The DEEP Projects:
10 years turning heterogeneity into modularity

Estela Suarez, Jülich Supercomputing Centre (JSC)

28.03.2022 – MAELSTROM Workshop

The Research leading to these results has received funding from the European Commission's FP7, H2020, and EuroHPC Programmes,
under Grant Agreements n° 287530, 610476, 754304, and 955606

The DEEP projects

2011-2021: The DEEP projects

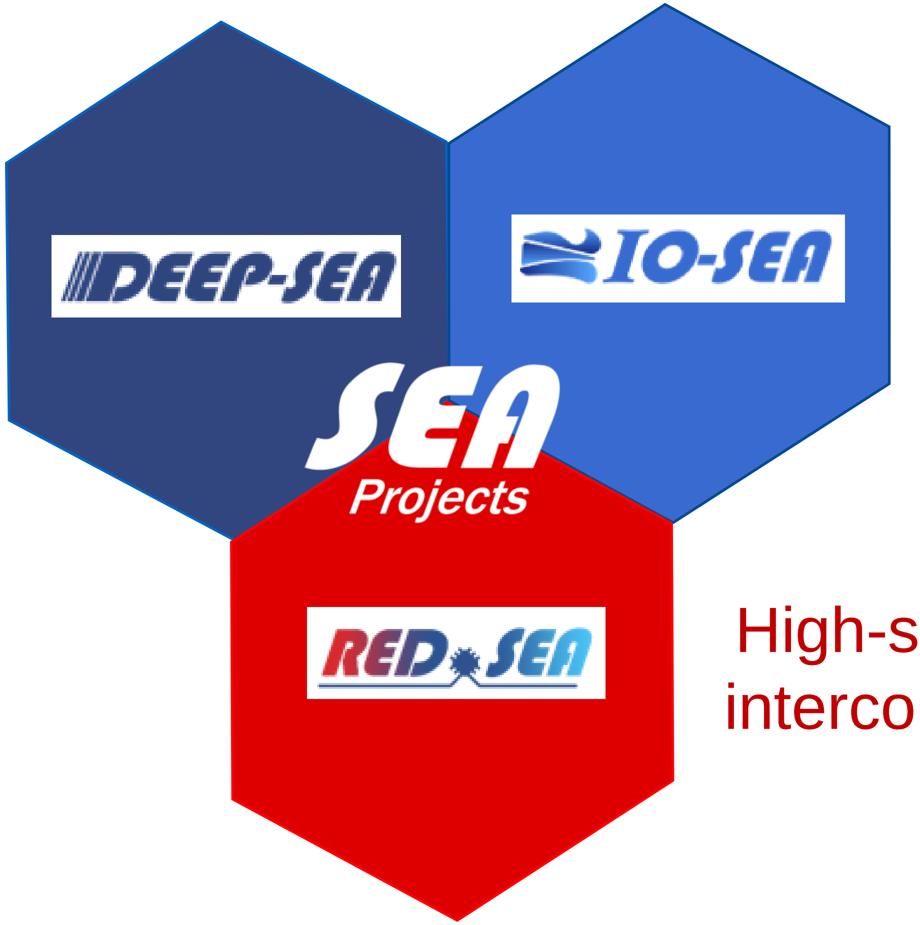
- **DEEP** (2011 – 2015)
  - Introduced Cluster-Booster architecture
- **DEEP-ER** (2013 – 2017)
  - Added I/O and resiliency functionalities
- **DEEP-EST** (2017 – 2021)
  - Modular Supercomputer Architecture

2021-2024 The SEA projects

- DEEP-SEA, IO-SEA, RED-SEA

SEA Projects family

Provide solutions for Modular Supercomputers of Exascale performance



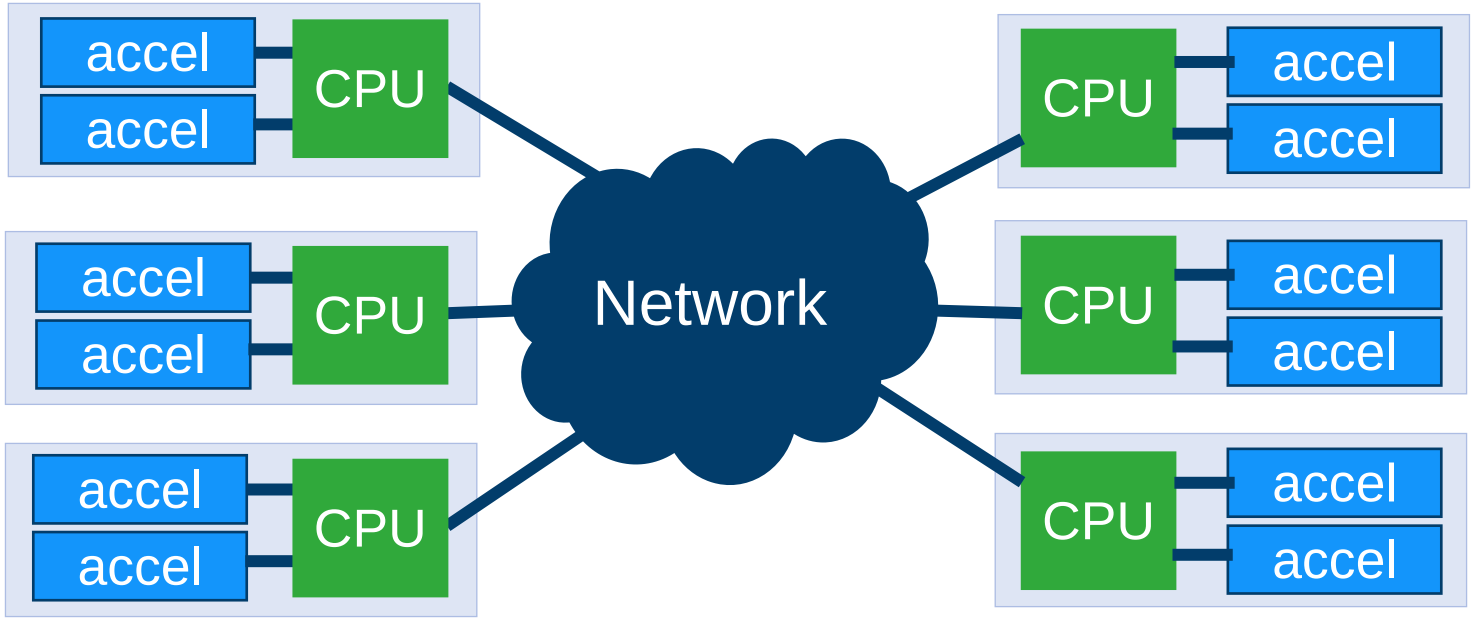
Software stack for heterogeneous compute and memory systems

