DEEP

Dynamical Exascale Entry Platform

2nd IS-ENES Workshop

on “High performance computing for climate models”

30.01.2013, Toulouse, France

Estela Suarez

The research leading to these results has received funding from the European Community's Seventh Framework Programme (*FP7/2007-2013*) under *Grant Agreement* n° 287530

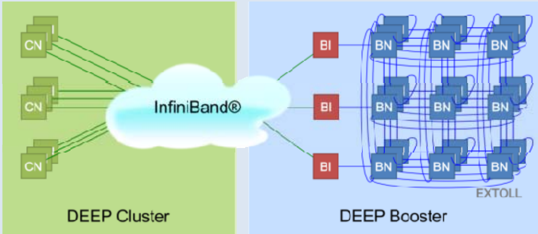
The DEEP project

- DEEP: “Dynamical Exascale Entry Platform”
- EU-project funded by the FP7 program:
  - 2011 call for Exascale
  - 3 Projects selected:
    - DEEP, MontBlanc, CRESTA
  - DEEP EU-funding: 8.03 M€
- Duration: 3 years
  - Dec 2011 – Nov 2014
- 16 Partners
  - Coordinator: JSC

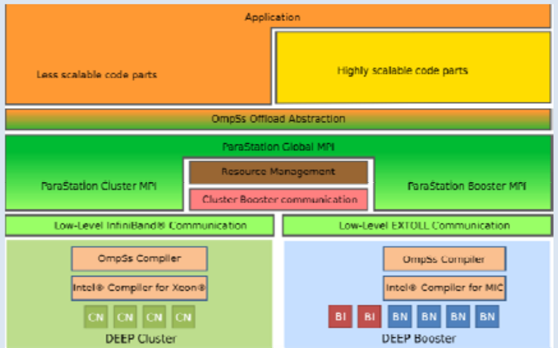
www.deep-project.eu

Goals

- **Build a prototype of an Exascale architecture:**
  - With accelerators that can react autonomously (→ “Booster”)
- **Hardware Development:**
  - Build a Booster with Intel® Xeon Phi™ and EXTOLL network
  - Energy efficient with “hot water” cooling
- **Software Development**
  - Ressource-Management System
  - Programming environment
  - Libraries and Performance analysis tools
- **Porting scientific applications**

