Cabana: Building on Kokkos for particles (and grids)

Sam Reeve (Oak Ridge National Lab)

CoPA team and collaborators:

Stuart Slattery, Damien Lebrun-Grandié, Lance Bullerwell, Pablo Seleson, Kwitae Chong, Austin Isner, (ORNL),

Susan Mniszewski, Christoph Junghans, Shane Fogerty, Guangye Chen (LANL),

Jim Belak, Lee Ricketson (LLNL),

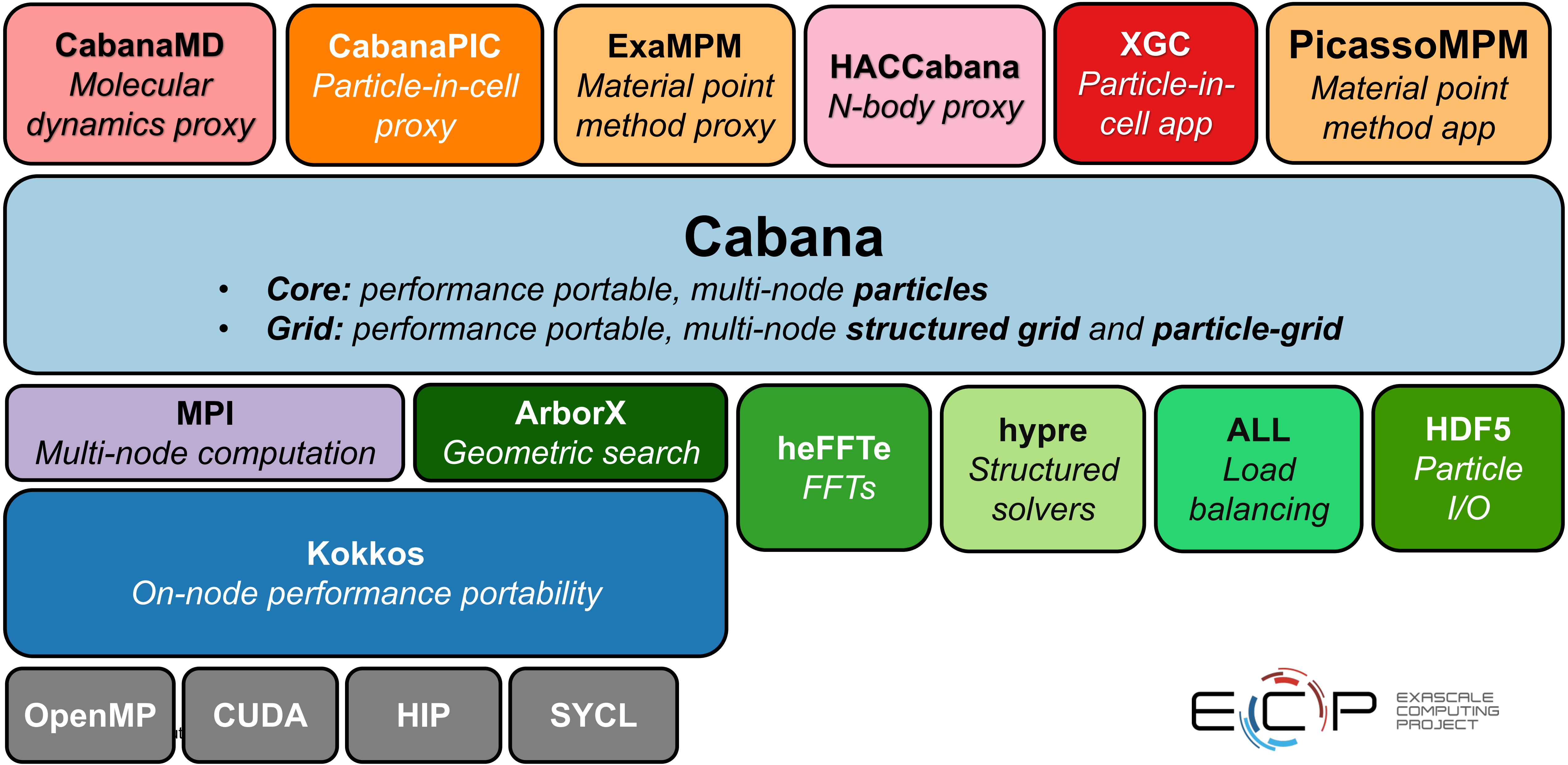
Adrian Pope, Steve Rangel (ANL)

Stan Moore (SNL),

Mark Stock (HPE), Evan Weinberg (NVIDIA),

Scot Breitenfeld (HDF Group), Aaron Scheinberg (Jubilee Dev.)

Slattery et al., (2022). Cabana: A Performance Portable Library for Particle-Based Simulations. Journal of Open Source Software, 7(72), 4115, https://doi.org/10.21105/joss.04115



CabanaMD
Molecular dynamics proxy

CabanaPIC
Particle-in-cell proxy

ExaMPM
Material point method proxy

HACCabana
N-body proxy

XGC
Particle-in-cell app

PicassoMPM
Material point method app

Cabana

- ***Core:*** *performance portable, multi-node* ***particles***
- ***Grid:*** *performance portable, multi-node* ***structured grid*** *and* ***particle-grid***

MPI
Multi-node computation

ArborX
Geometric search

heFFTe
FFTs

hypre
Structured solvers

ALL
Load balancing

HDF5
Particle I/O

Kokkos
On-node performance portability

OpenMP **CUDA** **HIP** **SYCL**

ECP EXASCALE COMPUTING PROJECT

Co-design center for Particle Applications (CoPA)

