DE LA RECHERCHE À L'INDUSTRIE

www.cea.fr

GOING ARM
A CODE PERSPECTIVE

ISC18 | Guillaume Colin de Verdière

JUNE 2018

GCdV

A history of disruptions

All dates are installation dates of the machines at CEA/DAM

The scalar era

- CDC 7600 1976 - 1987
  - Small central memory
    - Larger out of core memory
  - 60bits words

- **Fortran**
- **Scalar codes**

- Before
  - IBM 7094 1963 – 1966
  - IBM 360/50 1966 – 1973
  - CDC 6600 1974 - 1982

The first disruption: vectorization

- **CRAY 1S 1982 - 1990**
  - 64 bits words
  - 1 proc
    - 80 MHz
  - 160 Mflops

Stability period

- CRAY XMP 1990 – 1993
  - 4 procs, 0.96 Gflops

- CRAY YMP 1990 – 1997
  - 8 procs, 2.7 Gflops

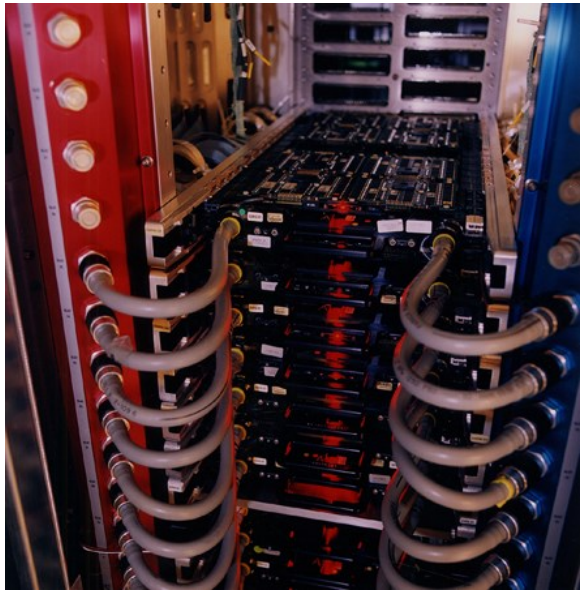
- CRAY T90 1996 – 2002
  - 24 procs, 1.8Gflops/proc, 454MHz
  - IEEE 754

- Main use : concurrent scalar jobs

Introduction of parallelism (R&D)

- CRAY T3D 1994 – 1997
  - 64 nodes, 128 procs (Alpha @ 150 MHz)
  - 2x64MB

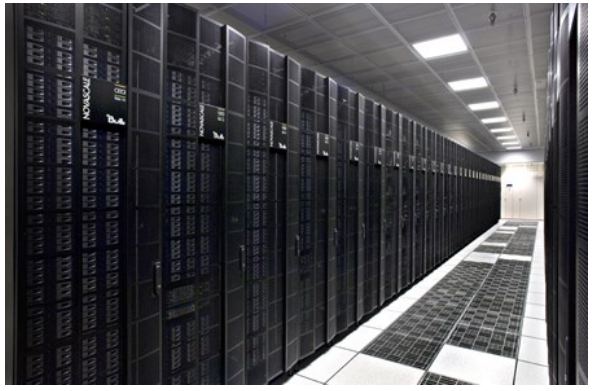
- CRAY T3E 1996 – 2001
  - 192 procs (Alpha)



- Mainly PVM, a bit of MPI

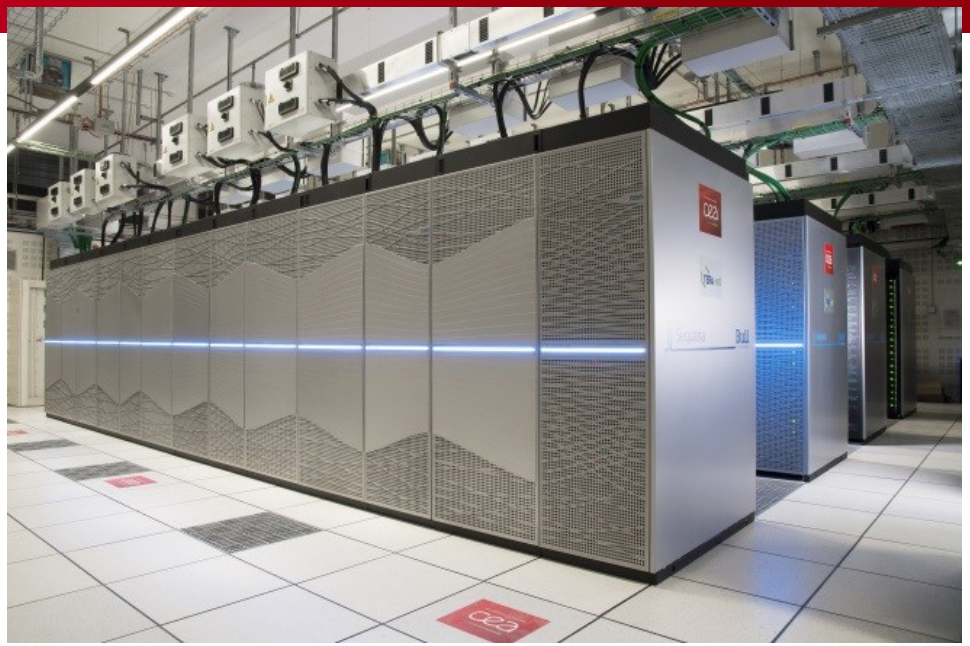
The second disruption: Cluster supercomputers