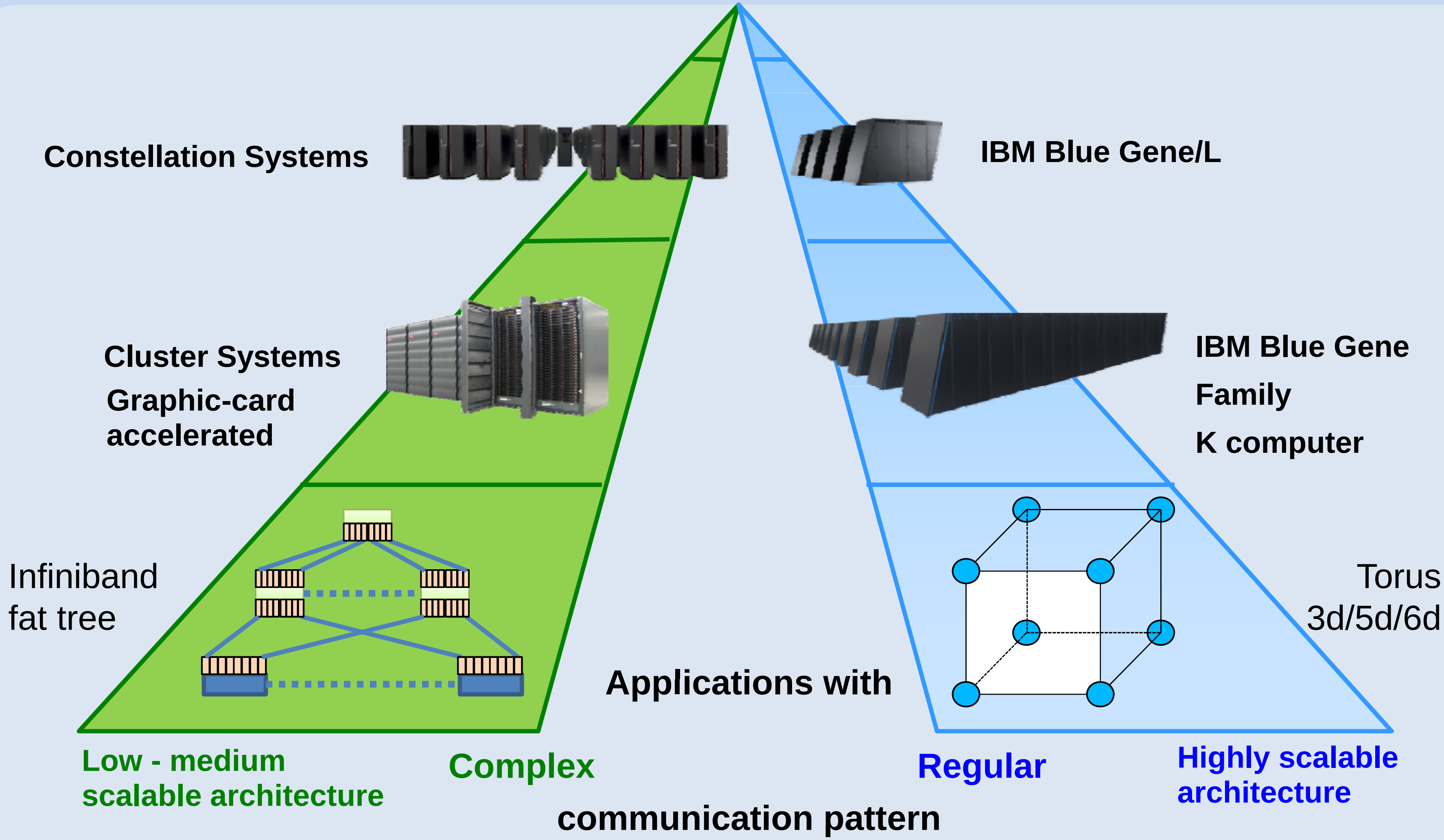


DEEP hardware architecture

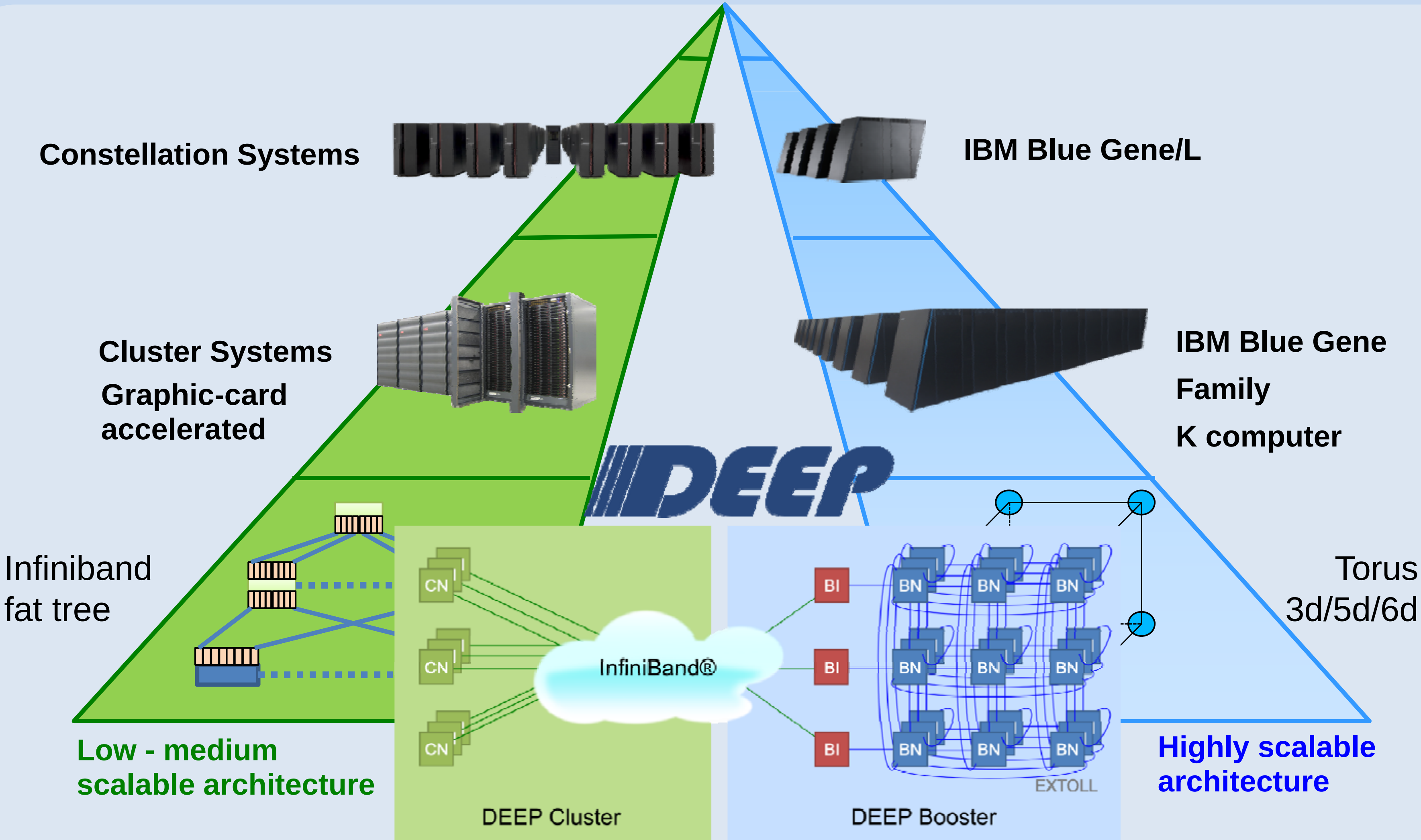


DEEP software architecture

DEEP Concept



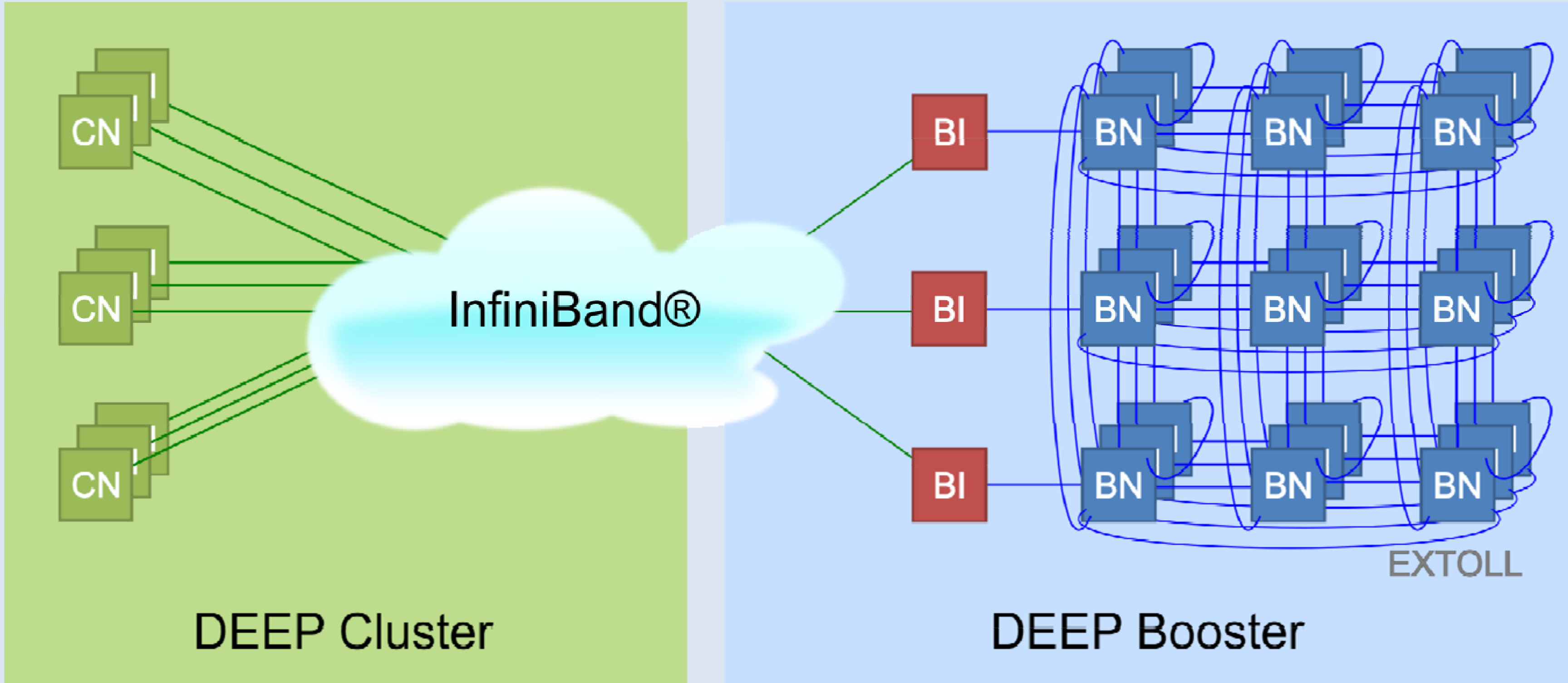
DEEP Concept



Hardware Architecture

