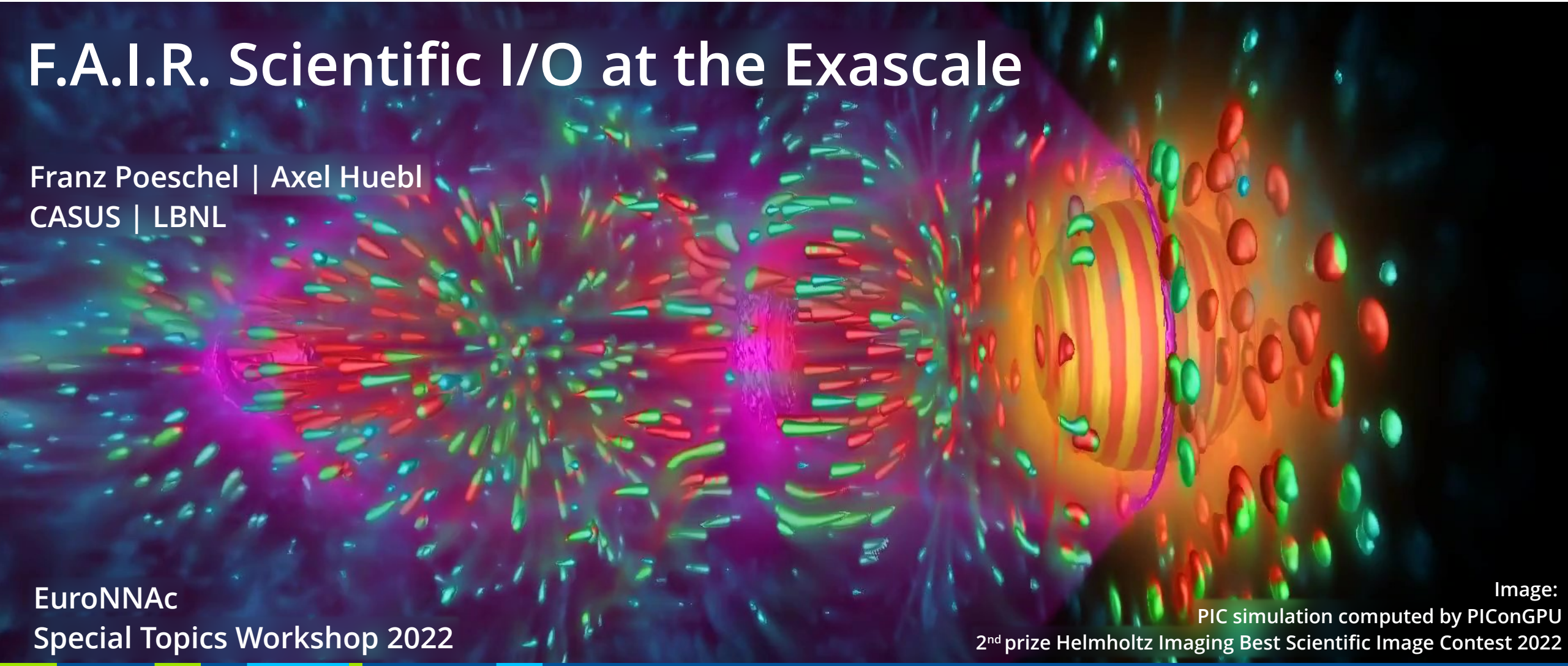
F.A.I.R. Scientific I/O at the Exascale

Franz Poeschel | Axel Huebl
CASUS | LBNL

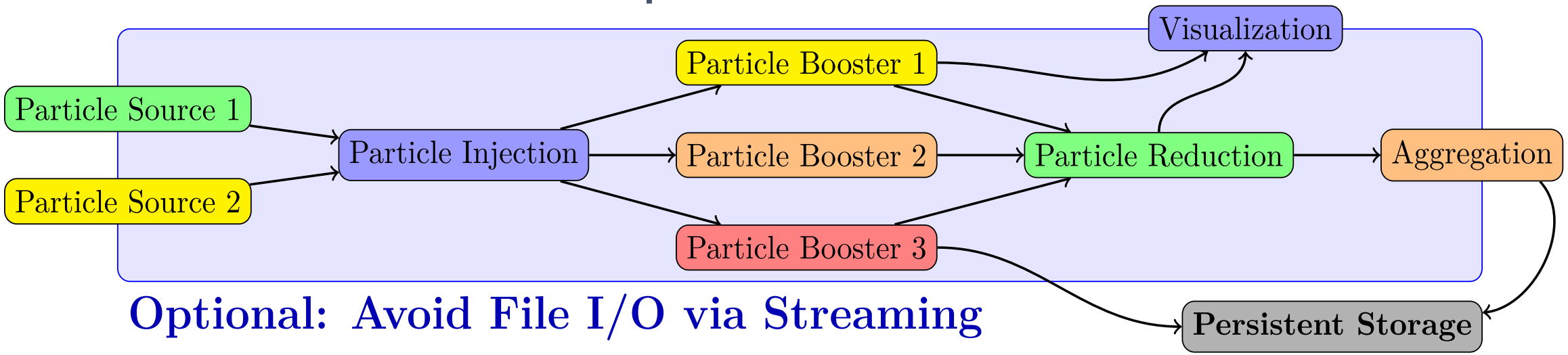


EuroNNAc
Special Topics Workshop 2022

Image:
PIC simulation computed by PIConGPU
2nd prize Helmholtz Imaging Best Scientific Image Contest 2022