<u>Lab leads</u>

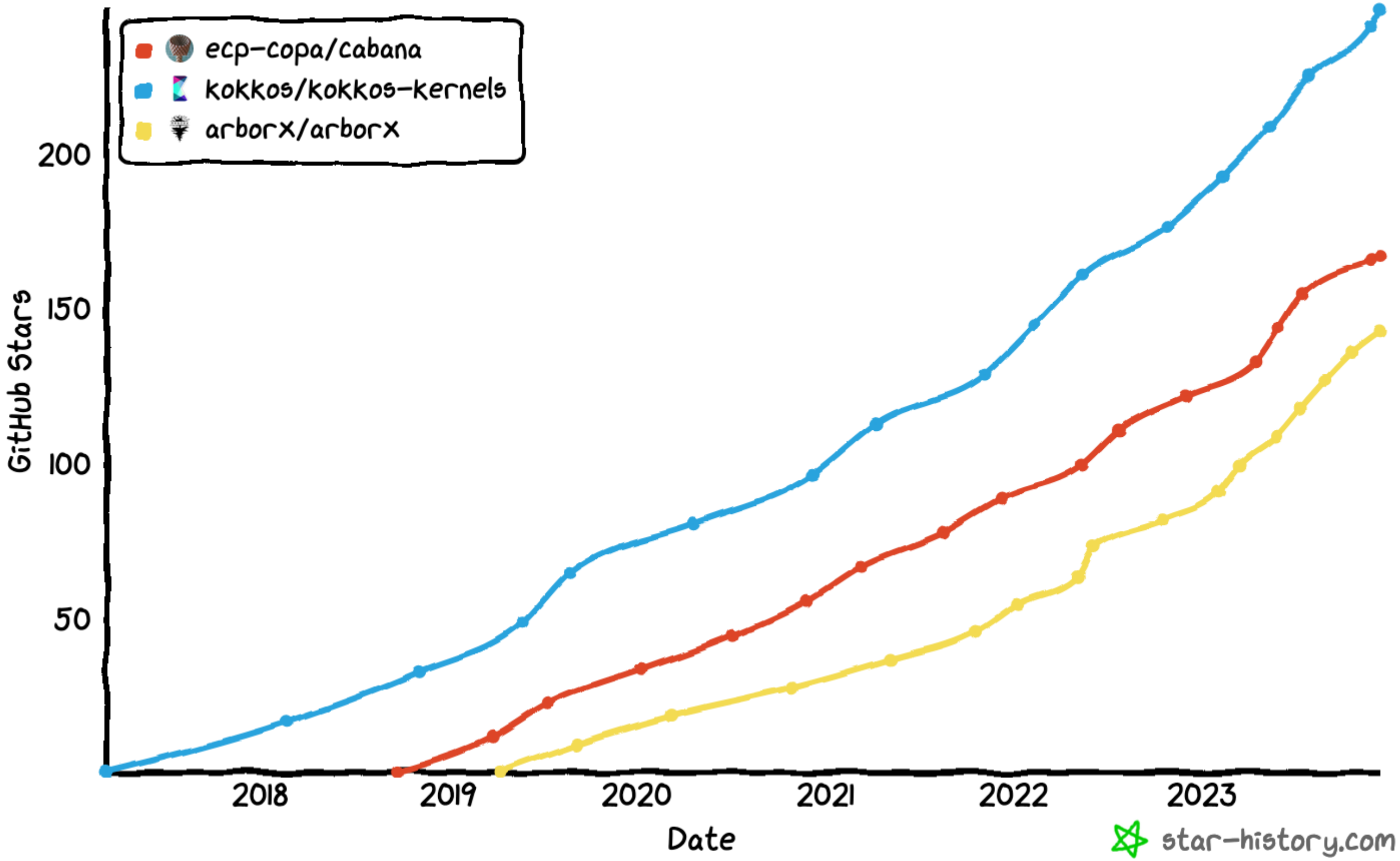
PI: Sue Mniszewski, LANL
Co-PI: Jim Belak, LLNL
Site Lead: C-S Chang, PPPL
Site Lead: Salman Habib, ANL
Site Lead: Steve Plimpton, SNL
Site Lead: Stuart Slattery, ORNL
FFTX Lead: Phil Collela, LBNL