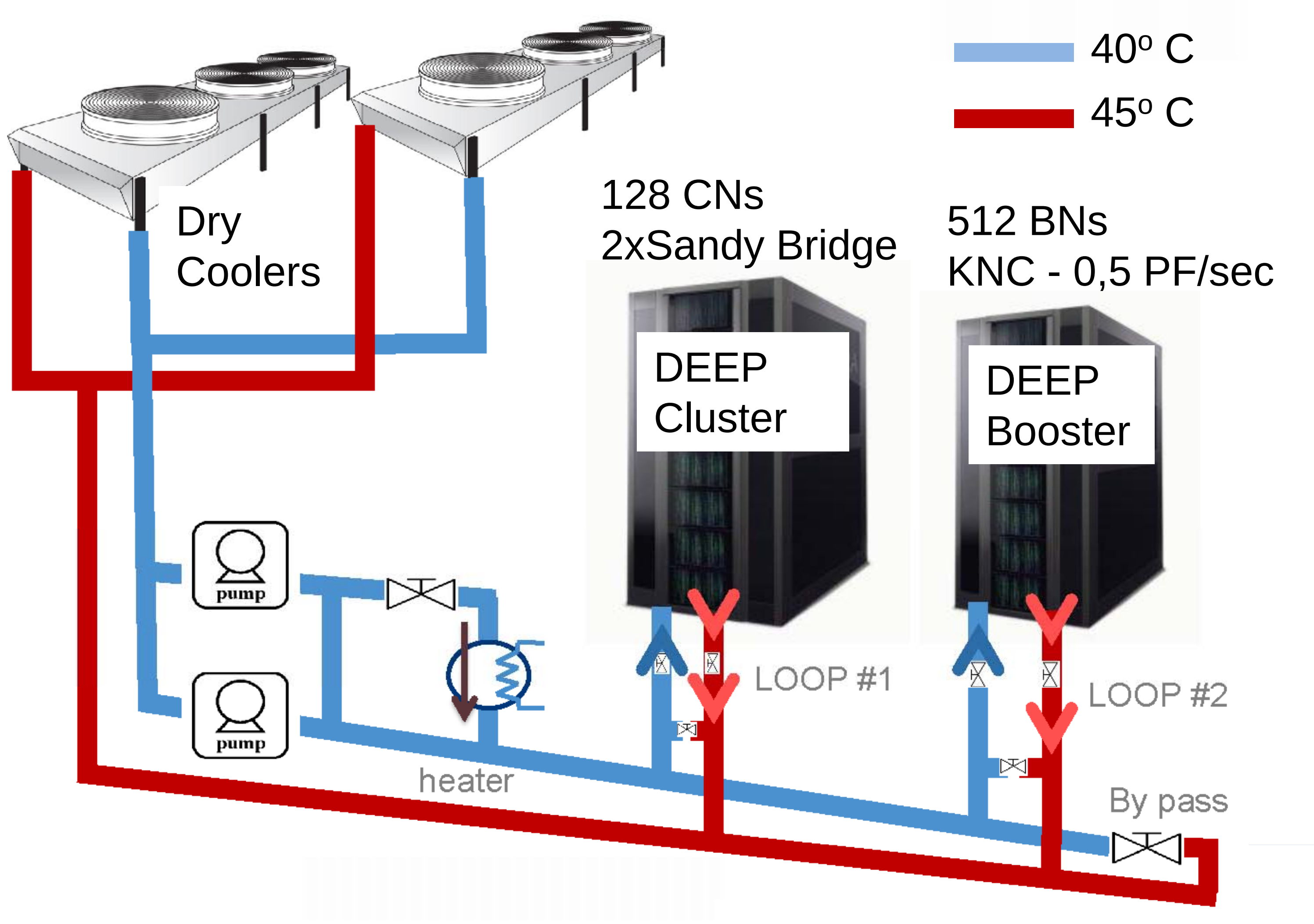
128 CNs - Sandy Bridge

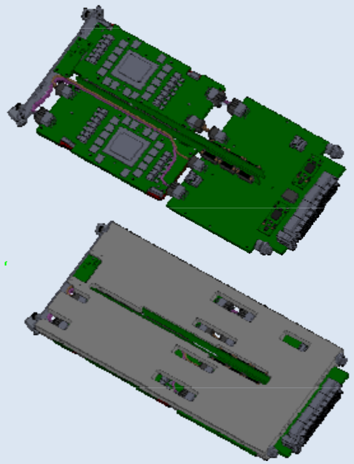
512 BNs – Intel® Xeon Phi™



Highly scalable code parts (HSCP) with regular communication patterns are off-loaded to the Booster