I/O and data management

High-speed interconnect

Heterogenous Monolithic

Every node contains accelerators (e.g. GPUs)



+: Energy efficient

+: Easy management

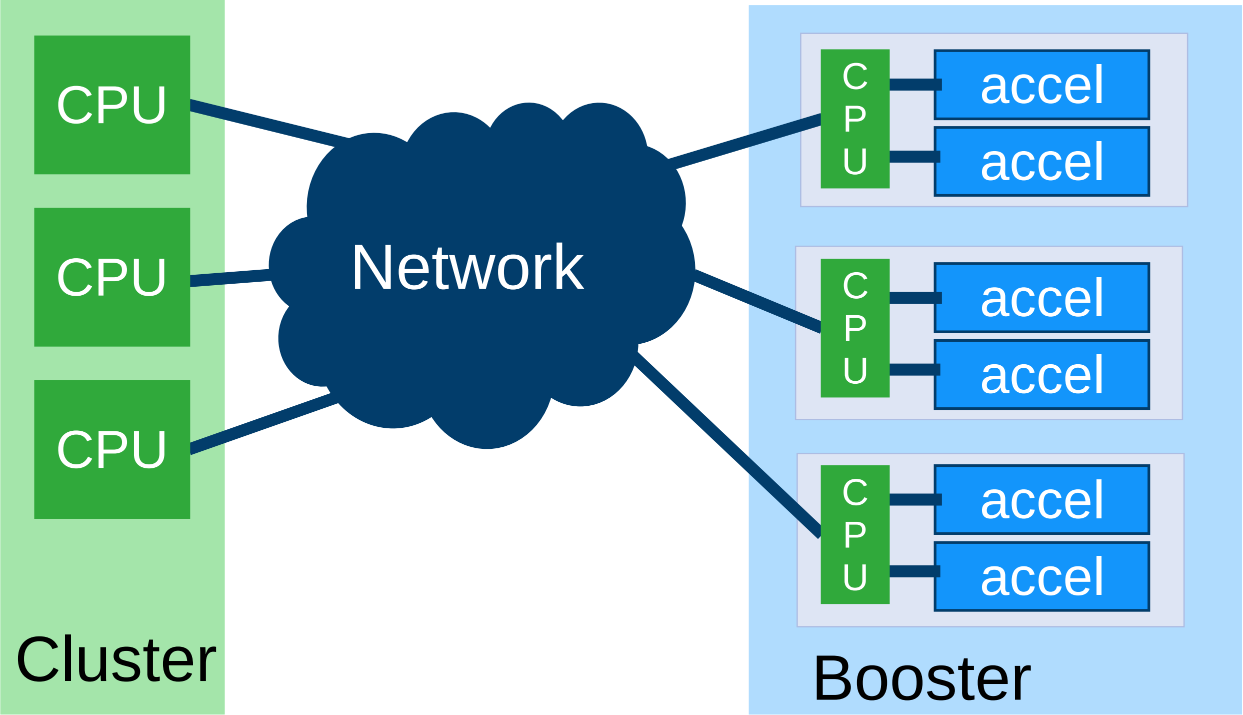
-: Static assignment of accelerators to CPUs

-: Difficult to efficiently share resources

- Every node contains CPU(s) and some accelerator
- All nodes are equal → "monolithic"

Heterogenous Modular

Different nodes are grouped in "modules"

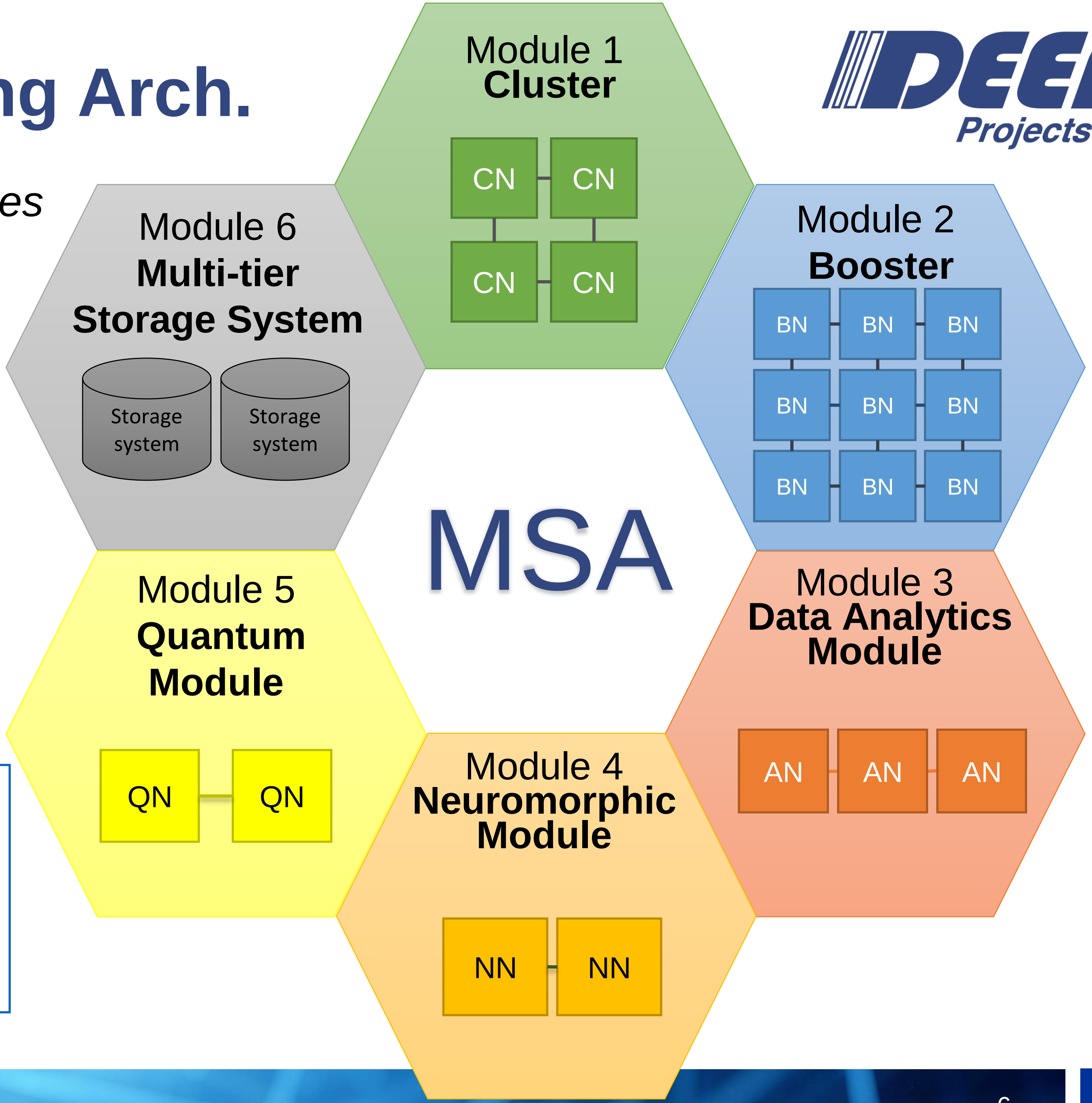


+: Energy efficient
+: Better scalability
+: High flexibility
+: Dynamic resource assignment
-: Complexity