Heterogeneity through Standardized Data

Particle accelerators are complex:



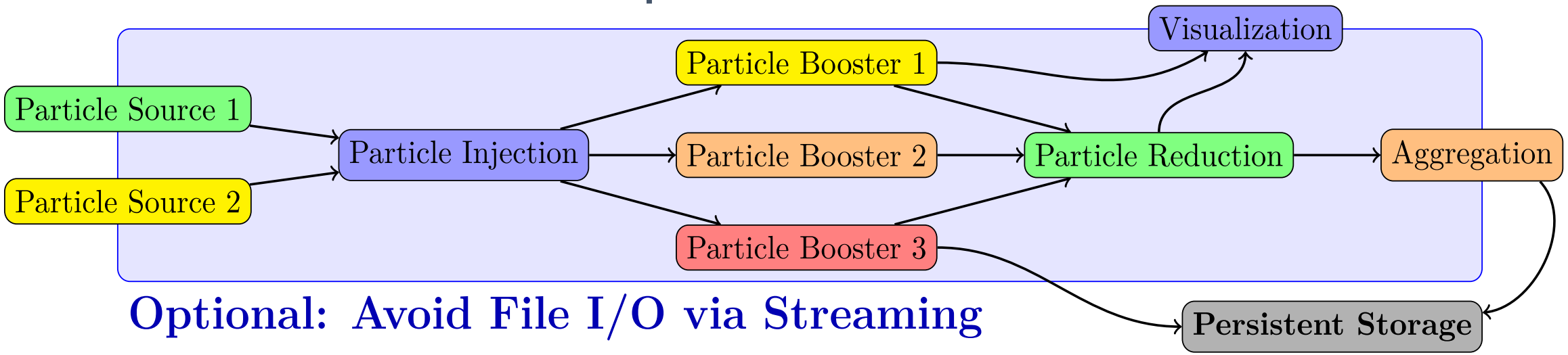
Optional: Avoid File I/O via Streaming

- need to span different **time** and **length scales**
- particle accelerator modeling requires **multiple codes**, collaborating in a **data processing pipeline**
- **bridge heterogeneous models** by standardization of data

Axel Huebl et al. “openPMD: A meta data standard for particle and mesh based data”. 2015. doi: 10.5281/zenodo.591699. url: https://openPMD.org

Heterogeneity through Standardized Data

Particle accelerators are complex:



Optional: Avoid File I/O via Streaming

→ openPMD standard for **p**article-**m**esh **d**ata as communication layer

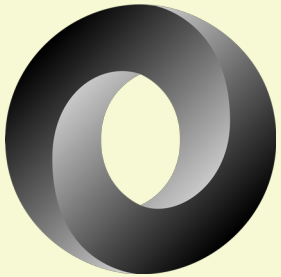
Axel Huebl et al. “openPMD: A meta data standard for particle and mesh based data”. 2015. doi: 10.5281/zenodo.591699. url: https://openPMD.org

openPMD – a FAIR standard

Findable: Standardized metadata to identify the data producer

```
string     /author                 attr    = "franz"
string     /software               attr    = "PIConGPU"
string     /softwareVersion        attr    = "0.5.0-dev"
```