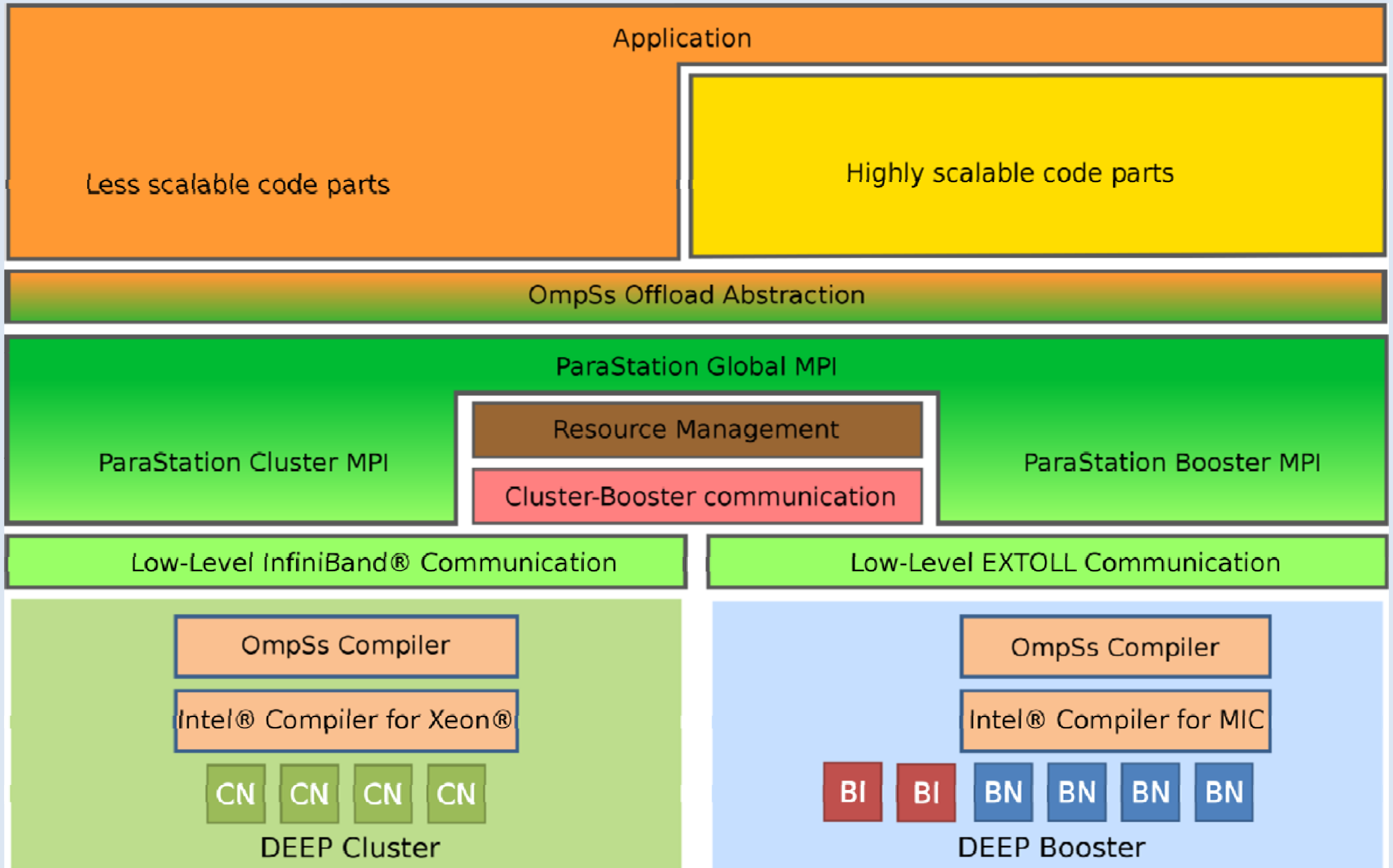
Hot-water cooling





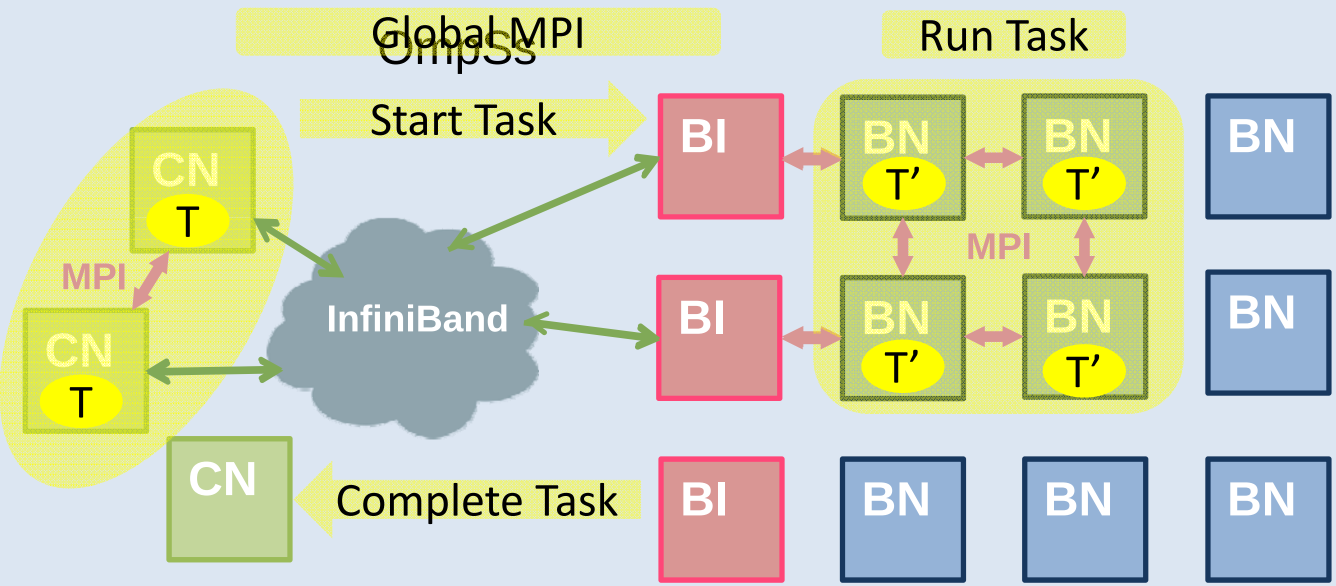
DEEP cold plate

Software-Architecture



DEEP Programming Model

- Combine task-based and MPI programming
  - Introduce highly parallel tasks implemented using MPI
  - Extend OmpSs to handle these tasks and data transfer between Cluster and Booster
  - Rely on MPI for scalability

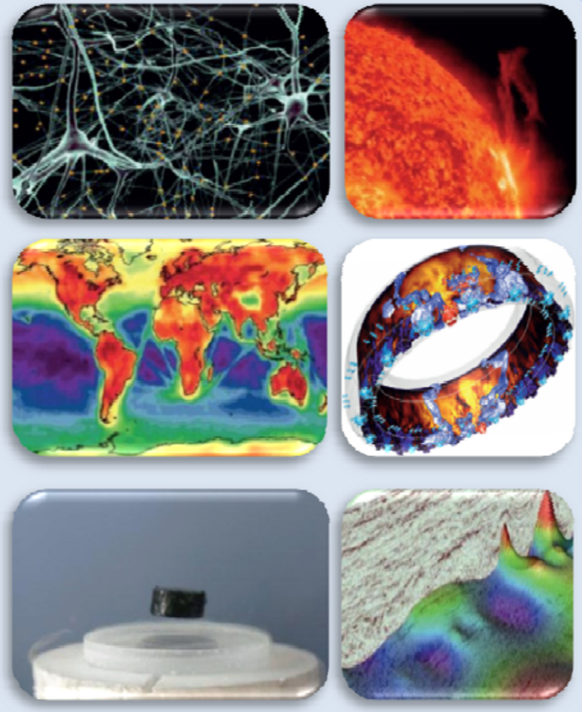


Advantages of DEEP with respect to standard accelerated architectures

- Booster Nodes do not need a host CPU
- Direct communication between BNs through EXTOLL
- Flexible assignment of groups of Cluster Nodes and Booster Nodes possible (also dynamically)
- Large blocks of code can be sent to the Booster
- Minimization of communication between CPU and accelerator
- Booster Nodes run standard Linux
- I/O is done through the Cluster

