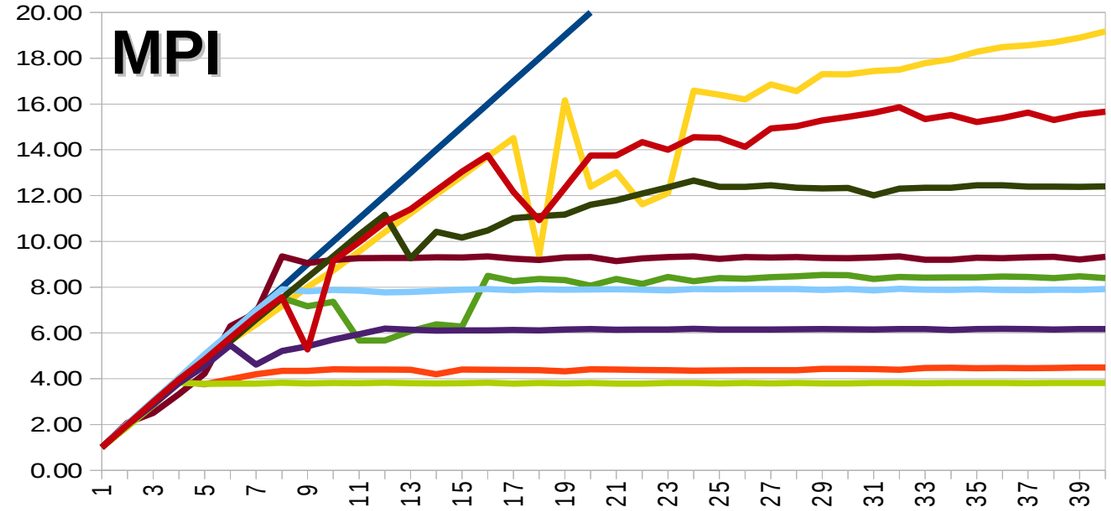




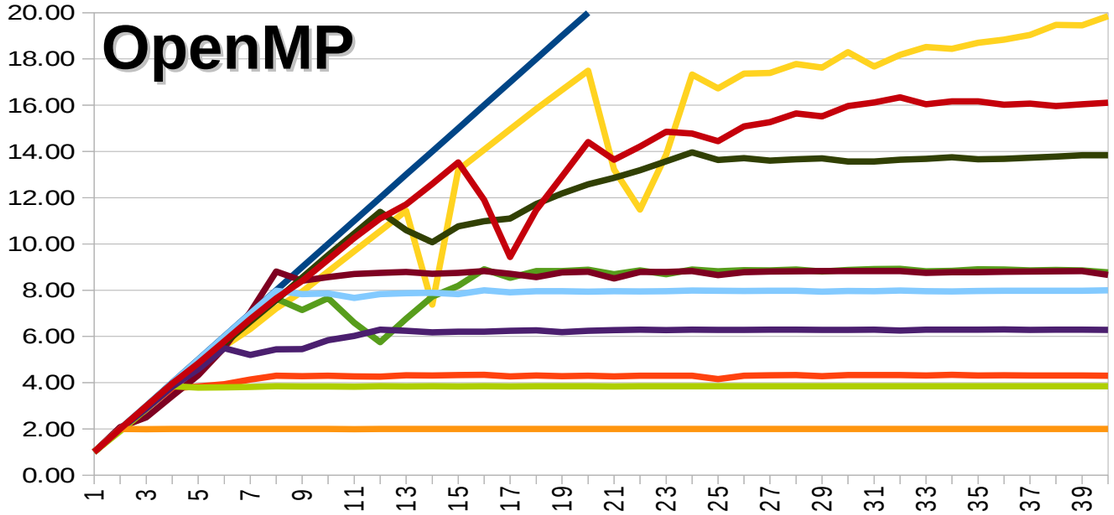


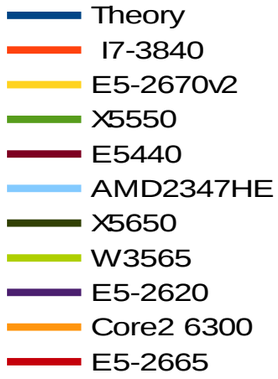
Morality : improve your experiments !