Accessible: Open standard, implementable in various formats

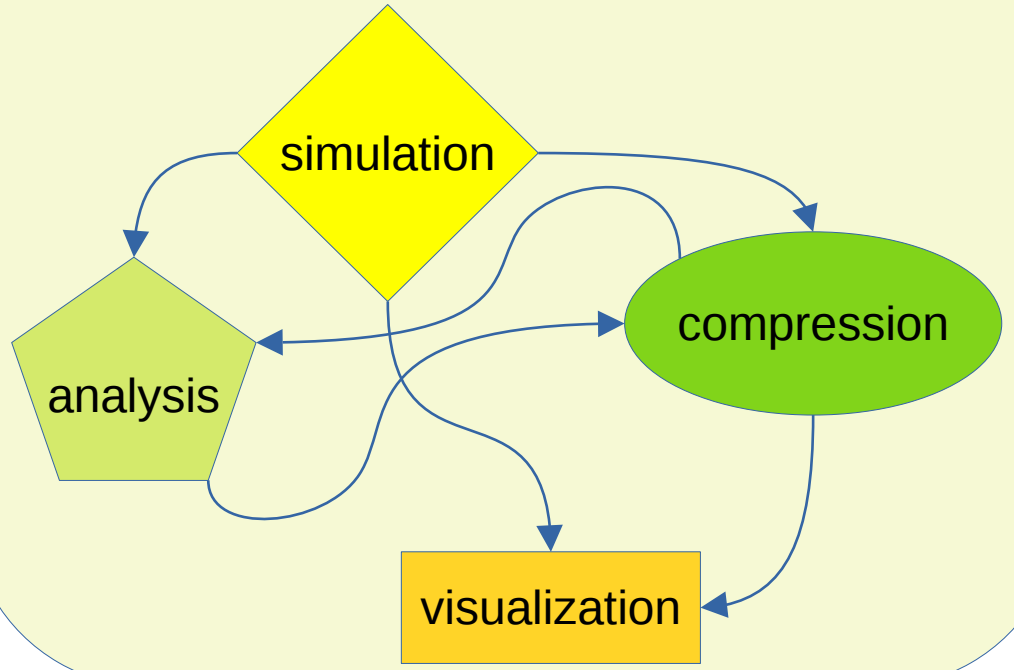


*currently implemented, but not limited to

"The FAIR Guiding Principles for scientific data management and stewardship" (Mark D. Wilkinson et al.)

openPMD – a FAIR standard

Interoperable:
Data exchange spans applications, platforms and teams



Reusable:
Rich and standardized description for physical quantities

Name	Value
axisLabels	[b'z' b'y' b'x']
dataOrder	b'C'
fieldSmoothing	b'none'
geometry	b'cartesian'
gridGlobalOffset	[0. 0. 0.]
gridSpacing	[4.252342 1.0630856 4.252342]
gridUnitSI	4.1671151662e-08
position	[0. 0. 0.]
timeOffset	0.0
unitDimension	[-3. 0. 1. 1. 0. 0. 0.]
unitSI	15399437.98944343

"The FAIR Guiding Principles for scientific data management and stewardship" (Mark D. Wilkinson et al.)

openPMD powered Projects and Users

CASUS
CENTER FOR ADVANCED SYSTEMS UNDERSTANDING

Documents:

- **openPMD standard** (1.0.0, 1.0.1, 1.1.0)
  the underlying file markup and definition
  A Huebl et al., doi: 10.5281/zenodo.33624

Language Binding:

- **openPMD-api** (HZDR, CASUS, LBNL)
  reference API for openPMD data handling
  maintainers: A Huebl, J Gu, F Poeschel et al.

Scientific Simulations:

- **PIConGPU** (HZDR)
  electro-dynamic particle-in-cell code
  maintainers: R Widera, S Bastrakov, A Debus et al.
- **WarpX** (LBNL, LLNL)
  electro-dynamic/static particle-in-cell code
  maintainers: JL Vay, D Grote, R Lehe, A Huebl et al.
- **FBPIC** (LBNL, DESY)
  spectral, fourier-bessel particle-in-cell code
  maintainers: R Lehe, M Kirchen et al.
- **SimEx Platform** (EUCALL, European XFEL)
  simulation of advanced photon experiments
  maintainer: C Fortmann-Grote
- **Wake-T** (DESY)
  fast particle-tracking code for plasma-based accelerators
  maintainer: A Ferran Pousa
- **HiPACE++** (DESY, LBNL)
  3D GPU-capable quasi-static PIC code for plasma accel.
  maintainers: M Thevenet, S Diederichs, A Huebl
- **Bmad** (Cornell)
  library for charged-particle dynamics simulations
  maintainers: D Sagan et al.

- and more…

see also: https://github.com/openPMD/openPMD-projects

openPMD powered Projects and Users

CASUS
CENTER FOR ADVANCED SYSTEMS UNDERSTANDING

Documents:

- **openPMD standard** (1.0.0, 1.0.1, 1.1.0)
  the underlying file markup and definition
  A Huebl et al., doi: 10.5281/zenodo.33624

Tools and converters:

- **file validators** (HZDR, LBNL)
  development scripts
  maintainer: A Huebl, R Lehe
- **XDMF creation** (TU Dresden, HZDR)
  xml meta file creation for (serial) reading in VTK
  maintainer: HZDR
- **HDF Compass** (third party + HZDR: ADIOS implementation)
  viewer for HDF5 files and related formats
  maintainer: HDF Group (HZDR: contribution)

