

Introduction to High Performance Computing

Piero Vicini (INFN Rome)

Perugia - June 6, 2017

- A (very) brief history of Supercomputing
- few key basic concepts: NUMA vs UMA parallel architecture, codes scaling, network,...
- Current technology survey and what is happening in HPC arena
- Next introduction ExaFlops HPC systems
- INFN activity in EU H2020 FET FP
 - ExaNeSt project: recap and status
 - Introduction to EuroExa project

High Performance Computing i.e. Supercomputer

From Wikipedia page (<https://en.wikipedia.org/wiki/Supercomputer>):

- A supercomputer is a computer with a high-level computational capacity compared to a general-purpose computer.
- Introduced in 1960 (Cray): from a few computing nodes to current *MPP* Massively Parallel Processors with 10^4 "off-the-shelf" nodes
- It's motivated by the search for solutions of *grand challenges* computing applications in many research fields (see PRACE scientific case for Europe HPC...)
 - quantum mechanics, weather forecasting, climate research, oil and gas exploration, molecular modeling, physical simulations...

High Performance Computing i.e. Supercomputer

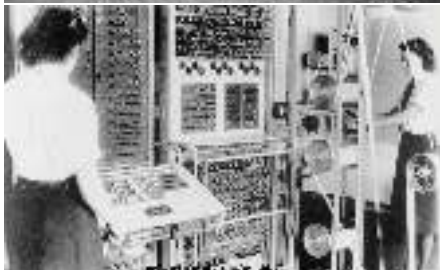
From Wikipedia page (<https://en.wikipedia.org/wiki/Supercomputer>):

- Performance of a supercomputer is measured in floating-point operations per second (FLOPS) instead of million instructions per second (MIPS).
 - T(era)Flops (10^{12}), P(eta)Flops (10^{15}), ExaFlops (10^{18}), Z(etta)Flops (10^{21})
 - Today $n * 10$ PFlops vs single workstation less than 1 TFlops...
- *Parallelism* is the key implemented with different approaches:
 - hundreds or thousands of discrete computers (e.g., laptops) distributed across a network (e.g., the Internet) devote some or all of their time to solving a common problem;
 - huge number of dedicated processors are placed in proximity and tightly connected to each other working in a coordinated way on a single task and saving time to move data around.

A very brief history of Supercomputing: the beginning...

Mainly motivated by military needs

- ENIAC (USA 1943), first stored-program electronic computer
- Colossus (UK 1943), successor of Bombe (designed by Alan Turing) and built in Bletchley Park to crack Enigma nazist codes.
- Supercomputers to design nuclear weapons



A very brief history of Supercomputing: 2nd generation

1964: Control Data Corporation releases **CDC 6600**, the first *supercomputer*



- Single CPU@40MHz, 1-3 MFlops, 4 racks Freon cooled, 10x faster than the powerful computer ever built (IBM 7030)...
- 8 M\$ cost (~ 60M\$ in today's money)
- designed by the *supercomputers guru* Seymour Cray...



1975-1990: *vector processors vs multiple scalar*

- The "One Million Dollars" question: few, big and fast, or many, small and slow?
- *"If you were plowing a field, which would you rather use? Two strong oxen or 1024 chickens?" (S.C.)*

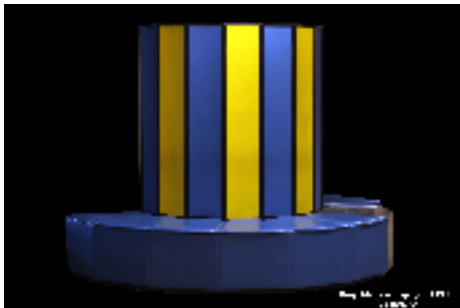
Cray-1

- vector processor
@80MHz, 133 MFlops
- 5-8 M\$ (25M\$ in today's money), 20-ton compressor for Freon cooling
- innovative shape for short wiring and fast clock



Cray X-MP, 1982

- 2 vector processors @105MHz, 400 aggregated MFlops
- memory shared



Cray Y-MP, 1988



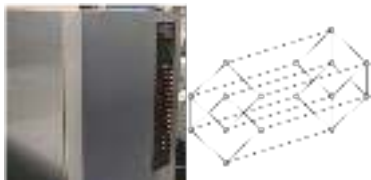
- 2, 4, or 8 vector processors with a peak of 333 MFlops each (-> 2.6GFlops!!!)
- memory shared
- dedicated OS: *UNICOS*
- followed by successful C90 series

A very brief history of Supercomputing: The MPP era...

1990 → *The attack of Killer Micros*, Eugene Brook's talk at Supercomputing90

- Massively Parallel Processor (**MPP**) era had begun, BUT vector processing do not allow to scaling to hundred processors systems
- from shared memory to *Distributed memory*, *Message passing* and "exotic" network

Intel iPSC Hypercube, 1985



- 32-128 nodes (286 + 287 coprocessor)
- 8 Eth ports per node → 5-Dim *hypercube* topology
- iPSC860 → Intel Paragon systems...

Connection Machine

CM-1 (1985), CM-5 (1991)

- CM-1: 65k SIMD processors arranged in Hypercube
- CM-5: MIMD, 1024 processors, 59.7 GFlops, 1st TOP500 leader...



A very brief history of Supercomputing: The MPP era...

INFN **APE** (*Array Processor Experiment*) is a 30 Years old project...

- Several generations of MPP systems (APE1, APE100, APEmille, apeNEXT)
- Custom FloatingPoint and 3D Torus interconnect optimized for LQCD application



APE1 (1988) 1GF, chipset Webtek



APEmille (1999) 128GF, 5P, Complex
Italy + France + Germany collaboration




apeNEXT (2004) 800GF, 1P, Complex



APE100 (1992) 25GF, 5P, REAL "Home made" VLSI processors

A very brief history of Supercomputing: The Cluster era...

- In 1994, Becker and Stirling built the first *PC Cluster*, with standard PCs and commodity network
 - Each PC has its own private memory space and, in principle, different OS
 - Low cost, scalable, leverages on CPU improvements, MP programming,...
- 
- From mid 2000 a *Cray inside my cellphone*: introduction of powerful *GPGPU* used as accelerators.
 - Most of current TOP500 ranking list are **Hybrid Supecomputers** based on clusters. Main differences due to:
 - CPUs (x86 multi-core, Power, custom)
 - FP accelerators (GPGPU, MIC, FPGA,...)
 - network technology (ethernet, Infiniband, Myrinet,...)
 - network topology (Fat-tree, Torus,...)

APEnet+



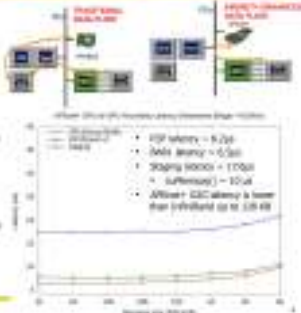
- APEnet+ Card:
 - 3D Torus: Spatially Close efficient
 - Active Switch (V FPGA (FPGA X8-Gen) board)
 - No Full host (Xilinx Z-400 Gbps QDR+)
- APEnet+ based on CNP:
 - Network Interface
 - 80Gbps (PCIe 3.0) / 40Gbps (PCIe 2.0) / 20Gbps (PCIe 1.1)
 - 80Gbps (PCIe 3.0) / 40Gbps (PCIe 2.0) / 20Gbps (PCIe 1.1)
 - 80Gbps (PCIe 3.0) / 40Gbps (PCIe 2.0) / 20Gbps (PCIe 1.1)
 - 80Gbps (PCIe 3.0) / 40Gbps (PCIe 2.0) / 20Gbps (PCIe 1.1)
 - 80Gbps (PCIe 3.0) / 40Gbps (PCIe 2.0) / 20Gbps (PCIe 1.1)
 - 80Gbps (PCIe 3.0) / 40Gbps (PCIe 2.0) / 20Gbps (PCIe 1.1)

GPUDirect RDMA in APEnet+

- Host-to-Host Mapping:**
- Data exchanged on the PCIe bus
 - Full source buffers at host

- APEnet+ RDMA support:**
- Existing edge RDMA technologies developed jointly with Nvidia
 - APEnet+ based around a peer
 - APEnet+ based on RDMA for existing GPU access

- Direct GPU access:**
- Specialized APEnet+ HW block
 - Latency saved for great data throughput



QUONG: GPU+3D Network FPGA-based

QUONG (QUantum chromodynamics ON Gpu) is a comprehensive initiative aiming to deploy an GPU-accelerated HPC hardware platform mainly devoted to theoretical physics computations.