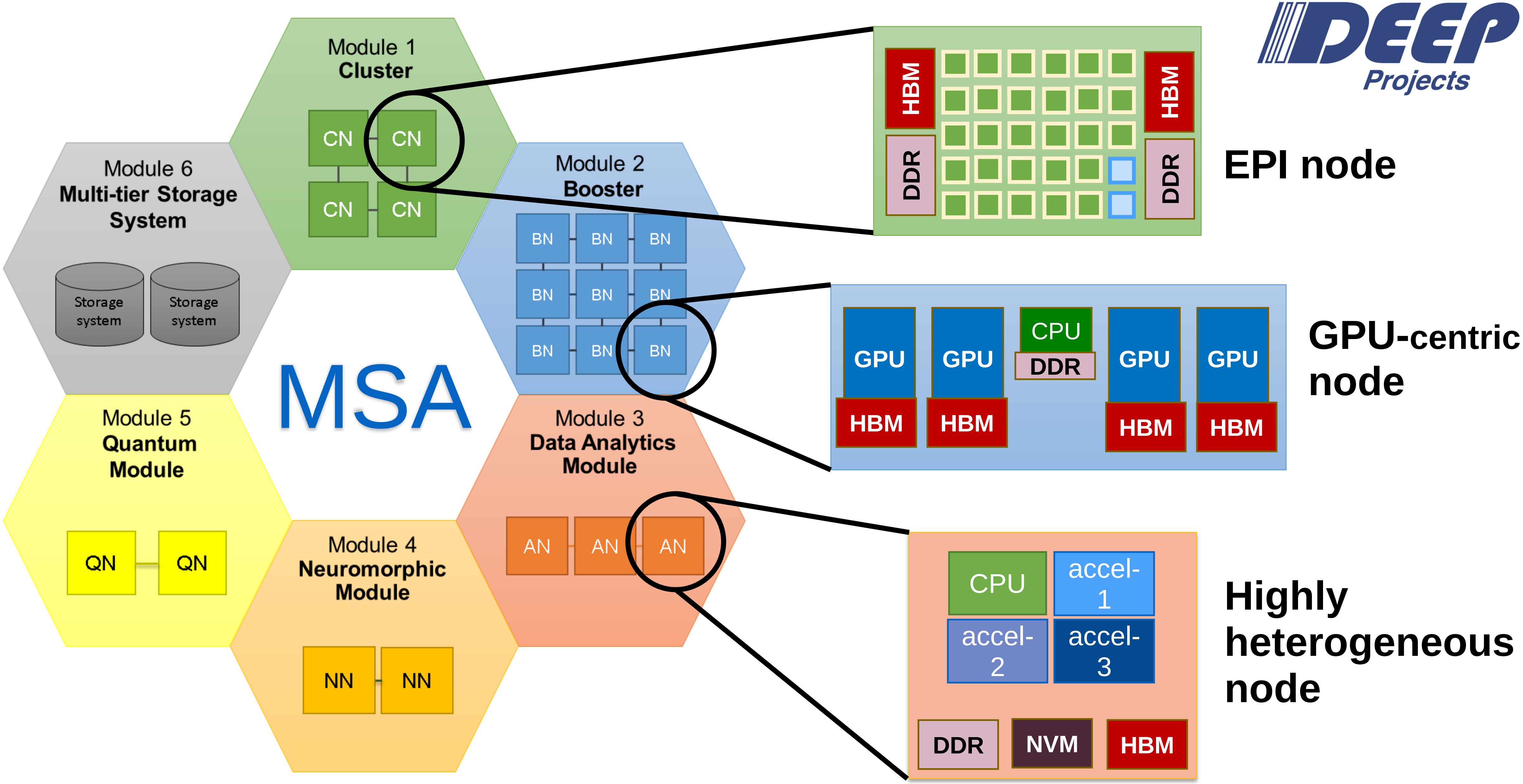
- All nodes within one module are equal
- Different modules have different configurations → "modular"

Modular Supercomputing Arch.

Composability of heterogeneous resources



- **E. Suarez**, N. Eicker, Th. Lippert, "*Modular Supercomputing Architecture: from idea to production*", Chapter 9 in Contemporary High Performance Computing: from Petascale toward Exascale, Volume 3, p 223-251, CRC Press. (2019)
- **E. Suarez**, N. Eicker, and Th. Lippert, "Supercomputer Evolution at JSC", Proceedings of the 2018 NIC Symposium, Vol.49, p.1-12, (2018)