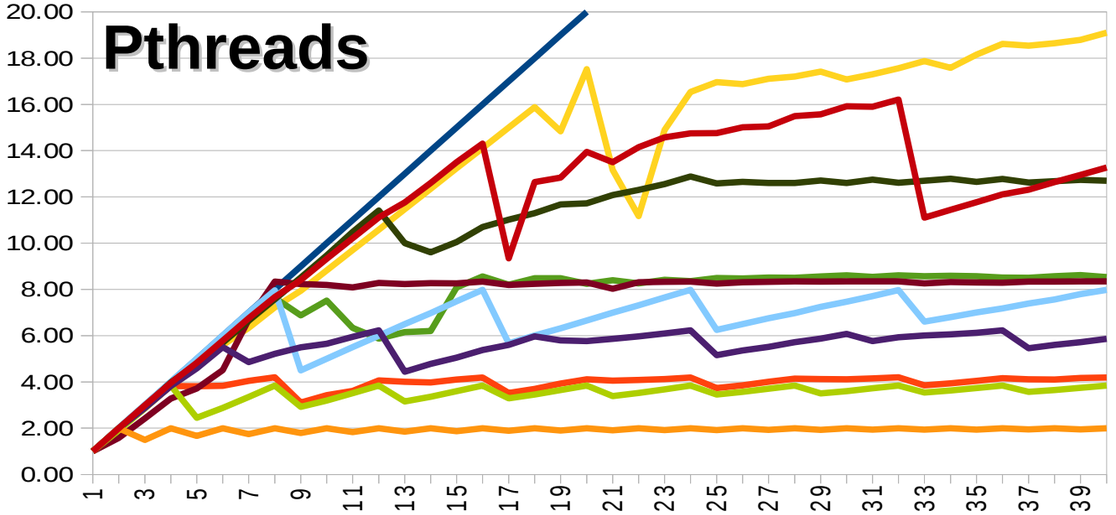
Amdahl Law in Real World
For « memory free » applications



- Morality :
  - Implementations comparable
  - Practical near Theory on Max
  - Hyper Threading interesting
  - Inevitable Glitches
  - **Always improve statistics !**







Parallelism : approaches Hardware/Software sides

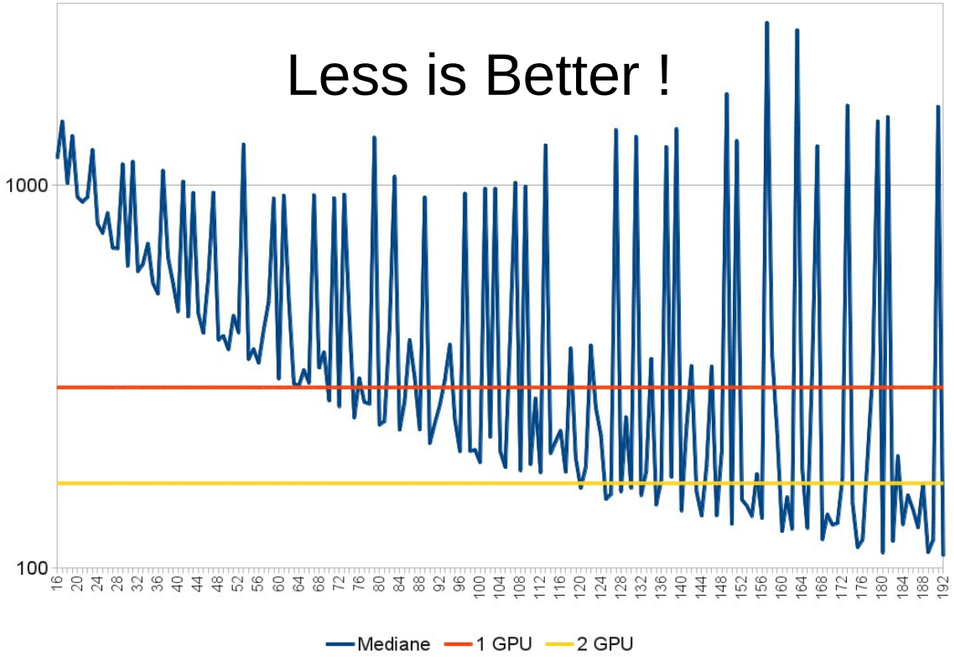
Parallel Programming Models

	Cluster	Node CPU	Node GPU	Node Nvidia	Accelerator
MPI	Yes	Yes	No	No	Yes*
PVM	Yes	Yes	No	No	Yes*
OpenMP	No	Yes	No	No	Yes*
Pthreads	No	Yes	No	No	Yes*
OpenCL	No	Yes	Yes	Yes	Yes
CUDA	No	No	No	Yes	No
TBB	No	Yes	No	No	Yes*

