How can applications benefit from NVRAM technology?

Evaluation methodology and testing

Dr Juan Rodriguez Herrera
EPCC - The University of Edinburgh

Glasgow Systems Seminar
Wednesday, 31st January 2018

Outline

- What is NEXTGenIO?
  - Hardware
  - Software
- Evaluation methodology
  - Objectives, applications, scenarios
  - Profiling tools
  - OpenFOAM, CASTEP, MONC, etc.
  - Best practices
- Ongoing work

What is NEXTGenIO?

- *Next Generation I/O for the Exascale*
- Addressing the I/O bottleneck of HPC workloads through exploitation of NVRAM technologies
- Aim: bridging the gap between memory and storage
  - Memory: fast read/writes – small capacity
  - Storage: slow read/writes – large capacity

What is NEXTGenIO?

- EC H2020 project
- 36-month duration
- 8.1 million € (50% hardware)
- 8 partners, covering:
  - Hardware
  - HPC centres and uses
  - Software
  - Tools development

nextgenio

epcc

THE UNIVERSITY OF EDINBURGH

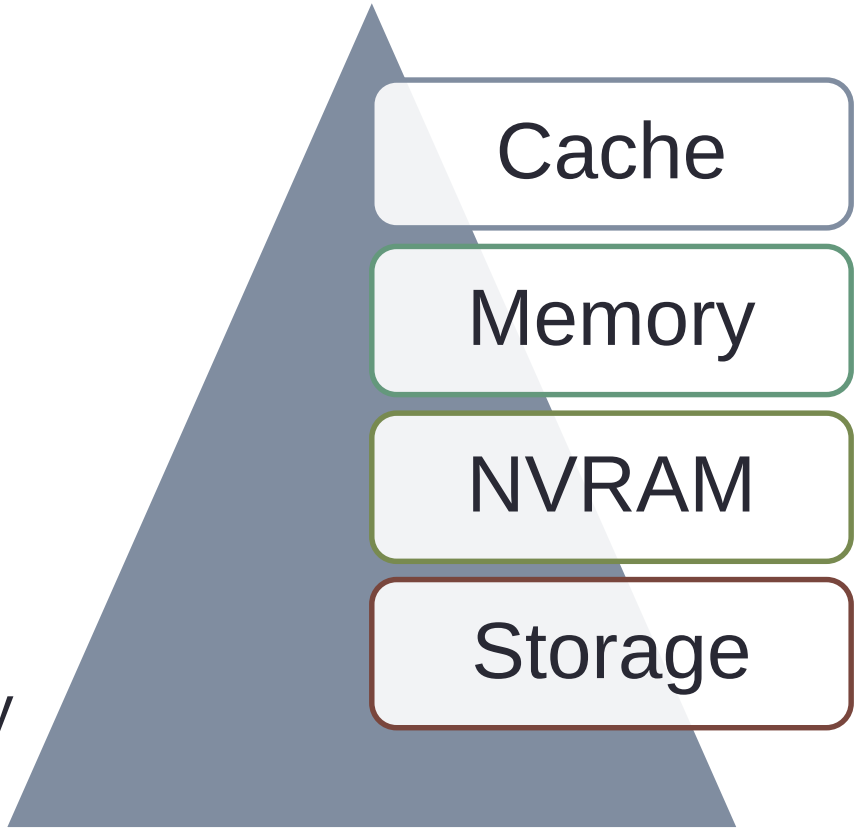
NEXTGenIO objectives

- Hardware platform prototype to investigate **applicability** for high performance and data-intensive computing
- Understand how best to **exploit NVRAM**
- Develop the necessary **systemware software** to enable (Exascale) application execution on the hardware platform
  - Systemware SW must understand extra level present in the memory hierarchy
- Study application **I/O profiles** and **I/O workloads**
  - How different I/O behaviour and scheduling policies will impact job throughput