- Heterogeneous cluster: PC mesh accelerated with high-end GPU (Nvidia) and interconnected via 3-D Torus network
- Added value:
 - tight integration between accelerators (GPU) and custom/reconfigurable network (DNP or FPGA) allows latency reduction and computing efficiency gain
 - Huge hardware resources in FPGA to integrate specific computing task accelerators
 - ASIP, OpenCL (in the future...)
- Communicating with optimized custom interconnect (APEX++), with a standard software stack (MPI, OpenMP, ...)
- Community of researchers sharing codes and expertise (LQCD, GWA, Laser-plasma interactions, BioComputing, Complex systems....)



HPC measure and compare: the TOP500 list

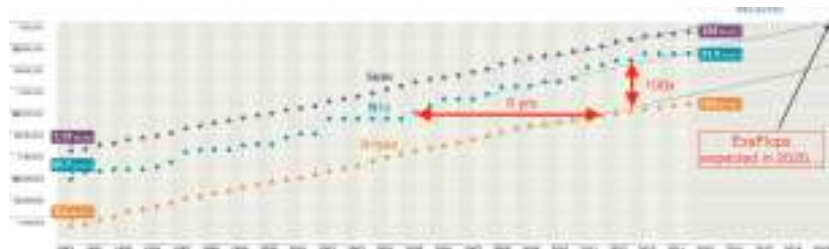
Web site: www.top500.org

- Based on a common benchmark: LinPack, a package for linear algebra
- www.top500.org/resources/posters – *and – materials/* for a bit of history (1993-...)

RANK	SITE	SYSTEM	CORES	FP32 TFLOPS	FP64 TFLOPS	POWER WATT
1	Microsoft Corporation Guangzhou China	Rank-1 (May/May-2) - 107 KOCPU Cluster Intel Xeon ES-2692 10C 2.70 THz / 10 Servers / 2.64 TFlops THERM MILK	315120	3066.7	648.74	11000
2	Intel/RTX/LLNL/IBM/STFC London United States	Mile - Intel Xeon E5-2697 12/27C 2.12 THz / 24 Servers / 2.6 TFlops NVIDIA 420x Trento	261344	1758.8	253.17	8278
3	CGGHI/ONLINE United States	Decade - Intel Xeon/Power-900 10C 1.08 THz / 24 Servers IBM	157664	1710.2	203.22	7478
4	Intel Advanced Hardware for Storage and Data (AHAD) Japan	Hi computer SPARC64 VII-X 2.00 THz / 76C Hi computer Fujitsu	76504	1070.8	1120.1	12560
5	CGGHI/Argonne National Laboratory Lanham PA/US	Mile - Intel Xeon E5-2697 12/27C 1.08 THz / 24 Servers IBM	76502	838.6	1050.0	8818
6	CGGHI/ONLINE/ILLINOIS Lanham PA/US	Triality - Intel Xeon E5-2697 12/27C 2.00 THz / 76 Servers / 76C Intel	76500	810.8	1179.1	

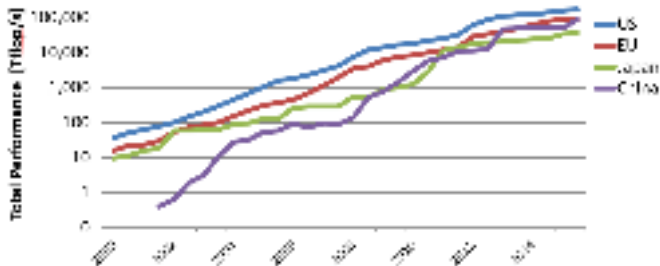


HPC measure and compare: the TOP500 list



PERFORMANCE OF COUNTRIES

TOP 500



TOP500 Web site: www.green500.org

The Green500 List

Direct links to the June 2014 The Green500 energy efficiency supercomputing list from the list:

Overall Rank	MFLOPS/W	Site*	Computer†	Total Power (MW)
1	4,036.36	U.S. LBNL (SDSC) Summit at Tennessee	CRAY XC30 2.1-2.15GHz Intel Xeon E5-2680 v2 2.15GHz InfiniBand QDR 100GbE 10000	2450
2	3,671.18	USNC/Argonne/LLNL	Cray XE6 E10000 2.66GHz Intel Xeon E5-2680 v2 2.66GHz InfiniBand QDR 100GbE 10000	2000
3	3,677.04	U.S. LBNL (SDSC) Summit at Tennessee, University of Utah	Cray XE6 E10000 2.66GHz Intel Xeon E5-2680 v2 2.66GHz InfiniBand QDR 100GbE 10000	2477
4	3,488.18	ORNL OLC	Cray XE6 E10000 2.66GHz Intel Xeon E5-2680 v2 2.66GHz InfiniBand QDR 100GbE 10000	2481
5	3,188.91	Centre National de Calcul (CNC) at Lille (France)	The Data Center for the High Performance Computing Center at the University of Lille	1,795.86
6	3,151.58	ORNL OLC	Cray XE6 E10000 2.66GHz Intel Xeon E5-2680 v2 2.66GHz InfiniBand QDR 100GbE 10000	2141
7	3,018.72	ORNL OLC	Cray XE6 E10000 2.66GHz Intel Xeon E5-2680 v2 2.66GHz InfiniBand QDR 100GbE 10000	2530
8	2,691.26	U.S. LBNL (SDSC) Summit at Tennessee	CRAY XC30 2.1-2.15GHz Intel Xeon E5-2680 v2 2.15GHz InfiniBand QDR 100GbE 10000	2200
9	2,672.14	University of California at Berkeley, U.C.A.	Cray XE6 E10000 2.66GHz Intel Xeon E5-2680 v2 2.66GHz InfiniBand QDR 100GbE 10000	1,284.49
10	2,672.11	University of Utah	Cray XE6 E10000 2.66GHz Intel Xeon E5-2680 v2 2.66GHz InfiniBand QDR 100GbE 10000	1781

TOP500: Tianhe-2

- 16000 nodes (2Xeon+3PHI);
- Peak: 54,9 PFlops, Sust: 33,8 PFlops; $\epsilon = 62\%$; Power: 17,8MW



TOP500: Titan

- 18000 nodes (1 Xeon+1 K20x);
- Peak: 27,1 PFlops, Sust: 17,6 PFlops; $\epsilon = 65\%$; Power: 8,2MW



Performance...

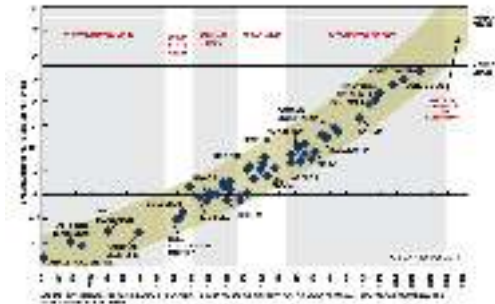
In the past, scaling to high(er) performances was an "easy" game...



