The DEEP-ER take on I/O

Wolfgang Frings
Jülich Supercomputing Centre

Workshop Exascale I/O: Challenges, Innovations and Solutions
SC16, Salt Lake City
18 November 2016

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 and n° 610476

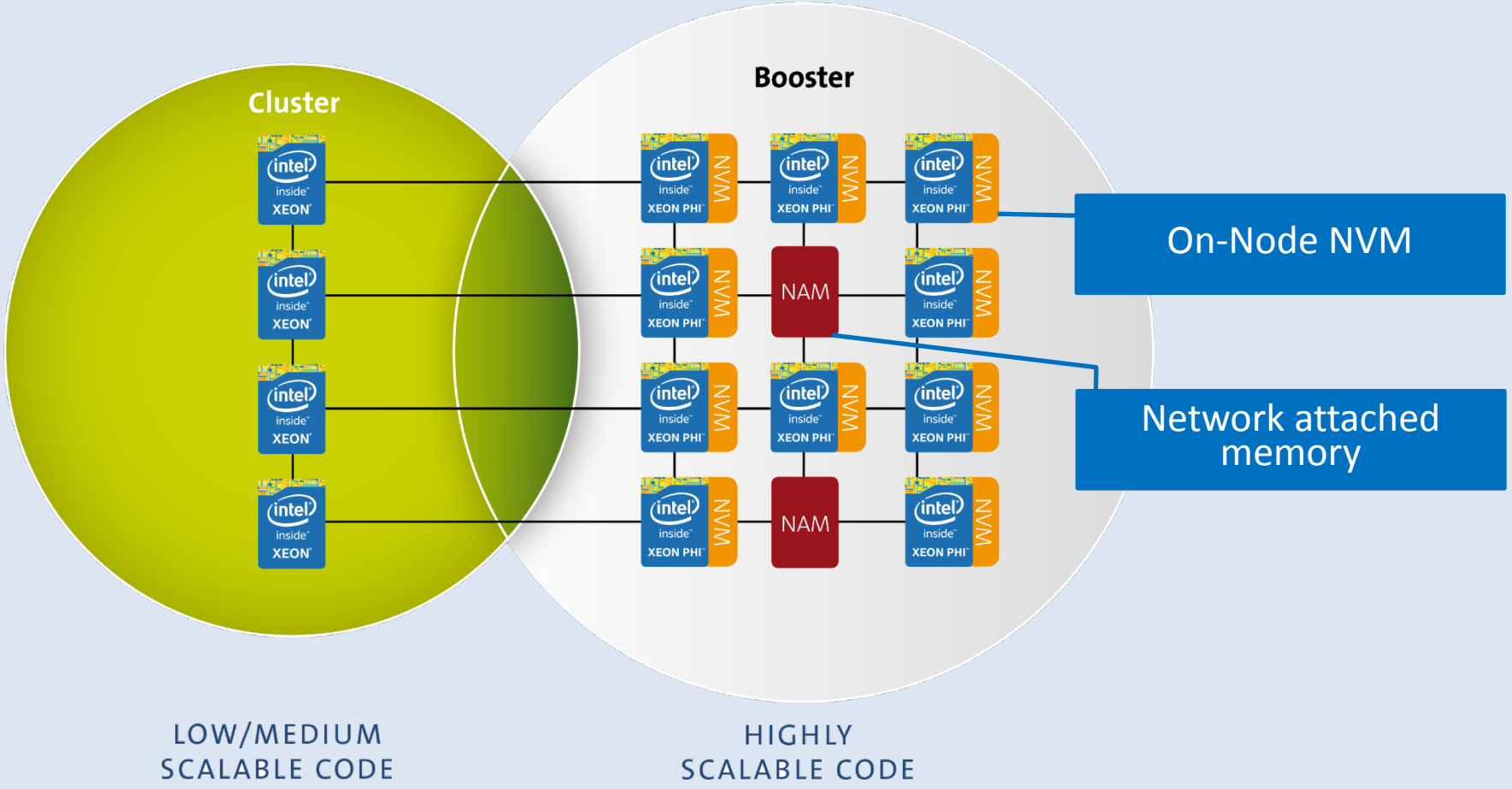
DEEP and DEEP-ER

EU-Exascale projects
20 partners
Total budget: 28,3 M€
EU-funding: 14,5 M€
Nov 2011 – Mar 2017