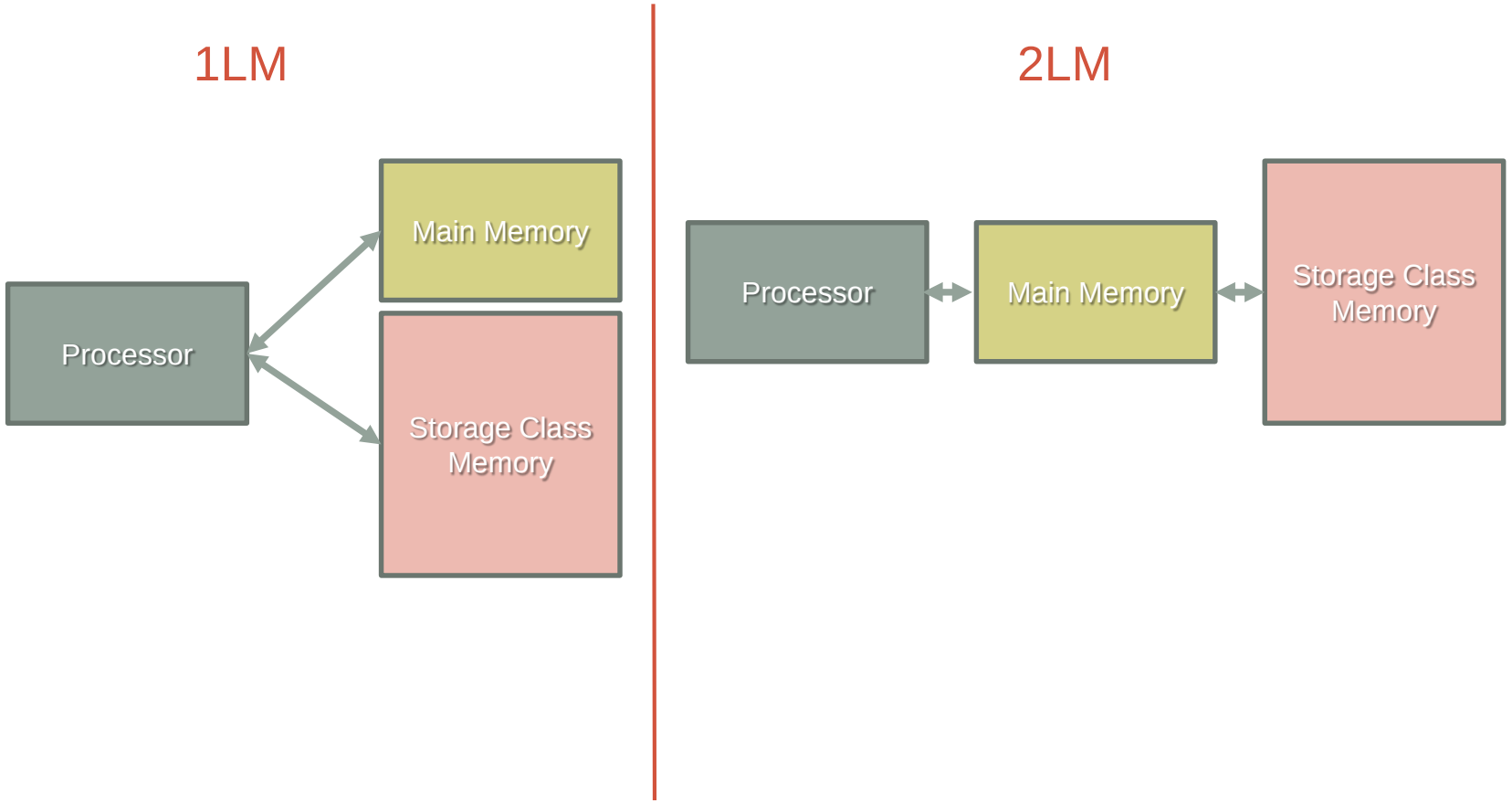
Hardware: NVRAM

Use of NVRAM:
Non-Volatile RAM

- 3D XPoint technology
- Much **larger capacity** than DRAM
  - Hosted in the DRAM slots, controlled by a standard memory controller
- **Slower than DRAM** by a small factor, but significantly **faster than SSDs**



NVRAM modes of operation



Software

- Systemware software
  - SLURM – job scheduler
  - Data scheduler
  - DAOS and dataClay – object stores as alternatives to file systems
  - echoFS – multi-node NVRAM file system
- Profiling tools
  - ARM Map
  - ScoreP / Vampir
- Applications

Evaluation methodology: objectives

- Define and maintain a **suite of applications and testcases** that will be used to evaluate the NEXTGenIO technology
- Carry out systematic tests and **evaluation** as technology results become available
- Facilitate co-design by providing clear and constructive **feedback** to the technology work packages
- Clearly document the **benefits** of the NEXTGenIO technology, indicate its impact and sketch future lines of development

epcc

Evaluation methodology: applications

Combination of traditional and novel HPC applications

- CASTEP: chemistry
- MONC: cloud modelling
- Halvade: genome sequencing
- OSPRay: ray-tracing
- OpenFOAM: CFD
- IFS: weather forecasting
- K-means: machine learning
- Tiramisu: deep learning