<u>Product leads</u>

Cabana: Sam Reeve
PROGRESS/BML: Jean-Luc Fattebert
FFTX: Phil Collela

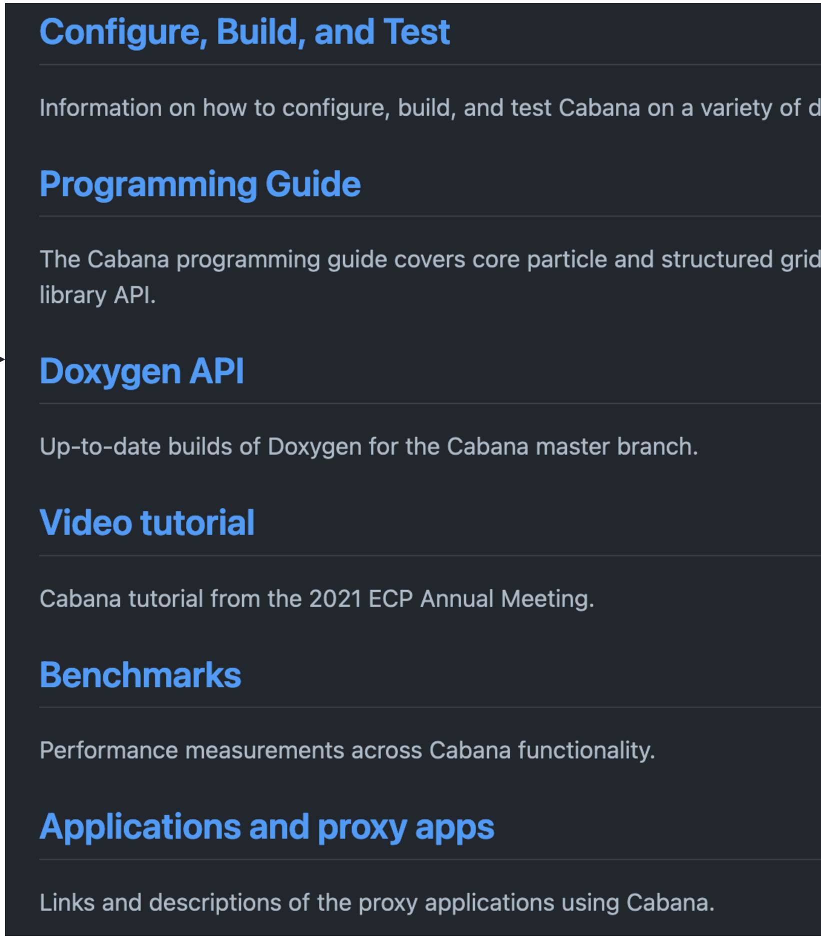
Cabana GitHub: contributions and usage

- Contributions from:
  - DOE labs
    - ORNL, LANL, LLNL, PPPL
  - ECP collaborators
    - NVIDIA, HDF Group, UCLA, Juelich
- Important design directions from:
  - Sandia, Argonne



Cabana resources

- GitHub: https://github.com/ECP-CoPA/Cabana
  - We welcome issues and pull requests
  - **Examples/tutorials in source code**
  - GitHub wiki
- YouTube tutorial (2021): https://youtu.be/VAS7JdprQqE
- Slack https://cabana-dev.slack.com
  - https://kokkosteam.slack.com **#cabana**
- Email (reevest@ornl.gov)



Configure, Build, and Test

Information on how to configure, build, and test Cabana on a variety of d

Programming Guide

The Cabana programming guide covers core particle and structured grid library API.

Doxygen API

Up-to-date builds of Doxygen for the Cabana master branch.

Video tutorial

Cabana tutorial from the 2021 ECP Annual Meeting.

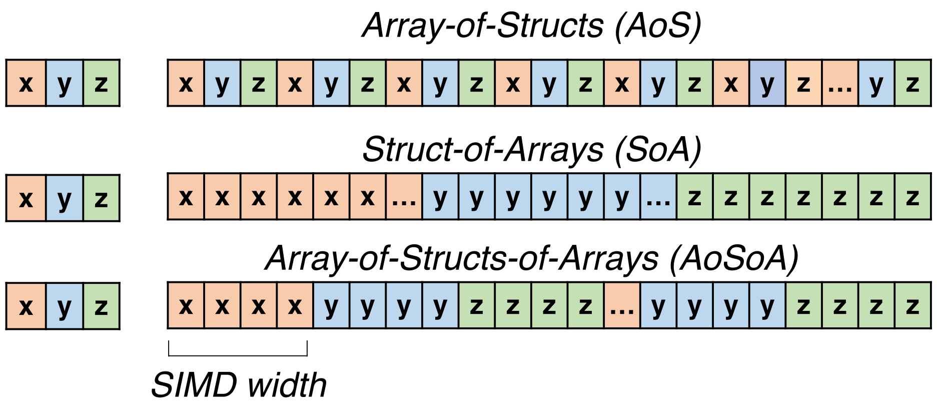
Benchmarks

Performance measurements across Cabana functionality.

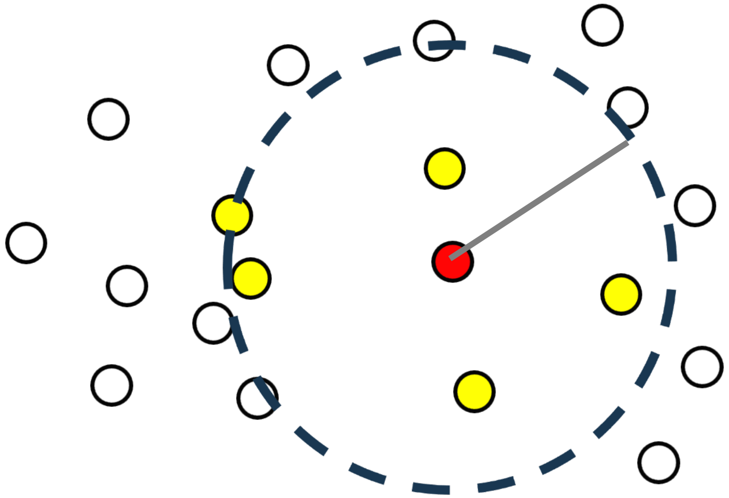
Applications and proxy apps

Links and descriptions of the proxy applications using Cabana.