MSA

Module 1
Cluster

CN CN CN CN

Module 2
Booster

BN BN BN BN BN BN BN BN BN

Module 3
Data Analytics Module

AN AN AN

Module 4
Neuromorphic Module

NN NN

Module 5
Quantum Module

QN QN

Module 6
Multi-tier Storage System

Storage system
Storage system

EPI node

HBM DDR HBM DDR

GPU-centric node

GPU GPU CPU DDR GPU GPU
HBM HBM HBM HBM

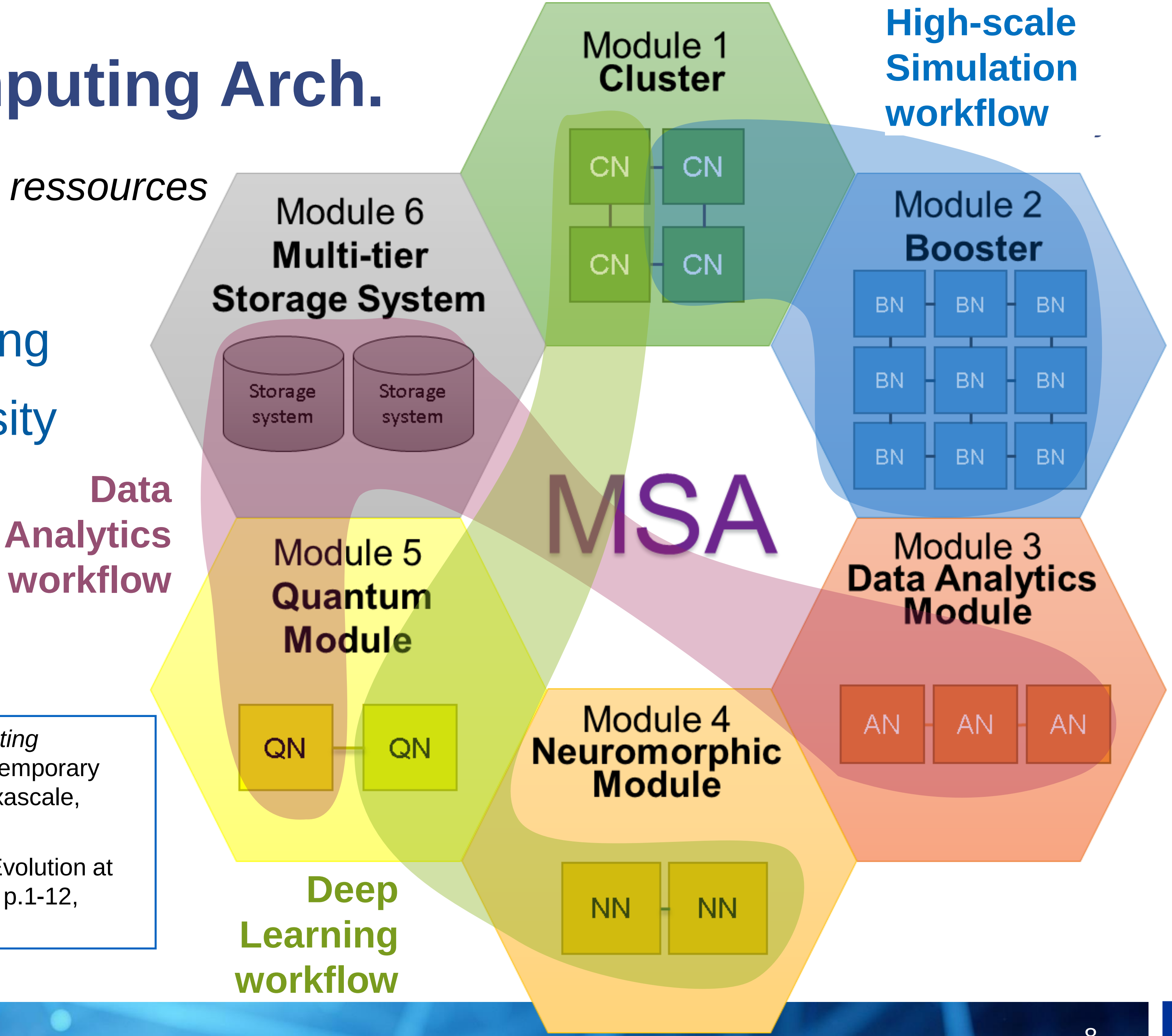
Highly heterogeneous node

CPU accel-1
accel-2 accel-3
DDR NVM HBM

Modular Supercomputing Arch.

Composability of heterogeneous ressources

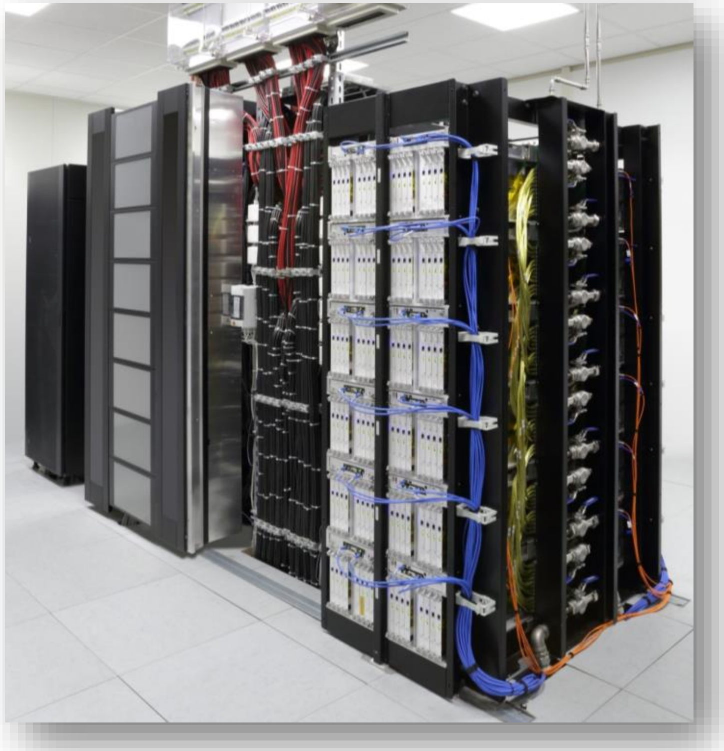
- Effective resource-sharing
- Match application diversity



- **E. Suarez**, N. Eicker, Th. Lippert, "*Modular Supercomputing Architecture: from idea to production*", Chapter 9 in Contemporary High Performance Computing: from Petascale toward Exascale, Volume 3, p 223-251, CRC Press. (2019)
- **E. Suarez**, N. Eicker, and Th. Lippert, "Supercomputer Evolution at JSC", Proceedings of the 2018 NIC Symposium, Vol.49, p.1-12, (2018)

The hardware Prototypes

2015



2016

2020

© FZJ

DEEP Prototype
128 Xeon + 284 KNC nodes
InfiniBand + 1.5Gbit Extoll
550 TFlop/s

DEEP-ER Prototype
16 Xeon + 8 KNL nodes
100Gbit Extoll
40 TFlop/s

DEEP-EST Prototype
55 Cluster + 75 Booster + 16 Data Analytics
100 Gbit Extoll + InfiniBand + Eth
800 TFlop/s

Software environment

- **Low-level SW:** Inter-network bridging
- **Scheduler**: Slurm, psslurm (ParaStation Modulo)
- **Filesystem**: BeeGFS, GPFS
- **Compilers**: Intel, GCC, NVIDIA HPC SDK
- **Debuggers**: Intel Inspector, TotalView
- **Programming**: ParaStation MPI, OpenMP, OmpSs, CUDA
- **Performance analysis tools**: Scalasca, Score-P Extrae/Paraver, Vampir, Intel Advisor, VTune…
- **Benchmarking tools**: JUBE
- **I/O Libraries**: SIONlib, SCR, HDF5,…