Evaluation methodology: scenarios

1. Baseline measurements in today's systems:
   - Use of ARCHER (Cray XC30)
   - Use of ECMWF cluster for IFS
   - Use of Marenostrum for BSC applications
2. Measurements on the NEXTGenIO platform without NVRAM.
3. Measurements on the NEXTGenIO platform with NVRAM.

Profiling tools

- Two profiling tools:
  - Map (ARM)
  - ScoreP / Vampir (TUD)
- Feedback has been provided to the developers on features that would be useful towards debugging and performance analysis in the prototype

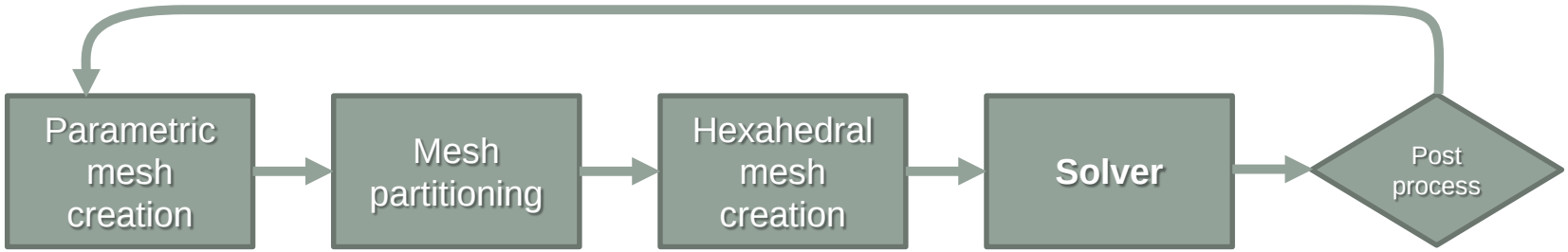
Profiling tools: features

- Ability to see the activity timeline of a specific rank
- Ability to distinguish I/O to disk and I/O to NVRAM
- Reporting memory usage of NVRAMs
- Both tools will extract memory usage information in 1LM and 2LM modes
- Reporting background I/O transfer between disk and NVRAM (echoFS)

OpenFOAM

OpenFOAM

- Solves CFD problems on arbitrary unstructured finite element meshes
- Important code for industrial (in particular SME) use
- C++ code of 1 million lines, using MPI parallelism
- I/O handling
  - Creates separate directory per MPI process and output time step
  - Stores mesh and field information
  - 4 096 processes and 5 outputs: 20 480 directories & 307 200 files
  - Not efficient for parallel filesystems

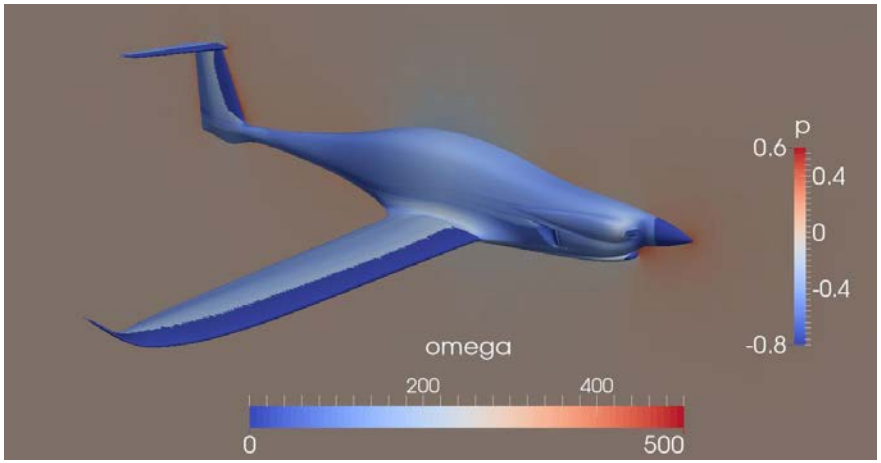


nextgenio

epcc

THE UNIVERSITY OF EDINBURGH

OpenFOAM – Evaluation on ARCHER



Output every # of timesteps	Simulated time 0.03 s	Simulated time 0.99 s
1000	15.75 GB	519.75 GB
100	157.5 GB	5.1975 TB