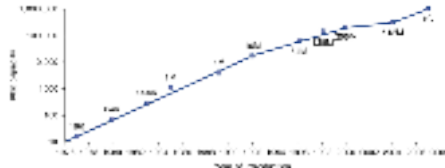
ENIAC 1943:
18000 tubes == 5000 transistors

Device	Year	Transistor Density (per cm²)
ENIAC	1946	~100
Intel 4004	1971	~2,300
Intel 8008	1972	~6,000
Intel 8080	1974	~29,000
Intel 8085	1976	~60,000
Intel 8088	1982	~275,000
Intel 80286	1985	~2.75 million
Intel 80386	1985	~2.75 million
Intel 80486	1989	~12 million
Intel Pentium	1992	~3.1 million
Intel Pentium Pro	1995	~5.5 million
Intel Pentium IV	2004	~291 million
Intel Core 2 Duo	2006	~291 million
Intel Core i7	2008	~731 million
Intel Core i9	2017	~62 billion

ENIAC vs current CPU



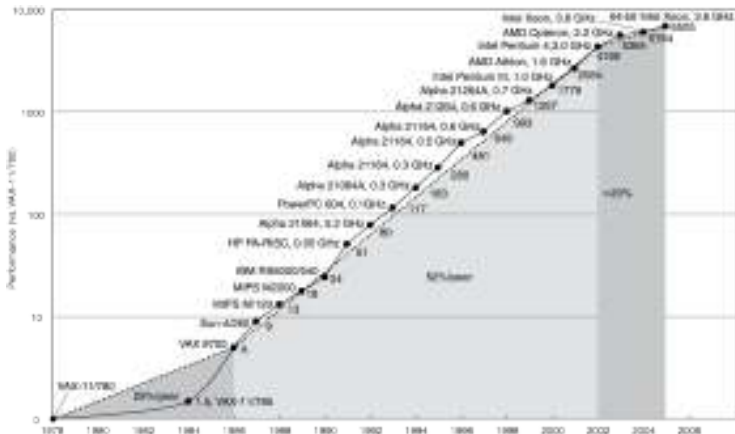
Moore's law: transistor density (i.e. computer performance) doubles every 18-24 (!) months



Memory density

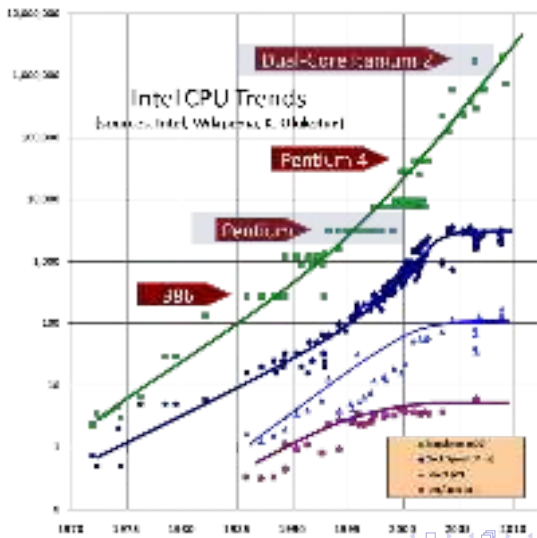
"Free lunch is over..."

- Once upon a time, and thanks to the Moore's law, performance scaled with the processor clock frequency...



"Free lunch is over..."

- From mid of 2000's it's no more true....



The Power Wall

- CMOS technology: current through junctions flows ONLY when (and during) transistor changes state

$$P = C \times V^2 \times (\alpha f)$$

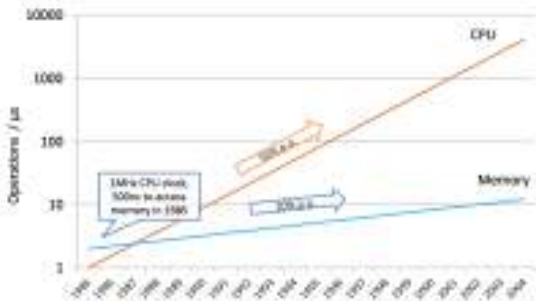
C = capacitance, V = voltage, f the switching frequency and α the fraction of gates switching per unit of time

- It exists a technological limit to surface power density. As a consequence:
 - processor clock frequency can not scale up freely...
 - supply voltage can not decrease too much (impact of leakage and errors due to fluctuations...)



The Memory Wall

- The GAP between CPU performance and memory devices bandwidth is growing
- The majority of application workloads are *memory limited* so the memory access is a real bottleneck



ILP Wall

- Instruction-level parallelism (ILP) is a measure of how many of the operations in a computer program can be performed simultaneously.
- The implicit parallelism in a single computing thread of a processor is quite limited.
 - Try to reorder instructions, reduce to sequence of micro-instructions, aggressive branch prediction but...
 - ... you can't feed the computing units if you are waiting for data memory
 - Additionally, adding functional units to exploit ILP parallelism increase HW complexity → increase the power dissipation (Power Wall)

Power wall + ILP wall + Memory wall → Serial hardware Game over...

- Use concurrency as much as you can → parallel architectures:
multiprocessors, multi-core, many-core
 - multi/many computing cores with "low" clock frequency
 - many multi/many cores processors interconnected by efficient networks
 - new programming model able to cope with parallel systems and able to distribute the workload in parallel
- Warning: effective parallel programming (performance next to the theoretical peak) is a BIG issue...
(luckily not fully covered in this talk ;-)

Amdahl's law gives the theoretical *speedup* of the execution of a task at fixed workload that can be expected of a system whose resources are improved.

- If P is the fraction of a computer program that can be parallelized on N computing nodes ($1 - P$ is the non-parallelizable part), the execution time, $T(N)$, is:

$$T(N) = T(1)\left(\frac{P}{N} + (1 - P)\right)$$

- so the **speedup** is $S(N) = \frac{T(1)}{T(N)}$ is equal to

$$S(N) = \frac{1}{(1-P) + \frac{P}{N}}$$

Amdahl's law gives the theoretical *speedup* of the execution of a task at fixed workload that can be expected of a system whose resources are improved.

- If P is the fraction of a computer program that can be parallelized on N computing nodes ($1 - P$ is the non-parallelizable part), the execution time, $T(N)$, is:

$$T(N) = T(1)\left(\frac{P}{N} + (1 - P)\right)$$

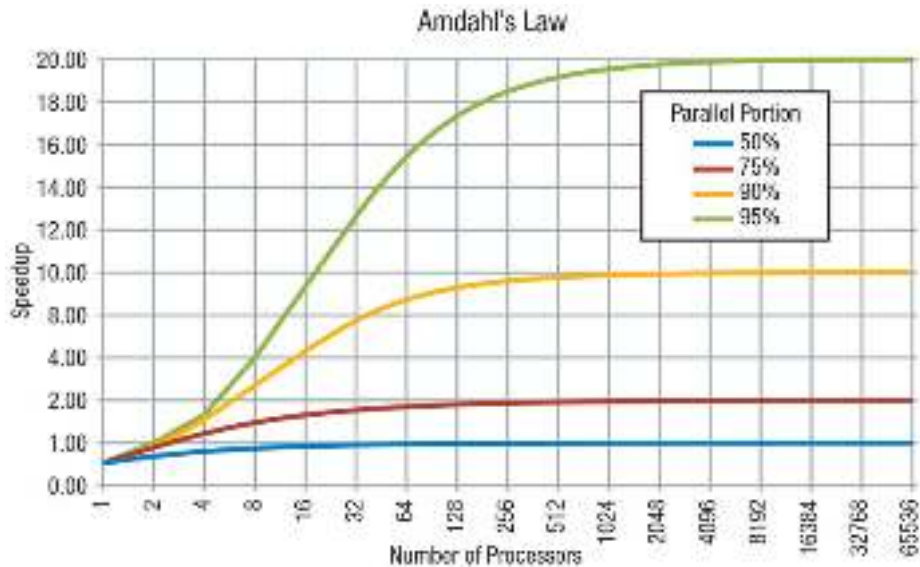
- so the **speedup** is $S(N) = \frac{T(1)}{T(N)}$ is equal to

$$S(N) = \frac{1}{(1-P) + \frac{P}{N}}$$

- Es: compute P to get 90% of speedup if the number of processing units of the computing system increases from 1 to 100.