- **Eicker et al.**, *Bridging the DEEP Gap - Implementation of an Efficient Forwarding Protocol*, Intel European Exascale Labs - Report 2013 34-41
- **Clauss et al.**, *Dynamic Process Management with Allocation-internal Co-Scheduling towards Interactive Supercomputing*, COSH@HiPEAC,(2016)

Heterogeneity from user’s PoV

- **Slurm supports the ability to submit heterogeneous jobs** (since v 17.11)
  - form **job pack (het-job)** allocation using colon notation for **salloc**, **sbatch**, **srun**
  - even allowing different executables

```
$ srun  -N 1 -p part1 ./first \
        : -N 2 -p part2 ./second
```

- **Full support for job packs in ParaStation psslurm**, with **unique features** for modular jobs:
  - Support for heterogeneous jobs with common MPI_COMM_WORLD, or with separated / interconnected MPI_COMM_WORLDS
  - For each job in the job pack, resources can be specified individually
  - Support global resources (e.g. gateways): **psgw** plugin to **psmgmt** + spank plugin
    - *Compensates for Slurm’s inability to handle global resources*
    - *Extends salloc, srun and sbatch*
- **ParaStation has further features that make is MSA-ware**
  - E.g. hierarchical collective operations

Source: Thomas Moschny

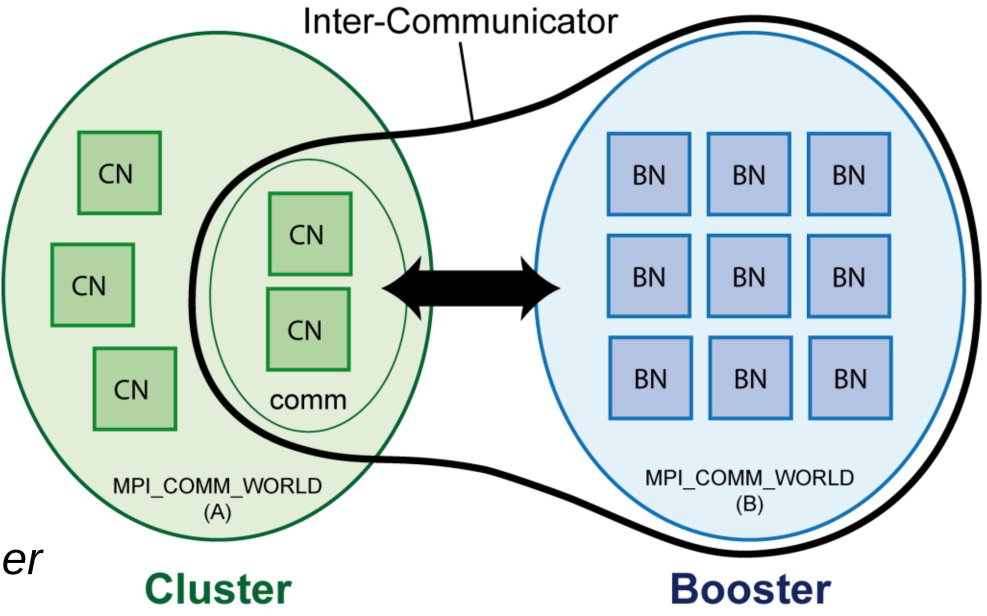
ParaStation Global MPI for MSA

- An MPI application can run:
  - Using only Cluster nodes
  - Using only Booster nodes
  - Distributed over Cluster and Booster
    - *In this case two executables are created*
    - *Collective offload process*
    - *Transparent data exchange via MPI*

- ParaStation Global MPI
  - Uses `MPI_Comm_spawn()`
    - *Collective spawn groups of processes from Cluster to Booster (or vice-versa)*
  - Inter-communicator
    - *Connects the 2 MPI_COMM_WORLD*
    - *Contains all parents on one side and all children on the other*
      - *Returned by MPI_Comm_spawn for the parents*
      - *Returned by MPI_Get_parent by the children*

- One can also start two parts of a code and connect them via `MPI_Connect()`

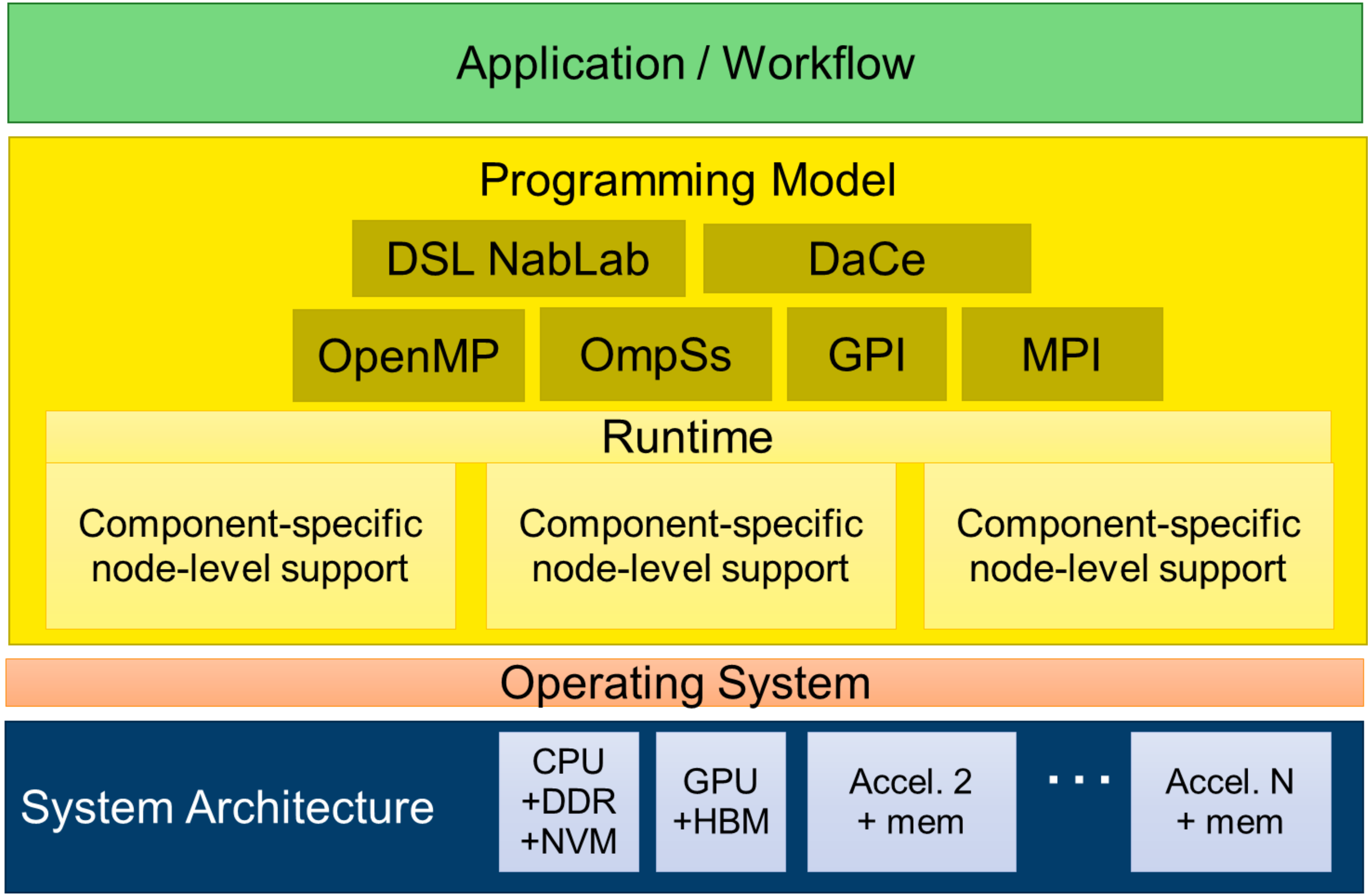
- Or have one single common `MPI_COMM_WORLD` and split it into subcommunicators via `MPI_Comm_Split()`