- Pipistrel light aircraft 3D airflow
- Mesh decomposed in 7 x 8 x 4
- pisoFOAM solver
- Run on 10 nodes (224 MPI ranks), 3,000 timesteps
- Output written every 1,000 timesteps
- 5.25 GB of data per output timestep
- Real use case runs up a 5TB data volume

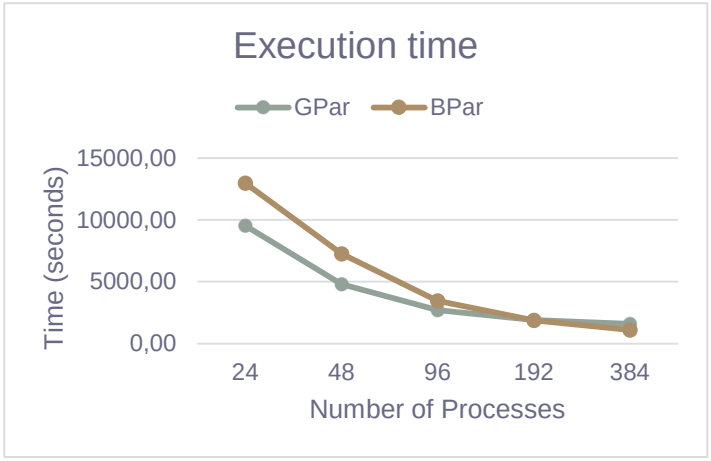
NEXTGenIO benefits for OpenFOAM

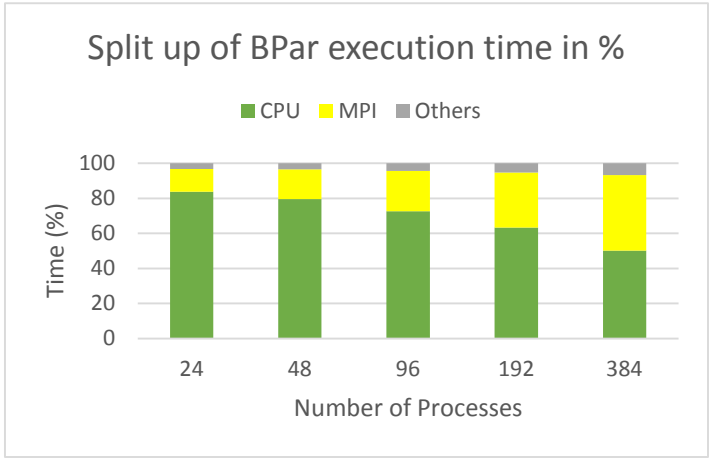
- 1LM mode
  - Use local SCM for output data – higher output rates
  - Use SCM to pass data between workflow steps (and to post-processing application) – reduce permanent FS I/O load and time between steps
- Overall benefits
  - Increased write frequency: more precise information for post processing
  - Improved strong scaling: shorter time to solution
  - Improved weak scaling: can run larger problems
  - Reduced time to completion: get solutions faster

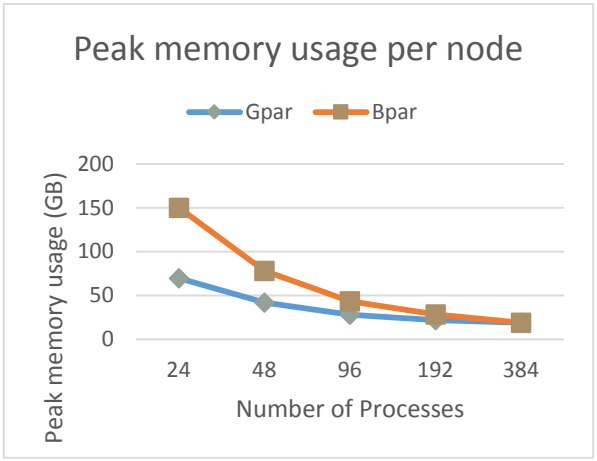
CASTEP

- Ab-initio density functional theory (DFT) application developed by a group of UK physics experts
- Describes electronic states (**bands**) of a material using a plane-wave vector (**g-vector**) basis at different points (**k-points**)
- Code is written in Fortran 95, uses MPI & OpenMP
- Compute and communication hotspots
  - Orthogonalisation (matrix operations) and FFTs
  - All-to-all communication
- Application uses a lot of DRAM
  - Typically not able to use all cores in a node
  - Wave functions are recomputed, since they cannot be stored in DRAM