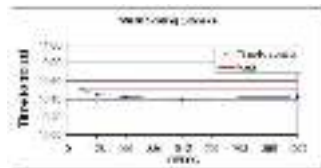
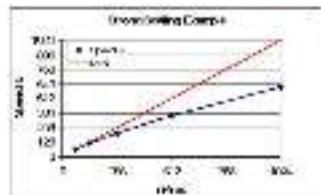
$$90 = \frac{1}{(1-P) + \frac{P}{100}} \rightarrow P = 0.999$$

- The sequential part (not-parallelizable) of program is to be less than **0.1%** (!!!)



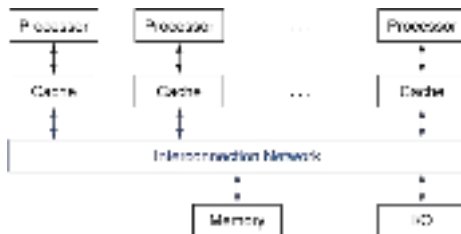
System scalability: strong e weak scaling

- **Scalability** Scalability of a system respect to an application measures how well latency and bandwidth scale with the addition of more processors.
- **Strong scaling:** given a fixed size computing problem the *strong scaling* represents its time to solution as a function of (increasing) number of execution processors.
- **Weak scaling:** *weak scaling* is the time to solution of a fixed size *per processor* problem varying the number of processors.
- System scalability is affected from **load balancing**
 - If the average computational load of a single processor is 2x, speedup may be halved...



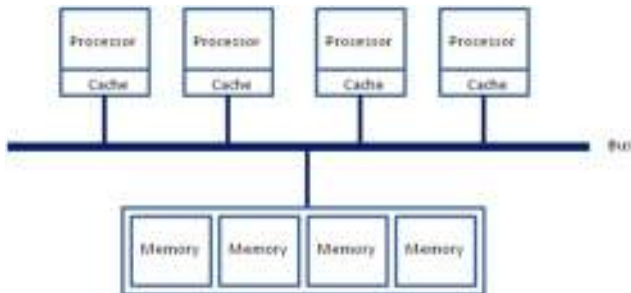
Parallel architecture: Shared Memory Multiprocessors (SMP)

- Parallel hardware architecture: all processors share *a single memory space*
- Parallel execution (coordination) and data exchange through *shared variables* located in shared memory.
 - Synchronisation primitives (*locks, barriers*) to handle memory access contention.



- Two different types of memory access:
- **UMA**: Uniform Memory Access vs **NUMA** Non-Uniform Memory Access

- **UMA**: i.e. *Symmetric Multi-Processing* architecture
- Any processor (core) can access any memory location with the same access time / latency (!!!)



- SMP is effective and easy to program but scaling is limited to few processors... (Multi-Core)
- It's really complex (almost unfeasible?) to ensure "uniform access" when hundreds of CPU compete to access data in memory...

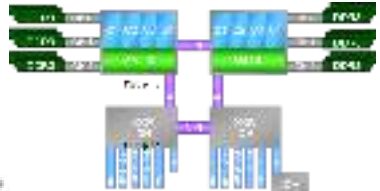
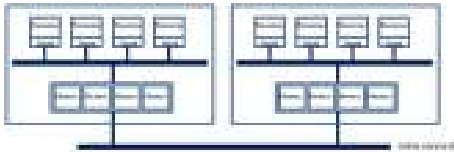
UMA vs NUMA

- **NUMA**

- Every processor (core) owns its (private) *local* memory and can access in parallel *remote* memory of others processors (cores), using high performance (hopefully, low latency) interconnection network.

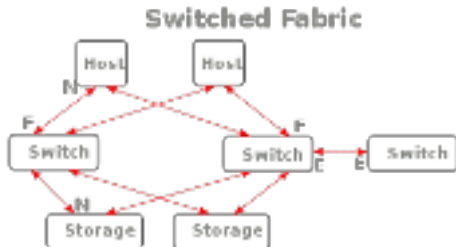
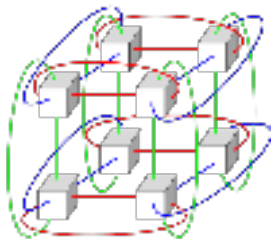


- Programming NUMA may be more difficult than programming UMA
- Very good scaling for "hybrid" architectures like NUMA SMP processors
 - New generation multi-proc of multi-core, AMD Hypertransport, INTEL QPI,...



Network Topology is how the processors are connected (Direct point-to-point and switched).

- 2D or 3D *Torus mesh*: is simple and ideal for programs with mostly nearest-neighbour communication.
- *Hypercube*: minimizes number of "hops" between processors, many wires/channels
- *Switched network*: all processors connected through hierarchical layers of high-speed switches. Overhead but quite fast especially for limited number of computing nodes



- Performance
 - Latency per message (unloaded network)
 - Throughput
 - Link bandwidth
 - Total network bandwidth
 - Bisection bandwidth
 - Congestion delays (depending on traffic)
- Cost
- Power
- Routability in silicon

- **OpenMP** (Open Multi Processing)
 - specification for a set *compiler directives, library routines, and environment variables* (compiler can skip them...)
 - target is shared memory → multi-core processors
 - support for Fortran and C/C++ programs.
- A simple example of OpenMP use...

```
#include <stdio.h>
#include <omp.h>
main(){
    int id;
    #pragma omp parallel
    {
        id = omp_get_thread_num();
        printf("Hello from process%d!\n", id);
    }
}
```

- ... and its output

```
Hello from process 0
Hello from process 1
Hello from process 2
Hello from process 3
```

- **MPI** (Message Passing Interface): an API for writing clustered applications
 - a library of "calls" to coordinate execution of multiple processes (optionally on multiple nodes)
 - provides *point-to-point* and *collective* communications in Fortran, C e C++
 - Multiple implementations (OpenMPI, MPICH, MVAPICH;...) leverage on 25 years of cluster computing and MPP practice
- MPI applications use the most common computing path in parallel programming:
 - Run the same program on P processing elements where P can be arbitrarily large.
 - Use the rank (an ID ranging from 0 to (P-1)) to select between a set of tasks and to manage any shared data structures.
- but needs explicit data movement coding...

```
int MPI_Send(void* buf, int count, MPI_Datatype datatype,
            int dest, int tag, MPI_Comm comm)
```

```
int MPI_Recv(void* buf, int count, MPI_Datatype datatype,
            int source, int tag, MPI_Comm comm,
            MPI_Status* status)
```

It's easy and straightforward...

```
#include <stdio.h>
#include <mpi.h>

int main (int argc, char **argv)
{
    int rank, size;

    MPI_Init (&argc, &argv); /* Init and build MPI. Required...*/
    MPI_Comm_rank (MPI_COMM_WORLD, &rank); /* Every process has
its own rank */
    MPI_Comm_size (MPI_COMM_WORLD, &size); /* Amount of total
processes in the job */

    printf( "Hello from process %d of %d\n", rank, size);

    MPI_Finalize(); /* Free MPI. Required... */
    return(0);
}
```

Parallel programming at a glance: MPI version of "parallel hello"

A bit more complicated: explicit data communications and src/dest selection..

```
#include <stdio.h>
#include <mpi.h>

int main (int argc, char **argv)
{
    int rank, size;
    MPI_Init (&argc, &argv);
    MPI_Comm_rank (MPI_COMM_WORLD, &rank);
    MPI_Comm_size (MPI_COMM_WORLD, &size);
    if (rank != 0){
        sprintf(message, "Hello from process %d!", rank);
        dest = 0;
        MPI_Send(message, strlen(message)+1, MPI_CHAR, dest,
                 tag, MPI_COMM_WORLD);
    }
    else { /* my_rank = 0 */
        for (source = 1; source < size; source++){
            MPI_Recv(message, 100, MPI_CHAR, source, tag,
                    MPI_COMM_WORLD, &status);
            printf("%s/n", message);
        }
    }
    MPI_Finalize();
    return (0);
}
```

