A Breakthrough in Non-Volatile Memory Technology
IT needs to accelerate time-to-market

**Situation:**
- End users and applications need instant access to data to progress faster and create better business outcomes

**Challenge:**
- Today most IT infrastructures use separate data storage in addition to the main computer systems
- Conventional storage systems can’t keep up with the speed of DRAM memory
- I/O operations increasingly impact application performance
- For I/O-intensive applications a system is only as fast as the storage, regardless of memory buildout
- In-memory computing can address this issue to an extent, but it is unproven and traditional memory is volatile

**Solution:**
- Servers with non-volatile memory, which provides speed and data persistence
Next Generation I/O Solutions for Exascale
EU Horizon 2020 NEXTGenIO Project

- Remove I/O bottlenecks through exploitation of non-volatile memory (NVRAM) technologies
- Bridge the gap between memory (fast access/small capacity) and storage (slow access/large capacity)
- Create NEXTGenIO hardware platform and validate NVRAM applicability for high performance and data-intensive computing
- Develop system software for HPC applications to exploit the NEXTGenIO platform NVRAM features. ECMWF runs a time-critical operational Weather Forecasting system that features a highly intensive HPC IO workload. As such, ECMWF is exploring the usage of NVRAM technology within HPC environments to tackle the ever-growing demands for high-density, high-contention IO systems.
- Validate the performance benefits of NVRAM using a mix of HPC workloads
- The entire project was driven by a strong co-design philosophy by the following partners:

Coordinator: epcc, FUJITSU, Intel
Hardware technology: FUJITSU, Intel
System software: BSC, epcc, FUJITSU
Software tools: ECMWF, arm
Applications: ECMWF, epcc, ARCturus, DLR, TUD, BSC, Intel

The NEXTGenIO project has received funding from the European Union’s Horizon 2020 Research and Innovation program under Grant Agreement no. 671951
A key output of NEXTGenIO will be a prototype system based on the new NVRAM technology

Fujitsu is responsible for the hardware prototype using Intel® Optane™ DC persistent memory technology and system software (e.g. BIOS, drivers, …)

Prototype system will be used to explore use of NVRAM technology for I/O intensive high-performance scientific computing

Development team and production located at Augsburg, Germany
Growing Need for New Class of Memory
Data Analytics, AI & Deep Learning

- Enormous amount of data to be processed
- Data/compute locality is important
- Keep data (large data sets) in fast memory close to compute for transfer and reinforcement learning
- Multiple users can share the same data
- Build in resilience thanks to NVRAM persistence
Growing Need for New Class of Memory
Engineering Apps & Computational Steering

- Simulations are extremely compute intensive. Engineers have no insight what is happening during simulation e.g. of a car in a wind tunnel (OpenFOAM®).
- Check what happens WHILE a simulation is running, to tweak and adjust designs on-the-fly.
- Typical use cases in car/aerospace design, such as OpenFOAM®, Star CCM+, Fluent (any engineering application with visualization and steering function would benefit).
- Workflow: keeps all data in NVRAM and avoids need for I/O to permanent storage.
Growing Need for New Class of Memory
Traditional & High-end HPC

- Typical use cases are weather and climate modelling which are very I/O intensive
- Weather modelling: precise forecasts are calculated in short cycles (e.g. 4× a day) producing enormous amounts of data
- I/O write (and read back) operations not required if data stays in NVRAM → significant reduction of runtime
- Additionally, streamed data (from different sources, e.g. mobile phones, sensors) can be used on the cluster / node WHILE the data is being processed
How it Works
NVRAM Technology Significantly Changes Memory and Storage Hierarchies

- NVRAM: Non-Volatile RAM
- In NEXTGenIO: Intel® Optane™ DC Persistent Memory
- With up to 512GB, significantly larger capacity than DRAM
- Hosted in the DIMM slots, controlled by a standard CPU memory controller
- Comparable performance to DRAM; significantly faster than PCIe-attached SSDs
# Software Stack

## NVDIMM Support in NEXTGenIO

<table>
<thead>
<tr>
<th>iRMC</th>
<th>DCPMM FW</th>
<th>BIOS</th>
<th>OS</th>
<th>DCIM</th>
<th>PMDK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Out-of-Band profiles/redfish</td>
<td>NVDIMM Firmware</td>
<td>Platform settings, ACPI Rev. ≥ 6.2 e.g. NFIT Table</td>
<td>Linux CentOS 7.5</td>
<td>ipmct (CLI, API, Mon) Manage device e.g. regions, FW</td>
<td>ndctl Manage namespaces</td>
</tr>
</tbody>
</table>

## Applications

**Unmodified Applications**
- High Capacity and affordable Memory
- High-Speed classic Storage paradigm

**Modified Applications**
- New storage paradigm In-Memory, Big-Data, HPC, etc.

## Tools & Libraries

- **SLURM** Data & Energy Aware Job Scheduler
- **iPMC** (CLI, API, Mon) Manage device e.g. regions, FW
- **ndctl** Manage namespaces
- **PMDK** Persistent memory libraries
- **PMDK** Persistent memory libraries
- **NDctl** Manage namespaces
- **IPMctl** Manage device e.g. regions, FW

## Operating Systems

- **OS** Linux CentOS 7.5

## Systemware

- **iRMC** Out-of-Band profiles/redfish
- **DCPMM FW** NVDIMM Firmware
- **BIOS** Platform settings, ACPI Rev. ≥ 6.2 e.g. NFIT Table