- **Clauss et al.**, *Dynamic Process Management with Allocation-internal Co-Scheduling towards Interactive Supercomputing*, COSH@HiPEAC, (2016)

DEEP-SEA: Extending the software stack

- **Support for accelerators & memory**
- **Malleability**
- **Composability**
- **Performance portability**
- **Resiliency**



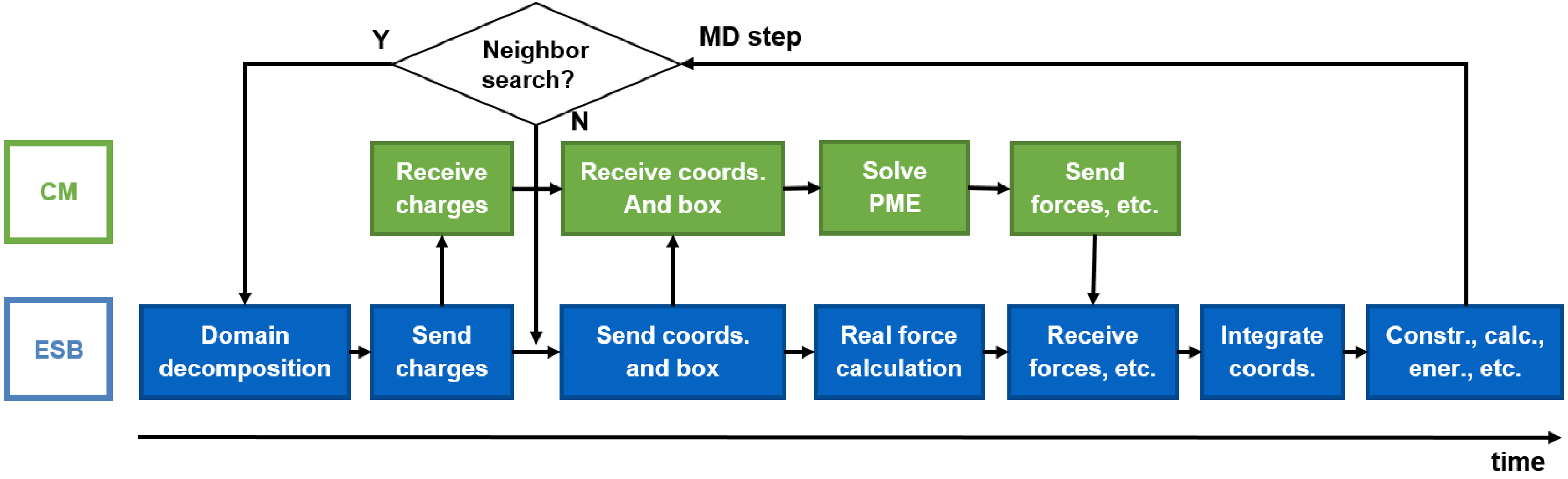
Co-design Applications

Feed user requirements in SW development, evaluate the SW-packets

- Application areas
  - Space Weather: xPic, AIDApy
  - Weather Forecast: IFS
  - Seismic imaging: RTM, BSIT
  - Molecular dynamics: GROMACS
  - Computational fluid dynamics: Nek5000
  - Neutron Monte Carlo transport for nuclear energy: PATMOS
  - Earth System Modelling: TSMP
- Additionally: low-level and synthetic benchmarks
- User support
- Early access program