Parallel Programming Libairies

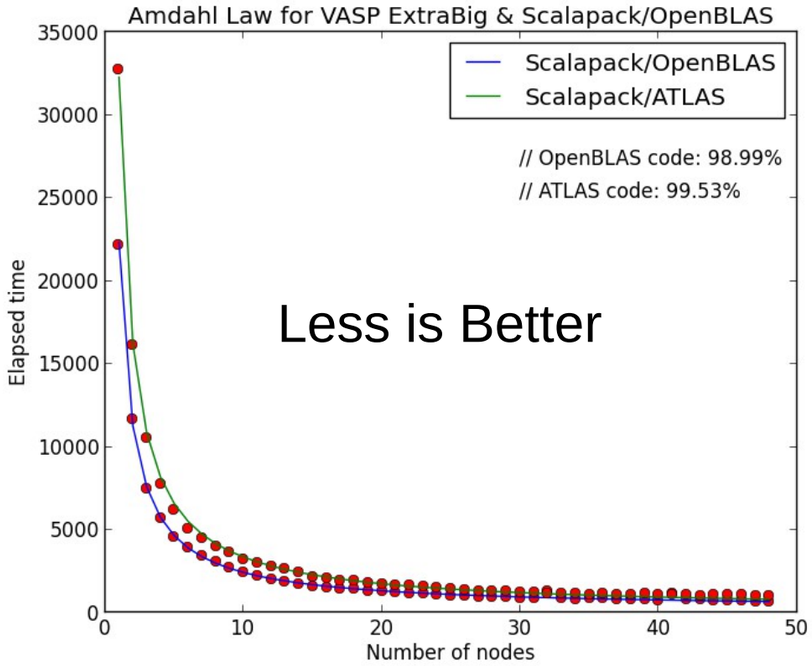
	Cluster	Node CPU	Node GPU	Node Nvidia	Accelerator
BLAS	BLACS MKL	OpenBLAS MKL	clBLAS	CuBLAS	OpenBLAS MKL
LAPACK	Scalapack MKL	Atlas MKL	clMAGMA	MAGMA	MagmaMIC
FFT	FFTw3	FFTw3	clFFT	CuFFT	FFTw3

Parallelism : Back to Science
2 applications where It works...



Lammps application

- From 16 to 192 NP
- Compared to GPGPU
- 99.96 % //ized (record!)
- 2GPU equal 120 NP



VASP application

- From 1 to 48 NP
- 99 % to 99.5 % //ized
- OpenBLAS vs ATLAS...

Parallelism on CP2K
From an user Point of View...

