SciDAC Projects

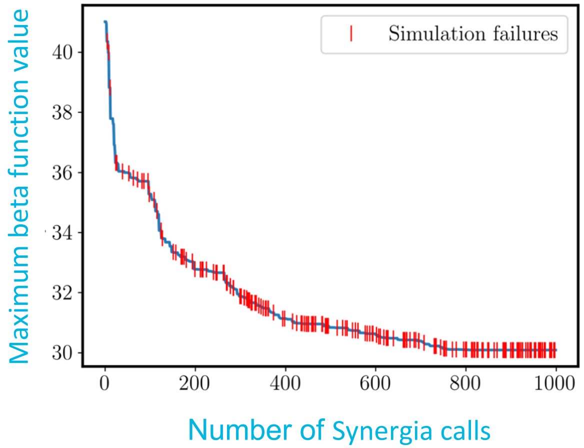
Marc Paterno
SCD Projects Meeting
16 July 2020

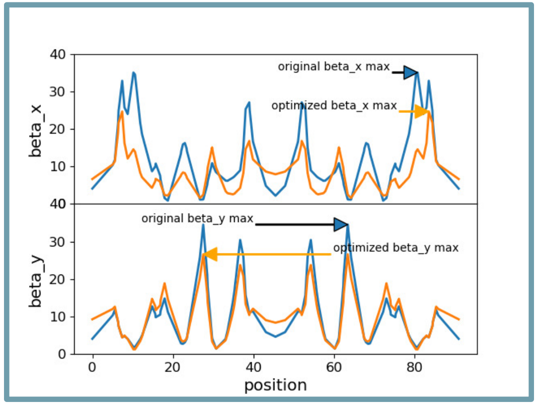
Scientific Discovery Through Advanced Computing (SciDAC)

- Overall program: https://www.scidac.gov/
  - DOE Program managers are Lali Chatterjee and Randall Laviolette
  - Combines ASCR and domain science programs
- ComPASS4: Accelerator Science and Simulation (Jim Amundson)
  - Conventional Beam Dynamics, Plasma-based acceleration
- HEP Data Analytics on HPC (Jim Kowalkowski)
  - Accelerate HEP analysis on HPC platforms with help from ASCR FASTMath and RAPIDS
  - https://computing.fnal.gov/hep-on-hpc/
- HEP Event Reconstruction with Cutting Edge Computing Architectures (Giuseppe Cerati)
  - Accelerate HEP event reconstruction using modern parallel architectures

ComPASS Accelerator Optimization

- Optimizing an accelerator design can be thought of as a massive multi-parameter problem of choosing 100s of magnet types, positions, and strengths.
- Work with ComPASS colleagues at Argonne to optimize accelerator designs by combining their POPAS (Platform for Optimization of Particle Accelerators at Scale) parallel optimizer framework with Synergia accelerator modeling to evaluate candidate designs.
- POPAS passes parameters for candidate accelerator designs to a Synergia Python script for evaluation and results are returned in a Python dict.





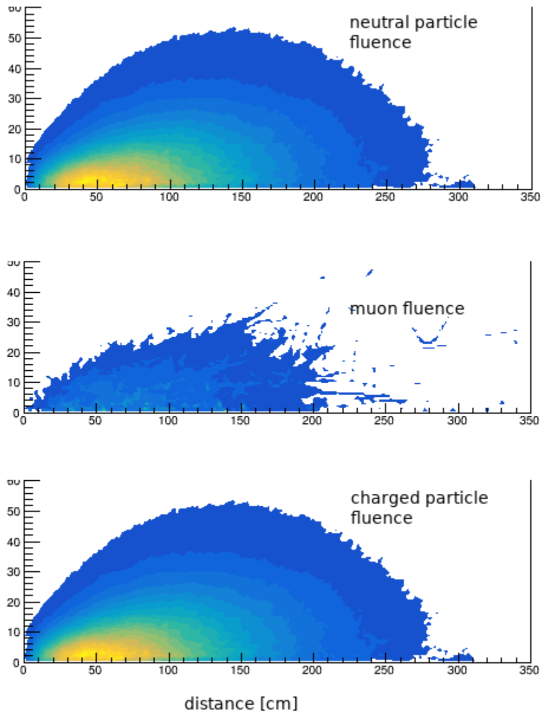
Optimization of a candidate RCS design with 6-fold symmetry with POPAS and Synergia for minimum beta functions. Continuing to optimize further designs for new 12-fold symmetry RCS.