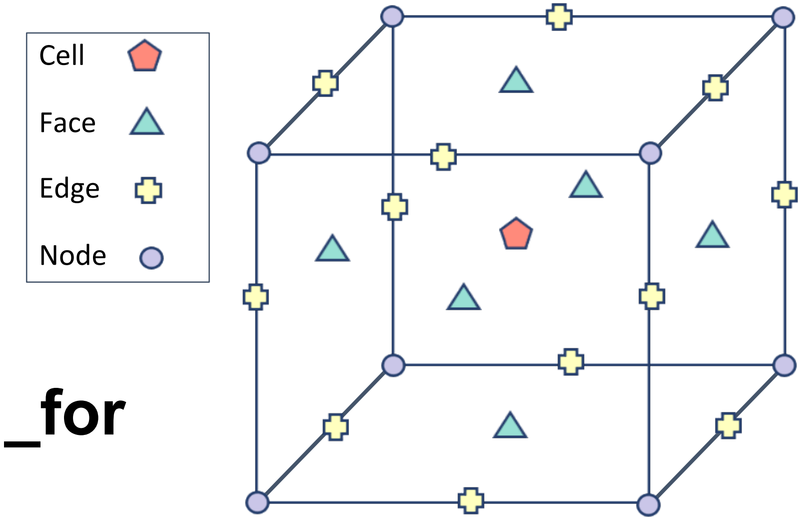
Primary Core library features


Array-of-Structs (AoS)
x y z
x y z x y z x y z x y z x y z x y z ... y z
Struct-of-Arrays (SoA)
x y z
x x x x x x ... y y y y y y ... z z z z z z z
Array-of-Structs-of-Arrays (AoSoA)
x y z
x x x x y y y y z z z z ... y y y y z z z z
SIMD width

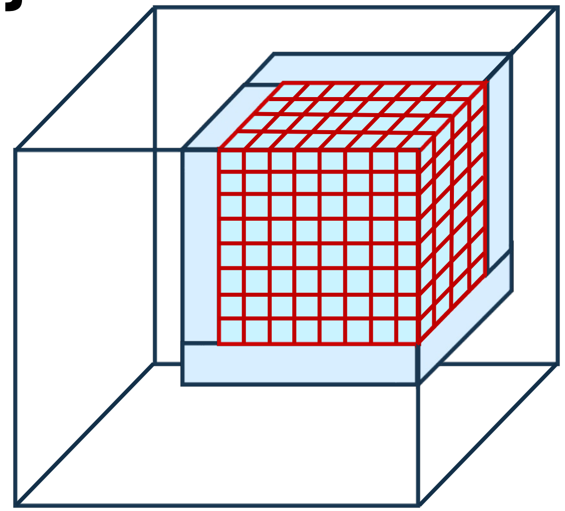
- Particle data structures (View extensions)
- Binning/sorting
  - Including spatial binning with **Cabana::LinkedCellList**
- Neighbor interactions
  - Neighbor lists and **Cabana::neighbor_parallel_for**
    - **Cabana::VerletList** (cell-accelerated N^2 search)
    - ArborX interface
    - LinkedCellList “neighbor-list free”)
- MPI Communication
  - Particle migration **Cabana::migrate**
  - Particle halo **Cabana::{Gather,Scatter}**