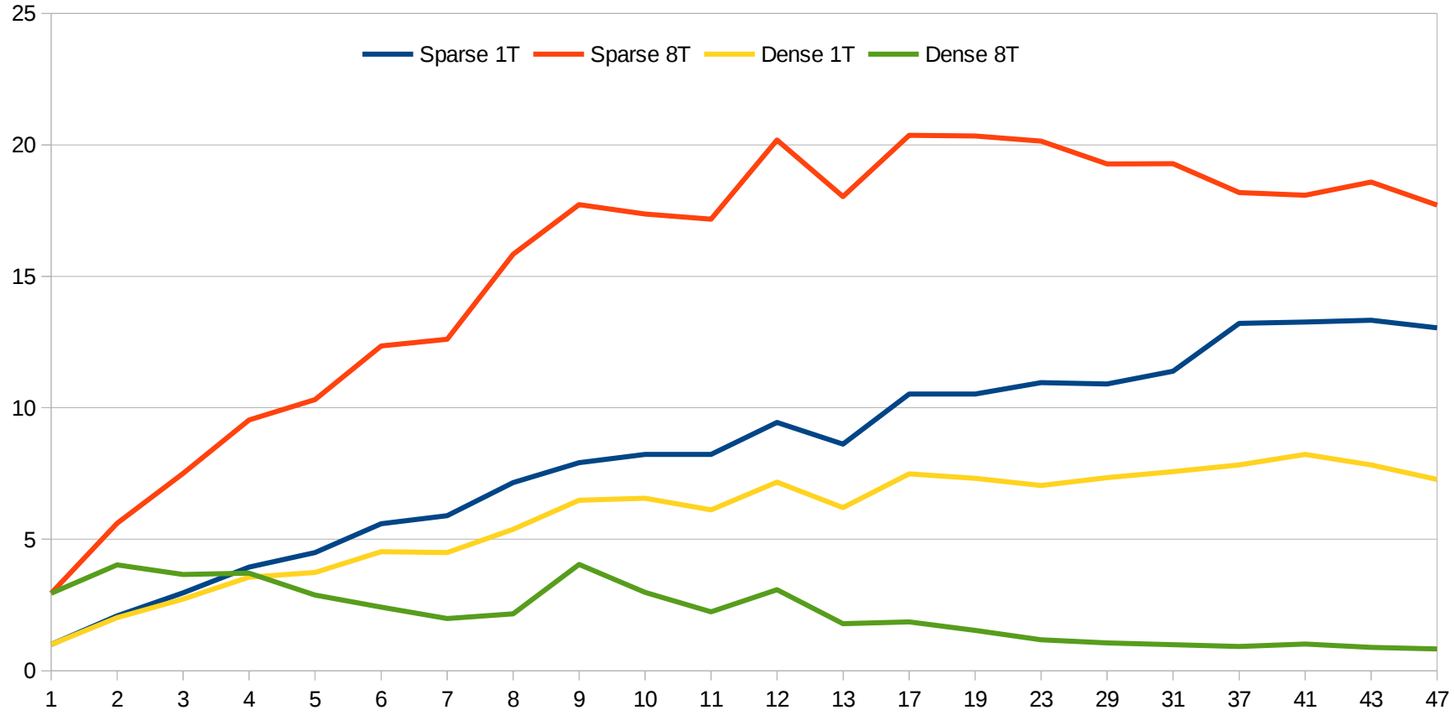
- 4 versions (in fact 8) :
  - Sequential : sopt
  - Parallelized on node (OpenMP & Pthreads) : ssmp
  - Parallelized on cluster (MPI) : popt
  - Parallelized on cluster & node : psmp
- 2 implementations on BLAS :
  - ATLAS : sequential BLAS routines
  - OpenBLAS : multithreaded BLAS routines
- How to « control » parallelization on P process :
  - MPI : mpirun -np P <MyCode>
  - OpenMP & Pthreads : OMP_NUM_THREADS=T <MyCode>
  - Hybrid : mpirun -np P -x OMP_NUM_THREADS=T <MyCode>

Parallelism on CP2K
Test Bench

- Cluster : 48 nodes R410, GE & Infiniband QDR
- Node : 2 sockets with 4 cores each, 24 GB RAM
- OS : Debian Wheezy on SIDUS, kernel 3.14
- Software backend : standard & backports
  - OpenMPI 1.4.5, FFTw3 3.3.2, OpenBLAS 0.2.8
- Code : CP2K 2.5.1, Hybrid version : OpenBLAS & psmp
- Input : H2O-128 benchmark
- Exploration from 1 to 48 nodes :
  - Sparse & Dense mode (how to fill nodes)
  - 1 Thread & 8 Threads (distribution on cores)

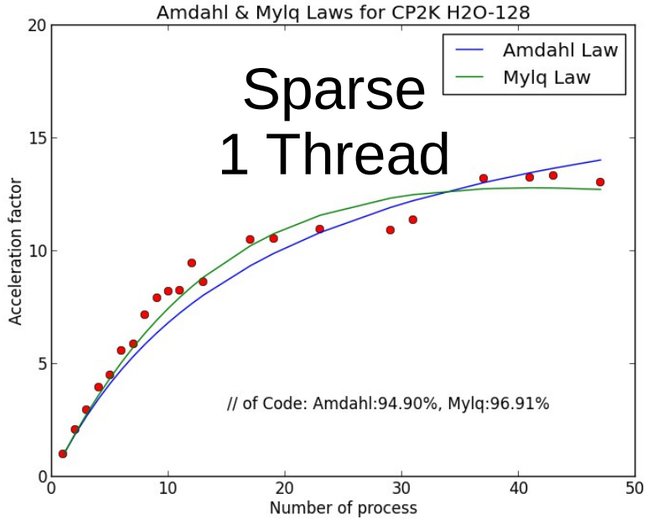
Parallelism on CP2K Results (& Conclusions) !