DEEP Pilot Applications

- **Scientific applications:**
  - Brain simulation (EPFL)
  - Space weather simulation (KULeuven)
  - Climate simulation (CYI)
  - Computational fluid engineering (CERFACS)
  - High temperature superconductivity (CINECA)
  - Seismic imaging (CGGVS)



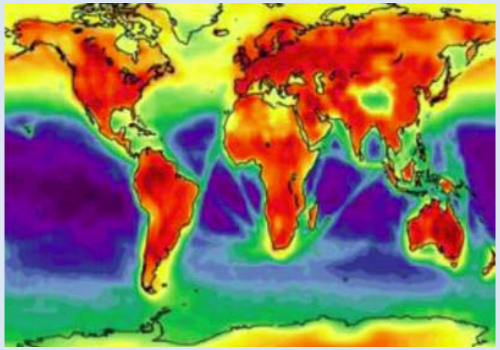
- **Goals**:
  - Evaluate the DEEP concept and its programmability
  - Compare its performance with standard architectures
  - Create best practice guidelines
  - Propose improvements to the DEEP System

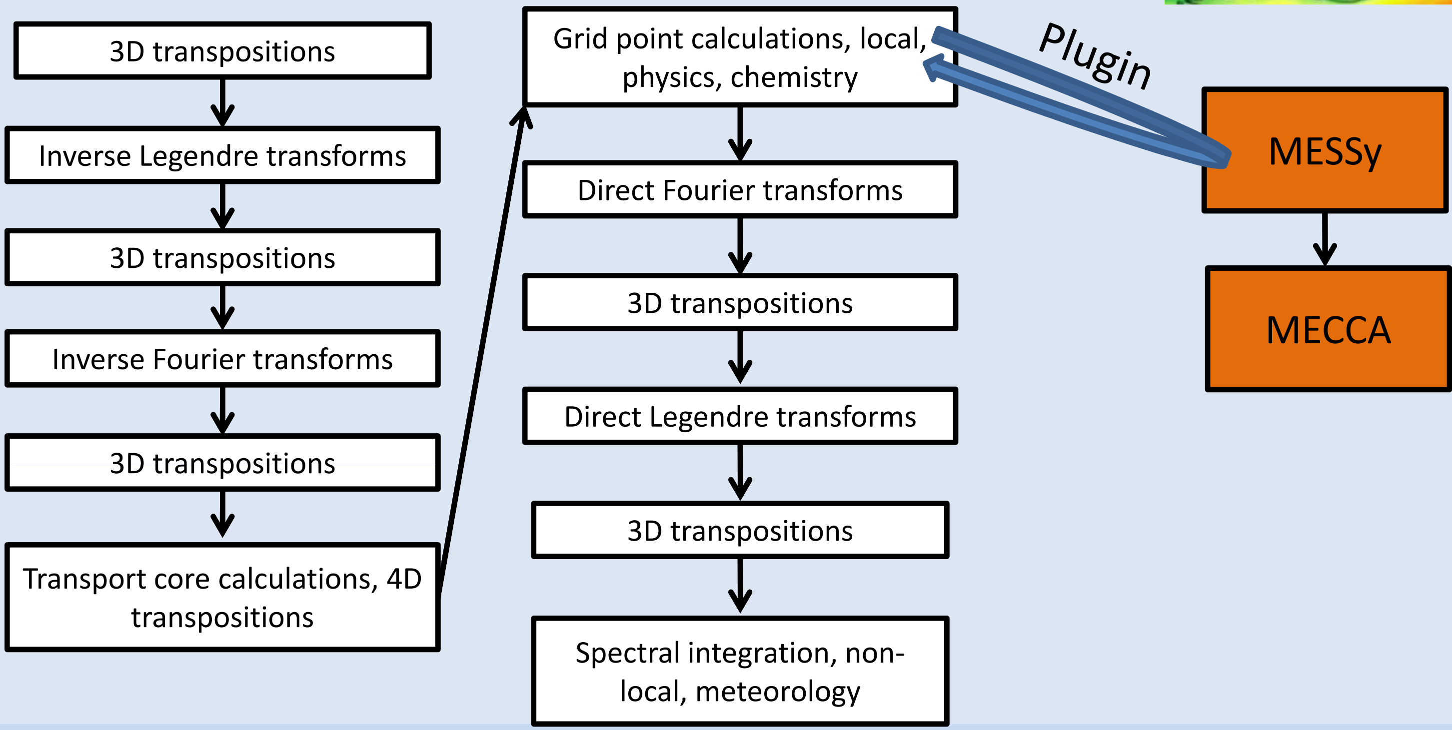
Climate Simulation (CYI)