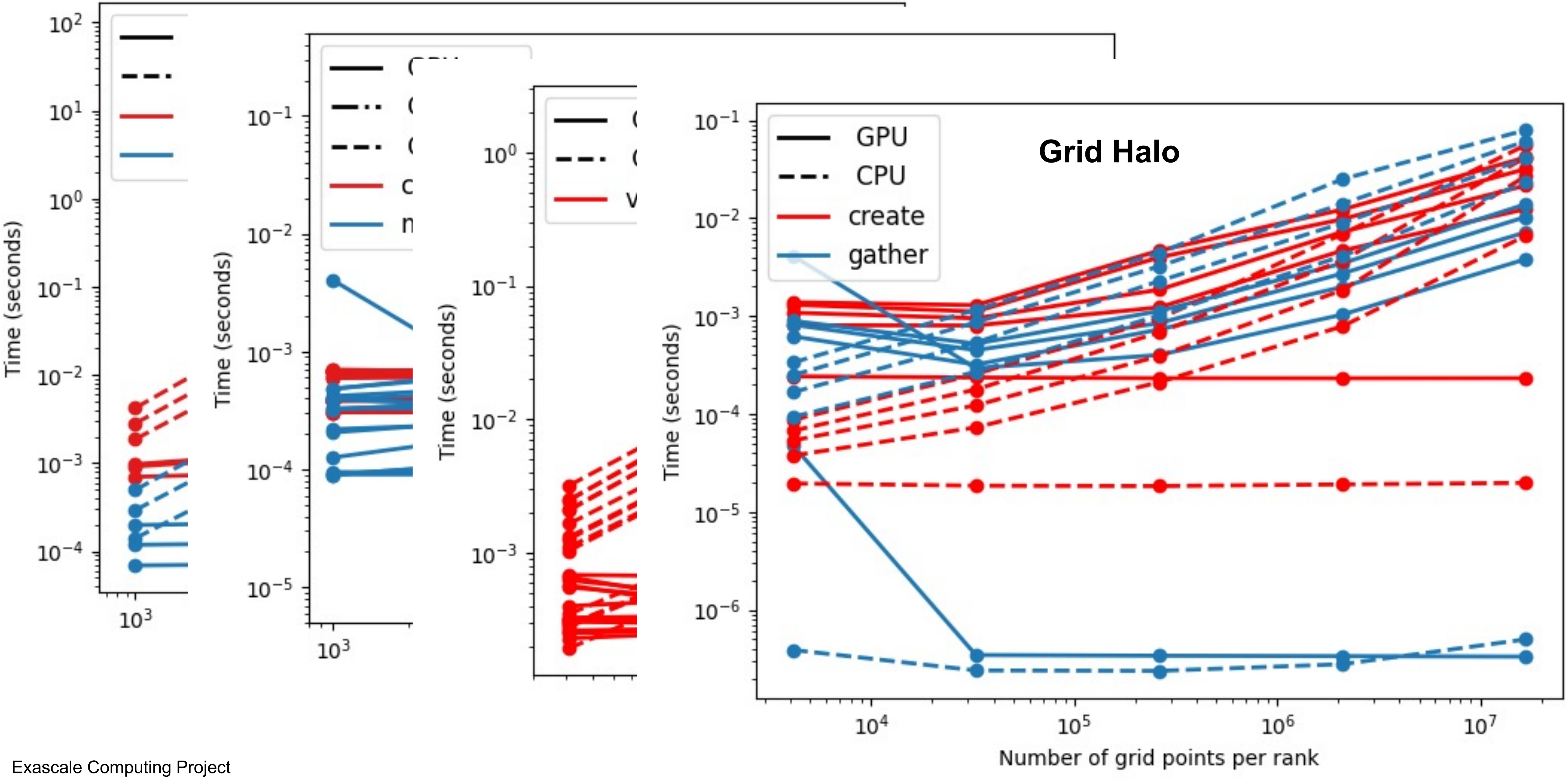
Primary Grid library features



- Domain decomposed structured grids
  - Grid objects: indices for **Cabana::Grid::grid_parallel_for**
  - Mesh objects: physical cell details
- Particle-grid interpolation: **Cabana::Grid::{p2g, g2p}**
- Algorithms
  - FFTs (heFFTe interface)
  - Solvers/preconditioners (HYPRE interface)
- MPI Communication: **Cabana::Grid::Halo**
- Sparse grids (logically dense)
  - Only allocate the portion of the grid fields where particles exist



Cabana benchmarks (Frontier 1 GCD/node)



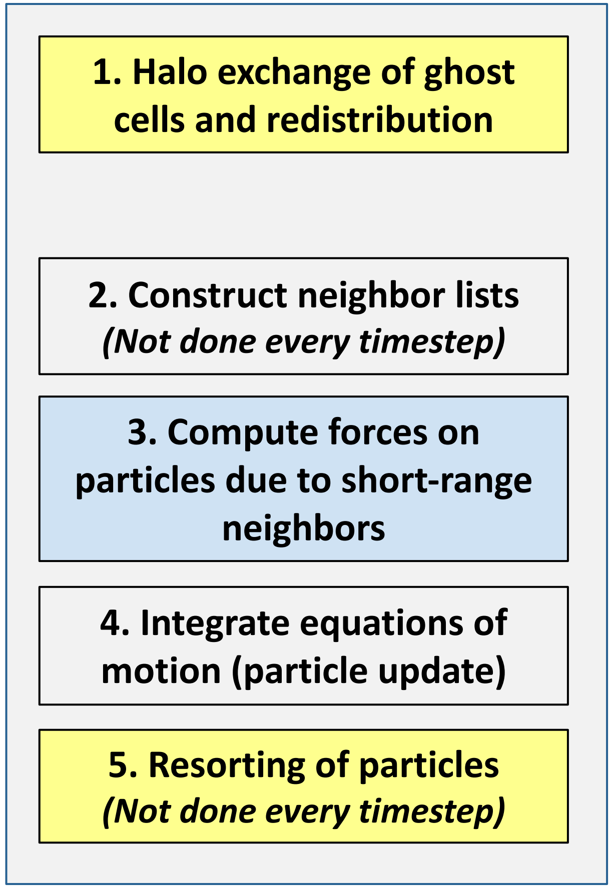
CabanaMD proxy app

🟦 Memory/ flop kernel

🟨 Communication

MD data

Single MD timestep



1. Halo exchange of ghost cells and redistribution

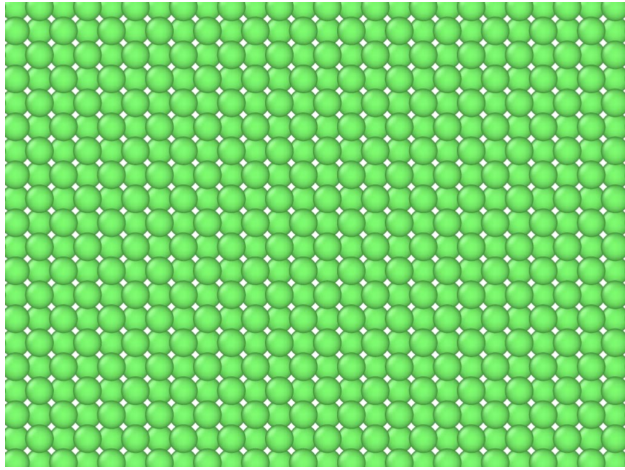
2. Construct neighbor lists *(Not done every timestep)*

