ESD Roundtable at European HPC Summit Week 2017
Input for the DEEP projects (DEEP, DEEP-ER, DEEP-EST)

Project Highlights

- **Architecture:**
  - ***Cluster-Booster architecture:*** The DEEP project has introduced the heterogeneous Cluster-Booster architecture that combines a general purpose "Cluster" with a highly scalable "Booster", that is a Cluster of Many-Core Accelerators.
  - **Multi-level memory hierarchy**: The DEEP-ER project has extended the Cluster-Booster concept by a multi-level memory hierarchy, which constitutes the basis for the project's comprehensive I/O and resiliency software stack.
  - ***Modular Supercomputer Architecture***: The DEEP-EST project willA generalise Cluster-Booster architecture to multiple compute modules with distinct characteristics to fit the needs of a diverse application portfolio. In particular, within DEEP-EST the Cluster and the Booster will be complemented by a Data Analytics Module, a system tailored to the needs of HPDA applications.
- **Innovative memory technologies:**
  - **Non-volatile memory** (NVM) integrated in a multi-level memory hierarchy. In DEEP-ER local NVMe storage is used to accelerate I/O and increase the scalability of the file system (BeeGFS) by using it as local cache.
  - **Network attached memory (NAM):** two prototypes built and demonstrated in DEEP-ER, using FPGA and a Hybrid Memory Cube (HMC). The HMC controller software has been released as Open Source, as it is planned to be done for the libNAM management software. In DEEP-EST this concept will be further developed in two directions: one focusing in more memory and computing capacity (which keeps the naming NAM), and one focusing in high-speed access to accelerate MPI collective operations (**Global Collective Engine**).
- **Efficient bridging network technologies**: DEEP implemented the "**Cluster-Booster protocol**" to seamlessly bridge between EXTOLL and InfiniBand protocols. In DEEP-EST equivalent bridging is planned for the network technologies chosen for different modules in the system.
- **Orchestration and dynamic scheduling of heterogeneous resources:**
- The DEEP-EST project will extend the Open Source software **SLURM** to support the heterogeneity of the Modular Supercomputing Architecture, enabling applications to use resources in all modules of the system, and maximising the overall use of the available hardware.
- **Software stack**
  - DEEP optimised **ParaStation MPI** to allow spawning MPI processes from Cluster to Booster and vice versa, so that large parts of an application (containing internal MPI-communication) can be efficiently offloaded from one side of the system to the other.
  - The **OmpSs** programming model has been extended to leave such application offloads to the runtime, by annotating the relevant tasks with pragmas.
- **I/O and resiliency:**
  - **Optimised BeeGFS:** development of a local-cache that increases the I/O performance and the scalability of the file system.
  - **SIONlib** improves the performance of applications performing task-local I/O.
  - **Exascale 10** improves the performance of applications using MPI-I/O
  - **Application-based checkpoint restart**: the Scalable Checkpoint/Restart Library (SCR, open source) has been extended to support the multi-level memory hierarchy of DEEP-ER. In combination with SIONlib

- **Co-design methodology**: demonstration of a new HPC architecture with hardware, software and application developments driven through intense co-design between all stakeholders.

Technology (HW/SW/methodology) for inclusion in an ESD project

The DEEP projects family provides very innovative individual technologies and also a global concept to efficiently integrate them into an overall HPC system. Details on several of the project technologies are given under “project highlights”. Here just a bullet list of some components is given

- **Modular Supercomputing Architecture** able to efficiently integrate highly innovative technologies fitting application needs
- **EXTOLL network**
- **Network Attached Memory** to perform global operations directly at the network.
- **Network bridging**
- **I/O stack based on BeeGFS, SIONlib and Exascale 10**
- **Resiliency features** for application-based checkpoint and task-based failure recovery
- **Co-design** as methodology reaching from hardware, through middleware/system ware to tools to applications

How would this technology be sued /integrated

Within a Modular Supercomputing System, with possible integration of further technologies that bring added value to the concept and EsD system itself.

Are there any pre- or co-requisite items

-

Any extra work / interaction (on top of the current project roadmap) needed to make them ready?

None

What information / actions are needed to best prepare for EsD projects

Overview of technologies and ideas interested to be integrated in a Modular Supercomputer; what would be their added value to the concept and EsD.