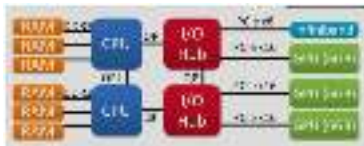
MPI	OpenMP
Same standard(s) available for different vendors and platforms	Specific implementation for different compilers
Targets both distributed as well as shared memory system	Targets shared memory systems only
Support for process and thread based parallelism (i.e. good for networked systems)	Only thread based parallelism (i.e. good for multi-core systems)
Messages based	Directives based
Overheads associated with transferring message from one process to another	No overheads, as thread can share variables
Flexible and expressive	Easier to program and debug

Issues for scalability

- *Access latency* to remote data memory.
 - MP introduces time overhead typically 10-100 μ S equivalent of $10^5 - 10^6$ FP operations...
- Communication *bandwidth* has to be large enough for feeding processors
 - first order evaluation: 3 64 bit words per operation \rightarrow
 $10^{11} \text{ Fops} * 24 \text{ Bytes} = 2 \text{ TB/s}$
 - to be multiplied for the number of nodes.....
 - many technics (algorithmic and technological) to mitigate the effects but not enough
- porting of sequential applications may be not easy since every data movement is explicit and source/target has to be identified

Hybrid Supercomputer: CPU + Accelerators

Most high-end HPC systems are characterized by *hybrid architecture*



- ASIP, FPGA or commodity components (GPGPU...)
- Better \$/PeakFlops: offload cpu task to accelerator able to perform faster
- May consume less energy and may be better at streaming data.
- —> warning!!!:
 - computing efficiency ϵ (Sustained/Peak) not impressive
 - it's a function of accelerator and network...

	Model	Year	Memory (Peak)	Topology (Ops)	Peak Perf. (Flops)	Impact/Perf. (Flops)	Efficiency	Power (MW)	Infrastructure
Titan-2	Dallas	1	16000 (16TB+3TB)	Hybrid (798)	54.5	33.5	50%	17.8	OpenStack
Titan	ORNL (Oak Ridge)	2	10000 (10PB+10TB)	Commodity (9000)	29.1	17.2	20%	6.2	OpenStack
FujiE10	OpenStack	10	8.2 (8.2PB+10TB)	Commodity (600)	2.4	1.0	40%	2.4	OpenStack
Summit	USA (ORNL)	7	5400	Hybrid (798)	11.5	5.1	90%	4.5	OpenStack

Accelerators: GPU

- Heterogeneous CPU/GPU systems: CPU for sequential code, (GP)GPU for parallel code
- Impressive use of state-of-the-art technologies
 - Example NVidia Tesla: 3D stacked mem, Proprietary GPU-GPU interconnect (NVLink), multi (10) TFlops/Proc, power effective...
- Processing is highly data-parallel (i.e. good for data parallel applications)
 - GPUs are SIMD-like and highly multithreaded: many parallel threads (up to 10^3 ...) distributed on many cores ($10^2 - 10^3$)
 - Graphics memory is wide ($N * 10^2$ bits) and high bandwidth ($N * Ghz$ per bit line).
- Programming languages standard (DirectX, OpenGL, OpenCL) or proprietary (NVidia Compute Unified Device Architecture (CUDA))



Intel Xeon
+ many caches - few processing



Nvidia Fermi GPU
more cores

Accelerators: GPU

NVIDIA Pascal P100 and the last generation Volta V100 (1.5x) recently announced...

TESLA P100 GPU: GP100

- Details
- GPU Architecture
- GPU Architecture
- GPU Architecture
- GPU Architecture
- GPU Architecture
- GPU Architecture



Model	Year	Process	Transistors	Memory	Power
V100	2017	16nm	21.1B	32GB	300W
P100	2016	16nm	10.3B	16GB	250W
V100	2017	16nm	21.1B	32GB	300W
P100	2016	16nm	10.3B	16GB	250W
V100	2017	16nm	21.1B	32GB	300W
P100	2016	16nm	10.3B	16GB	250W
V100	2017	16nm	21.1B	32GB	300W
P100	2016	16nm	10.3B	16GB	250W
V100	2017	16nm	21.1B	32GB	300W
P100	2016	16nm	10.3B	16GB	250W
V100	2017	16nm	21.1B	32GB	300W
P100	2016	16nm	10.3B	16GB	250W
V100	2017	16nm	21.1B	32GB	300W
P100	2016	16nm	10.3B	16GB	250W

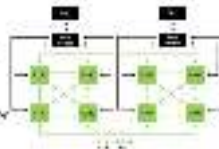
HBM2 : 720GB/SEC BANDWIDTH

And 200 to 400x



NVLINK - GPU CLUSTER

- Highly scalable and efficient
- 100% utilization of GPU resources
- Low latency and high bandwidth
- Low power consumption
- High performance and reliability
- Supports up to 64 GPUs



Knights Landing Overview

Tile		
1 Core	1 DRAM	1 GPU
Cache	Local L3	Cache

Chip 36 Tiles interconnected by 3D Mesh
Tile: 2 Cores + 2 DRAMs + 1 GPU

Highway HCDRAM: 16 GB on-package High BW
DRAM: 8 channels @ 1000 up to 18GB

ID: 30 lanes PCIe Gen 3, 3 lanes of DDI for display

Nodes: 1-socket only
Fabric: Direct-Path on-package (not shared)

Vector Pass Part: 2x TF DP and 4x TF DP
Scaler Part: 2x over Knights Corner
Streams Total (88%) HCDRAM: 400x, DDI: 100x

Knights Landing Products

Knights Landing Products

Knights Landing Products

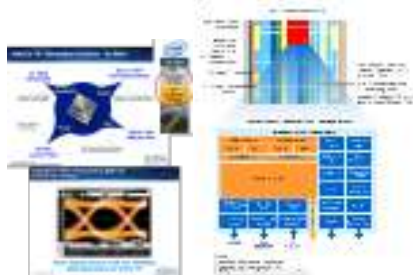
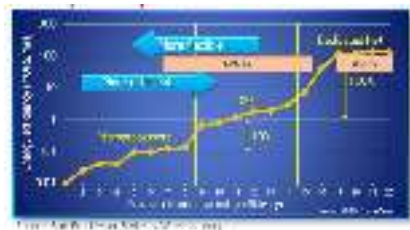
3 Knights Landing Products

3 Knights Landing Products

Knights Landing Products

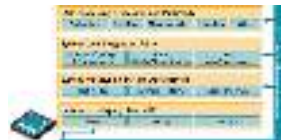
An emerging new player in hybrid HPC: FPGA

- Stratix10 high-end, introduction 2016
- INTEL TriGate 14nm -> 30% less than old generation power consumption
- 96 transceivers @32Gbps (56Gbps?) for chip-to-chip interconnection and @28Gbps for backplane/cable interconnection
- Many industrial standards supported included CAUI-x (Nvlink)
- tons of programmable logic @1GHz
- ...and "for free"
 - 10 Tflops of DSP single precision FP
 - HMC (3D mem, high bandwidth) support
 - Multiple (4->8) ARM Cores (a53/57) @1.5GHz
- Similar in performance: XILINX Zynq UltraScale+ MPSoC Devices



- **ARM** is the only "European" CPUs maker
- Innovative business model: ARM sell Intellectual Properties hw/sw instead of physical chip;
 - 1100 licenses signed with over 300 companies and royalties received on all ARM-based chips
 - Pervasive technology: Android and Apple phones and tablets, RaspberryPI, Arduino, set-top box and multimedia, ARM-based uP in FPGA, ...
 - From 1990, *60 billion* of ARM-based chips delivered

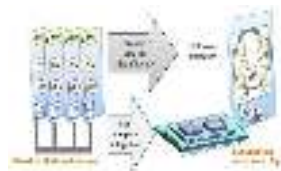
ARM



- Architecture specialised for embedded/mobile processors:
 - low power, low silicon area occupation, real-time, scalable, energy-efficient
- few generations of high end (64 bits) processors delivered
 - current **Cortex Axx ARM V8-A** enabling multi-core ARM-based processors
 - complete IP portfolio

Several attempts to use ARM low power processors in high end computing

- Server and micro-server ARM-based
 - AMCC X-gene 3, 32 v8-A cores@3GHz,
 - CAVIUM ThunderX SoCs up to 48 v8-A cores@2.4GHz
 - Broadcom/Qualcomm multi-core, Samsung SoC Exynos
- EU-funded projects
 - Mont-blanc project (BSC)
 - UniServer
 -
- INFN COSA project measured energy efficiency of low power architecture ARM based for scientific computing (Astrophysics, Brain simulation, Lattice-Boltzmann fluid-dynamics,..). On average:
 - $\sim 3x$ ratio x86 core / ARM core performances
 - but $\sim 10x$ ratio x86 core / ARM power consumption
 - \rightarrow **ARM architectures 3x less energy to solution for scientific applications**



The needs for ExaScale systems in science



- HPC is mandatory to compare observations with theoretical models
- HPC infrastructure is the theoretical laboratory to test the physical processes.
- HPC for Big Data...