3. Compute forces on particles due to short-range neighbors

4. Integrate equations of motion (particle update)

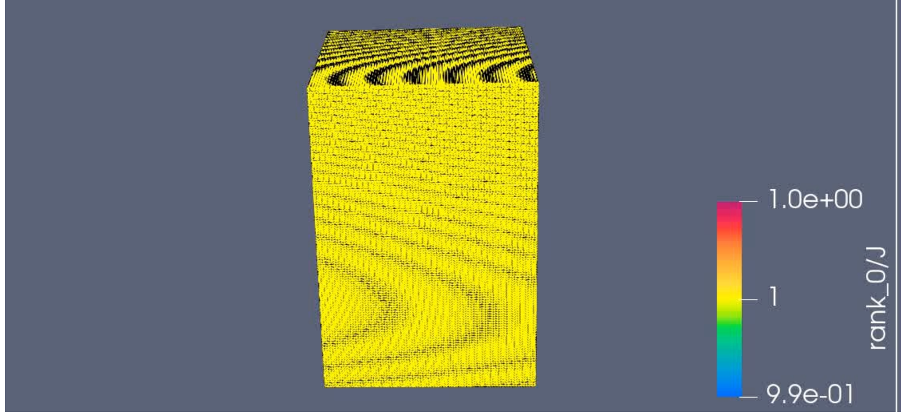
5. Resorting of particles *(Not done every timestep)*

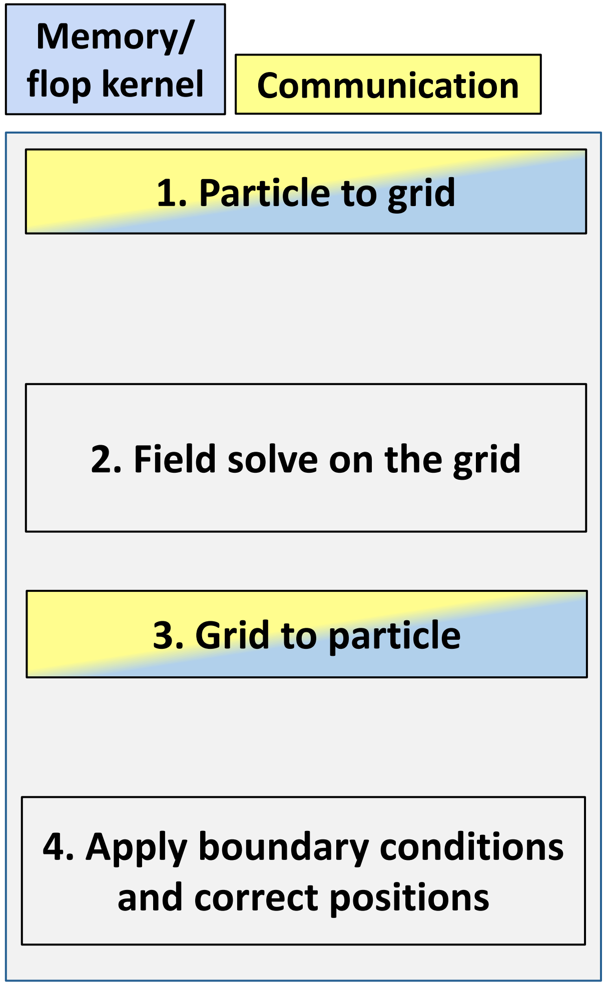
- Cabana::AoSoA, Kokkos::Views, and Cabana::Grid::UniformMesh



- **Cabana::migrate(Cabana::Distributor(…), AoSoA); Cabana::gather(Cabana::Halo(…), x);**
- **Cabana::VerletList(x, …);**
- **Cabana::neighbor_parallel_for(…); *// with x, f***
- **Kokkos::parallel_for(…); *// with x, v, f***
- **Cabana::permute(Cabana::LinkedCellList(x, AoSoA), AoSoA);**

ExaMPM proxy app





Memory/ flop kernel

Communication

1. Particle to grid

2. Field solve on the grid

3. Grid to particle

4. Apply boundary conditions and correct positions

- **Kokkos::parallel_for** with **Cabana::Grid::P2G::{gradient, value}**
  - And **Cabana::Grid::Halo::gather**

- **Cabana::Grid::grid_parallel_for**

- **Kokkos::parallel_for** with **Cabana::Grid::G2P::{gradient, value}**
  - And **Cabana::Grid::Halo::scatter**

- **Kokkos::parallel_for** and **Cabana::Grid::grid_parallel_for**

* Could replace with **Cabana::Grid::{p2g, g2p}**, if fusing the kernels is not performance critical

