



Contribution ID: 183

Type: Oral

## Harnessing the power of supercomputers using the PanDA Pilot 2 in the ATLAS Experiment

*Tuesday 5 November 2019 11:00 (15 minutes)*

The unprecedented computing resource needs of the ATLAS experiment have motivated the Collaboration to become a leader in exploiting High Performance Computers (HPCs). To meet the requirements of HPCs, the PanDA system has been equipped with two new components; Pilot 2 and Harvester, that were designed with HPCs in mind. While Harvester is a resource-facing service which provides resource provisioning and workload shaping, Pilot 2 is responsible for payload execution on the resource.

The presentation focuses on Pilot 2, which is a complete rewrite of the original PanDA Pilot used by ATLAS and other experiments for well over a decade. Pilot 2 has a flexible and adaptive design that allows for plugins to be defined with streamlined workflows. In particular, it has plugins for specific hardware infrastructures (HPC/GPU clusters) as well as for dedicated workflows defined by the needs of an experiment.

Examples of dedicated HPC workflows are discussed in which the Pilot either uses an MPI application for processing fine-grained event level service under the control of the Harvester service or acts like an MPI application itself and runs a set of job in an assemble.

In addition to describing the technical details of these workflows, results are shown from its deployment on Cori (NERSC), Theta (ALCF), Titan and Summit (OLCF).

### Consider for promotion

No

**Primary authors:** NILSSON, Paul (Brookhaven National Laboratory (US)); BENJAMIN, Doug (Argonne National Laboratory (US)); Mr OLEYNIK, Danila (Joint Institute for Nuclear Research (RU)); ANISENKOV, Alexey (Budker Institute of Nuclear Physics (RU)); GUAN, Wen (University of Wisconsin (US)); JAVUREK, Tomas (CERN)

**Presenter:** NILSSON, Paul (Brookhaven National Laboratory (US))

**Session Classification:** Track 3 –Middleware and Distributed Computing

**Track Classification:** Track 3 –Middleware and Distributed Computing