Let's talk of Basic Science...

- High Energy & Nuclear Physics
 - LQCD (again...), Dark-energy and dark matter, Fission/Fusion reactions (ITER)
- Facility and experiments design
 - Effective design of accelerators (also for Medical Physics, GEANT...)
 - Astrophysics: SKA, CTA
 - ...
- Life science
 - Personal medicine: individual or genomic medicine
 - Brain Simulation ← HBP (Human Brain Project) flagship project

Just to name a few....

- Power efficiency and compute density
 - huge number of nodes but limited data center power and space
- Memory and Network technology
 - memory hierarchies: move data faster and closer...
 - increase memory size per node with high bandwidth and ultra-low latency
 - distribute data across the whole system node set but access them with minimal latency...
- Reliability and resiliency
 - solutions for decreased reliability (extreme number of state -of-the-art components) and a new model for resiliency
- Software and programming model
 - New programming model (and tools) needed for hierarchical approach to parallelism (intra-node, inter-node, intra-rack....)
 - system management, OS not yet ready for ExaScale...
- Effective system design methods
 - CO-DESIGN: a set of a hierarchical performance models and simulators as well as commitment from apps, software and architecture communities

- General agreement on the fact that data center power budget is less than 20 MW
 - half for cooling -> only 10MW for active electronics
- Current processors performances are
 - multi-core CPU: 1 TFlops/100W
 - GPGPU: 5-10 TFlops/300W but worst sustained/peak (and needs CPU) so only a factor 1.5 better
 - add few tens of watt for distributed storage and memory per node
- ExaScale sustained (where $\epsilon = 50\% - 70\%$)
 - 10^6 computing nodes
 - **100 MW** of power -> *low power* approach is needed

- Current computing node assembly:

- 8 processors into 1U box
- ~30 1Uboxes per 42U rack (25% of volume dedicated to rack services)

- Summing up

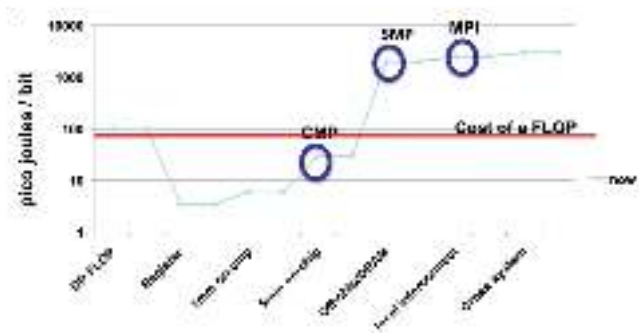
- 4000 racks per ExaFlops sustained
- 6000 m^2 of floor space
- service racks (storage, network infrastructure, power&controls, chillers,...) not included (!!)



- It needs:

- New mechanics for denser systems
- New cooling technology (liquid/gas cooling) for reduce impact of cooling system on power consumption and data center space

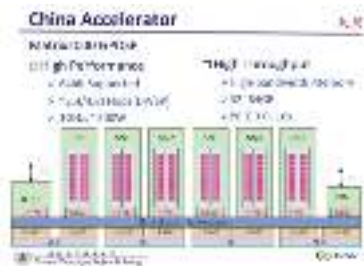
Big numbers, big problems: data locality



- Needed improved hierarchical architectures for memory and storage
 - distributed hierarchical memory
 - zero-copy through R(emote)DMA, P(artitioned)G(lobal)A(ddress)S(pace) leveraging on affinity to exploit data locality
- low latency, high bandwidth network

Next (almost) ExaScale systems around the World

- US **CORAL** (Collaboration of Oak Ridge, Argonne, and Livermore) project, 525+M\$ from DOE, for 3 100-200 PetaFlops systems in 2018-19 (Pre-Exascale system), ExaScale in 2023
 - *Summit/Sierra* OpenPower-based (IBM P9 + NVidia GPU + Mellanox) 150(300) PFLops/10MW
 - *Aurora* Intel-based (CRAY/INTEL, Xeon PHI Knights Hill, Omnipath) 180(400) PFLops/13MW
- JAPAN **FLAGSHIP2020** RIKEN + Fujitsu
 - derived from Fujitsu K-computer, SPARC64-based + Tofu interconnect, delivered in 2020
- CHINA **???** , NUDT + Government
 - ShenWei and FeiTang CPUs plus proprietary GPU and network... delivered in 2020





European Commission President
Jean-Claude Juncker



"Our ambition is for Europe to become one of the top 3 world leaders in high-performance computing by 2020"

French-German Conference on Digital
Paris, 27 October 2015

—> EuroHPC: 7 countries agreement on pushing HPC development in Europe (Digital Day, March 2017)

What next in Europe?

HPC Objectives (1)

- **Acquisition** (in 2020-2021) of 2 operational **pre-exascale** and (in 2022-2023) two full **exascale** machines (of which one based on European technology)
- **Interconnection and federation** of national and European HPC resources and creation of an HPC and Big Data service infrastructure facility
- **Demonstrating and testing** technology performance towards exascale through scientific & industrial compute-intensive applications

HPC Objectives (2)

Build a world-class European High Performance Computing (HPC), Big Data and Cloud Ecosystem

Enabled by the Convergence of 3 Big technologies



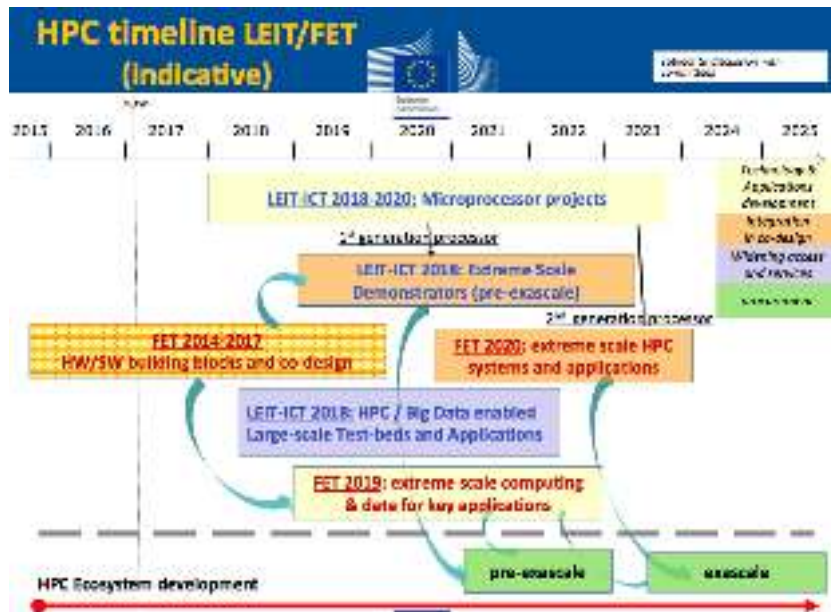
- Major investments so far both at MS and EU level (FP7, Horizon)
- Numerous research players (academic and industry)
- HPC and Big Data PPPs, PRACE, GCRIT, etc.