Cabana applications

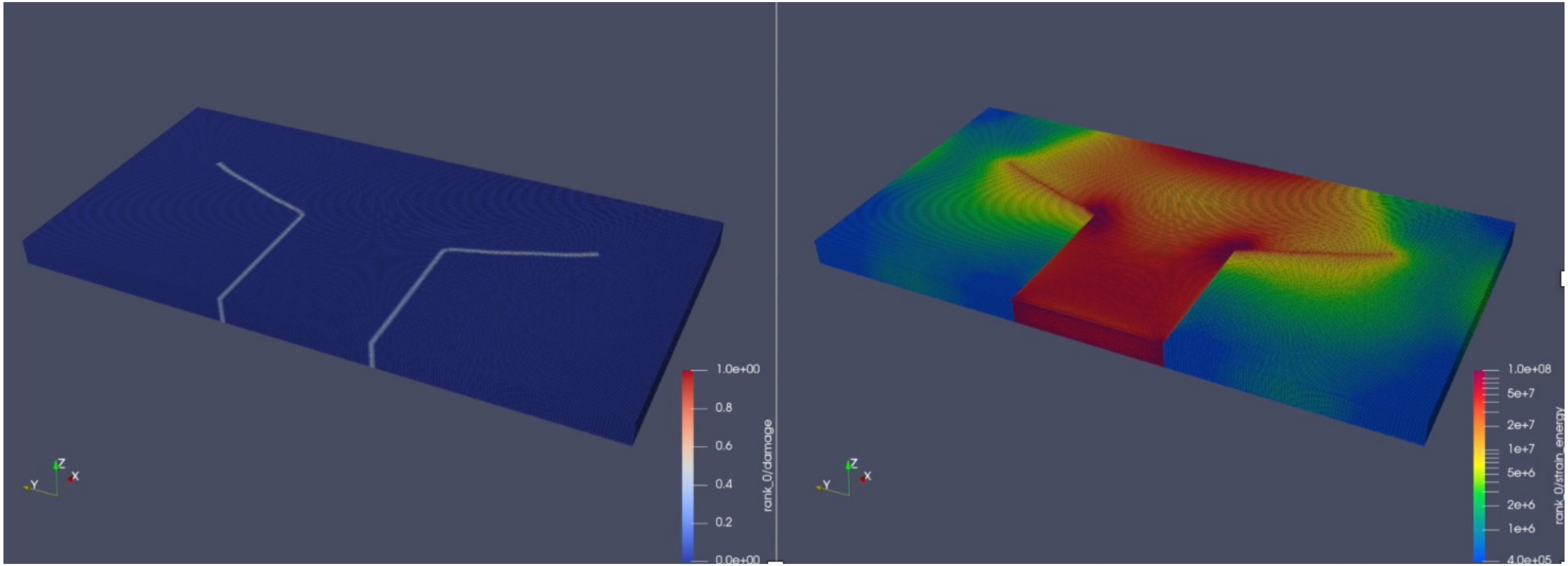
- ECP partner apps
  - XGC (WDMApp, PPPL)
    - Plasma particle-in-cell (PIC) using Cabana particles on unstructured mesh
    - Established code migrated from FORTRAN to Kokkos+Cabana
  - PicassoMPM (ExaAM, ORNL)
    - New application for additive manufacturing based on the material point method
- ECP proxy apps
  - CabanaMD (EXAALT) [LAMMPS]
  - HACCabana (ExaSky, ANL) [HACC]
    - Potential exploratory use alongside production HACC
  - ExaMPM [PicassoMPM]
  - CabanaPIC (LANL) [VPIC]

Cabana applications, continued

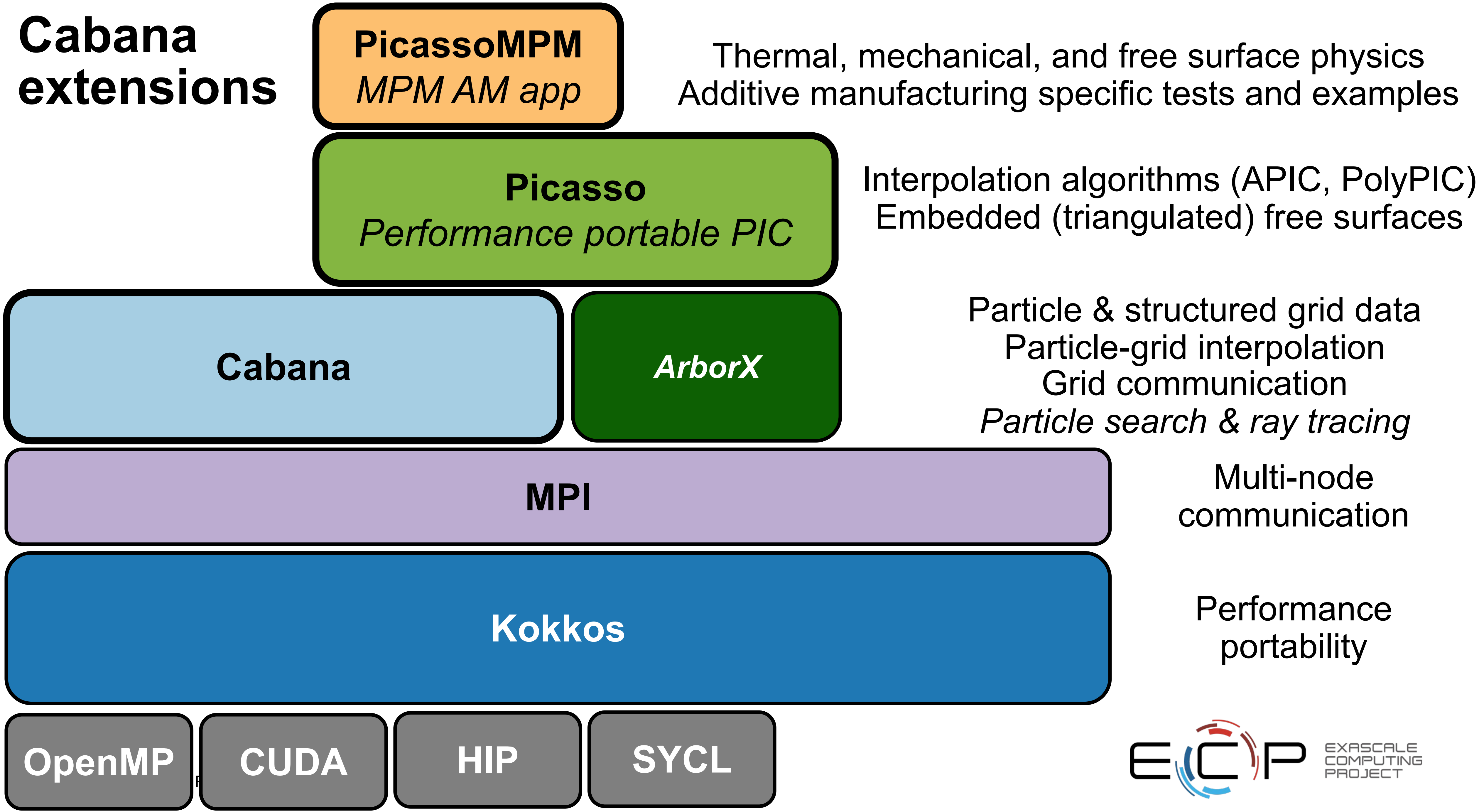
- Non-ECP applications
  - **Beatnik (UNM PSAAP):** Z-model interface solver
  - **CabanaPD (ORNL LDRD):** Peridynamics for fracture mechanics
  - **CabanaPF (ORNL SULI):** Pseudospectral phase field
  - **CabanaMPCD (Julich):** Multi-particle collision dynamics
  - **Finch (ORNL MDF):** Heat transfer for additive manufacturing
  - **Hyperion (LANL LDRD):** multi-physics hybrid PIC
  - **MRMD (Max Planck)**: multi-resolution molecular dynamics
  - **MultiSim (UCLA):** Material point method for computer graphics
  - **PUMI-PIC (RPI):** PIC library using Cabana data structures