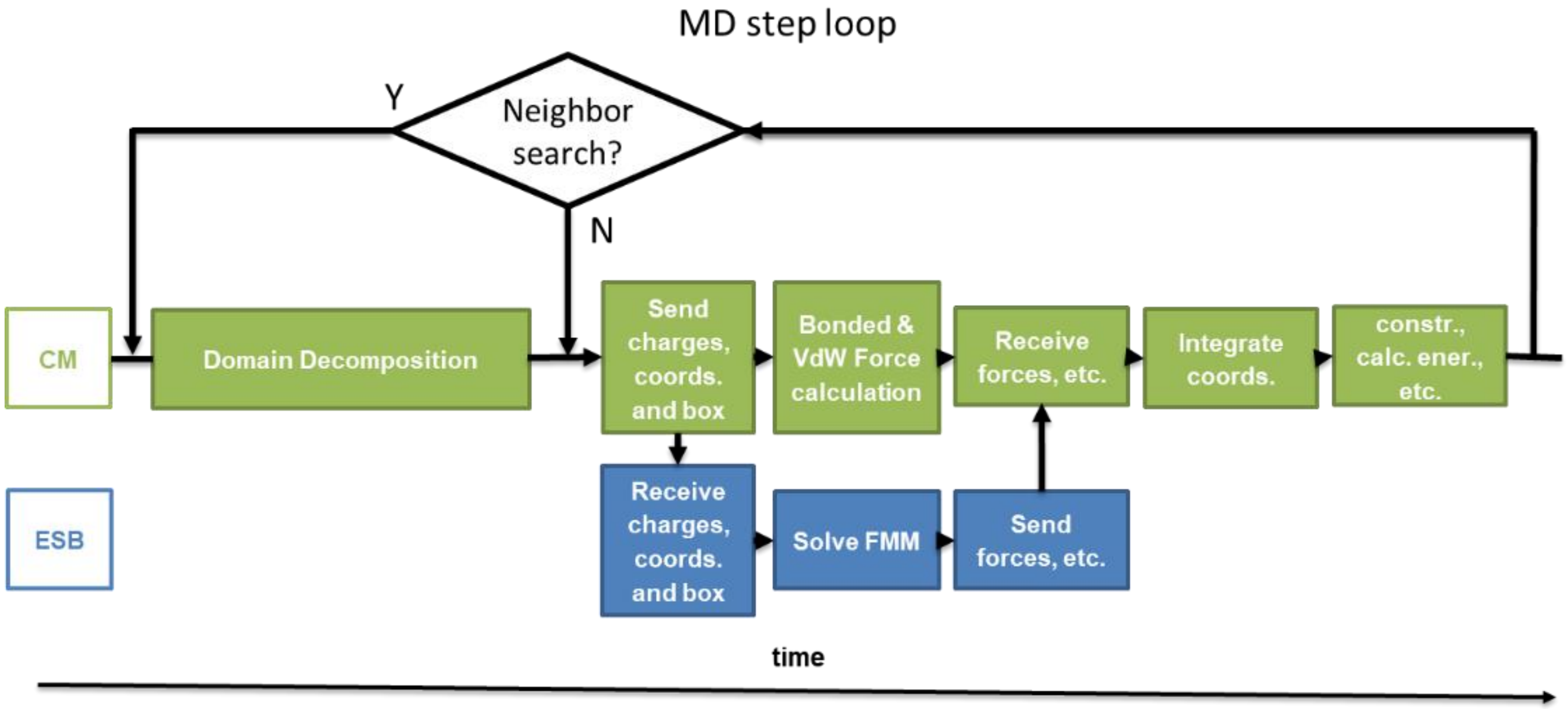
GROMACS: multi-module usage in MD simulations

- Best mapping on MSA depends on the problem size and aims at optimizing the computational load
  - $<10^4$ particles: only on Cluster (CPU)
  - ~ 10^5 particles: Booster or DAM (Data Analytics Module)
  - $>10^6$ particles (large macromolecules): pair interactions on GPU, run PME on CPUs



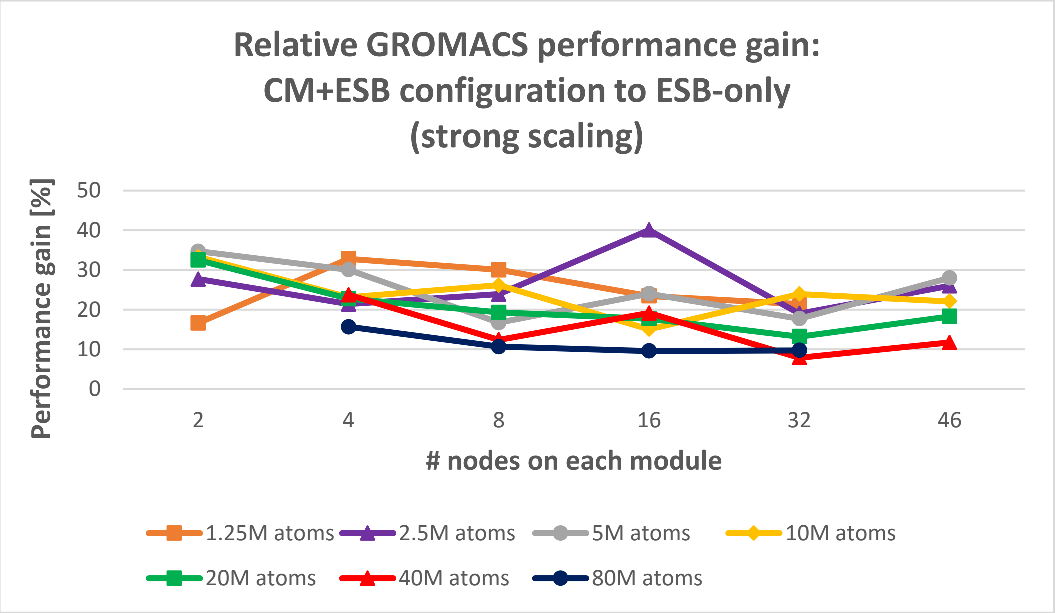
GROMACS: multi-module usage in MD simulations

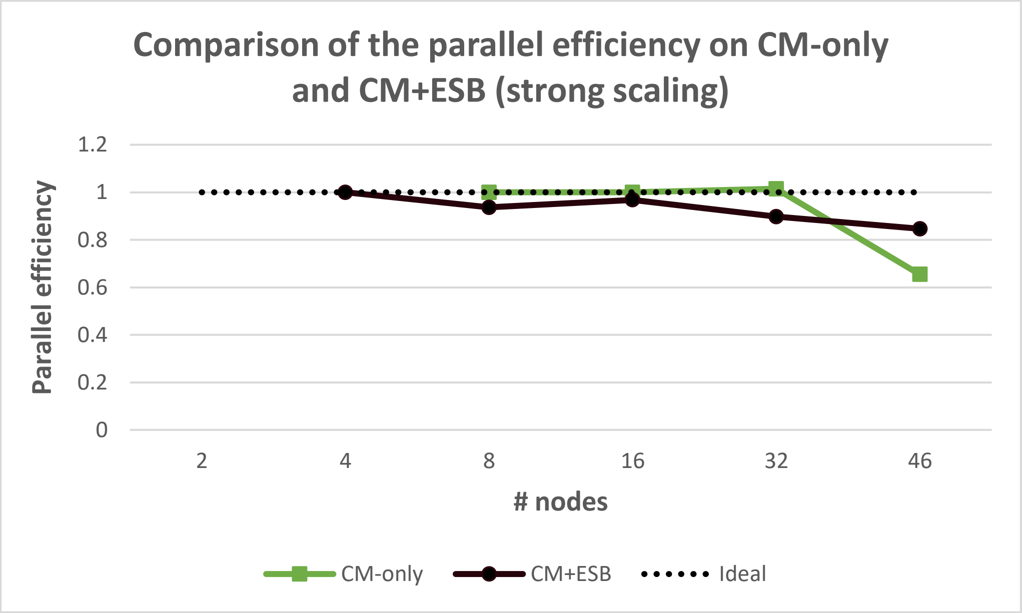
- Best mapping on MSA depends on the problem size and aims at optimizing the computational load
  - <10^4 particles: only on Cluster (CPU)
  - ~ 10^5 particles: Booster or DAM (Data Analytics Module)
  - >10^6 particles (large macromolecules): pair interactions on GPU, run PME on CPUs
  - Very large volume (>10^6 nm^3): Replace PME with FMM (Fast Multipole Method) running on ESB