CASTEP – Evaluation on ARCHER







- Crambin testcase (1 k-point)
- Each process has 2 OpenMP threads
- Significant and growing MPI overhead
- Band & g-vector parallelisation scales better, yet uses even more memory
- I/O behaviour: regular writes of 7.5 GB

NEXTGenIO benefits for CASTEP

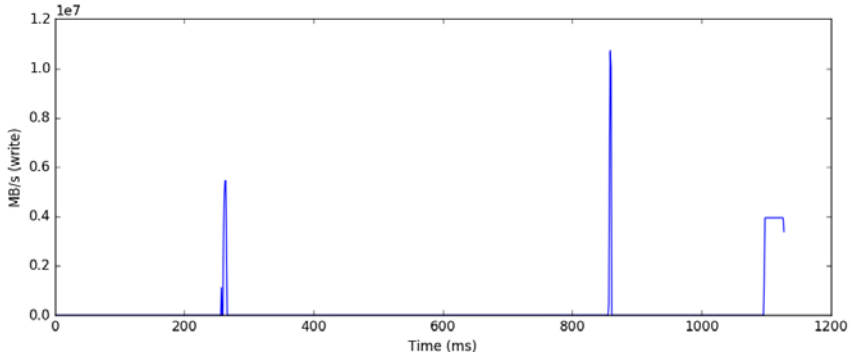
- Use SCM as application memory (2LM mode)
  - Much larger memory space available
  - Can run larger problems on given system
  - Run more processes per node and reduce MPI collective overhead
  - Achieved performance will depend on access patterns vs. memory-side caching policies
- Store output data (checkpoints) in local FS on SCM (1LM mode)
  - Significant reduction of I/O time, less energy use
  - Faster time to solution
- Store computed wave functions in local SCM (1LM or 2 LM mode)
  - Significant reduction of computation
  - Faster time to solution, less energy use

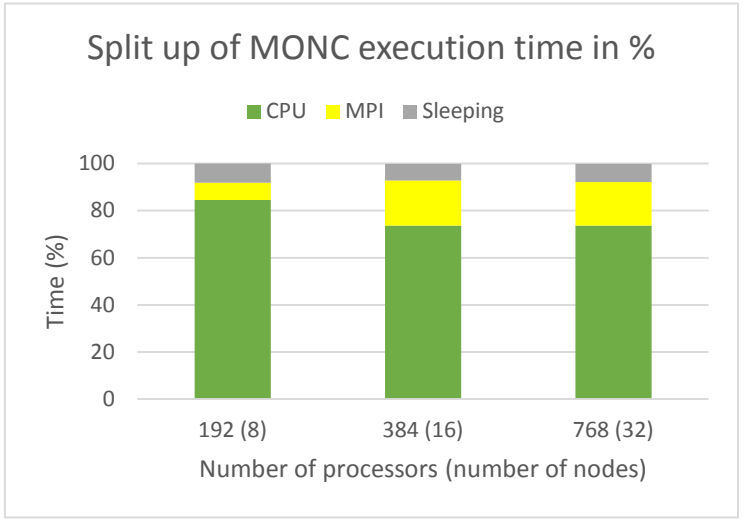
MONC

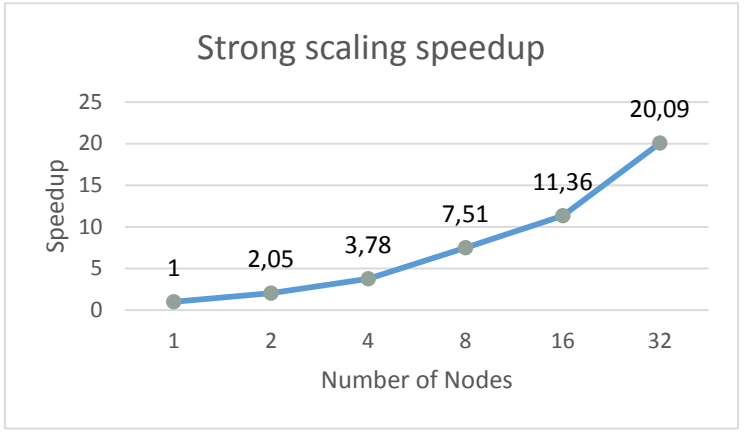
- Very high resolution (~2 to 50 meters), flexible and portable cloud modeling framework
- UK Met Office and EPCC are collaborating to develop MONC
- Fortran 2003 code using MPI for parallelism, about 50K lines, modular architecture
- I/O handling:
  - Code uses the NetCDF libraries and data format
  - Distinct I/O server processes, ratio to compute processes configurable
  - Compute processes send raw data to I/O servers at dynamic intervals
  - I/O servers process raw data and write at configured intervals
  - I/O servers are both communication & I/O bound