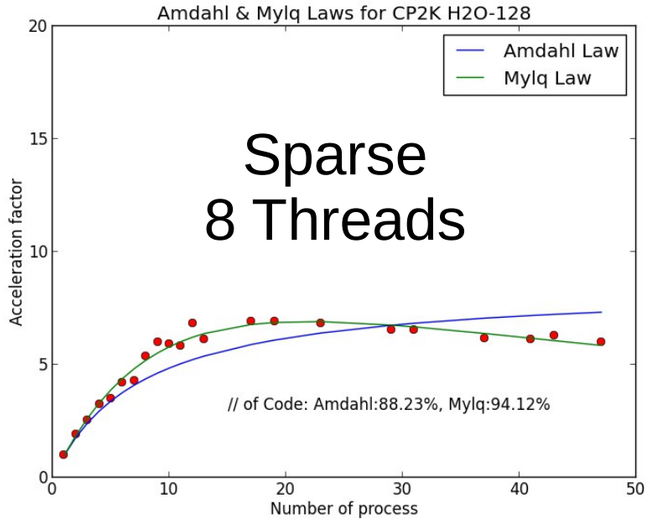
- A « smart » prime number discoverer :
  - Only (booouuuh !) : 1 to 13, 17, 19, 23, 29, 31, 37, 41, 43, 47
- Curves in vrac, and estimation of Amdahl p !
  - Sparse 1T : 95 %
  - Sparse 8T : 88 %
  - Dense 1T : 90 %
- Morality :
  - Hybrid better
  - Sparse better
  - Acc up to 20
  - Amdahl Law is bullsh*t

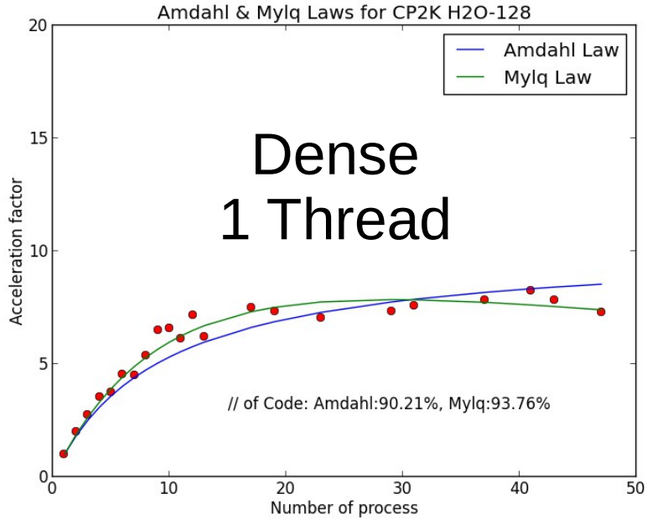


Parallelism of CP2K
How to « fit » better ?

- Add correction factor in Amdahl law :
  - From *1/(1-p+p/N)* to *1/(1-p+p/N+c*n)*
  - c : Mylq Constant
  - Best parallel factor, simplest linear correction
- Finally, use Mylq Law ;-)