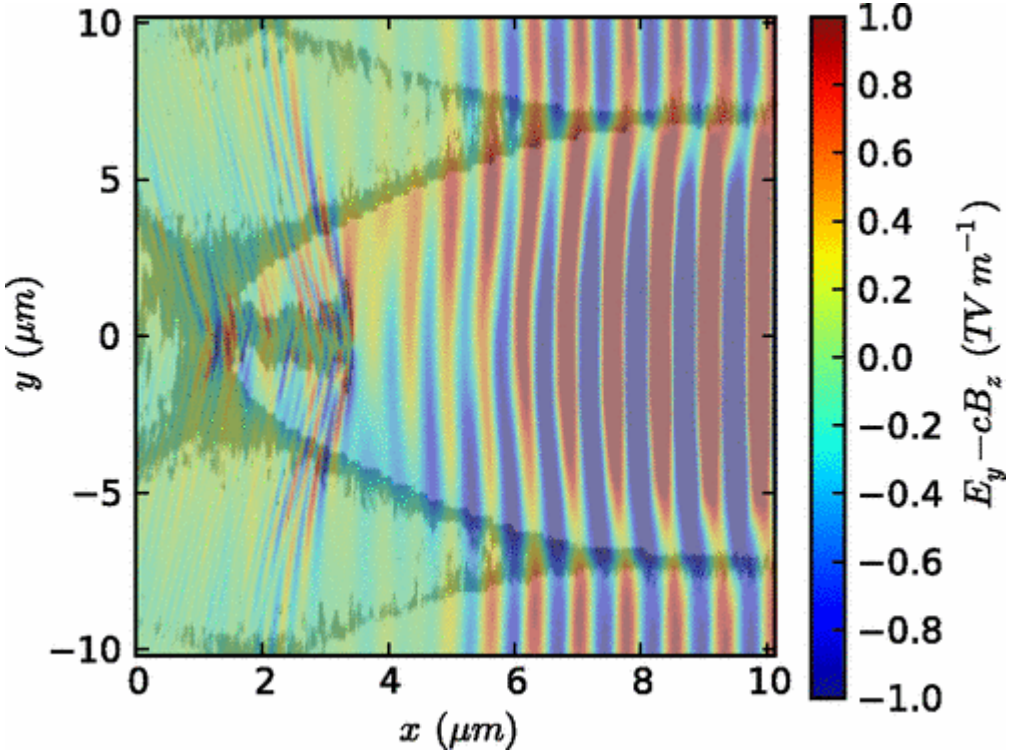
Data processing and visualization:

- **openPMD-viewer** (LBNL, DESY)
  high-level python API & interactive jupyter notebook GUI
  maintainer: R Lehe
- **VisualPIC** (DESY)
  post-processing and visualization for particle-in-cell data
  maintainer: A Ferran Pousa
- **postpic** (IOQ Jena)
  post-processing and visualization for particle-in-cell data
  maintainer: S Kuschel
- **yt project** (third party + HZDR: reader implementation)
  framework for parallel analysis and visualization
  maintainer: the yt team (HZDR: contribution)
- **VisIt** (LLNL)
  parallel post-processing and 3D visualization
  maintainer: LLNL (NERSC: contribution)
- **ParaView** (Kitware)
  analysis and visualization
  maintainers: Kitware (reader plugin by B Geveci and A Huebl)

see also: https://github.com/openPMD/openPMD-projects

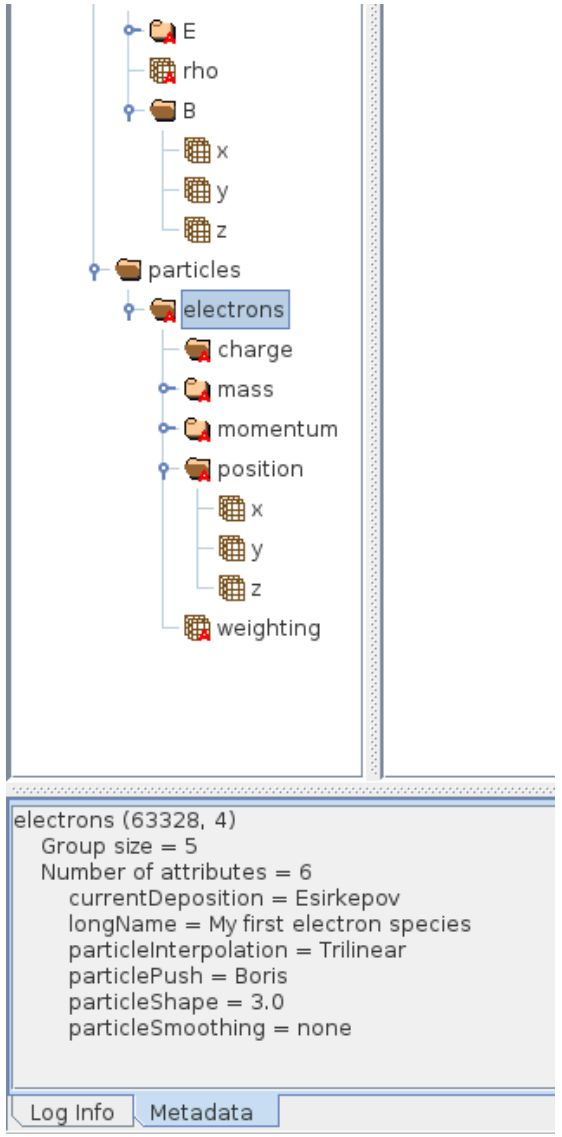
Extensions: e.g. ED-PIC

Field image → field solver, smoothing



similar:

Emittance → particle push, field solver, shape



openPMD-api – I/O for Scientific Compute Workflows

openPMD-api

Software Stack

planned / open for contributions

Packaging	**Spack**	**conda-forge**	**Homebrew**	**Python .whl**	
Frontend	**C++17**	**Python3** +numpy	**Julia** under development		
Middleware	**C++17 library**				
I/O Libraries parallel / serial	**HDF5**	**ADIOS1**	**ADIOS2**	**JSON**	**TOML**
Platforms	**Linux/Unix**	**OSX**	**Windows**		

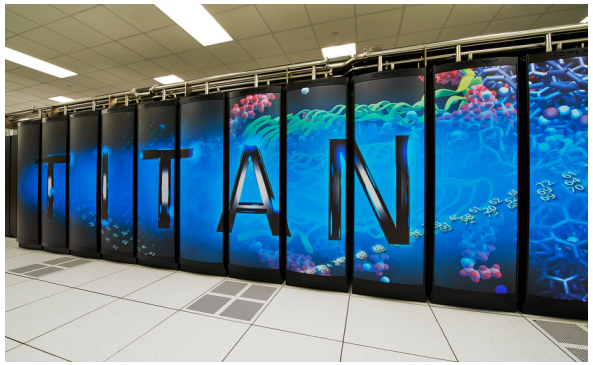
describe data workflow once

pick and configure backend
at runtime without recompiling

```python
import openpmd_api as io

# pick backend by filename extension
series = io.Series("simOutput.h5",   io.Access.create)
series = io.Series("simOutput.bp",   io.Access.create)
series = io.Series("simOutput.sst",  io.Access.create)
series = io.Series("simOutput.json", io.Access.create)
```

Compute Performance Outpaces Storage Performance



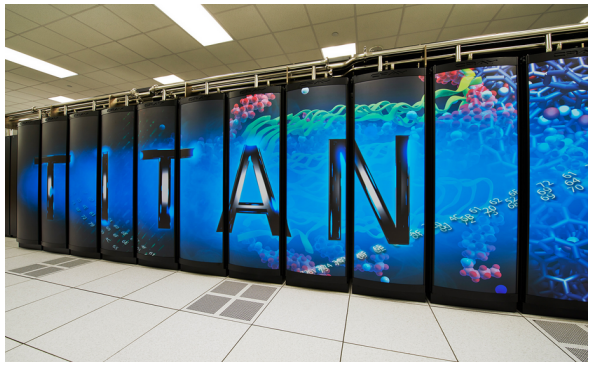
	Titan	Summit	Frontier	*Growth Factor*
Peak Performance:	27 Pflop/s	200 Pflop/s	1.6 Eflop/s	~60
FS Throughput:	1 TiByte/s	2.5 TiByte/s	5~10 TiByte/s	5~10
FS Capacity:	27 PiByte	250 PiByte	500~1000 PiByte	18~37

→ parallel bandwidth insufficient for HPC at full scale

→ filesystem capacity insufficient for HPC at full scale

Franz Poeschel et al. “Transitioning from file-based HPC workflows to streaming data pipelines with openPMD and ADIOS2”. 2022. doi: 10.1007/978-3-030-96498-6_6.

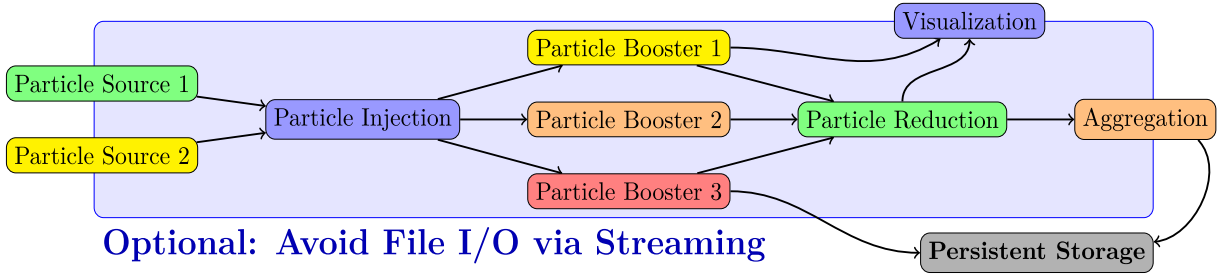
Compute Performance Outpaces Storage Performance



	Titan	Summit	Frontier	Growth Factor
Peak Performance:	27 Pflop/s	200 Pflop/s	1.6 Eflop/s	~60
FS Throughput:	1 TiByte/s	2.5 TiByte/s	5~10 TiByte/s	5~10
FS Capacity:	27 PiByte	250 PiByte	500~1000 PiByte	18~37