Fermilab

Optimization with Synergia and MARS

- Optimize accelerator for maximum delivered beam with minimum uncontrolled particle loss.
- MARS is developed and maintained in the Target Systems department in AD.
- Mature program since 1974. Is **the** accepted radiation and beam loss simulation tool for the lab Rad Safety for at least 25 years.
- Lost Synergia particles are transported by MARS to generate radiation maps as ROOT histograms.
- pyROOT can read histograms and make energy deposition quantities available to POPAS.
- The next step would be to run simulate something closer to an accelerator environment that would be used for optimization.

Connect toy Synergia simulation to MARS



Fermilab

Issues moving forward

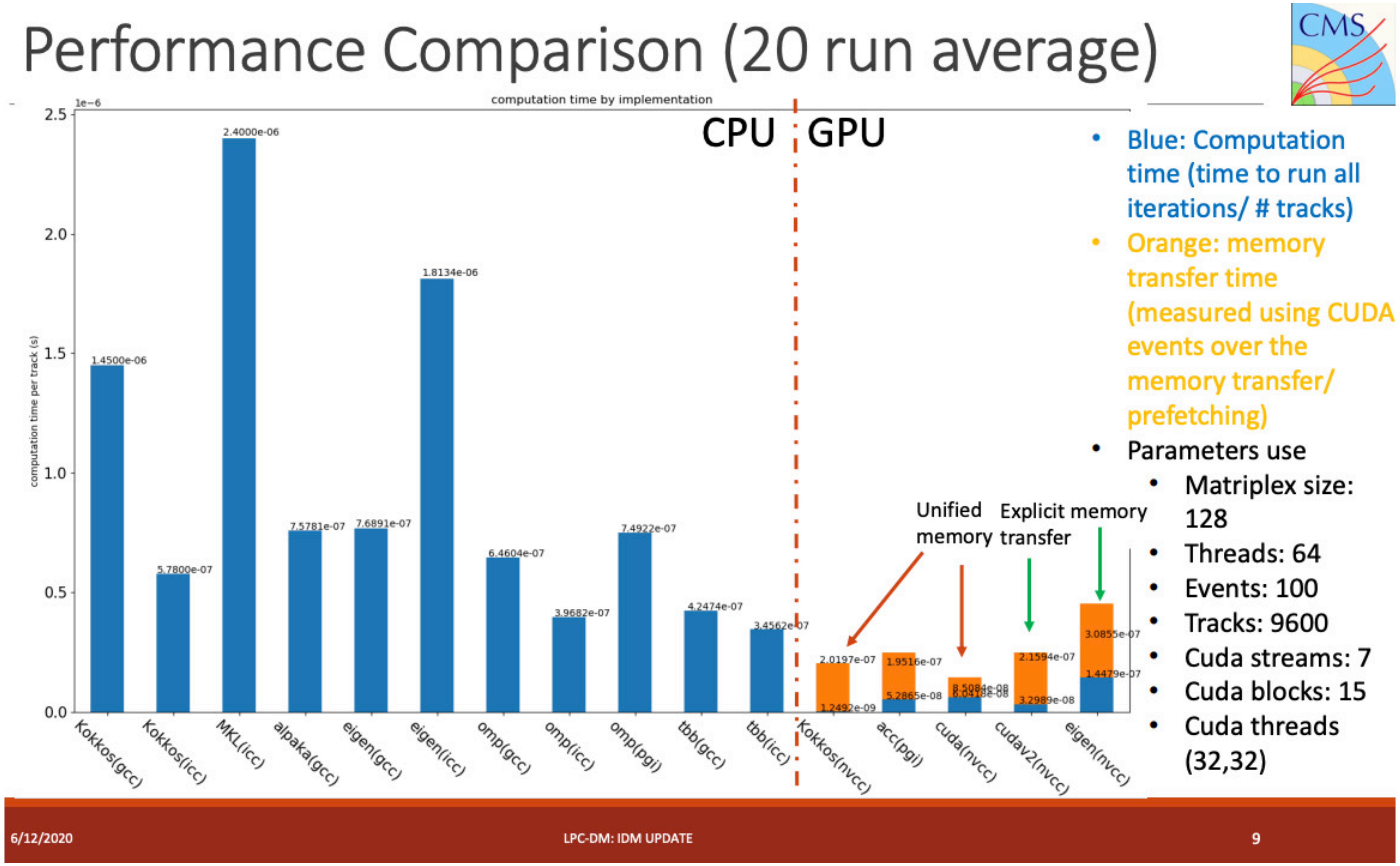
- MARS is developed and run by a small group of 4 (previously 5) people who are involved in all major lab projects. They are all overcommitted.
- MARS source code is not available. It is old Fortran code.
- Our users have difficulty finding useful documentation. We find that the support mailing list is not always helpful.
- MARS does not run on systems that Synergia runs on. MARS only executes on FermiGrid nodes.
  - This is not a technical issue; historically there had been export control issues.
  - There is a plan to be able to run MARS at NERSC but that has been delayed indefinitely.
- Our plan had been to take an existing MARS model of an accelerator component and modify it to our needs, asking for minimal support from the MARS team.
- The MARS group has made it clear that they will not assist us in this path. They tell us that we will have to construct our accelerator model from scratch by ourselves.
- This hugely increases the amount of work to progress in this effort beyond the 20% FTE that we have available.