www.deep-project.eu

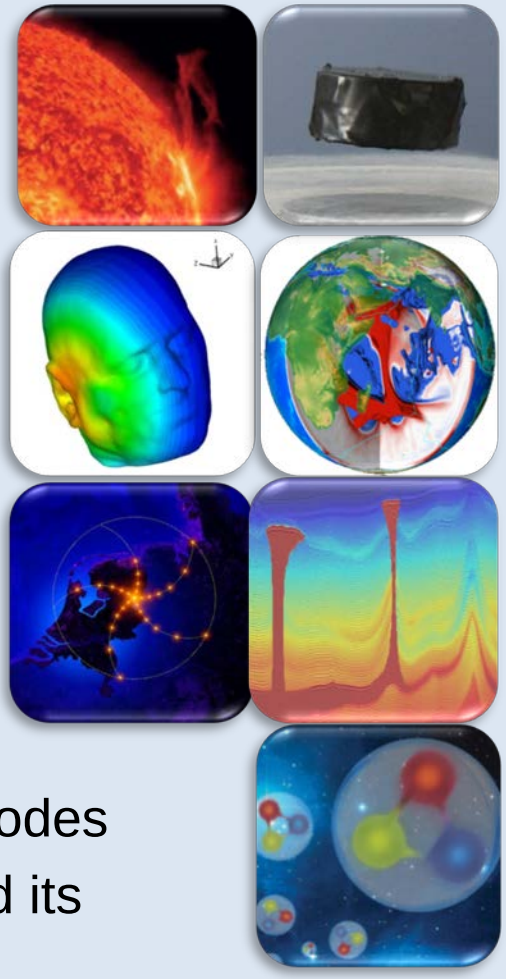
www.deep-er.eu

I/O Hardware Innovation

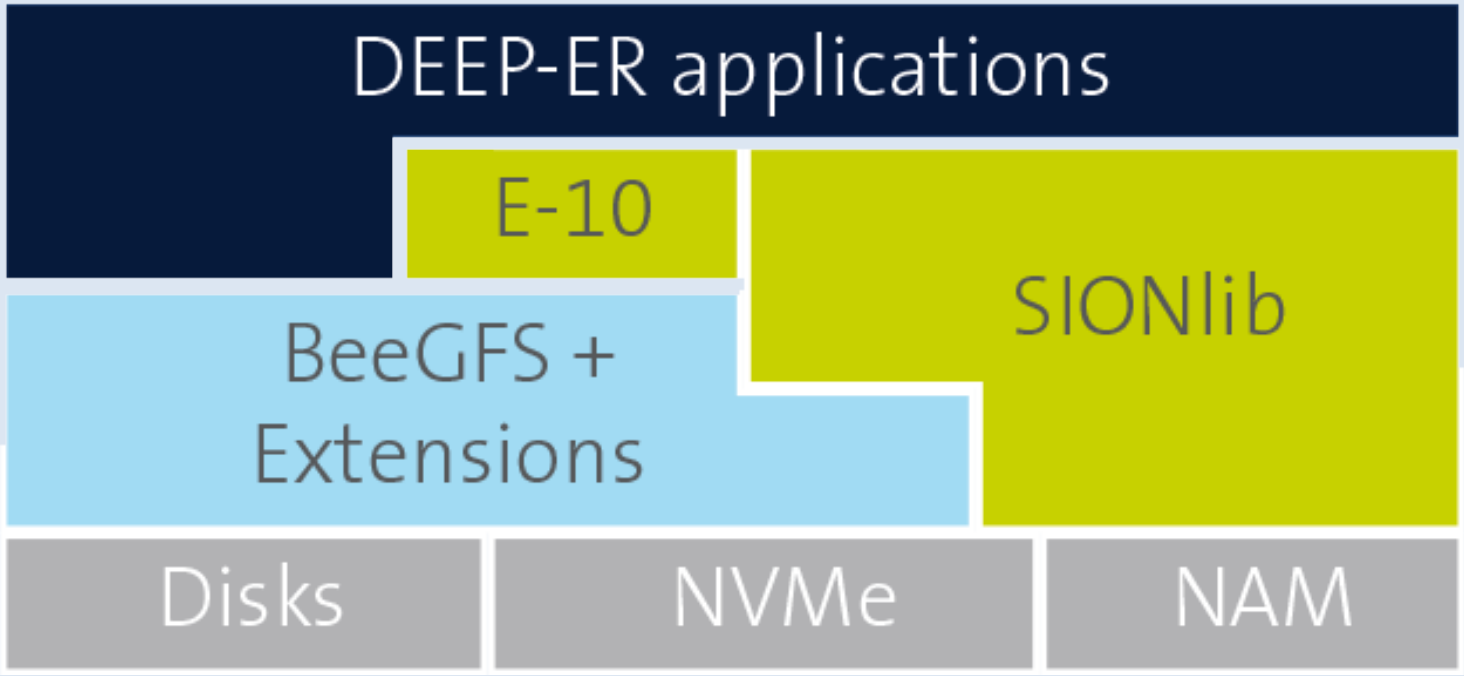


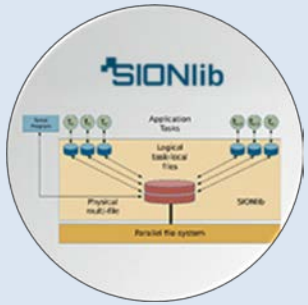
Driven by Applications

- **Pilot applications:**
  - Space weather simulation (KULeuven)
  - High temperature superconductivity (CINECA)
  - Human exposure to electromagnetic fields (Inria)
  - Geoscience (BADW-LRZ)
  - Radio astronomy (ASTRON)
  - Oil exploration (BSC)
  - Lattice QCD (UREG)
- **Goals:**
  - Analyse I/O and resiliency requirements of HPC codes
  - Evaluate DEEP-ER I/O and resiliency concept and its programmability