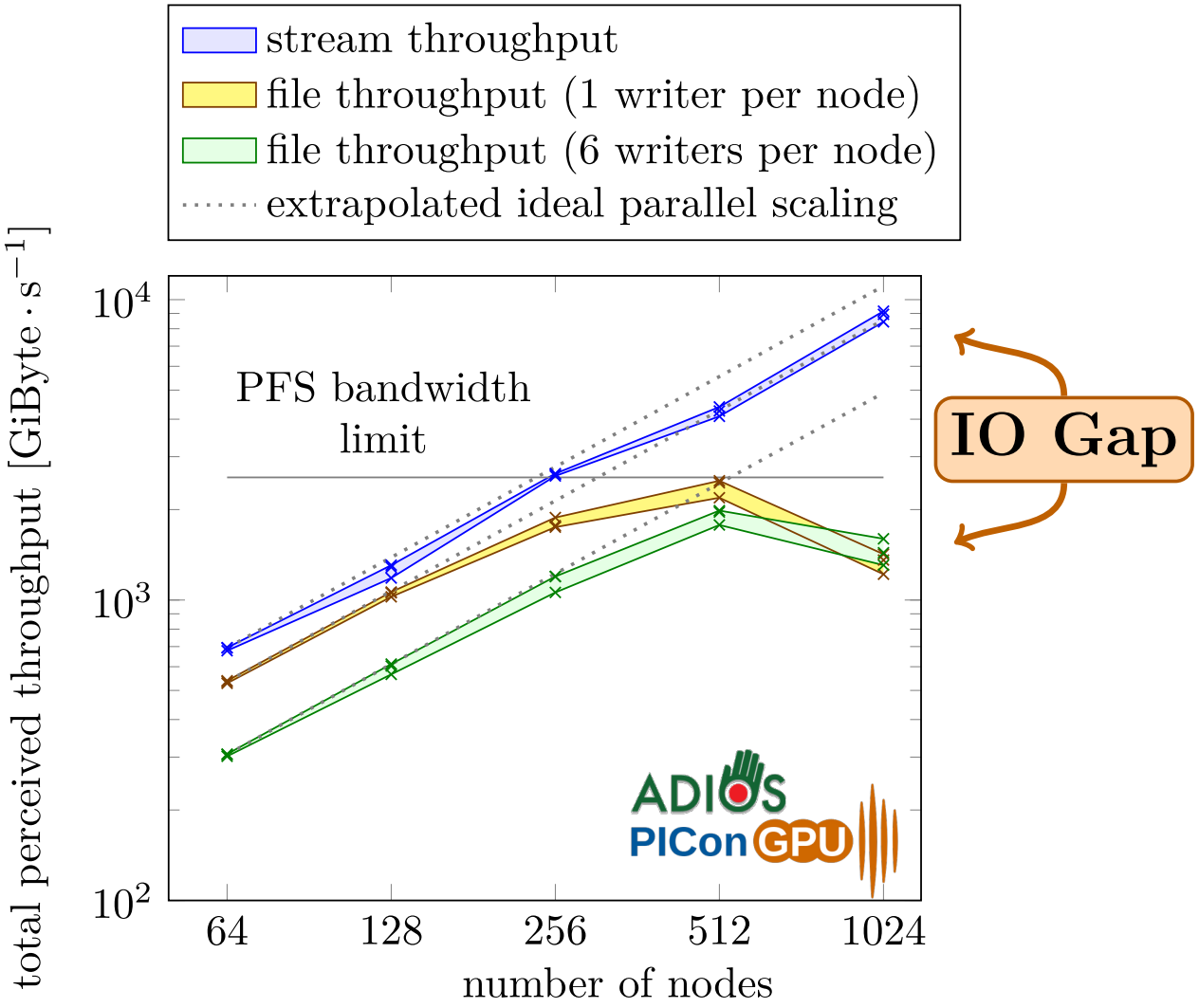
Why does this concern us?

- Heterogeneous data processing pipelines traditionally have large I/O usage
- Scalable alternative: Streaming



Franz Poeschel et al. “Transitioning from file-based HPC workflows to streaming data pipelines with openPMD and ADIOS2”. 2022. doi: 10.1007/978-3-030-96498-6_6.