Structure

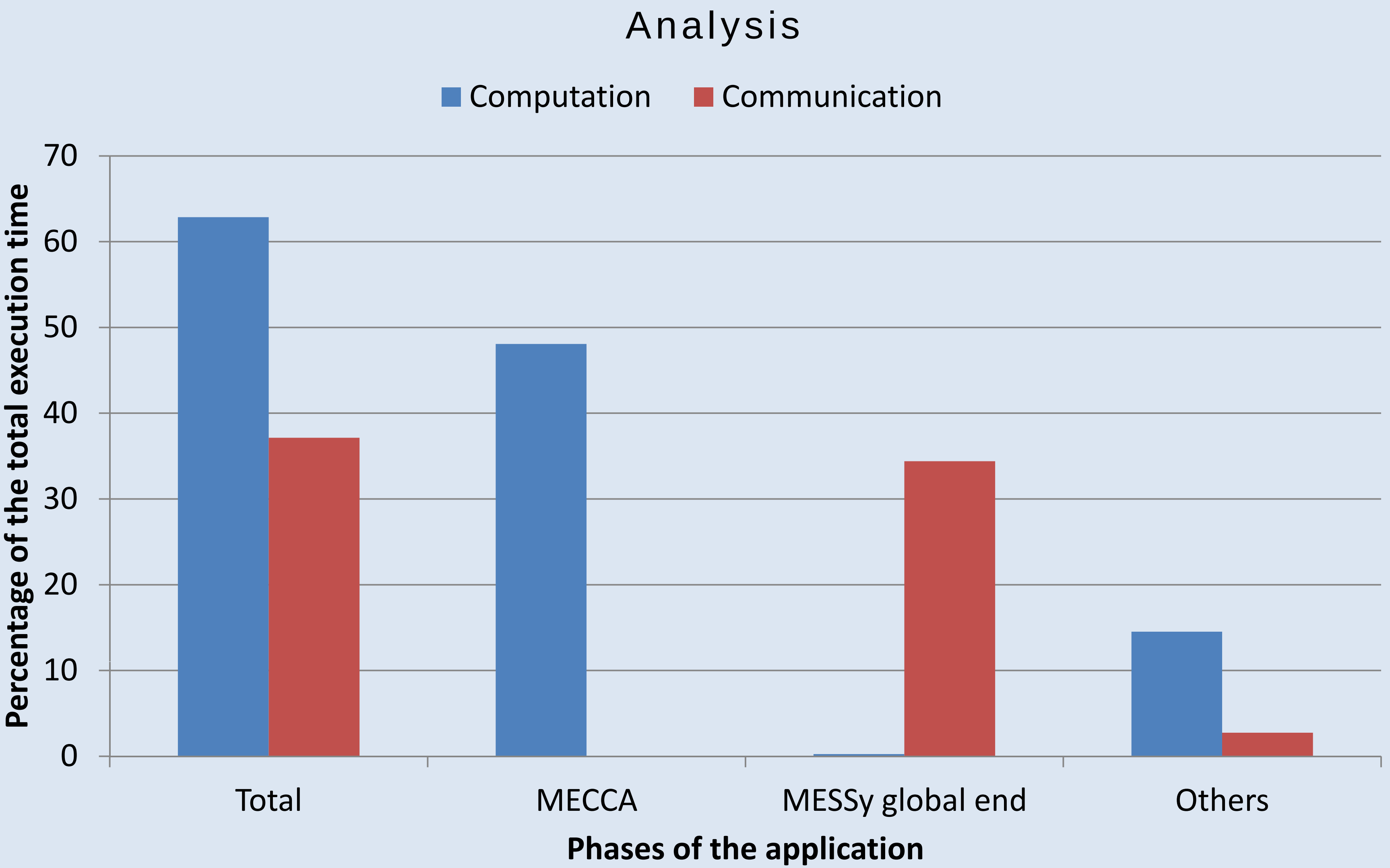
- Two coupled models:
  - ECHAM (Atmospheric)
  - MESSy (Physicochemical interactions)

EMAC = ECHAM/MESSy Atmospheric Chemistry



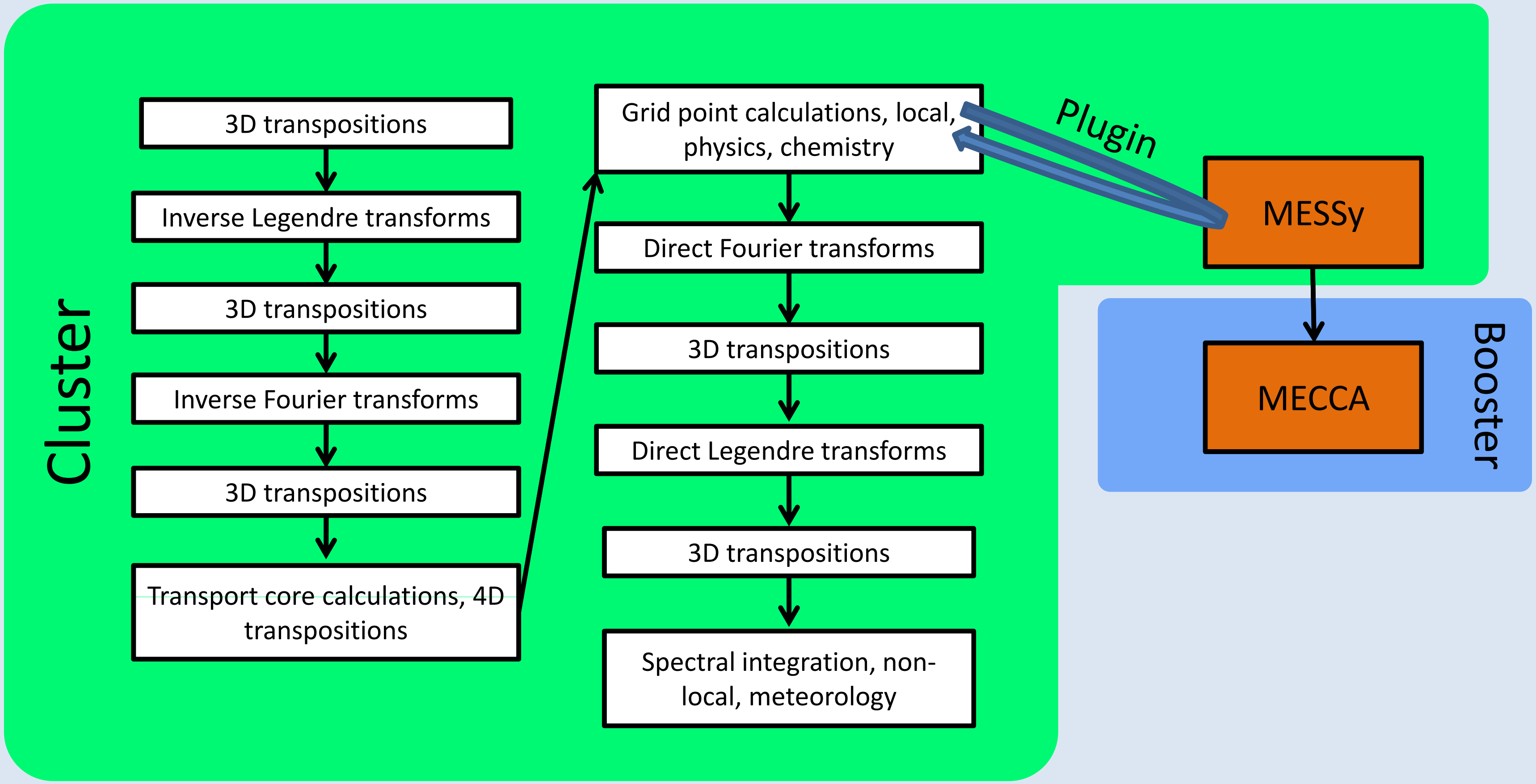


Climate Simulation (CYI)