MONC – Evaluation on ARCHER

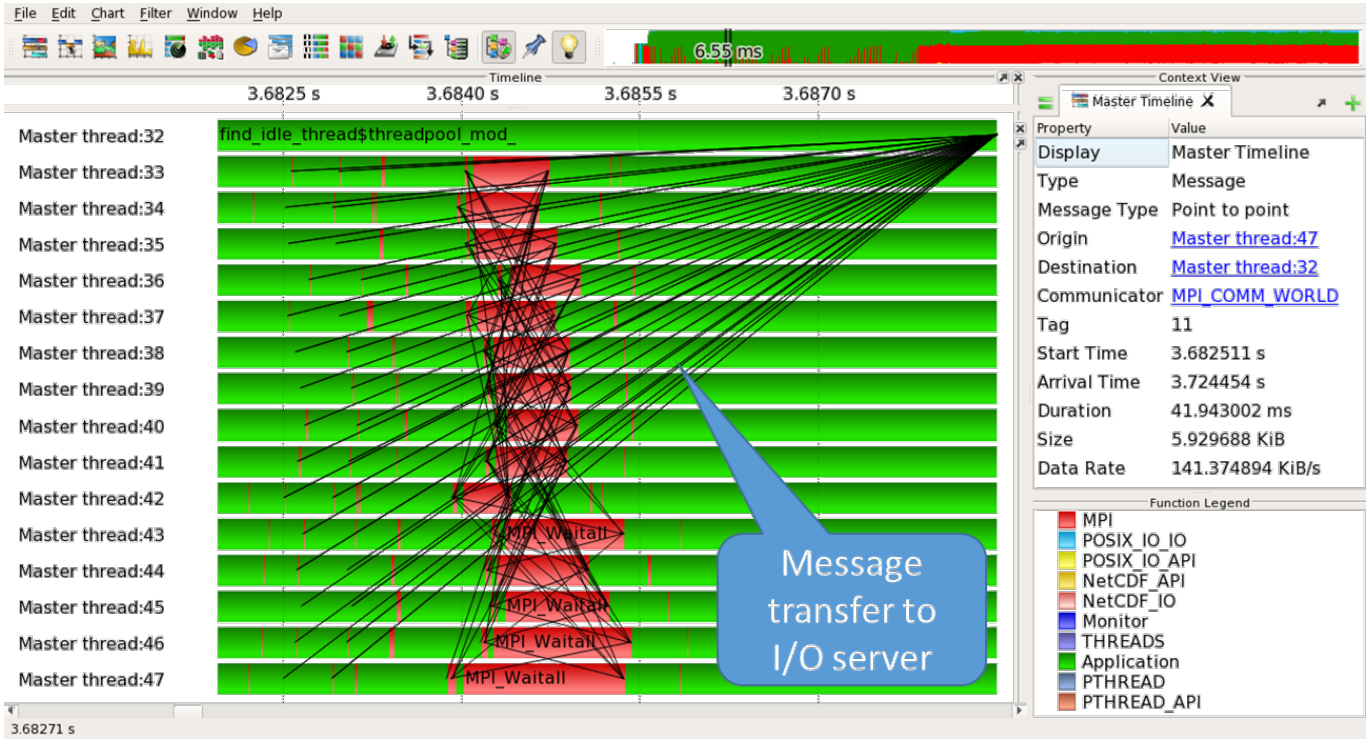
- Stratocumulus test dataset
- 22 compute and 2 I/O processes per node
- Regular writes of about 2 GBytes of data
- Significant amount of time spent in MPI communication
- All-to-All and All-reduce operations are most significant here



