HL-LHC Related Software R&D in the BNL Nuclear and Particle Physics Software (NPPS) Group (Supplementary Slides)

Torre Wenaus (BNL)

IRIS-HEP Workshop on HL-LHC Computing R&D
Catholic University of America
October 23 2019

HL-LHC software R&D in BNL NPPS

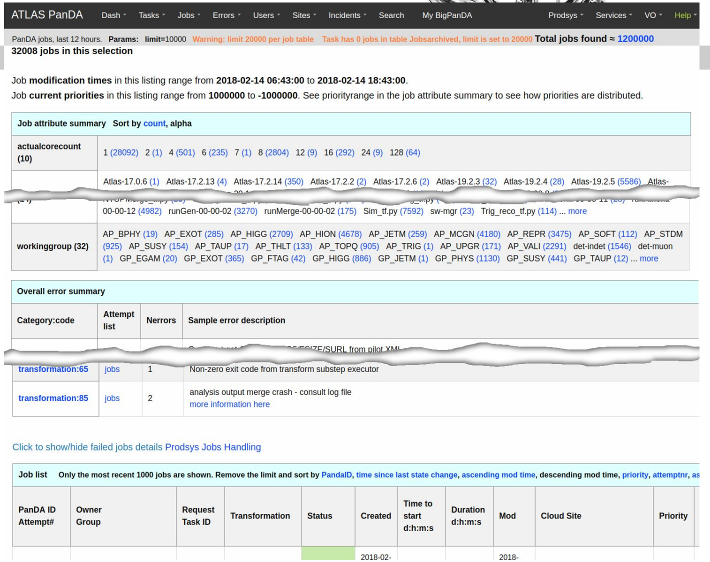
These slides summarize software R&D projects in the BNL NPPS group that aren't covered elsewhere

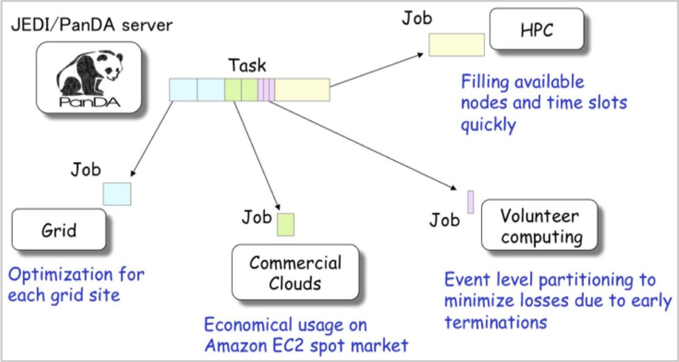
Nuclear and Particle Physics Software (NPPS) Group in the BNL Physics Department consolidates much of the NPP software development in the department

- Currently 19 people, 14 HEP, 5 NP
- ATLAS, Belle II, DUNE, EIC, PHENIX, sPHENIX, STAR
- More info at https://npps.bnl.gov