Climate Simulation (CYI)

Division



DEEP Status (I)

- **Project is now in Month 14**
- **Hardware status:**
  - DEEP Cluster installed at JSC (Juelich)
  - 3x Intel Xeon Phi workstations installed at JSC
  - Booster backplane design finished
  - BNC prototype ready and under test

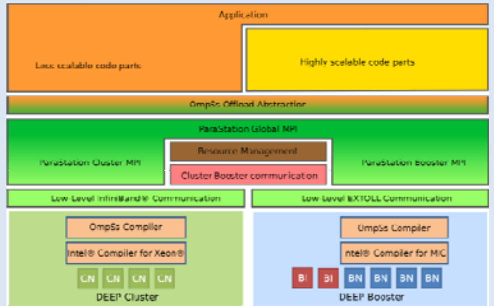
DEEP Cluster

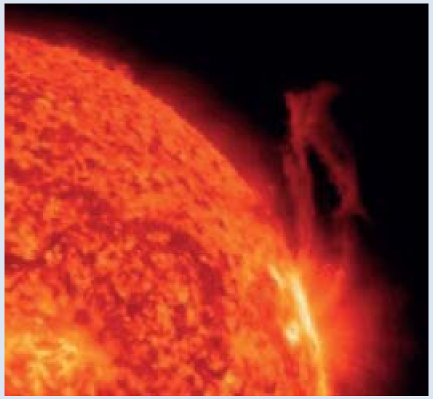
BNC (Booster Node Card) prototype

DEEP Status (II)

- **Software status:**
  - Programming model definition completed:
    - Global MPI + OmpSs
  - OmpSs runtime (Nanos++) ported to MIC
  - ParaStationMPI port to MIC and EXTOLL ongoing
  - Low-level Cluster-Booster protocol implemented
  - Mathematical libraries under evaluation
- **Scientific Applications:**
  - Structure of applications analysed
  - First ansatz on application division (between Cluster and Booster) already defined
  - First experiences with Intel Xeon Phi (KNC)

DEEP software architecture





Space Weather application

Thank you for your attention

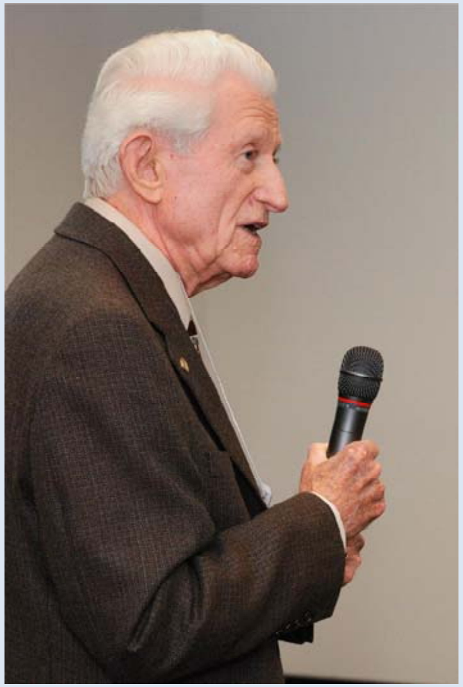
Dynamical Exascale
Entry Platform

www.deep-project.eu

The research leading to these results has received funding from the European Community's Seventh Framework Programme (*FP7/2007-2013*) under *Grant Agreement* n° 287530

BACK

Concurrency Levels



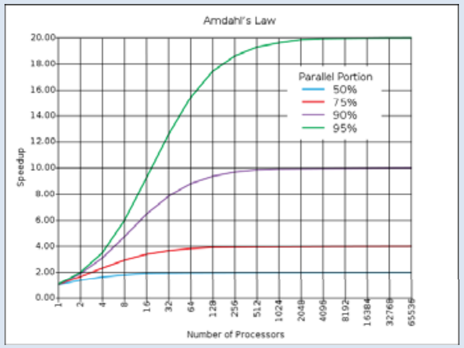
scalar + parallel

strong scaling

$$S_N = \frac{1}{s + \frac{p}{N}}$$

weak scaling

$$W = s + pN$$



2 concurrency levels, K and N

$$S_{K,N} = \frac{1}{\frac{1-p}{Kf} + \frac{p}{N}}$$

p_r=.50, N = 500.000, K = 10.000, f = 1:
→ S = 20.000

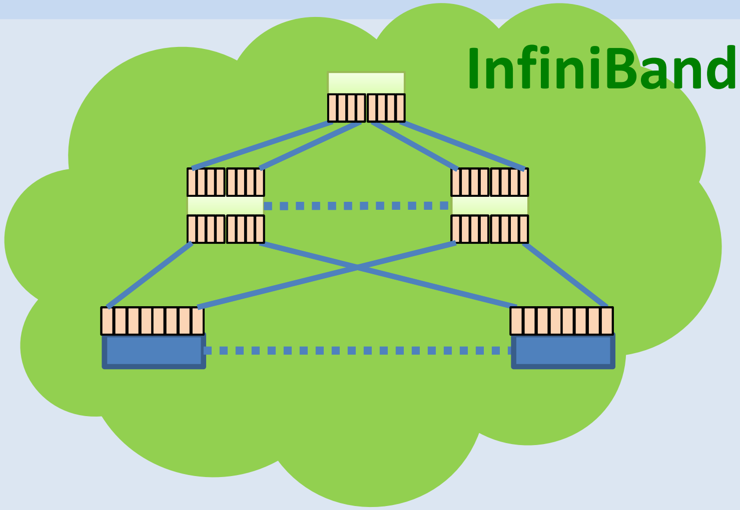
p_r=.95, N = 500.000, K = 10.000, f = 4:
→ S =320.000