Scalable I/O



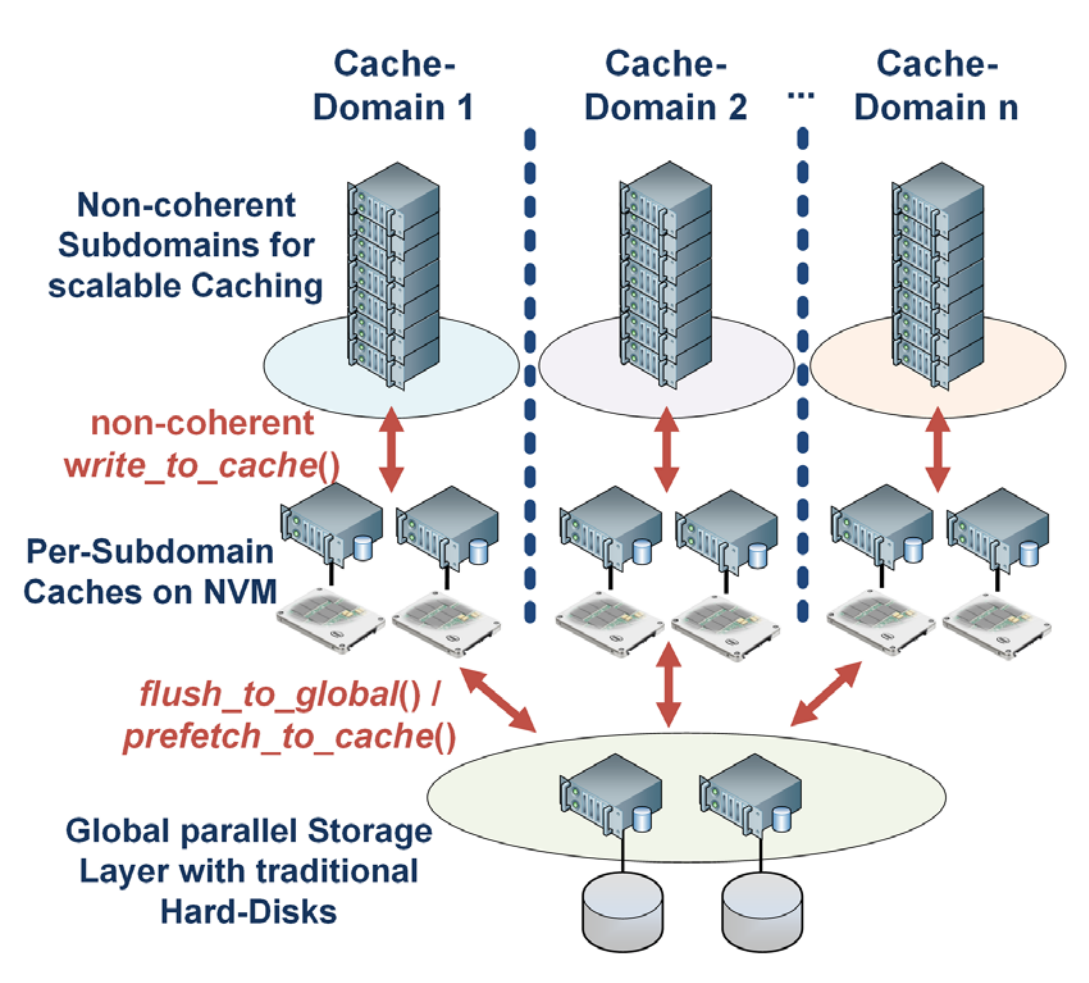


- Improve I/O scalability on all usage-levels
- Used also for checkpointing

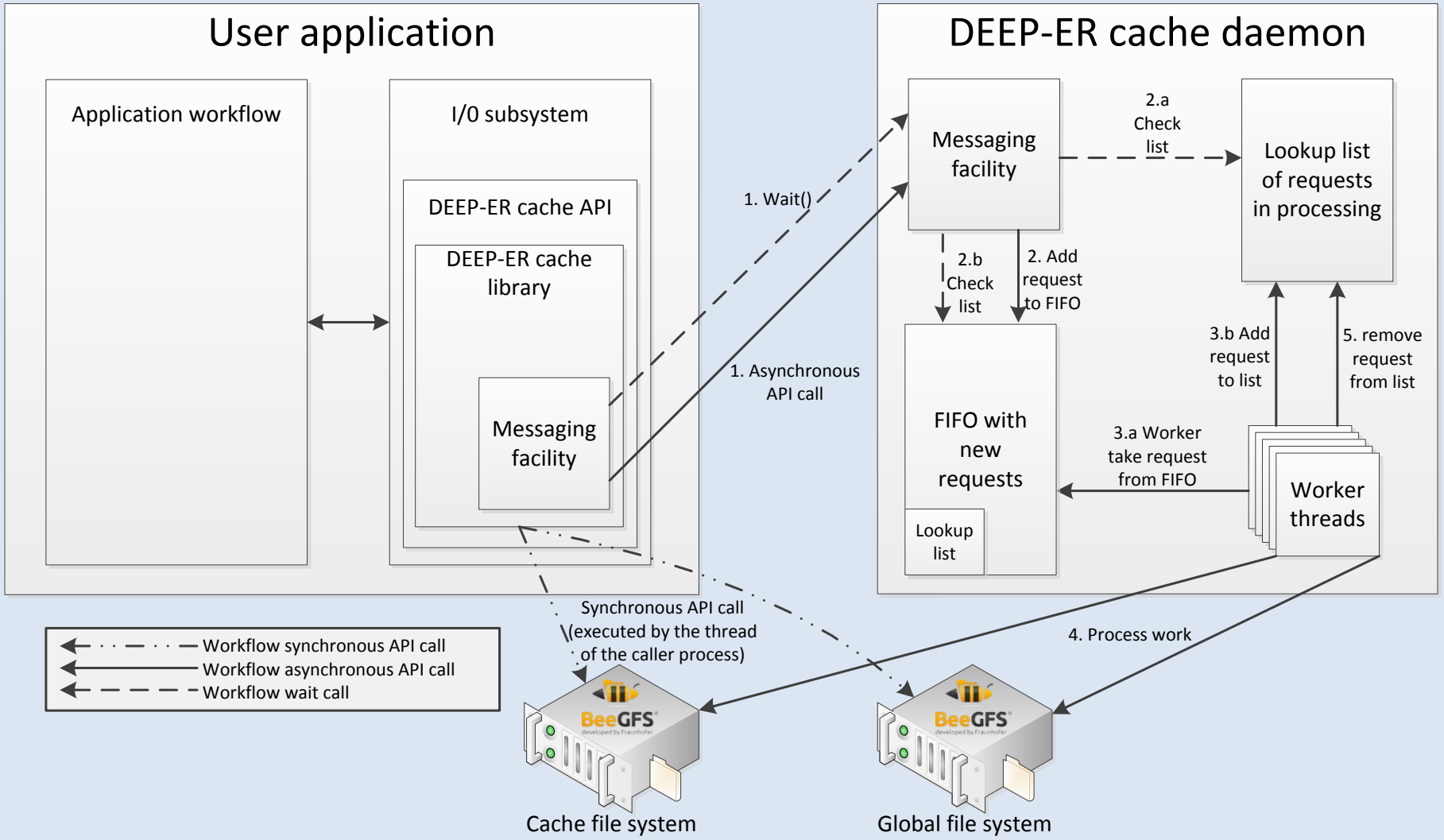
BeeGFS and Caching

- Two instances:
  - Global FS on HDD server
  - Cache FS on NVM at node
- API for cache domain handling
  - Synchronous version
  - Asynchronous version

BeeGFS®
developed by Fraunhofer



Architecture of the asynchronous cache API



DEEP-ER Cache Integration in ROMIO

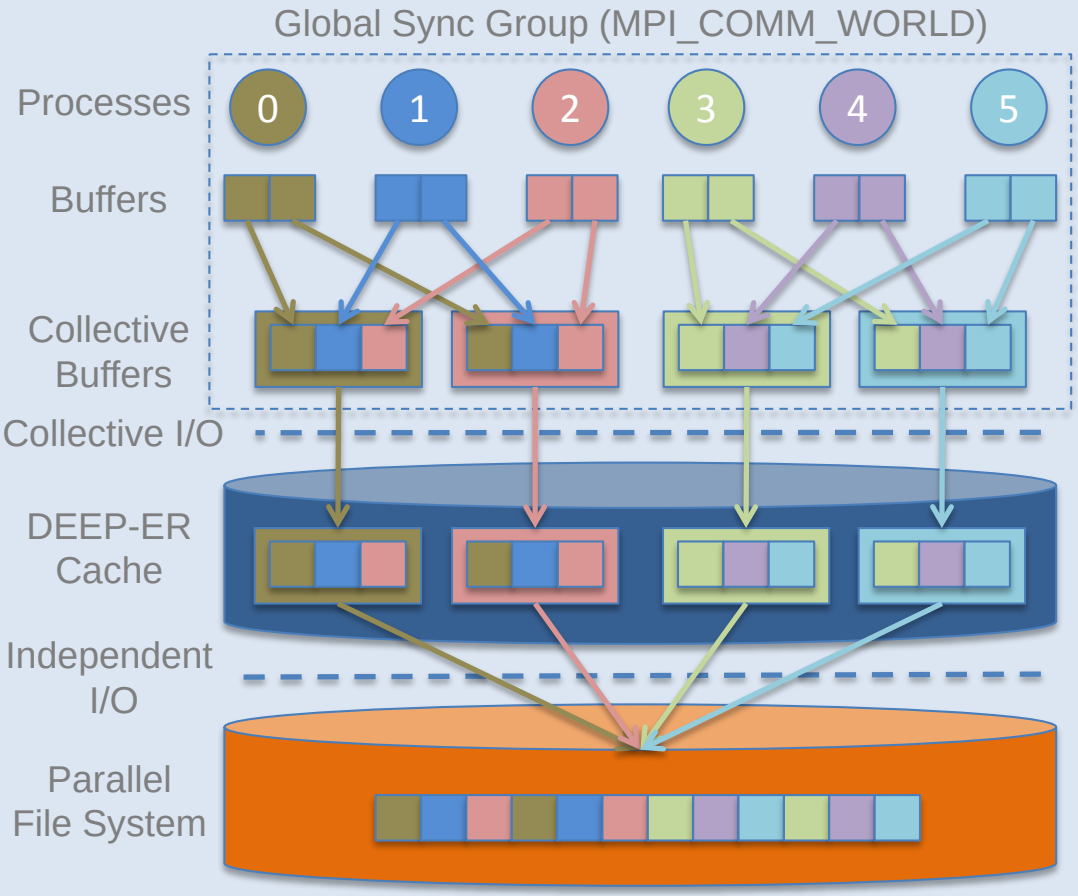
New MPI-IO Hints
e10_cache ★
e10_cache_path ★
e10_cache_flush_flag ★
e10_cache_discard_flag ★
e10_cache_threads ★