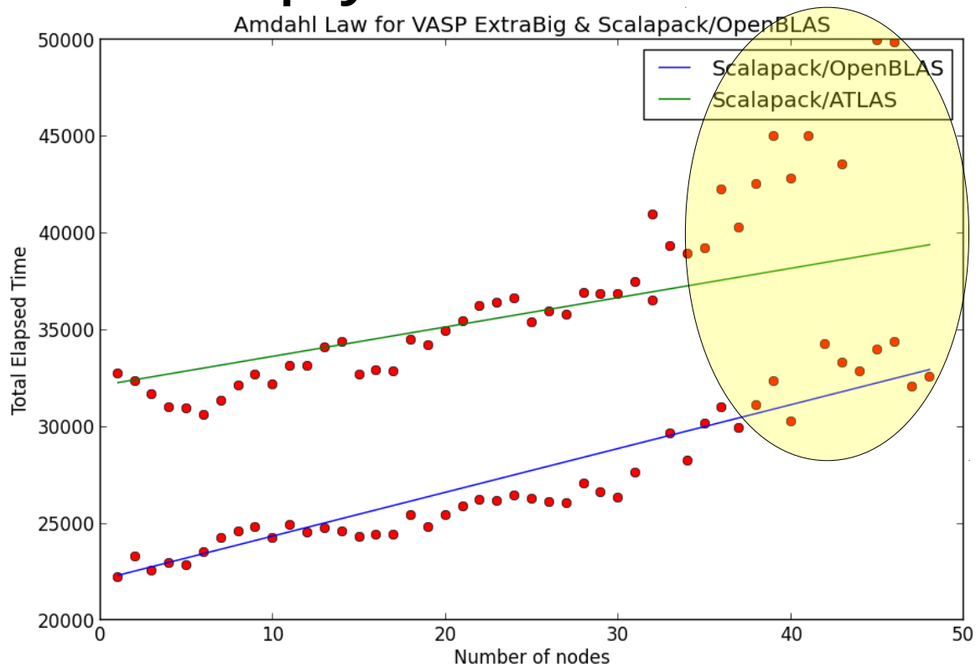






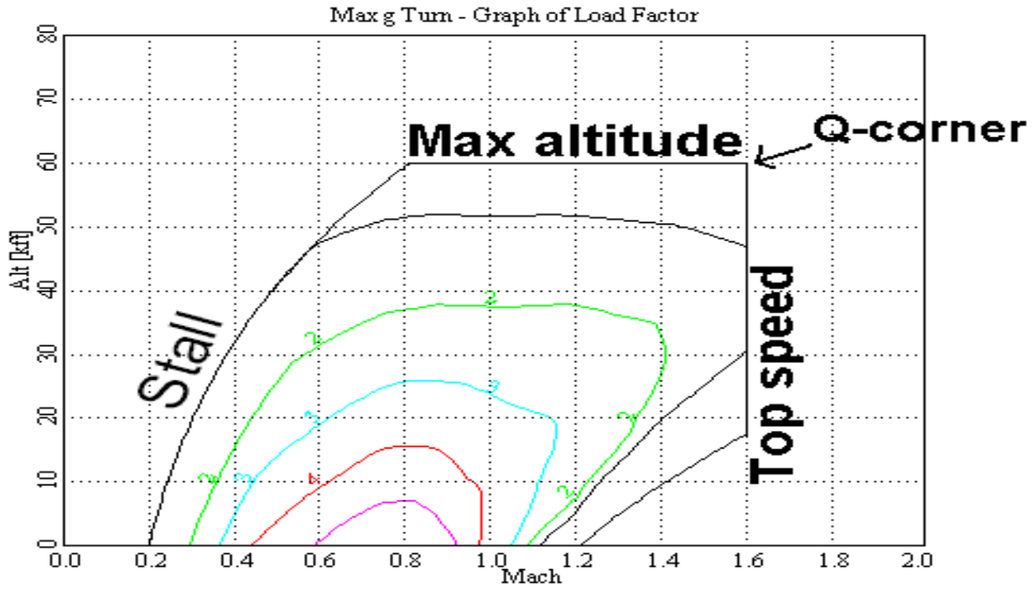
Observables in Parallelism
Elapsed is one, NRJ another...

- From Elapsed Time to Total Elapsed Time
  - Cost : Material Resources (space) & Elapsed Time
  - Multiply Elapsed time by nodes used and look…
- Don't multiply the nodes used : **it kills polar bears**



From Aeronautics To Computing Sciences

- Flight envelope:
  - Airpeed, Altitude,
  - Load factor
- Defines :
  - Best practices not to « crash »
- Back to computing sciences
  - Choose your system, select your observables
  - Experiments your representative statistics
  - Build your own « **parallel envelope** » for your system !