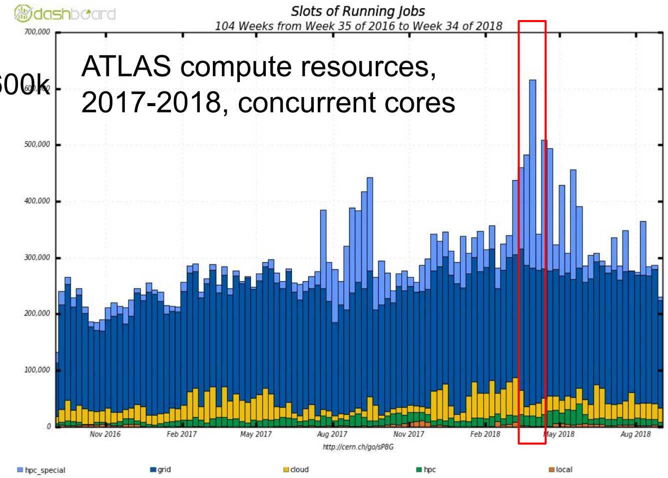
PanDA evolution

- The PanDA workload manager has evolved continuously since it entered ATLAS wide production in 2008
- Expanding capability on a stable, scalable functional foundation
- Manages ATLAS distributed computing, analysis and production
- Global orchestration of 400-600k cores 24x7 across ~140 sites, peaking > 1.2M cores
- Uses resources of all types: grid (distributed computing clusters), HPCs & LCFs (including no outbound access), clouds, volunteer computing (ATLAS@Home)
- Key features are its *scalability* and its *close coupling with large scale data management* (the Rucio data manager of ATLAS)
- Integrated workflow and dataflow moving >1 PB, >20 GB/s, 1.5-2M files per day, > 400PB, 1B files
  - Over 1.5 Exabytes processed per year
- Powerful monitoring system for operators and analysis users
- Sophisticated allocation and prioritization among dozens of activities and over 1000 individual analysis users
  - ATLAS physics coordination drives the prioritization and sharing on a weekly and sometimes daily basis
- ATLAS aggressively harvests *opportunistic resources* because our physics is compute-limited -- more cycles, more physics
  - PanDA makes this possible and manageable
- Strong *HPC support* good for DOE's strong "**use HPCs"** mandate
- Much of the rest of these slides is concerned with how we are building on the PanDA foundation with innovative R&D ultimately directed at solving HL-LHC computing challenges

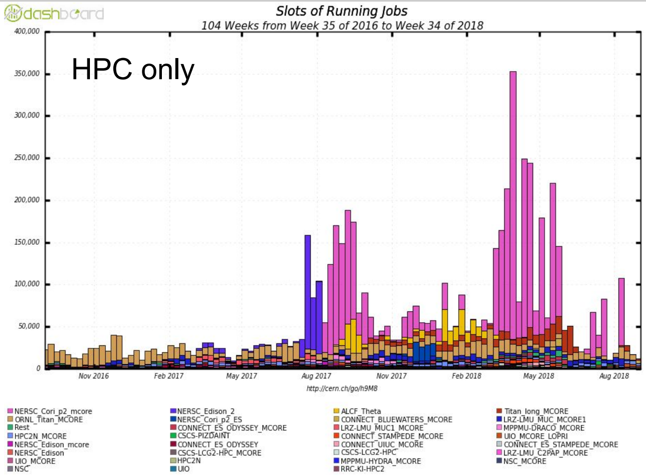




HPCs in ATLAS via PanDA



A long history but a new era from 2018: very large facilities



Light blue: “special” HPCs, where special means big, difficult to use, US DOE

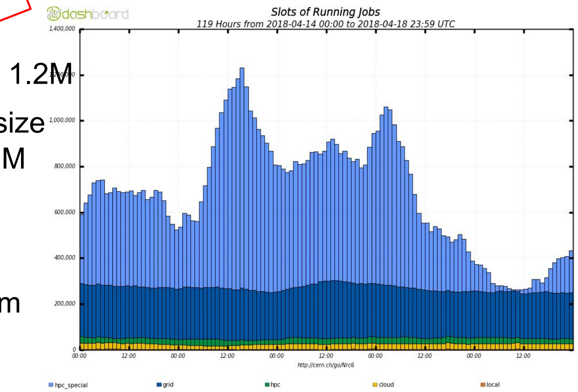
Dark blue: the grid

Yellow: cloud resources including (dominantly) HLT

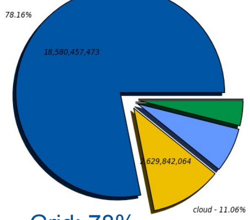
Green: “regular” HPCs, meaning easier to use, European or US NSF

Zoom showing full size of scaling peak: 1.2M concurrent cores.

Our workload management system is highly scalable!



