HEP-CCE Quarterly Report, Q2/FY20

Project Management (Habib/Calafiura):

HEP-CCE has been set up as a collaboration of four national labs with involvement from six HEP experiments and four ASCR facilities. The main focus of the management team during the first quarter has been to set up the collaboration infrastructure, and an initial set of priorities for the first year of the project. The management structure of the project is shown in Figure 1. A representative from the NSF IRIS-HEP effort will be added to the Steering Board; is is expected that a reciprocal position will be created for HEP-CCE within IRIS-HEP.

Mailing lists, a Slack channel, and a GitHub repository have been set up to facilitate internal communication and collaboration. All HEP-CCE technical meetings are open to the HEP community and accessible from indico [1]. A website hosted at Argonne is in the process of being set up.

The management team converged on the initial set of technical priorities in collaboration with the leads of the four technical areas, and the experiment POCs. Setting priorities was an iterative process that culminated in the well-attended HEP-CCE project kickoff meeting [2]. The meeting was hosted by Argonne on March 9-10 and had 42 registered participants from 13 institutions. During the kickoff meeting, experiment requirements were matched with the four sub-projects defined in the proposal, and the expertise of HEP-CCE collaborators. As a result, the "Portable Parallelization Strategies" and "Fine-Grained I/O and Storage" sub-projects were chosen as the top priorities for the first year of the project. The two sub-projects are already making progress towards their Year 1 deliverables, while onboarding more personnel. Following a phased start-up strategy, the "Complex Workflows" sub-project is getting organized and is expected to begin its activities during the second quarter. The fourth sub-project, "Event Generators", is expected to become active during the second half of the year.

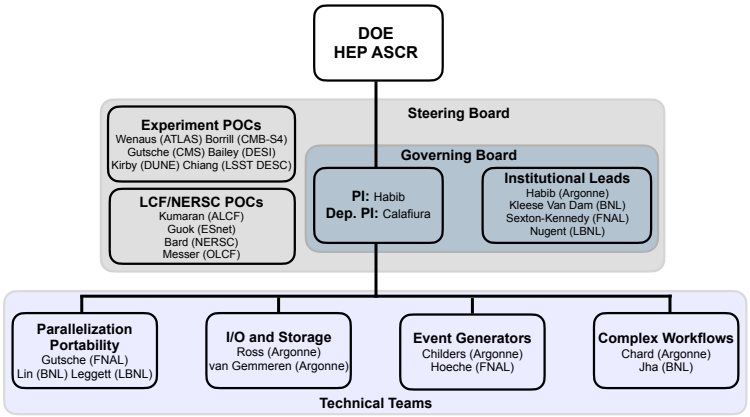


Figure 1: *HEP-CCE project organization.*

Portable Parallelization Strategies (Gutsche/Lin/Leggett):

The focus of the PPS activity is to investigate different portable programming models for the main HEP workflows in anticipation of diverse exascale architectures. The PPS working group has been having weekly meetings since January 2020, with about 10-15 regular participants of high energy physicists, software engineers, and computational scientists. A Slack channel has been set up to facilitate technical discussions. A private PPS GitHub repository is used to share documents of relevance, such as meeting notes, presentations, and instructions for obtaining, compiling and running the codes. This is in addition to the individual code repositories that already exist in their prior locations.

The goal is to provide recommendations to the experiments as to which portability solution would allow them to execute the same algorithmic code on a variety of CPU and accelerator architectures. To be able to measure the advantages and disadvantages of each programming model, we have put together a set of evaluation metrics, including software productivity, ease of use, ease of debugging, long-term sustainability, performance, etc. The working version will be made available once finalized.

We have identified three use cases to investigate code portability strategies: FastCaloSim from ATLAS, the Wire-Cell Toolkit from DUNE and Patatrack from CMS. Each of these use cases already has an implementation using the CUDA API for NVIDIA GPUs, and shown a potential to benefit from compute accelerators, a result partly due to the previous HEP-CCE funding. Initial performance profiling analysis has been carried out to identify the performance hot spots and bottlenecks in each use case. Instructions on how to obtain and run the codes on Cori [3] have been shared with the group in the PPS git repository. We have also discussed a variety different portable programming models to explore, including Kokkos, RAJA, SyCL, OpenACC and OpenMP. We have begun to investigate the use of Kokkos in all three use cases.

Fine-Grained I/O and Storage (Ross/van Gemmeren):

The goal of the Fine-Grained I/O and Storage (IOS) activity has been set up to investigate options for improving the performance and parallelism of I/O, understand the implications of these options for event data models and representations, and demonstrate promising options in real-world HEP workflows. We have identified an initial group of 15 domain and computer scientists to dive into this challenge. The group began meeting weekly, following the kickoff meeting, starting with a series of short presentations designed to foster discussion on key topics (e.g., ROOT I/O, HPC parallel I/O libraries).

Special attention will be paid to ROOT [4], which is used for I/O (including object serialization and compression) and data storage for most HEP experiments.

The team agreed that to focus on "production" workflows: Simulation, full/GEANT (CPU intense), fast (I/O intense) and including EventService, and Reconstruction. These workflows will be discussed in more details in future meetings. The team has agreed to initially target instrumentation of ATLAS production workflows running in containers on the Cori system at NERSC. The Darshan I/O characterization toolkit [5] will be used to capture salient I/O behavior at MPI and POSIX levels. Concurrently, we will continue structured discussion to learn about each other's technologies and workloads, identify gaps in instrumentation, and begin to sketch possible alternatives.

---

[1] `https://indico.fnal.gov/category/1053/`.

[2] `https://indico.fnal.gov/event/23262/`.

[3] We have decided to use Cori at NERSC as a common development platform to begin with. Access to other platforms is being planned.

[4] `https://root.cern.ch/`.

[5] `https://www.mcs.anl.gov/research/projects/darshan/`.