Break through Filesystem Bandwidth with Streaming

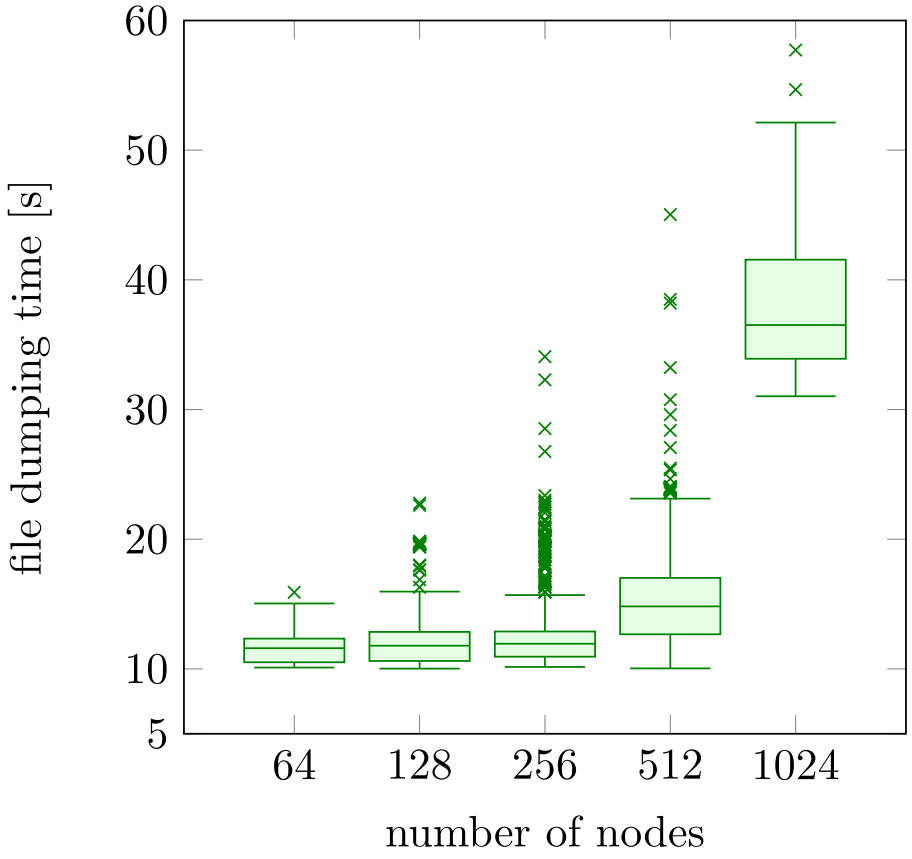


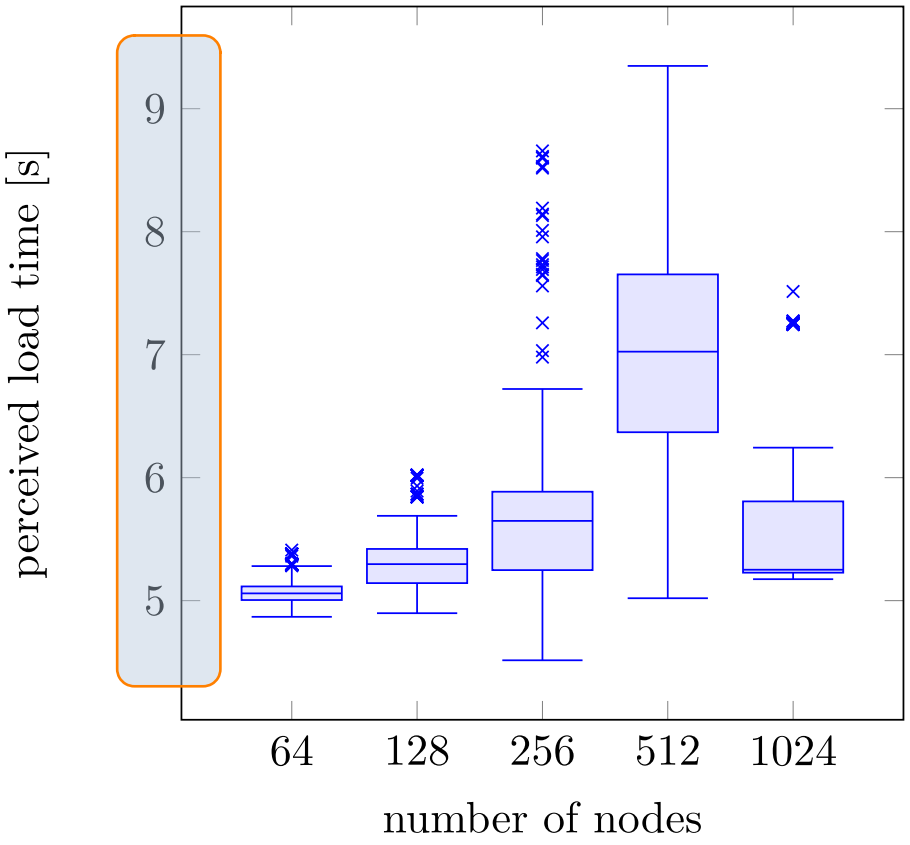
Memory-bound simulations reach the I/O system limits at a fraction of full scale

- Summit FS bandwidth (2.5TiByte/s) reached at 512 nodes (~11% of system size)
- Streaming workflows unaffected by filesystem bandwidth, use Infiniband hardware to scale beyond it