SciDAC HepReco: News

- Presented plans for a snowmass LOI focusing on our SciDAC work, so the outcome of our studies will be an input to the planning of the future of HEP computing
- The mkFit paper received comments asking for a "minor revision" from JINST; comments were addressed and the new version has been submitted... looking forward for the notification of acceptance!
- mkFit continues to work towards improving the algorithm performance and integration/deployment in CMS workflows
- Initial exploration of code portability options is in advanced state, more next slide
- Icarus reconstruction workflow: work on spack build is continuing (now testing on a non-FNAL system), more in next week's talk by Steve White

SciDAC HepReco: Portability studies

- Slide by Tres Reid
  - Cornell student collaborating with us
- Broad survey of portability tools applied to a single kernel of mkFit is nearly complete
  - top missing piece is completing alpaka on GPU
- Large variability of results, with Kokkos and alpaka generally performing reasonably well
  - reasonably well means same order of magnitude of architecture-specific options such as TBB or cuda
  - these are the favorite options for different reasons, e.g. alpaka seems to be the most CMSSW-compatible option
- Next steps:
  - finish studies based on current kernel and produce CS paper
  - add complexity to the tested kernel (growing into a full application for e.g. fitting) and focus on selected options



Fermilab U.S. DEPARTMENT OF ENERGY Office of Science

SciDAC HEP Data Analytics on HPC

PandAna: An environment for scalable high-level HEP analysis on HPC

Achievements