- TERA-1 2002 – 2006
  - 640 nodes, 4 x EV68
  - 5TFlops
- TERA- 10 2006 – 2011
  - 624 nodes, Intel Montecito @ 1.5GHz, 48GB/node
- TERA-100 2011 – 2018
  - 4370 nodes, Intel Xeon 7500
  - **5 MW**
  - HPL: 1.05Pflops
- MPI
  - Domain decomposition mainly



The third disruption: multi level parallelism

- TERA1000-2
  - 8004 nodes, KNL
  - **4MW**
  - HPL: 11.96 Pflops



- Strong impact on codes: 3 level parallelization
  - MPI across nodes
  - OpenMP across cores
  - Vectorization inside a core

- The dawn of the **fourth disruption**
  - **Energy Awareness**

Reasons to go Arm

Code portability is essential

- Our code are long lived
  - 20 – 30 years
    - = Several generations of supercomputers
  - Most are mission critical

- Tera 10 – 100 – 1000 are Intel based
  - Need an alternative architecture to **validate** codes
    - Different compilers
      - Standards interpretations
    - Different optimized libraries
      - Rounding influences
    - Different ISA
      - Difference in optimizations

Codes in the future

- Energy is becoming more and more important
  - Our focus is on exascale class machines for ~2022
  - Balanced between Energy to Solution and Time to Solution
    - Certain classes of codes are better suited to E2S
      - Older ones are more T2S
- Codes will have to adapt to this new constraint
  - Better dialog with the system
    - hints to SLURM, frequency regulation, …
  - New energy aware algorithms
    - Minimize data movements
    - Make the developers conscious of the resources used.

Arm Proof of Concept future partition

- Goal: study emerging high efficiency architectures
- Architecture based on the **Mont-Blanc3**(*) project results
- To be installed later in 2018

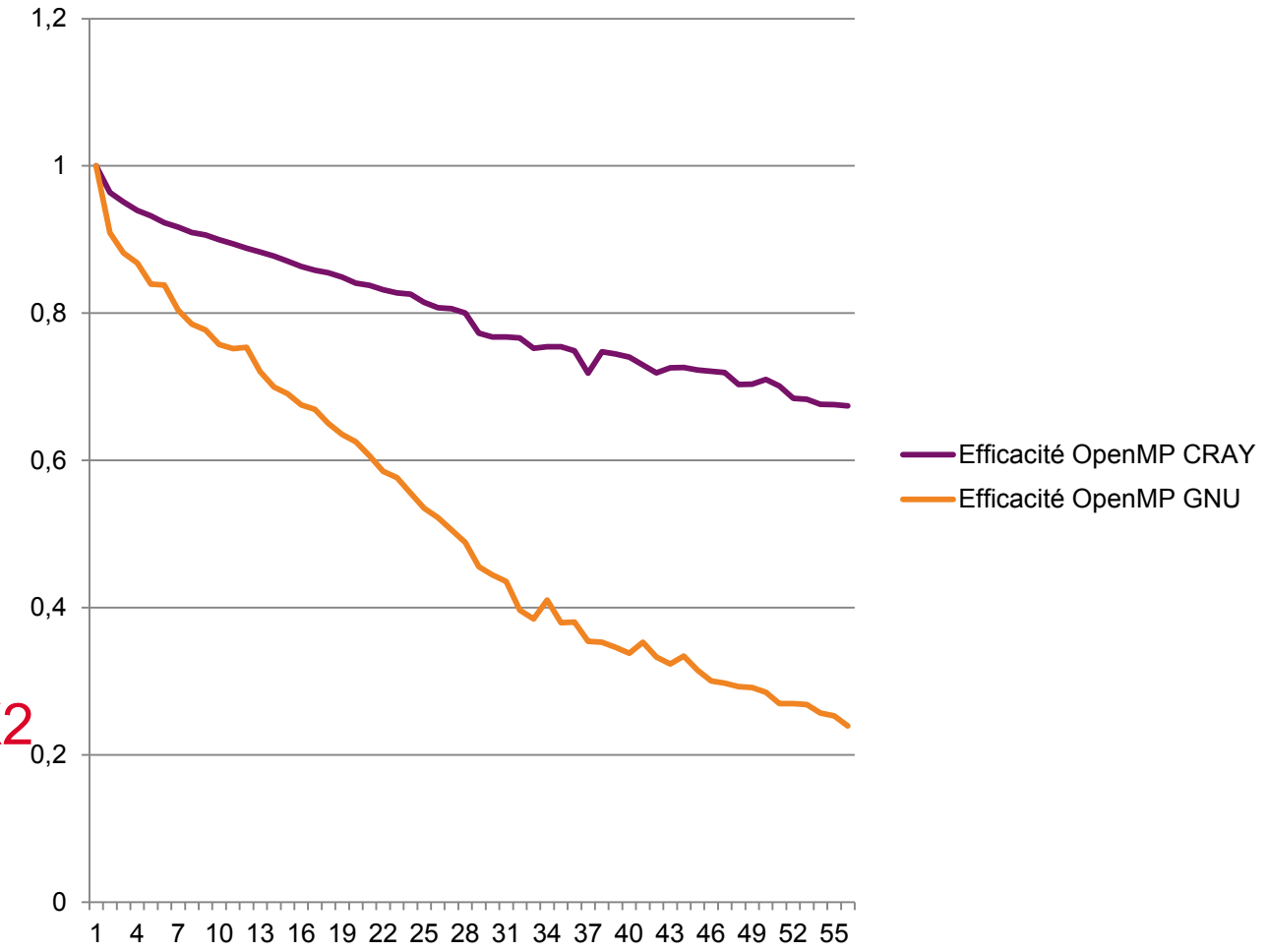
	Future partition
Node type	2* THX2 (30 cores @ 2.2 GHz -- tbc)
# compute nodes	160 (tbc)
Memory size	256 GB / node
I/O router	20 GB/s
Interconnect	EDR pruning 1:2
Cooling	DLC (**Sequana** infrastructure)