## Usage Model

- **Intel® Optane™ DC**
- **FS DAX** File-system with DAX option
- **DEVDAX** Usage pattern
- **SLURM** Data & Energy Aware Job Scheduler
- **PMDK** Persistent memory libraries

## Hardware

- **Huge Memory Regions** Hardware managed, DRAM as cache
- **High Speed Storage** gekkoFS, echofs, DAOS, ext4, fs without DAX option
- **High Speed Storage** File-system with DAX option

## Impact

- **High Capacity and affordable Memory**
- **High-Speed classic Storage paradigm**
- **New storage paradigm In-Memory, Big-Data, HPC, etc.**
Persistent Memory Introduction

What is persistent memory?
- Persistent memory modules based on Intel® Optane™ DC persistent memory
- Breakthrough in non-volatile memory for servers using future Intel® Xeon® Scalable CPUs
- Memory-like performance, byte-addressable
- Accessed through load/store instructions

Why does it matter?
- Adds a new tier between DRAM and block storage
- Larger capacity, higher endurance, consistent low latency
- Just a few instructions instead of ten thousands of instructions to get data persistent stored
- Application can direct read/write from/to Intel® Optane™ DC persistent memory without use of I/O links
How fast is Fast Enough
Goods for Production, Data for the CPU

container ship grouped shipment

Express train grouped shipment

high-bay storage
Goods can be taken individually

Manufacturing

Shanghai

Hamburg

Berlin

~3.000-10.000 Kilometer.
~80-300 Kilometer.
100-300 Meter

~3-10 msec.
~80-300 µsec.
100-300 nsec.

HDD Block Device

NVMe SSD Block Device

*source Windmöller & Hölscher
Why is NVRAM so Much Faster?

1. CPU prepare I/O in Main Memory (Buffer)
2. Send write command to I/O Controller
3. DMA transfer from Main Memory to I/O Controller
4. DMA Transfer from I/O Controller to HDD
5. HDD confirms data persistence
6. I/O Controller uses interrupt to inform CPU about data persistence

Depending on the importance of data persistence e.g. Database the CPU can not go forward in the application until persistence is committed.
RAID can avoid single point of failure for storage media
A 2nd persistence confirmation from a remote storage might be required for HA purpose.
System Software

- The system software stack (developed in NEXTGenIO) sits between the hardware and the user applications, exposes the infrastructure to the user level and enables direct interaction
  - Job scheduler (SLURM)
  - Object store as alternatives to file systems
    - DAOS (Distributed Application Object Storage)
    - dataClay
  - Multi-node NVRAM file system
    - echoFS & GekkoFS (a collaboration with JGU university)

Key goal: Platform must be usable “as is” for legacy applications
Application object store as alternative to file systems: **dataClay**

- Persistent data can be stored using the same data model used by the application in memory.

**dataClay** is a platform that:
- enables storing and computing data in persistent memory
- facilities the location of the required data by an application after a failure
- facilities the sharing of persistent data by several applications using the same memory data model

Multi-node NVRAM file system: **echoFS & GekkoFS** (collaboration with Johannes Gutenberg University Mainz)

- Leverages NVM providing a local and distributed filesystem for legacy applications.
- Allows automatizing stage-in and stage-out using the job-scheduler to provide new scheduling and coordination options.
- Interfaces with applications using a POSIX interface with some features disabled to increase performance
Access Through User-Level Filesystem

- Allow legacy applications to transparently benefit from new storage layers
- Make new layers readily available to applications
- Hide I/O stack complexity from applications
- Construct collaborative burst buffer assigned to a batch job scheduler
NEXTGenIO Prototype Rack

Login / Management Node
- Xeon Platinum 8260M
  - 24C 2.4GHz 165W
- 6x 16GB RAM
- 2x 256GB NVDIMM
- 1x 1GE (Mgmt)
- 1x 10GE
- 1x Omni-Path
- 1x NVIDIA Quadro M4000 (FH, 120W)
- 3x SATA-SSD ~2TB

Boot Node
- Xeon® Platinum 8260M
  - 24C 2.4GHz 165W
- 3x 16GB RAM
- 1x 1GE (Mgmt)
- 1x 10GE
- 1x NVIDIA Quadro M4000

Compute Node
- Xeon® Platinum 8260M
  - 24C 2.4GHz 165W
- 12x 16GB RAM
- 12x 256GB NVDIMM
- 2x 1GE (1x Mgmt / 1x LAN)
- 2x Omni-Path
- Remote Boot / (no internal boot device HDD / SSD / M.2)

Gateway Node
- Xeon® Platinum 8260M
  - 24C 2.4GHz 165W
- 12x 16 GB RAM
- 2x 1GE (1x Mgmt / 1x LAN)
- 2x Omni-Path
- 2x 2-ch IB-FDR (LP)

Rack / 42 HU populated
- CN: 32 x 1 = 32
- L/M: 2 x 1 = 2
- B/N: 2 x 1 = 2
- 1/10GE: 2 x 1 = 2
- OPA: 2 x 1 = 2
- GW: 2 x 1 = 2

NVM up to ...
32x12x1MIOPs = 384MIOPs @ 1/1µs (r/w)
32x12x0.25TB = 96TB ➔ 12MIOPs, 3TB per Node
shaping tomorrow with you