ROMIO

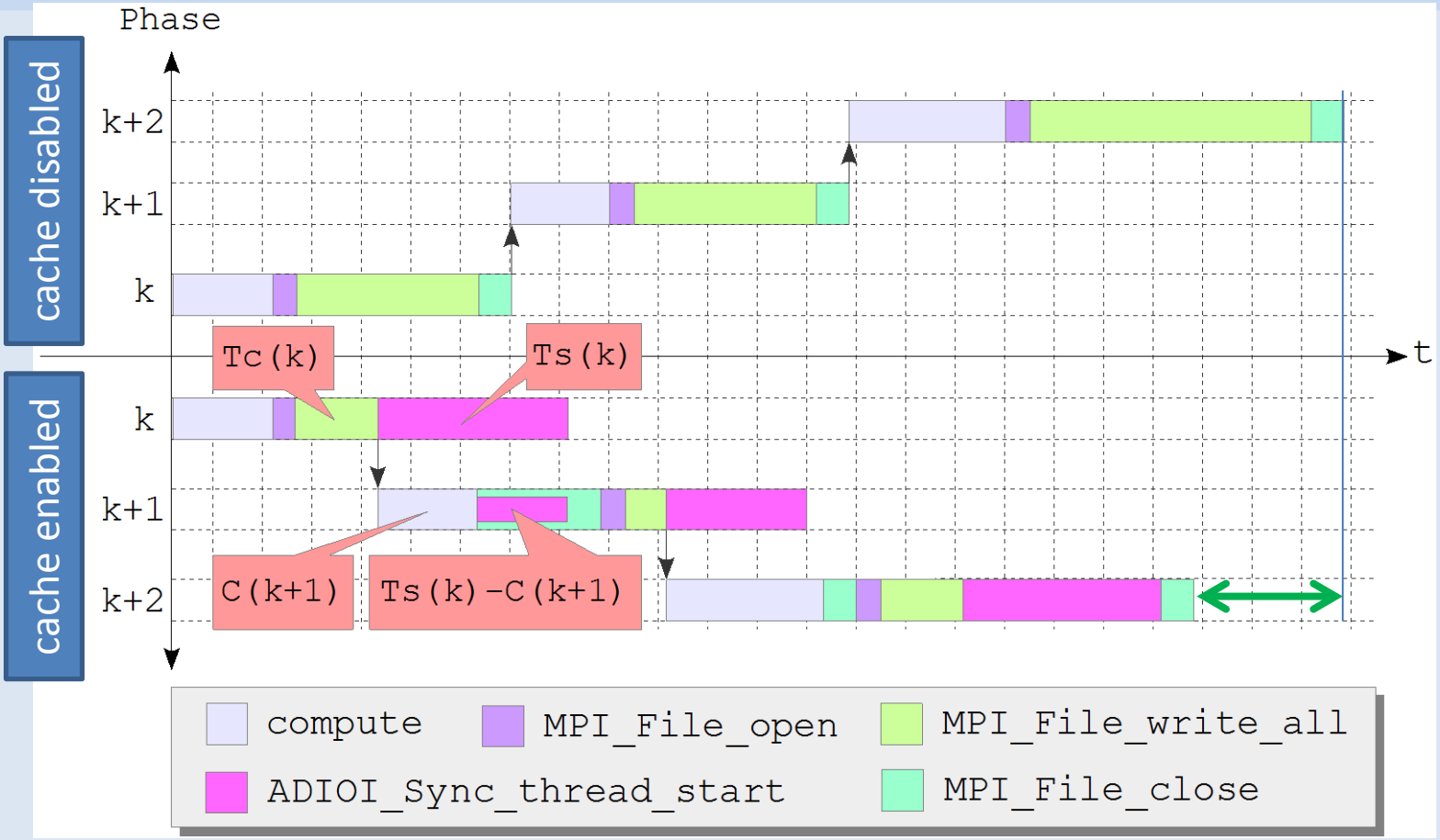
- MPI
- MPI-IO
- ADIO (Abstract Device IO)
  - Lustre Driver
  - GPFS Driver
  - UFS Driver
  - BeeGFS Driver ★
- DEEP-ER Cache ★
- Parallel File System

EXASCALE IO

★Tested in DEEP cluster



Workflows Using DEEP-ER Cache



S(k): amount of data written to the file at phase k (here constant)