(benchmarks at 1024 nodes done after Summit system upgrade)

Summit: Reproducible Performance via Streaming





File I/O:

Single writers need
up to a minute to finish

Streaming I/O:

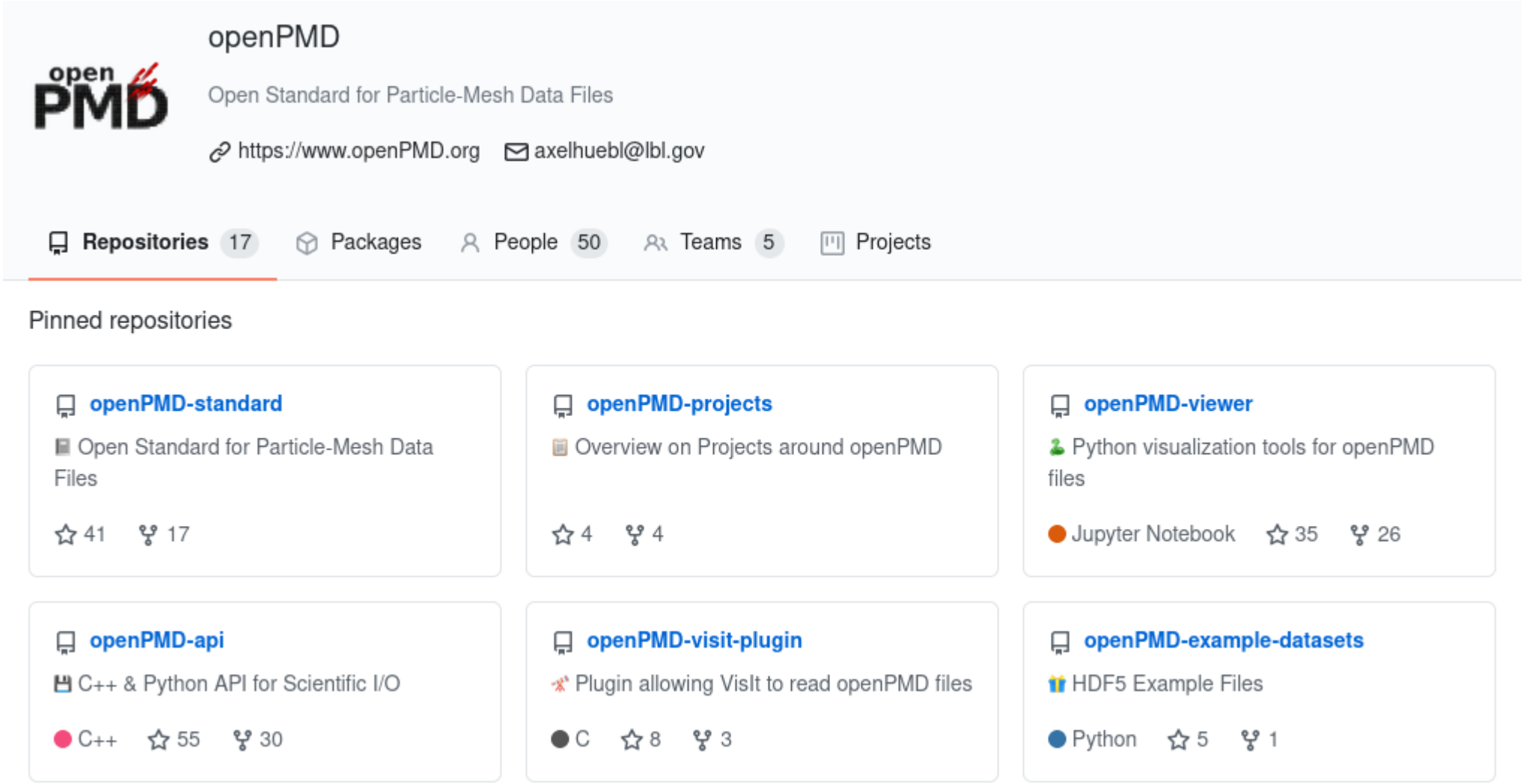
All measurements
within 5~9 seconds

Acknowledgements

This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725. Supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of two U.S. Department of Energy organizations (Office of Science and the National Nuclear Security Administration). Supported by EC through Laserlab-Europe, H2020 EC-GA 871124. Supported by the Consortium for Advanced Modeling of Particles Accelerators (CAMPA), funded by the U.S. DOE Office of Science under Contract No. DE-AC02-05CH11231. This work was partially funded by the Center of Advanced Systems Understanding (CASUS), which is financed by Germany's Federal Ministry of Education and Research (BMBF) and by the Saxon Ministry for Science, Culture and Tourism (SMWK) with tax funds on the basis of the budget approved by the Saxon State Parliament.

First Steps

→ head to https://github.com/openPMD/



...and of course https://openpmd-api.readthedocs.io/