CabanaPD *Pablo Seleson and Sam Reeve*

- New fracture mechanics code using mesh-free peridynamics
- Current extension to contact problems and more complex constitutive models



Cabana extensions



PicassoMPM
MPM AM app

Thermal, mechanical, and free surface physics
Additive manufacturing specific tests and examples

Picasso
Performance portable PIC

Interpolation algorithms (APIC, PolyPIC)
Embedded (triangulated) free surfaces

Cabana

ArborX

Particle & structured grid data
Particle-grid interpolation
Grid communication
Particle search & ray tracing

MPI

Multi-node
communication

Kokkos

Performance
portability

OpenMP **CUDA** **HIP** **SYCL**

Cabana interoperability

- Cabana has been built in order to connect and unify across disparate scientific communities (still a subset of the Kokkos universe)
- We are completely intertwined with Kokkos and MPI
  - But we do support interfaces with other portability strategies
    - heFFTe & hypre
- Primary current effort is better separation of data structure and algorithm
  - Accept **`Kokkos::View`** anywhere we use **`Cabana::slice`**
  - The particle data could be **`Unmanaged`** and therefore wrap user data without significant code intrusion

Current status and future possibilities

- ORNL science areas for Cabana-based applications
  - Fracture mechanics (fusion, materials science)
  - Plasma physics (fusion)
  - Materials and manufacturing (additive)
  - Mesh generation

- Develop at the application level
  - Generalize and move code to Cabana where appropriate
    - Generalize and move code to Kokkos where appropriate

Potential futures and collaborations

- Transition to a Kokkos project?
  - KokkosParticles:: and KokkosGrid::

- Create/cultivate motif library consortiums?
  – ECP co-design libraries”, as well as some ST libraries and other projects with similar needs and goals

- Centralized support from multiple separate applications???
  – Something like SciDAC supported by multiple offices

Moving code upstream

- From ExaAM Picasso into Cabana
  - **Cabana::particleGridMigrate** to determine particle destinations based on mesh
  - **Cabana::ParticleList** for user type-tagged particle fields
  - **Cabana::createParticles(Cabana::InitRandom, …)**
  - **Cabana::Grid::createParticles(Cabana::{InitRandom, InitUniform}, …)**
- *Planned* from CabanaPD into Cabana
  - **Cabana::particleGrid{Gather, Scatter}**
- *Potentially* Cabana::AoSoA into Kokkos
  - View extension with extra dimension for SIMD/SIMT width
  - Supports packing multiple fields together
  - Associated execution policies and parallel_for

Continuing work

- Cabana 1.0 release
  - “at the end of ECP”
  - Promote some optional dependencies from Experimental
  - Finalize **Cabana::particleGridGather**
  - Finalize **Cabana::NeighborPolicy**
  - Full support for Views in core algorithms
  - Remove deprecations (notably **Cajita** namespace wrappers)
- Beyond 1.0
  - Complete backend support in optional interfaces
  - Generalize load balancing interfaces
  - Generalize sparse grid for cases without particles
    - Consider approaches for adaptive meshes with sparse grids

Work performed under the auspices of the U.S. DOE under contract DE-AC05-00OR22725 for ORNL.

This research was supported by the Exascale Computing Project (17-SC-20-SC), a joint project of the U.S. Department of Energy's Office of Science and National Nuclear Security Administration, responsible for delivering a capable exascale ecosystem, including software, applications, and hardware technology, to support the nation's exascale computing imperative.

ECP

EXASCALE COMPUTING PROJECT

Cabana references

- Slattery, S., Reeve, S. T., Junghans, C., Lebrun-Grandié, D., et al. (2022). Cabana: A performance portable library for particle-based simulations. *Journal of Open Source Software*, *7*(72), 4115. https://doi.org/10.21105/joss.04115
- Mniszewski, S. M., Belak, J., et al. (2021). Enabling particle applications for exascale computing platforms. *The International Journal of High Performance Computing Applications*, *35*(6), 572-597.
- Qiu, Y., Reeve, S. T., Li, M., Yang, Y., Slattery, S. R., & Jiang, C. (2023). A Sparse Distributed Gigascale Resolution Material Point Method. *ACM Transactions on Graphics*, *42*(2), 1-21.
- Desai, S., Reeve, S. T., & Belak, J. F. (2022). Implementing a neural network interatomic model with performance portability for emerging exascale architectures. *Computer Physics Communications*, *270*, 108156.