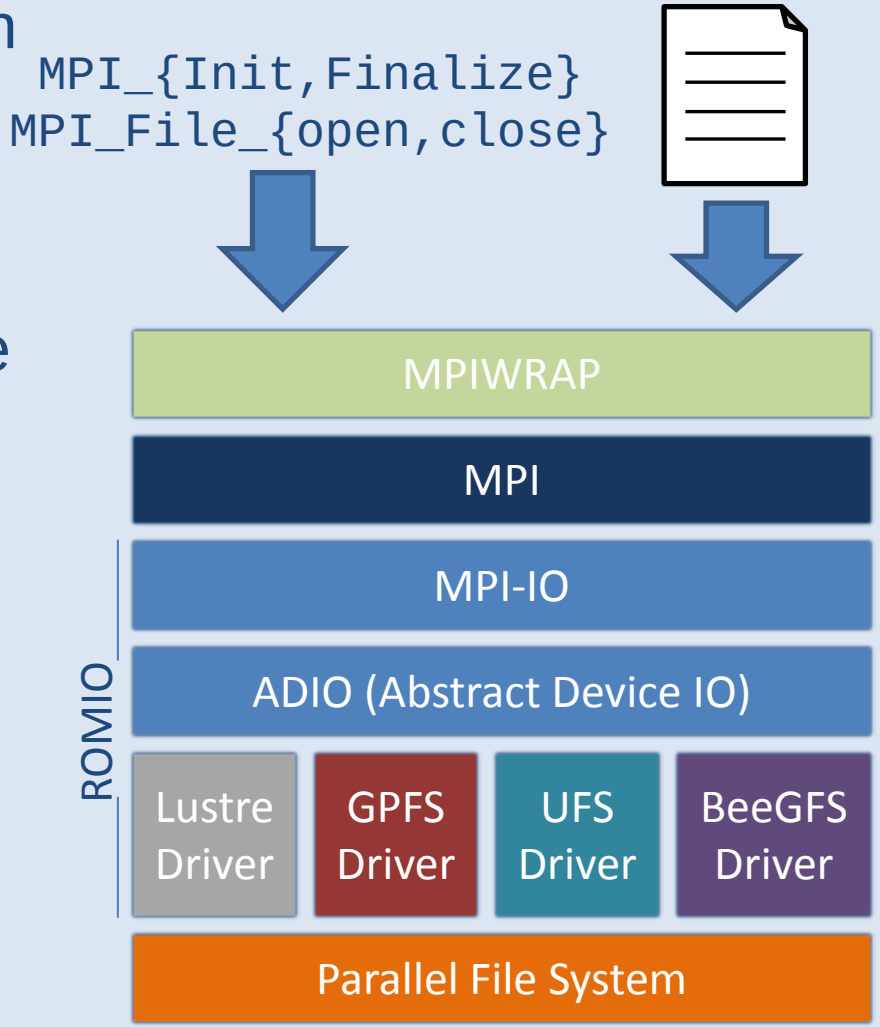
Tc(k): time to write S(k) to the cache

Ts(k): time to sync S(k) with the parallel file system

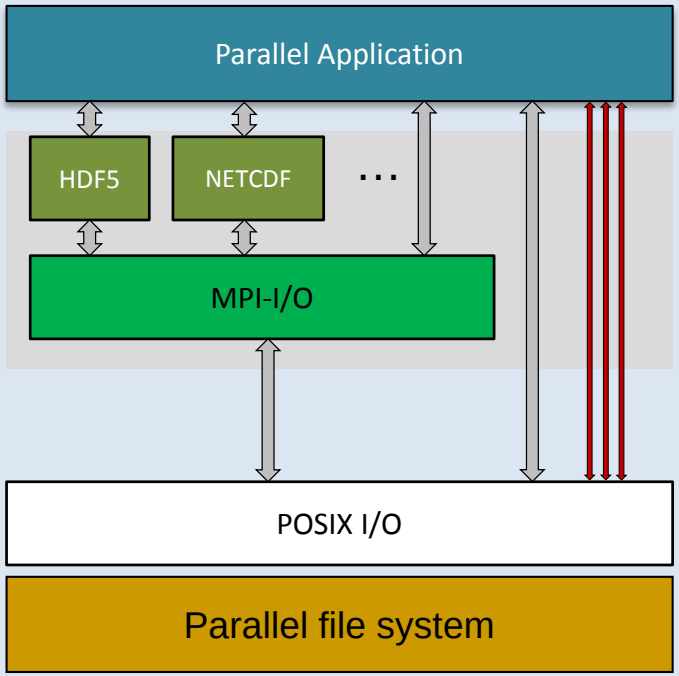
C(k): compute time at phase k

MPIWRAP Support Library

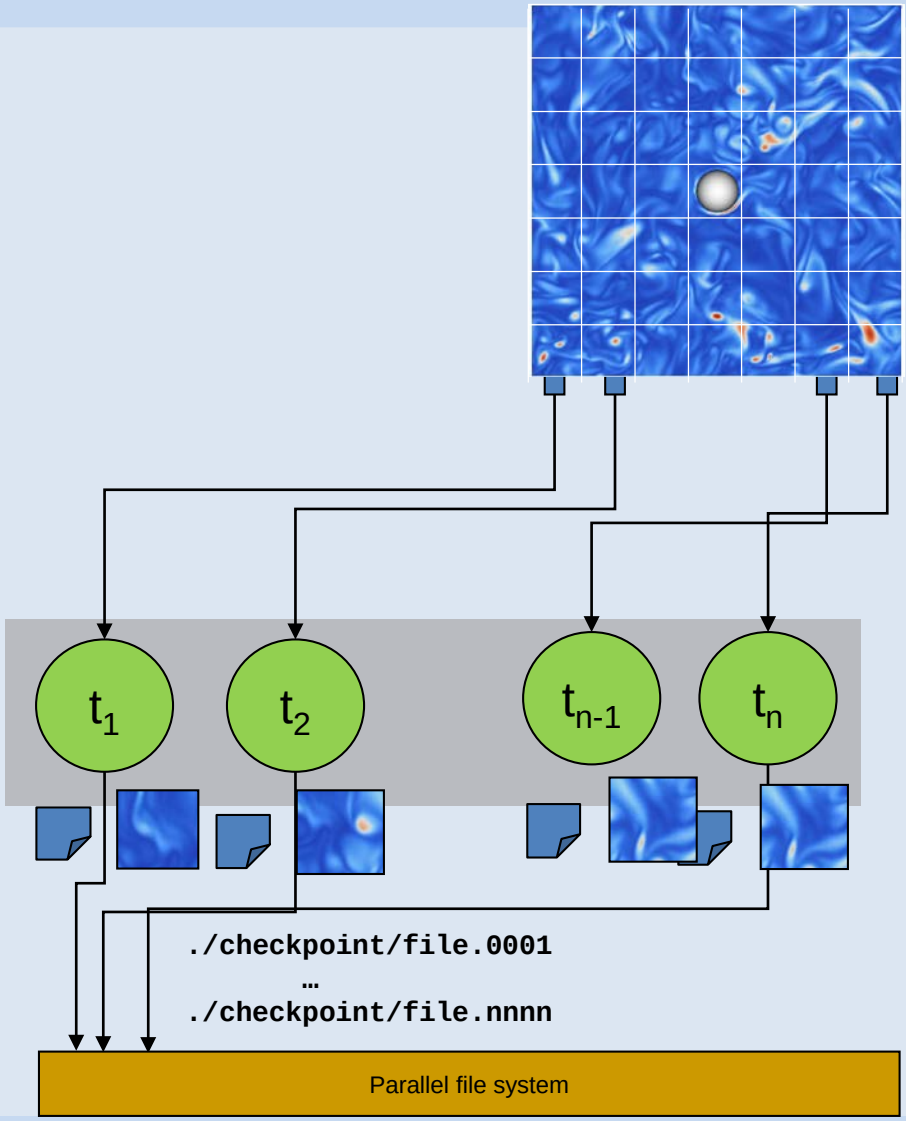
- MPI-IO hints are defined in a config file and injected by libmpiwrap into the middleware
- Provides deeper and more flexible control of MPI-IO functionalities to the users
- Provides transparent integration of E10 functionalities into applications
- Works with any high level library (e.g. pHDF5)



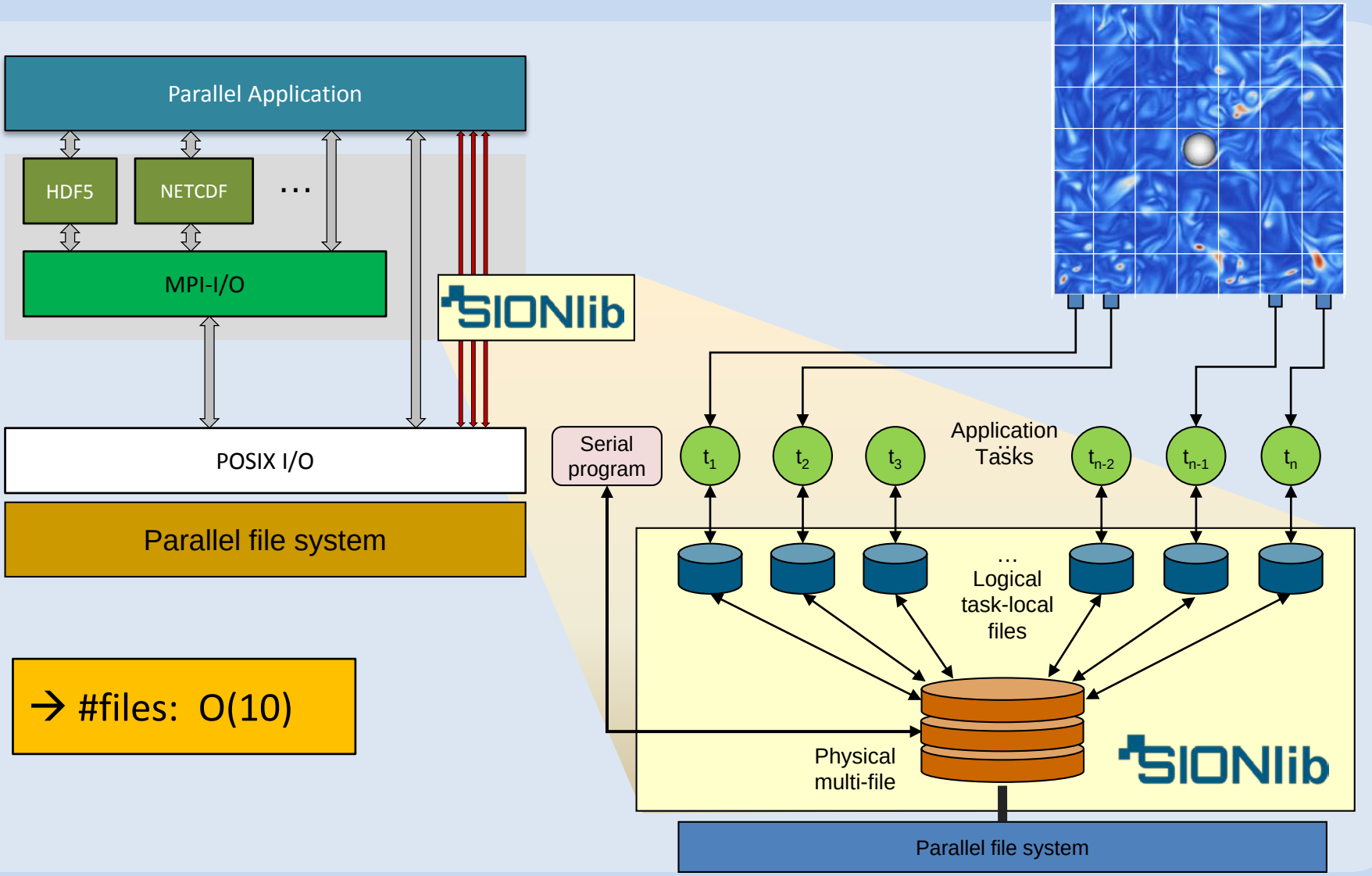
SIONlib: Shared Files for Task-local Data



→ #files: O(10000)