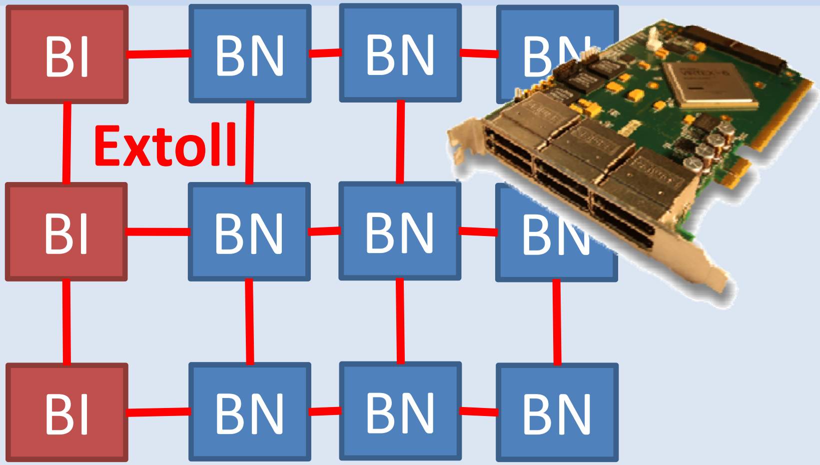
EXTOLL performance comparison

Technology	Implementation	Internal Clock Frequency	Link Bandwidth [Gb/s]	Latency [µs]	Effective maximum Bandwidth [GB/s]	Message Rate [msg/s]
EXTOLL VENTOUX	Xilinx Virtex6 FPGA	200 MHz	16 Gb/s	1.2 µs	Up to 1.4 GB/s	~ 25 millions
EXTOLL TOURMALET	65 nm ASIC	750 MHz	120 Gb/s	0.6 µs	Up to 10 GB/s	~ 100 millions
IB FDR*	45 nm ASIC	Unknown	56 Gb/s	0.8 µs	Not measured	Not measured
Typical 1GE	ASIC based	e.g. 125 MHz	1 Gb/s	e.g. 40 µs	0.11 GB/s	~0.5 millions
10GE	ASIC based	~125 to 312 MHz	12,5 Gb/s	12.5 µs	1.2 GB/s	< 2.5 millions
Cray Gemini	90 nm ASIC	650/800 MHz	75 Gb/s	1.5 µs	Up to 5.9 GB/s	~ 2 millions
Tianhe-1a	ASIC based	Unknown	80 Gb/s	2.5 µs	Up to 6.34 GB/s	~ 1-3 millions
TOFU (K-Computer)	65 nm ASIC	312.5 MHz	50 Gb/s	1.5 µs	Up to 4.76 GB/s	> 8 millions

* Mellanox literature data

InfiniBand vs. Extoll

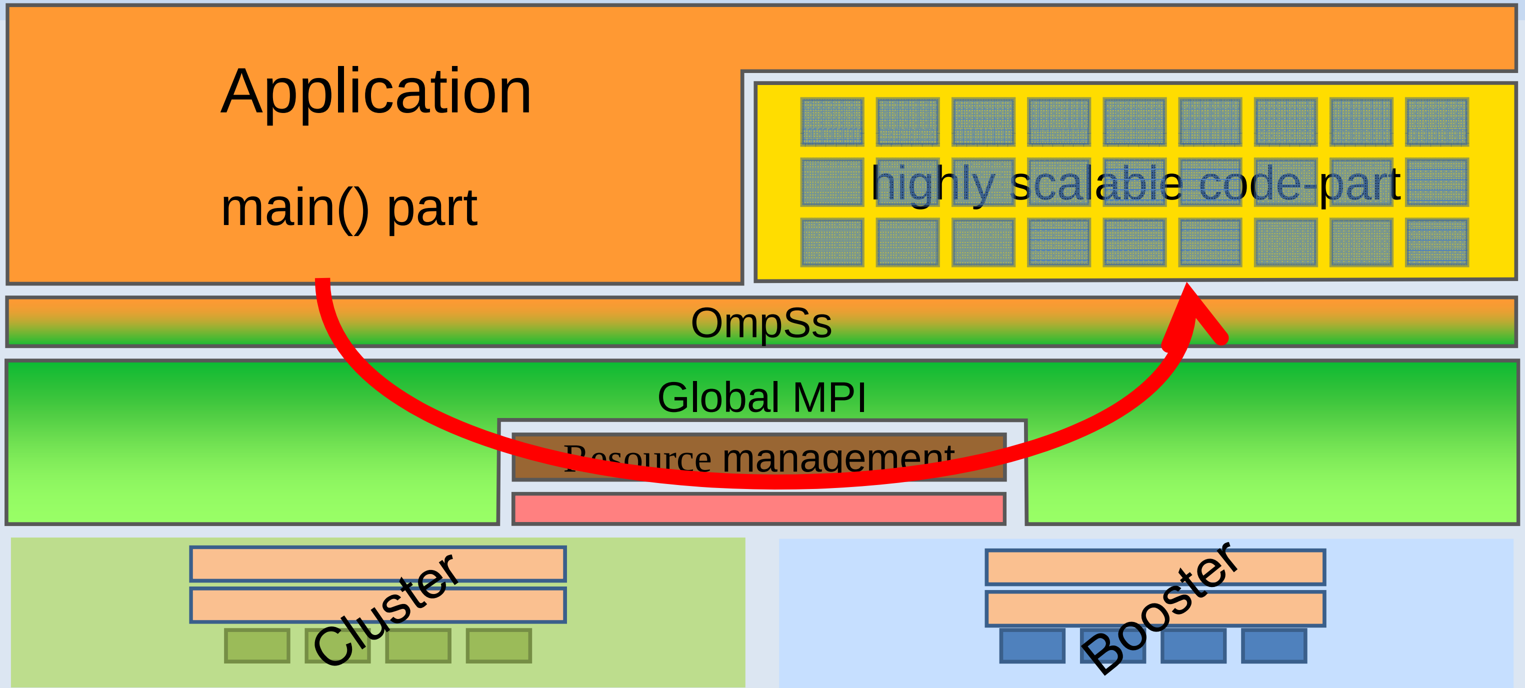




- **Low latency**
  - 1-2 µsec
- **Large Bandwidth**
  - 32 Gbit/s
- **Clos-Topologie (fat-tree)**
  - 36-port Switches
  - Very flexible
  - But not so scalable

- **Very low latency**
  - <1 µsec
- **Large Bandwidth**
  - 32 Gbit/s
- **3D-torus Topology**
  - 10-port Switches
  - Highly scalable
  - But low flexibility

Application startup in detail



- Application's main() part runs on Cluster Nodes (CN) only
- Resources managed statically or dynamically
- OmpSs acts as an abstraction layer
- Actual spawn done via Global MPI
- Spawn is a collective operation of Cluster processes
- Highly scalable code-parts (HSCP) utilise multiple Booster Nodes (BN)