Conclusion

- Efficiency in parallelism demands
  - Vectorization, Distribution, Pipelining, ... and **Locality** !
- Hardware, OS, Software, Applications, Input Data, ...
  - Are NOT independent : they **form** One Unique system
  - A system has observables : choose yours !
- To One Problem, One System (& its parallelization)
- One simulation ~ One experiment : unique...
- Intrumentation is **absolutely** necessary
  - Inside the code (beware observable effects)
  - Outside the code (simple monitoring functions)
- Let's have some pratical work ?

Iconography & links

http://www.enigme-facile.fr/wp-content/uploads/2013/08/quel-est-le-comble-du-cuisinier.jpg

- http://whatisyourfoodworth.com/wiyfw-review-of-behind-the-kitchen-door/
- http://levelx.me/technology/from-sand-to-silicon-how-a-intel-cpu-is-built/
- http://upload.wikimedia.org/wikipedia/commons/e/e0/Cell-Processor.jpg
- http://www.fordfoundation.org/Images/newsroom/728-hero.jpg
- http://g-ecx.images-amazon.com/images/G/01/ciu/18/a4/c04570868f010ad8976f56.L._V146160188_SL290_.jpg
- http://silzel.com/images/tux-laptop-md.png
- http://openclipart.org/image/800px/svg_to_png/188285/CPU-20131103.png
- http://www.tech-faq.com/wp-content/uploads/images/Quad-Core-Processor.jpg
- http://images.bit-tech.net/content_images/2011/01/intel-sandy-bridge-review/sandy-bridge-die-map.jpg
- http://www.realworldtech.com/includes/images/articles/sandy-bridge-5.png?71da3d
- http://origin.arstechnica.com/reviews/hardware/nehalem-launch-review.media/NehalemDie.jpg
- http://blogs.pcmag.com/miller/assets_c/2011/08/Buldozer%20Die%20Floorplan%20HC23-thumb-450x357-24701.jpg
- http://hothardware.com/articleimages/Item1742/bulldozer-die.jpg
- http://www.realworldtech.com/includes/images/articles/sandy-bridge-5.png?71da3d
- http://upload.wikimedia.org/wikipedia/commons/c/cf/Samsung_Galaxy_S5.png
- http://openclipart.org/detail/167173/ear-by-johnny_automatic
- http://openclipart.org/image/800px/svg_to_png/140689/brain_draw.png
- http://openclipart.org/image/800px/svg_to_png/140683/brain2colors.png
- http://www.mo5.com/musee/histoire/images/jacquard.gif
- http://upload.wikimedia.org/wikipedia/commons/b/b4/Pascaline_calculator.jpg
- http://fr.wikipedia.org/wiki/Pascaline#mediaviewer/Fichier:Arts_et_Metiers_Pascaline_dsc03869.jpg
- http://fr.wikipedia.org/wiki/Machine_d%27Anticyth%C3%A8re#mediaviewer/Fichier:Antikythera_model_front_panel_Mogi_Vicentini_2007.JPG
- http://fr.wikipedia.org/wiki/Von_Neumann#mediaviewer/Fichier:Von_Neumann_architecture.svg
- http://diit.cz/clanek/nvidia-tesla-k40-s-2880-cuda-jadry-a-12-gb-gddr5-je-tu
- http://1.bp.blogspot.com/_VR3XpzDWOHo/SwbPinfyIVI/AAAAAAAAEVU/jqR-TMt__X4/s400/interieur-plaque-induction.jpg
- http://openclipart.org/image/800px/svg_to_png/24912/Anonymous_Globe_01.png
- http://en.wikipedia.org/wiki/Amdahl's_law#mediaviewer/File:AmdahlsLaw.svg
- http://images.anandtech.com/doci/7457/HawaiiArch.png
- http://avvesione.files.wordpress.com/2011/10/tamayura_hitotose-03-norie-komachi-cooking-kitchen-food.jpg
- http://upload.wikimedia.org/wikipedia/commons/2/20/Ursus_maritimus_us_fish.jpg
- http://cdn.itproportal.com/photos/Xeon-Phi-3Aubrey_Isle_die-640x480_original.jpg
- http://media.bestofmicro.com/T/W/328964/original/xeon-e5-die.jpg
- http://hothardware.com/articleimages/Item1989/small_gk110-die-shot.jpg