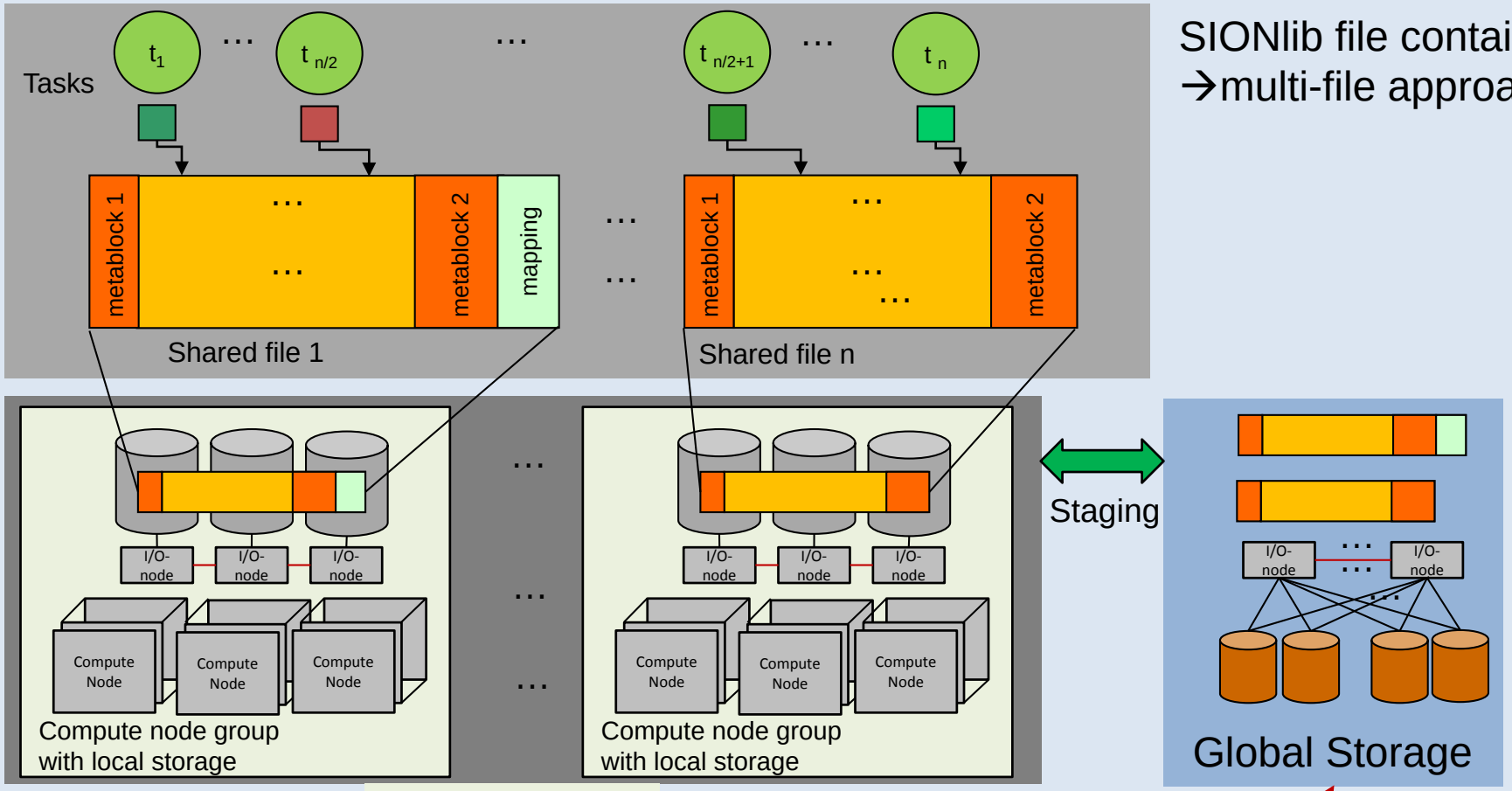


SIONlib: Shared Files for Task-local Data



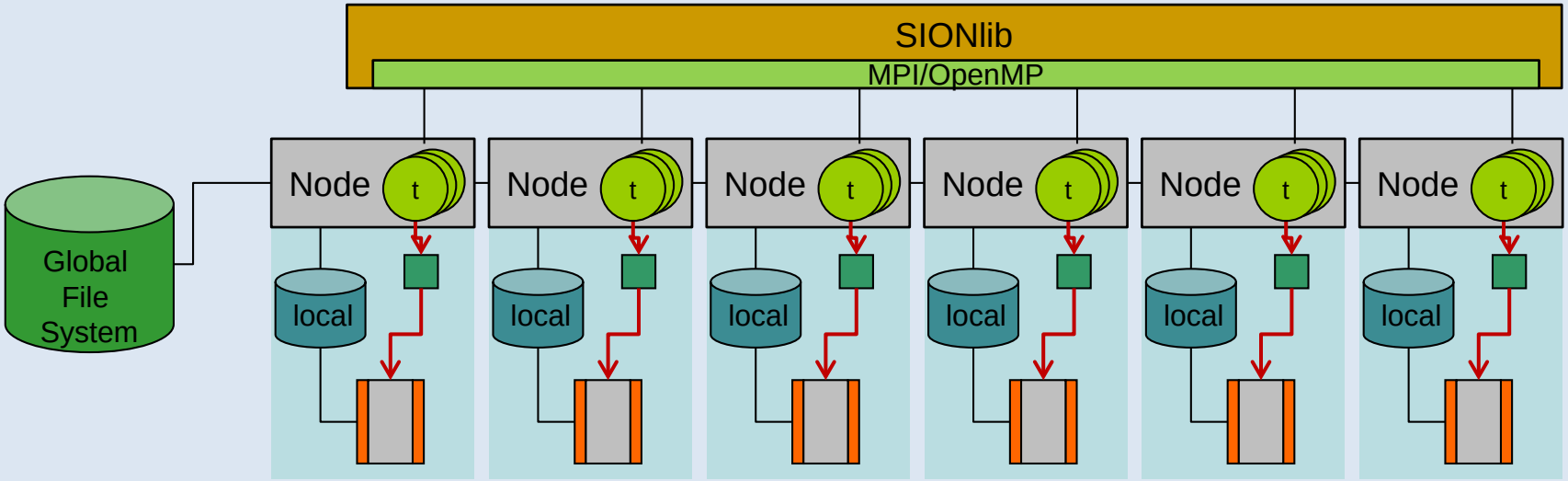
→ #files: O(10)

SIONlib: Strategy for Local Storage



SIONlib file container
→multi-file approach

SIONlib: Local checkpointing



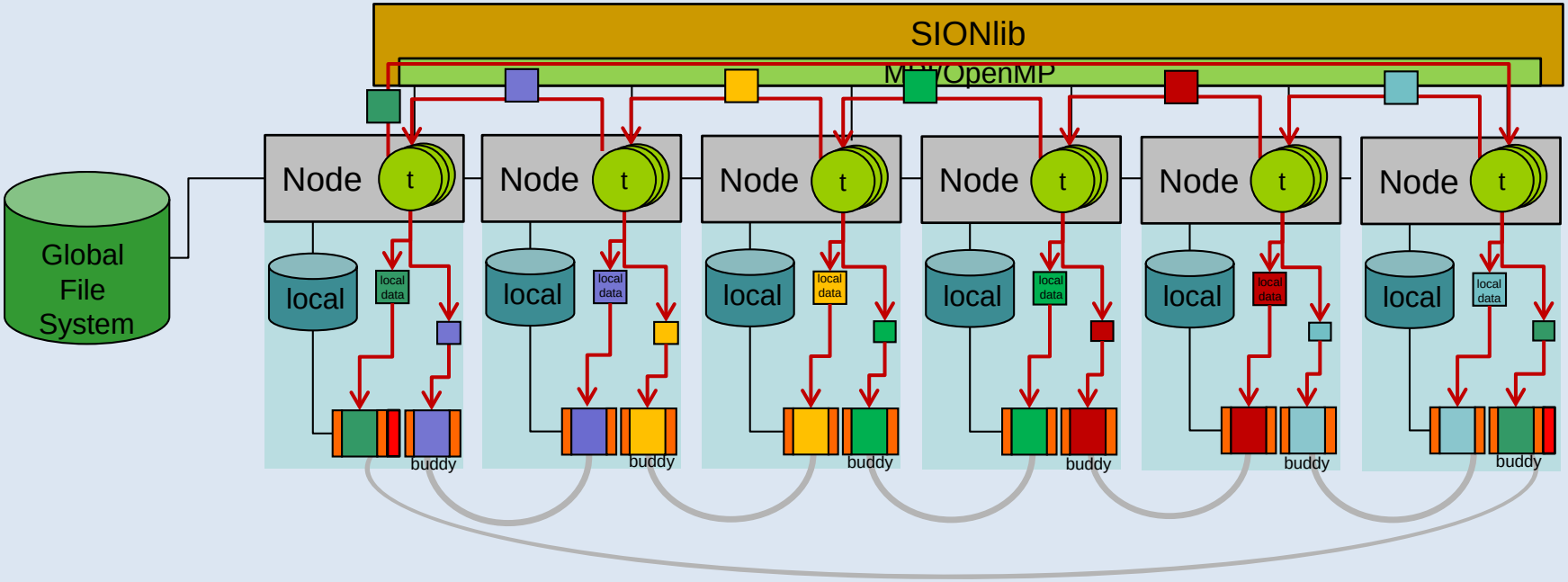
- Mapping: local task writes to local storage, using one physical file of SIONlib file container