GROMACS: multi-module usage in MD simulations

- Best mapping on MSA depends on the problem size and aims at optimizing the computational load
  - <10^4 particles: only on Cluster (CPU)
  - ~ 10^5 particles: Booster or DAM (Data Analytics Module)
  - >10^6 particles (large macromolecules): pair interactions on GPU, run PME on CPUs
  - **Very large volume (>10^6 nm^3): Replace PME with FMM running on ESB**

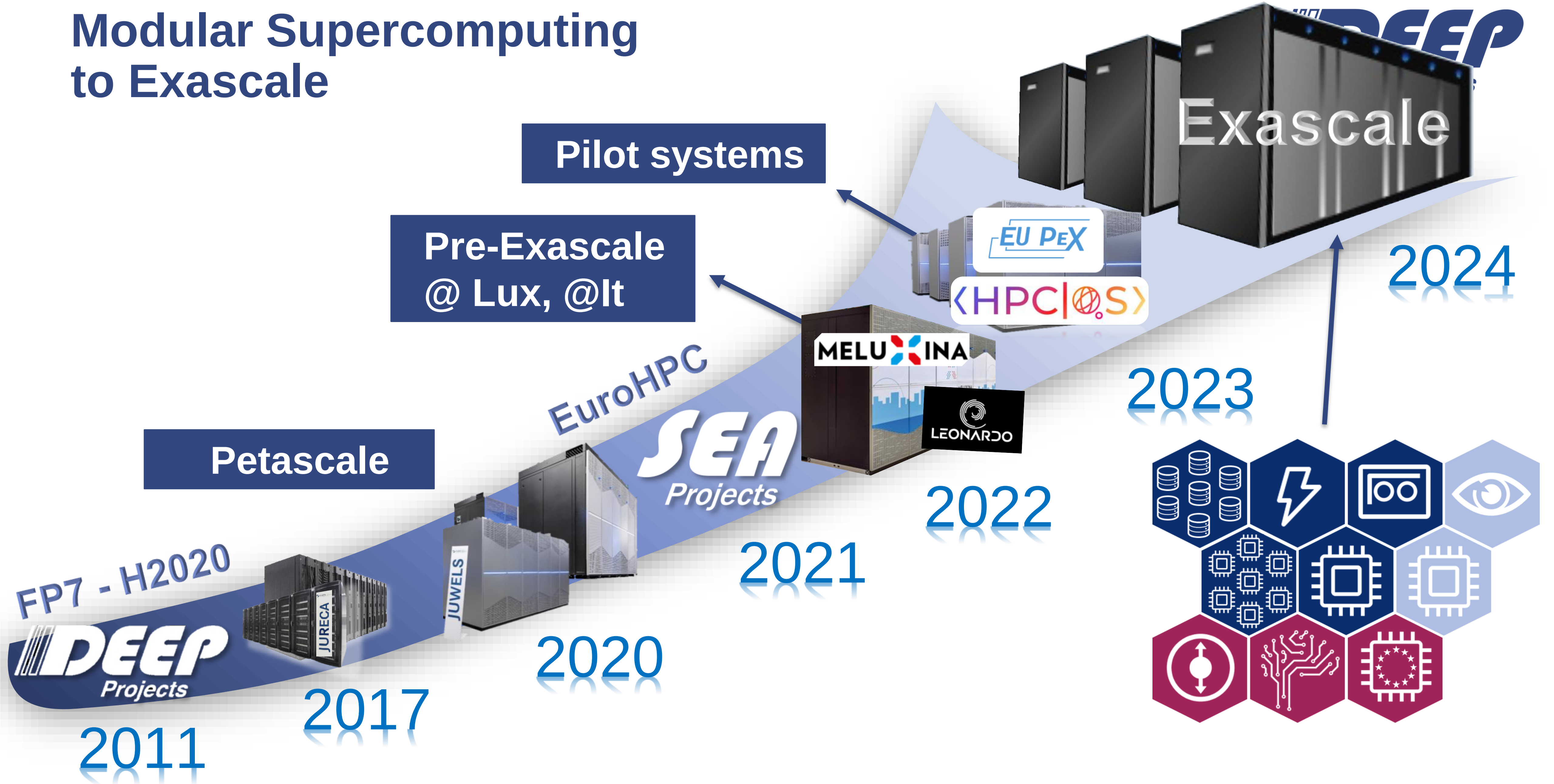




Conclusions

- **The Modular Supercomputing Architecture (MSA)**
  - Orchestrates heterogeneity at system level
  - Serves very diverse application profiles
    - *Maximum flexibility for users, without taking anything away (still can use individual modules)*
- **Distribute applications on the MSA give each code-part a suitable hardware**
  - Straight-forward implementation for workflows
  - Partition at MPI-level interesting for multi-physics / multi-scale codes
  - Monolithic codes do not need to be divided
- **Current / Upcoming implementations of MSA**
  - DEEP system, JURECA, JUWELS
  - MELUXINA (Luxembourg EuroHPC Petascale system)
  - EUPEX and HPCQS pilots
  - … Exascale !

Modular Supercomputing to Exascale



Petascale

Pre-Exascale @ Lux, @It

Pilot systems

Exascale

FP7 - H2020

EuroHPC

2011

2017

2020

2021

2022

2023

2024

Funding Acknowledgement

DEEP Projects

EuroHPC Joint Undertaking

SPONSORED BY THE

Federal Ministry of Education and Research

fwo

bpifrance

ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΥΠΟΥΡΓΕΙΟ
ΑΝΑΠΤΥΞΗΣ ΚΑΙ ΕΠΕΝΔΥΣΕΩΝ

ΓΓΕΚ
ΓΕΝΙΚΗ ΓΡΑΜΜΑΤΕΙΑ
ΕΡΕΥΝΑΣ ΚΑΙ ΚΑΙΝΟΤΟΜΙΑΣ

MINISTERIO
DE CIENCIA
E INNOVACIÓN

Financiado por
la Unión Europea
NextGenerationEU

Plan de Recuperación,
Transformación y
Resiliencia

AGENCIA
ESTATAL DE
INVESTIGACIÓN

Swedish
Research
Council

The DEEP Projects have received funding from the European Commission's FP7, H2020, and EuroHPC JU Programmes, under Grant Agreements n° 287530, 610476, 754304, and 955606. The DEEP-SEA project receives also support from Belgium, France, Germany, Greece, Spain, Sweden, Switzerland, and United Kingdom

www.deep-projects.eu

@DEEPprojects