HPC/EDI – Funding needs (COM(2012) 178 of 16/4/2012)

- **1.5 BE** for 2 pre-exascale and 2 exascale machines
- **1.7 BE** for the interconnection and federation of supercomputing infrastructures
- **0.5 BE** for processor and for wider access to HPC facilities for SMEs
- **1.0-1.5 BE** for demo and testing of industrial applications

- Total: 4.7 - 5.2 BEuro needed....
- mainly from National and Regional funds...
- 1.5 BEuro for systems procurement
- 0.15 BEuro for European Processor NRE

What next in Europe?





ExaNeSt: European Exascale System Interconnection Network & Storage

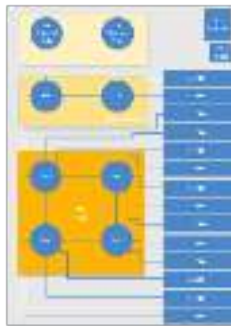
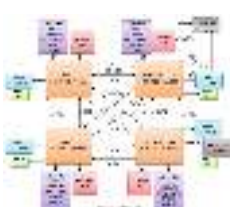
- EU Funded project H2020-FETHPC-1-2014
- Duration: 3 years (2016-2018). Overall budget about 7 MEuro.
- Coordination FORTH (Foundation for Research & Technology, GR)
- 12 Partners in Europe (6 industrial partners)

"...Overall long-term strategy is to develop a European low-power high-performance Exascale infrastructure based on ARM-based micro servers..."

- System architecture for datacentric Exascale-class HPC
 - Fast, distributed in-node non-volatile-memory
 - Storage Low-latency unified Interconnect (compute & storage traffic)
 - RDMA + PGAS to reduce overhead
- Extreme compute-power density
 - Advanced totally-liquid cooling technology
 - Scalable packaging for ARM-based (v8, 64-bit) microserver
- Real scientific and data-center applications
 - Applications used to identify system requirements
 - Tuned versions will evaluate our solutions

- **EuroServer**: Green Computing Node for European microservers
 - UNIMEM PGAS model among ARM computing nodes
- INFN **EURETILE** project: *brain inspired* systems and applications
 - APEnet+ network on FPGA + brain simulation (DPSNN) scalable application
- **Kaleao**: Energy-efficient uServers for Scalable Cloud Datacenters
 - startup company interested in commercialisation of results
- **Twin** projects: **ExaNode** and **EcoScale**
 - ExaNode: ARM-based Chipllets on silicon Interposer design
 - EcoScale: efficient programming of heterogenous infrastructure (ARM + FPGA accelerators)





- Computing module based on Xilinx Zynq UltraScale+ FPGA...
 - Quad-core 64-bit ARM A53
 - ~1 TFLOPS of DSP logic
- ... placed on small Daughter Board (QFDB) with
 - 4 FPGAs, 64 GB DDR4,
 - 0.5-1 TB SSD,
 - 10x 16Gb/s serial links-based I/O per QFDB
- mezzanine(blade) to host 8 (16 in second phase) QFDBs
 - intra-mezzanine QFDB-QFDB direct network
 - lots of connectors to explore topologies for inter-mezzanine network

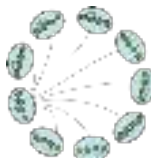
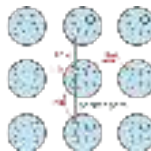
- ExaNeSt high density innovative mechanics...
 - 8(16) QFDBs per mezzanine
 - 9 blades per chassis
 - 8-12 chassis per rack
- ...totally liquid cooling
 - track 1: immersed liquid cooled systems based on convection flow
 - track 2: phase-change (boiling liquid) and convection flow cooling (up to 350 kW of power dissipation capability...)



- $\sim 7PFlops$ per racks and $20GFlops/W$
- Extrapolating from current technology, ExaNeSt-based Exascale system with 140 racks, $21M$ ARM cores and $50MW$

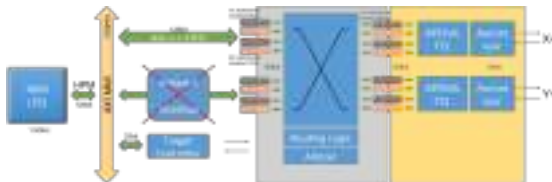
ExaNeSt is working testbed FPGA-based to explore and evaluate innovative network architectures, network topologies and related high performance technologies.

- **Unified** approach
 - merge interprocessor and storage traffic on same network medium
 - PGAS architecture and RDMA mechanisms to reduce communication overhead
- innovative routing functions and control flow (congestion managements)
- explore performances of **different topologies**
 - Direct blade-to-blade networks (Torus, Dragonfly,...)
 - Indirect blade-switch-blade networks
- **All-optical switch** for rack-to-rack interconnect (ToR switch)
- Support for **resiliency**: error/detect correct, multipath routing,...
- Scalable network **simulator** to test large scale effects in topologies

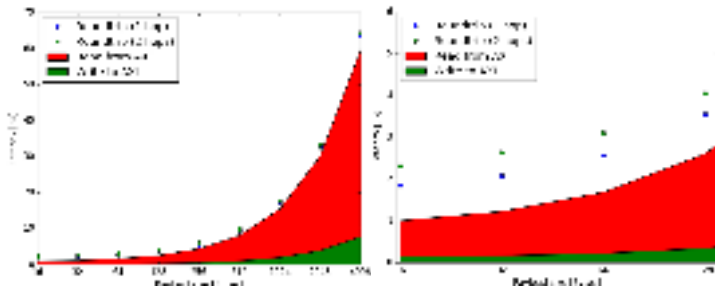


ExaNeSt highlights: network userspace results

First sketch of test (user space) writes commands/data to the hardware

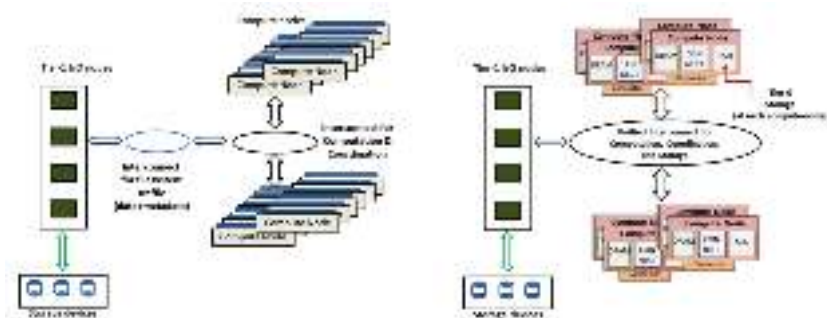


- single and dual hops test; no DMA, no interrupts, no system-wide locking and no fast virtual-to-physical address translation: sub- μ S latency



Co-design approach

- Applications **define** quantitative requirements for the system under design
- Applications **evaluate** the hw/sw system
- Applications list:
 - Cosmological n-Body and hydrodynamical code(s) (INAF)
 - Large-scale, high-resolution numerical simulations of cosmic structures formation and evolution
 - **Brain Simulation: DPSNN** (INFN)
 - Large scale spiking behaviours and synaptic connectivity exhibiting optimal scaling with the number of hardware processing nodes (INFN).
 - Mainly multicast communications (all-to-all, all-to-many).
 - Weather and climate simulation (ExactLab)
 - Material science simulations (ExactLab and EngineSoft)
 - Workloads for database management on the platform and initial assessment against competing approaches in the market (MonetDB)
 - Virtualization Systems (Virtual Open systems)



- **Distributed storage:** NVM close to the computing node to get low access latency and low power access to data
- based on **BeeGFS** open source parallel filesystem with caching and replication extensions
- Unified interconnect infrastructure per storage and inter-node data communication
- Highly optimized I/O path in the Linux kernel

Project Full Title: Co-designed Innovation and System for Resilient Cloud Computing in Europe: From Applications to Silicon