→ Transparent access to local data from application

→ Transparent access to data when files migrated to global storage

→ Re-distribution of task-local data in SIONlib layer possible (Buddy-CP)

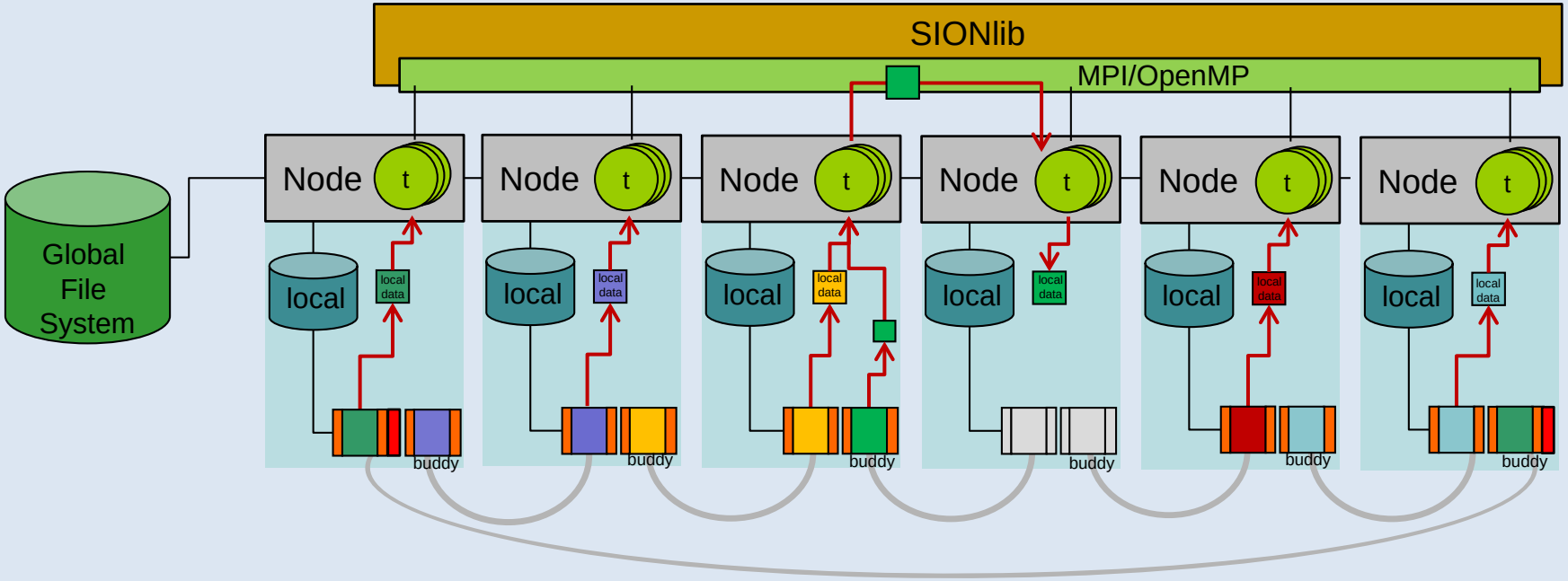
SIONlib: Buddy-CP Logical mapping



- Mapping: 1:1 task writes to local storage and storage of buddy node
  1:m task writes to local storage and storage of m buddy nodes
- Data exchange to buddy node is done via SIONlib MPI/OpenMP layer
- Collective checkpoint calls required

```
Open: sid=sion_paropen_mpi(…,"bw,buddy=m",MPI_COMM_WORLD, … )
Write: sion_coll_write_mpi(data,size,elements,sid)
Close: sion_parclose(sid)
```