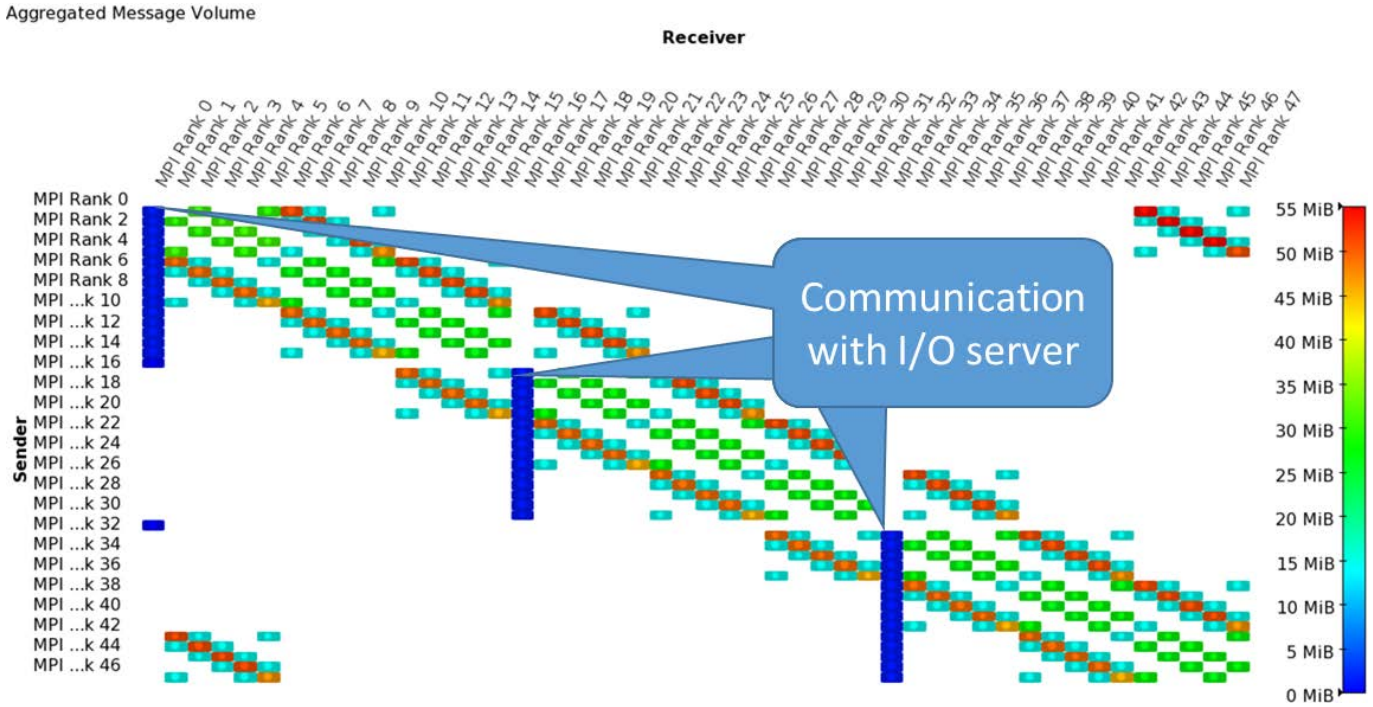




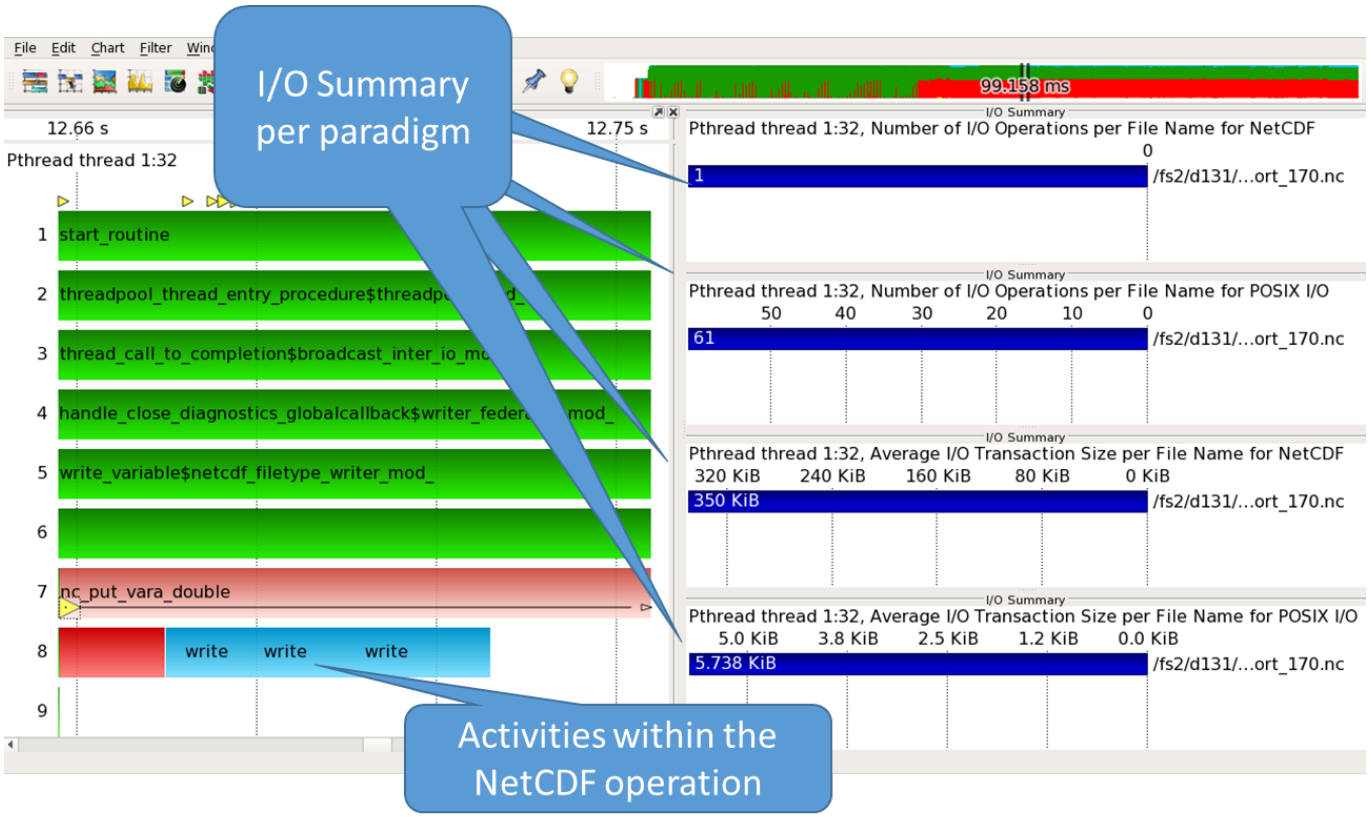
MONC I/O profile



MONC I/O profile



MONC I/O profile



monc

Score-P
Scalable performance measurement infrastructure for parallel codes

VAMPIR

nextgenio

epcc

THE UNIVERSITY OF EDINBURGH

NEXTGenIO benefits for MONC (1/2)

- Rewrite I/O server or replace it (XIOS) to stage results of data analysis in SCM and effect asynchronous transfer to disk
  - Can reduce I/O times and overlap with data processing
  - Can handle larger results data than use case 1 and tackle larger problems plus resolve results better
  - Additional improvements in scaling compared to use case 1 due to maximum overlap

NEXTGenIO benefits for MONC (2/2)

- I/O server stores data analysis results in SCM and visualization step (OSPRay) picks them up – 1LM mode
  - Fast data transfer and no I/O load on PFS
  - Post-processing starts & proceeds faster, reducing time to workflow completion
  - Enables concurrent, co-scheduled visualization with minimal impact on computation and system load

Other applications

- Halvade
  - Map Reduce – Hadoop
  - DNA testcase (available online)
- OSPRay
  - Use of OSPRay library to visualise MONC output
- IFS, K-means, Tiramisu.

Best practices

- Save compilation info (use of an internal wiki)
  - Modules loaded for compilation
  - Source code (version used for the annual reports)
  - Makefile (if edited for ARCHER)
- Store execution results (shared folder on ARCHER)
  - Output logs
  - Profile archives

Ongoing work

- Complete baseline measurements for all applications
- Identify larger use cases for experiments for the NEXTGenIO platform
- Work in Progress: keep an eye to www.nextgenio.eu

EOF

Thanks for your attention!
Any questions?

Juan F. R. Herrera
Applications Developer
EPCC – The University of Edinburgh
j.herrera@epcc.ed.ac.uk

The NEXTGenIO project received funding from the EU Horizon 2020 research and innovation programme under grant agreement No 671591.