Acronym: EuroExa

Web Page: www.fethpc.eu

Trade Name: Research and Innovation Action (RIA)

Name Of Coordinator: Prof. Dr. Giovanni Costantini



EuroEXA

Resilient Eascale Computing in Europe:
From Applications to Silicon

Lead Or Participants

Part. No.	Participant Organization name	Start Month	Country
1	Institute of Communications and Computer Systems	2015	GR
2	University of Manchester	2014/05	GB
3	Research Supervising Center	2015	ES
4	Frankfurt Institute for Advanced Technology (FIAT)	2014	DE
5	Politecnico di Milano	2015	IT
6	University of Applied Sciences Technikum Wien	2015	AT
7	Intel	2015	US
8	University of Cambridge	2015	GB
9	Altran	2015	FR
10	Systemic System Architecture and Deployment Monoproprietor	2015	IT
11	Microsoft	2015	US
12	Intel	2015	US
13	Intel	2015	US
14	Intel	2015	US
15	Intel	2015	US
16	Intel	2015	US
17	Intel	2015	US
18	Intel	2015	US
19	Intel	2015	US
20	Intel	2015	US

... EuroEXA brings a *holistic foundation* from multiple European HPC projects and partners together with the industrial SME (MAXeler for FPGA data-flow; ICEotope for infrastructure; ARM for HPC tooling and ZPT to collapse the memory bottleneck)...

-> Computing platform as a whole thanks to consortium based on SME and key European academic partners

... co- design a ground-breaking platform capable of scaling peak performance to *400 PFlops* in a peak system power envelope of *30MW*
... we target a PUE parity rating of 1.0 through use of *renewables and immersion-based cooling*... modular-integration approach, novel *inter-die links* and the tape-out of a resulting *EuroEXA processing unit* with integration of *FPGA for prototyping and data-flow acceleration*.

-> challenging targets achievable through adoption of beyond-state-of-the-art tech.

... *a homogenised software platform* offering heterogeneous acceleration with scalable shared memory access...

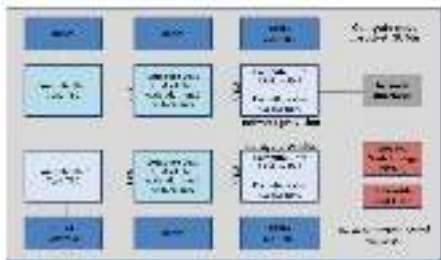
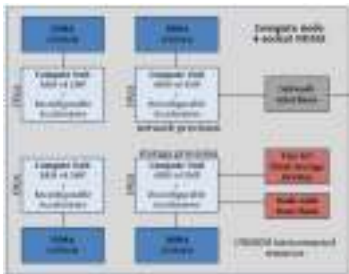
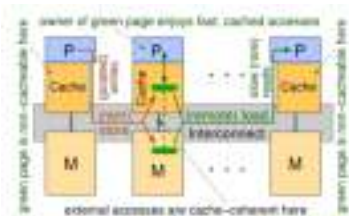
... a unique *hybrid, geographically-addressed, switching and topology interconnect* within the rack offering low-latency and high-switching bandwidth...

... a rich mix of *key HPC applications* from across climate/weather, physics/energy and life-science/bioinformatics domains

... deployment of an *integrated and operational peta-flop level prototype* hosted at STFC, monitored and controlled by *advanced runtime capabilities*, equipped by *platform-wide resilience mechanisms*.

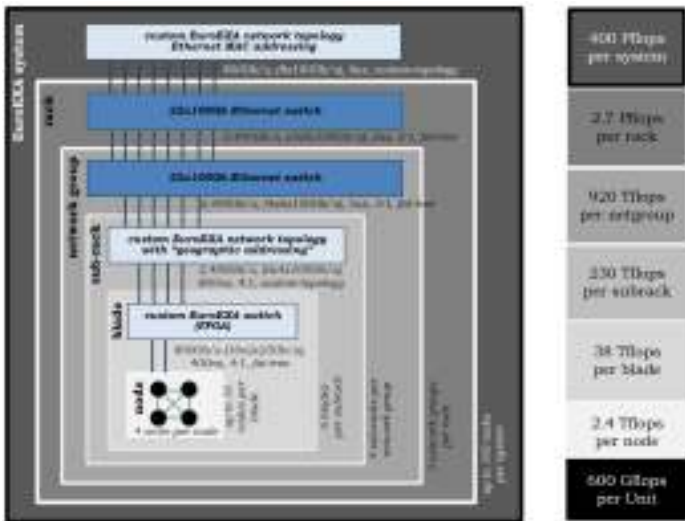
EuroExa (few) details

- high efficiency computing node with low latency (local and remote) memory access...

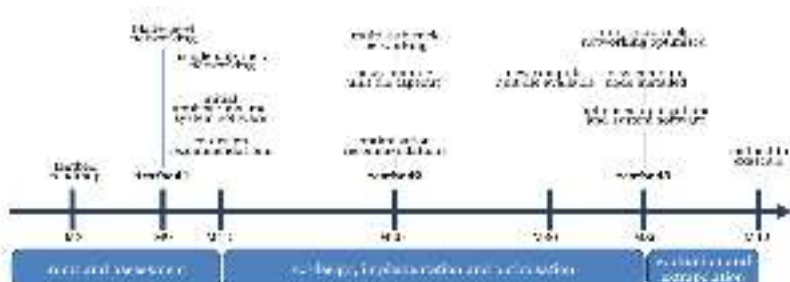
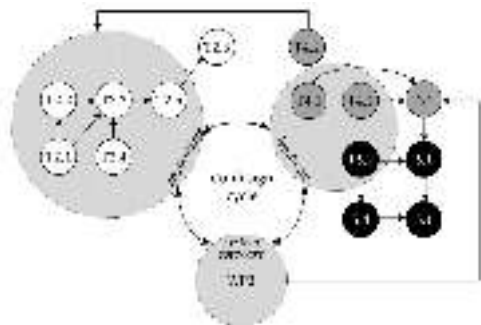


EuroExa (few) details

- Balanced, hierarchical network...



- EuroExa will use a strong co-design approach and incremental system design and integration



	WPI	WPI	WPI	WPI	WPI	WPI	Total WPI
ICD	18	65	20	3	0	0	118
TECH/217	15	15	67	161	25	5	334
SWP	15	60	91	5	0	0	171
DOCT11	1	27	86	19	37	6	256
STTC	1	36	8	5	35	1	186
UNIL	1	20	5	3	0	1	42
SPT	1	5	4	12	0	1	63
FA	1	1	2	14	67	10	135
UNIN	1	11	14	2	0	1	42
STH	1	11	36	3	0	5	75
MAY	1	6	34	5	0	1	136
SCUE	1	40	11	3	0	1	75
UNIN	1	20	24	13	67	1	138
STTC	1	40	11	2	0	1	94
STTC/217	1	30	2	3	0	1	72
UNIN	1	11	27	3	0	1	72
Total FTEs							1789

- Start date and duration: September 1st, 2017, 42 months
- Total budget: 20MEuro (>7MEuro for hardware procurement and NRE for silicon);
- INFN and UniFE mainly in :
 - benchmarking through applications: neural network simulator (RM1, link with HBP projects), LBE simulation (UniFE)
 - Network design at sub-rack level (RM1)
- INFN budget: 730 kEuro, 3 FTEs for the whole project duration

- HPC has a long and successful history (mainly not-European...)
- Fundamental scientific and engineering computing problems needs (again) ExaScale computing power
- The race toward ExaScale is started and Europe is trying to compete with established and emerging actors (USA, Japan, China,...) pushing for HPC technologies developments (EuroHPC, EXDCI, IPCEI,...)
- Many challenging issues require huge R&D efforts: power, interconnect, system packing and effective software frameworks
- ExaNeSt and EuroExa will contribute to the evaluation and selection of ExaScale enabling technologies, leveraging on Europe traditional expertise: embedded systems (ARM), excellence in scientific programming, design of non-mainstream network architecture