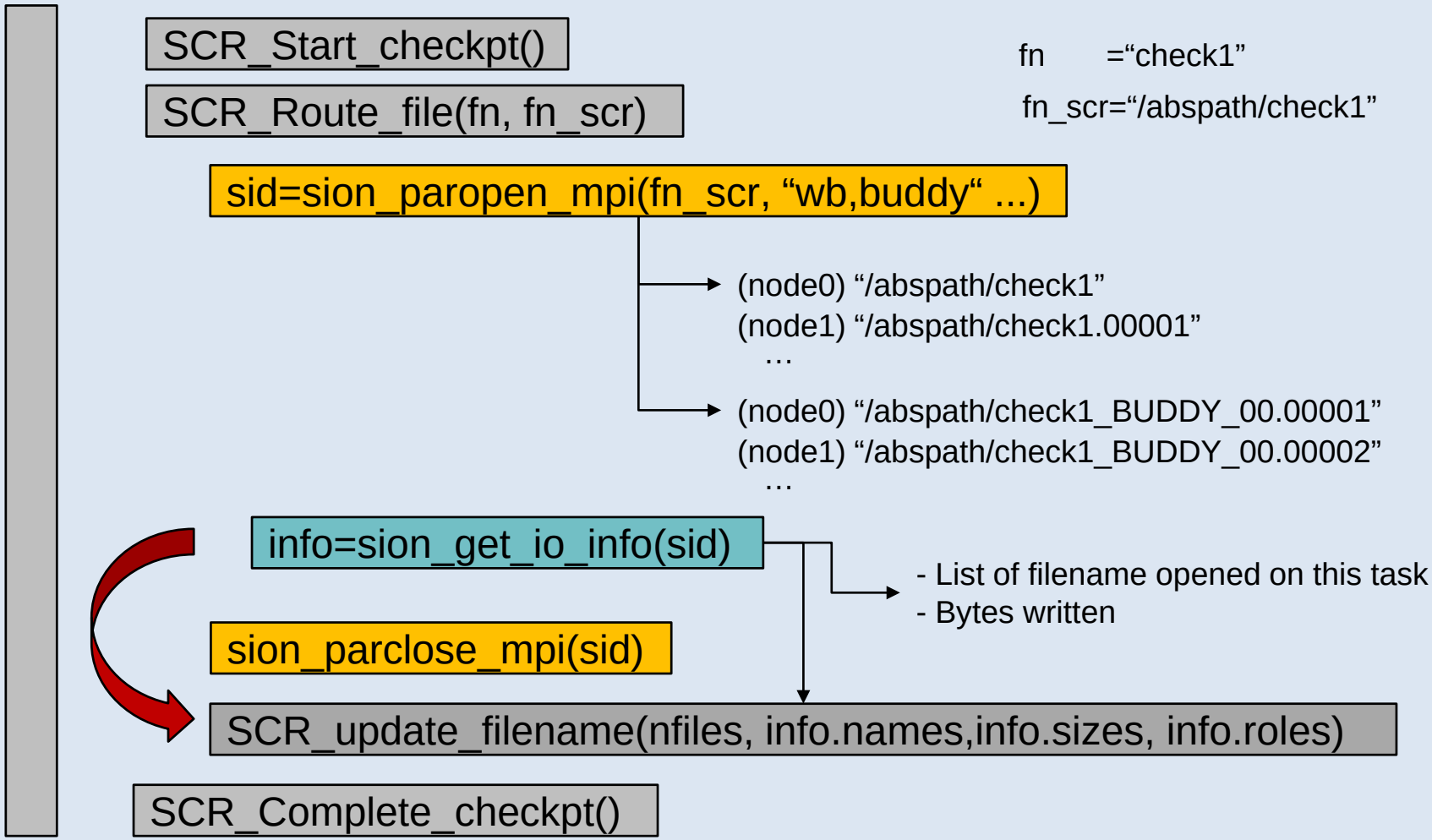
Restore checkpoint after failure



- Automatic analysis of SIONlib data availability on local storage
- On missing data: falls back to buddy data if first open fails

```
Open: sid=sion_paropen_mpi(…,"br,buddy=m",MPI_COMM_WORLD, … )
Read: sion_coll_read_mpi(data,size,elements,sid)
Close: sion_parclose(sid)
```

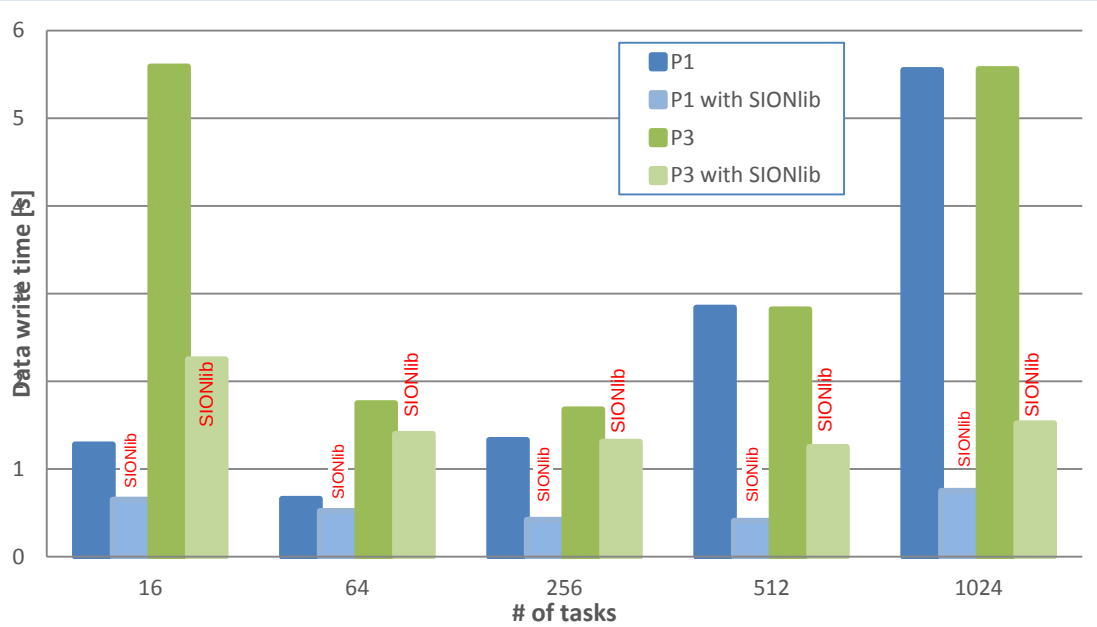
SIONlib and SCR interoperability



Preliminary results SIONlib

MAXW-DGTD: Human exposure to electromagnetic fields (Inria)

SIONlib reduces number of files
→ less filesystem overhead



# of tasks	P1	P1 with SIONlib	P3	P3 with SIONlib
16	1.28	0.65	5.59	2.25
64	0.66	0.52	1.75	1.4
256	1.33	0.42	1.68	1.31
512	2.84	0.41	2.82	1.25
1024	5.55	0.75	5.56	1.52

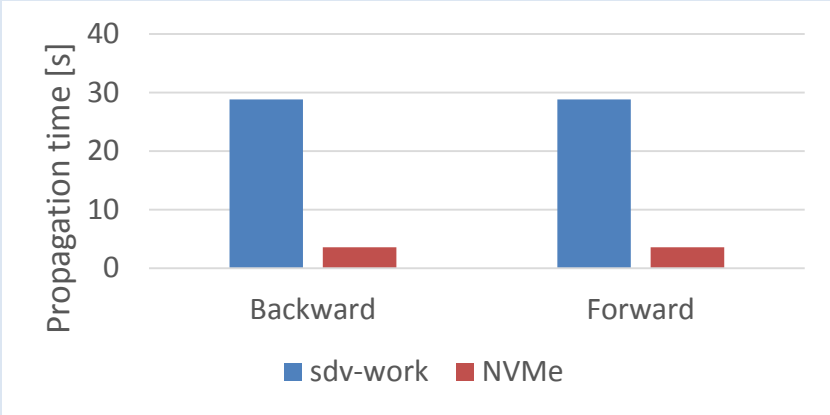
output times on DEEP

Preliminary results NVMe

Oil exploration (BSC)

	sdv-work	NVMe
Backward	28.84	3.6
Forward	28,84	3.6

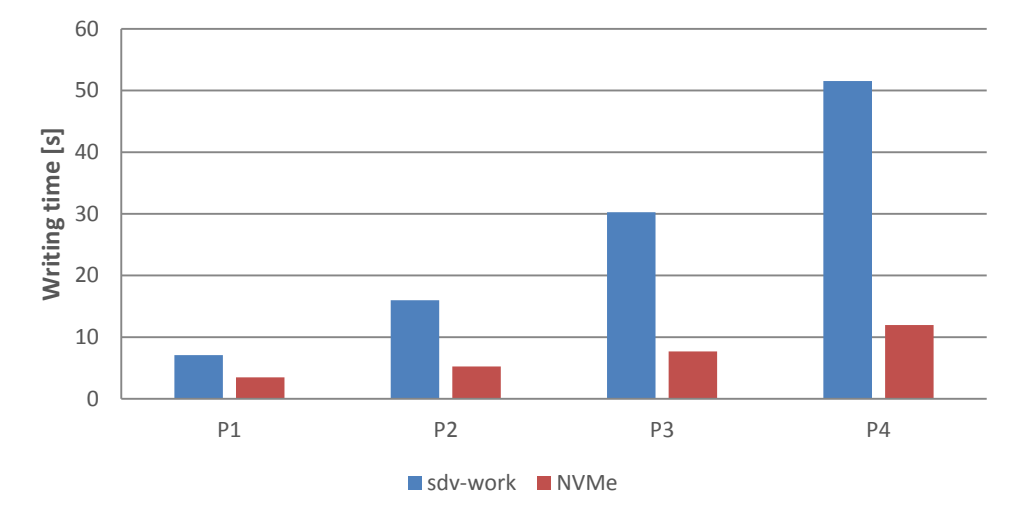
Impact of NVMe



Human exposure to electromagnetic fields (Inria)

	sdv-work	NVMe
P1	7.1	3.49
P2	15.99	5.24
P3	30.27	7.69
P4	51.54	11.98

I/O performance of MAXW-DGTD



Summary

- DEEP-ER explores future directions of I/O
  - On POSIX optimization level → SIONlib
  - On filesystem level → BeeGFS
  - On MPI-IO level → E10
- Aims to be able to test and combine the approaches
- Exploration and validations of new hardware
  - NVMe
  - NAM
- More information on http://www.deep-er.eu