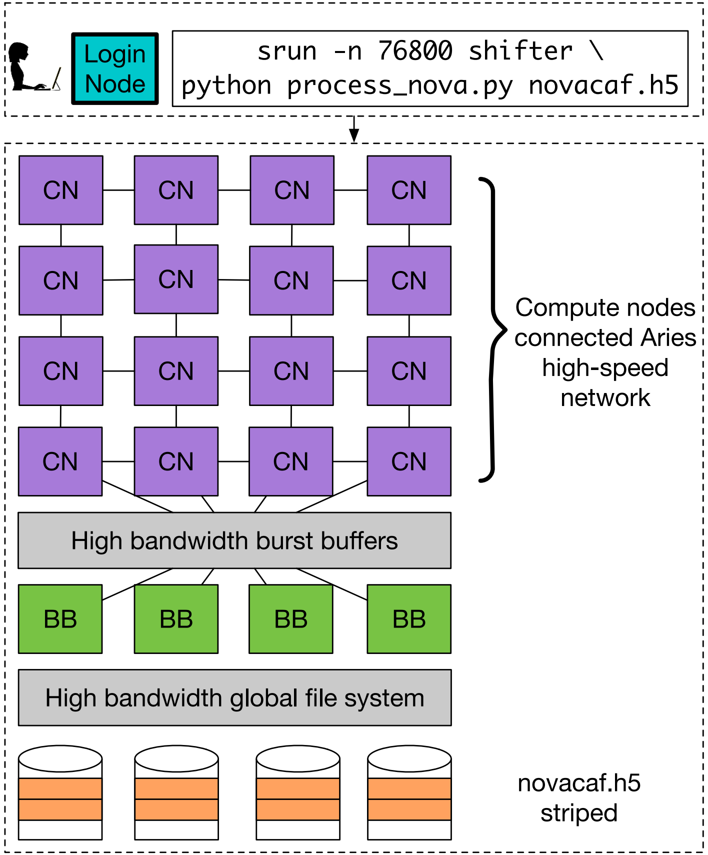
Demonstration of scalable parallelization of an analysis code from NOvA by replacing serial IO mechanism with parallel IO.

Significance and Impact

Allows existing analysis code developed by experimenters to be deployed at HPC sites for processing of large datasets.

Research Details

- Provide an easy-to-use environment for fast and scalable HEP high-level data analysis
  - Users can develop on laptops or local clusters and deploy code to HPC
- **Use HDF5** for fast parallel reading of large amounts of data
- **Use Python** and popular Python data science tools (numpy, pandas)
- **Introducing to HEP the "tidy data" analysis model**, using large data matrices and distributed data parallelism
  - Use MPI for distributed parallelism
  - The parallelism in user code is implicit



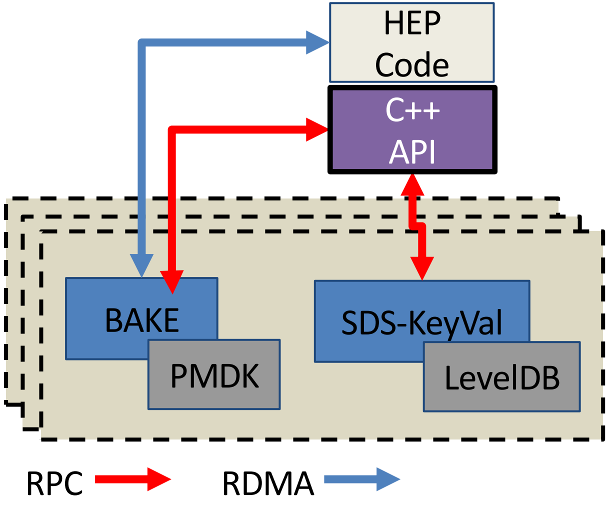
HEPnOS: Fast Event-Store for High-Energy Physics (HEP)

Goals

- Manage physics event data from simulation and experiment through multiple phases of analysis
- Accelerate access by retaining data in the system throughout analysis process
- Reuse components from Mochi ASCR R&D project

Properties

- Read in data at start of run and write results to persistent storage at the end of a campaign
- Hierarchical namespace matching physics concepts (datasets, runs, subruns)
- C++ API (serialization of C++ objects)
- Write-once, read-many



The HEPnOS event store is built using Mochi, a framework for developing specialized data services for use in HPC. Mochi allows use of state of the art libraries (e.g., LevelDB for key/value storage) while providing convenient APIs to scientists.