(*) This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement n° 671697

Challenges going Arm

The compiler and runtime matter

- Lulesh
  - V2.0.3, **OpenMP**
- GCC 7.3
- Cce/8.6.2
- Parallel efficiency
- Cavium ThunderX2
  - A2 stepping



Credit: JC Weill

Optimized libraries matter (1/2)

- A standard Arm version of Intel SVML
  - SVML is automatically used by Intel compiler

- https://developer.arm.com/products/software-development-tools/hpc/documentation/vector-math-routines
  - **-fsimdmath** for C/C++
    - Based on SLEEF Vectorized Math Library http://sleef.org/

What about optimized FTN?
is **flang** up to the task?

Optimized libraries matter (2/2)

- Intel MKL library is heavily used in production
  - Blas, Lapack, FFT, …
- In most cases hand coded algorithms don't beat MKL

- In the Arm world, optimized libraries start to exist
- https://developer.arm.com/products/software-development-tools/hpc/arm-performance-libraries
  - BLAS - Basic Linear Algebra Subprograms (including XBLAS, the extended precision BLAS).
  - LAPACK - a comprehensive package of higher level linear algebra routines.
  - FFT - a set of Fast Fourier Transform routines for real and complex data.
  - Math Routines - Optimized exp, pow and log routines

Multithreaded versions ?
TBB ?
Tasks in general ?

Prepare our code port: CEA-Arm/Allinea collaboration

- Longstanding collaboration between CEA and Allinea

- CEA has funded quite a lot DDT, MAP and Performance Report
  - Scalability (large number of cores, large number of libraries, KNL, …)
  - Robustness
  - Thread support (MPC, OpenMP)
  - C++ friendliness
  - Allinea Metric Plugin Interface compatibility with OpenSource profiling tools like MALP (http://malp.hpcframework.com)

- Collaboration extension to Arm (WiP)

- Idea: have the same developer experience on Arm and on X86
- New items for co-design
  - Compiler
    - MPC support in LLVM, linker optimization, …
  - Optimized scientific libraries
  - Profiling and debugging tools for Arm
    - MPI, perf counters, vectorization, …
    - Thread debugging
  - OS support (Work in Progress)

Conclusion

- Energy to Solution will gain importance
  - Arm based solutions are to be investigated seriously
- Arm based (super)computers are available now
  - Thanks to **MontBlanc3** (Atos) but also CRAY, HPE, …
- The software ecosystem is maturing fast
  - Compiler and libraries provided by Arm
  - Allinea tools
- It is time to start porting codes to Arm
  - And investigate new algorithms that take Energy into account

Commissariat à l'énergie atomique et aux énergies alternatives
Centre DAM Ile-de-France | 91297 Arpajon Cedex, France
T. +33 (0)1 69 26 40 00

DAM
DSSI
ED

Etablissement public à caractère industriel et commercial | RCS Paris B 775 685 019

GCdV