CPU HS06 shares, 2018



Grid: 78%
Cloud, HLT: 11%
HPC special: 7%
HPC regular: 4%

BROOKHAVEN

ATLAS Event Service (AES)

Efficient processor and memory utilization
In ATLAS production for several years

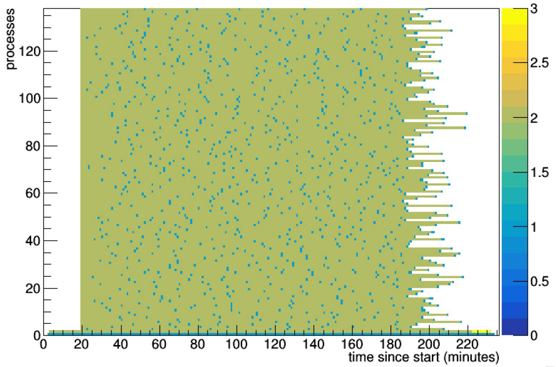
Without event service, each core processes N events. Once a core has finished its allocation, it idles (white)

With event service, each core is allocated events to process until the scheduler slot ends.

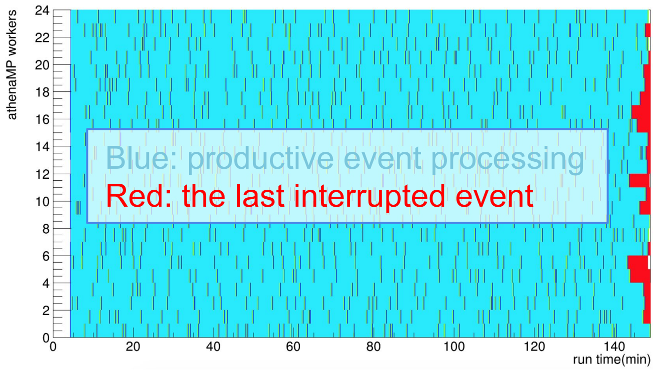
If the job is suddenly killed by preemption, all processed events are preserved except the last few minutes (all are lost in a conventional job)

Next phase: fine grained event delivery via Intelligent data delivery system (iDDS)

NERSC utilization per core, no AES

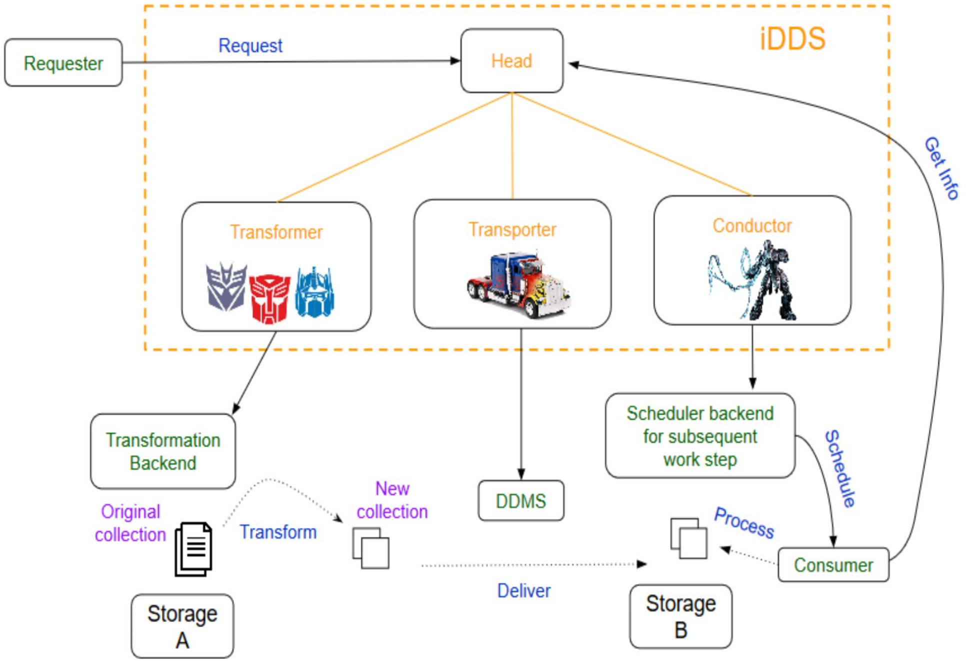


NERSC utilization per core with AES



AES elastically fills all cores for the full job lifetime

Intelligent data delivery system (iDDS)



- Experiment agnostic
- Deliver full or partial events efficiently, promptly, dynamically over the wire, leveraging caching
- Avoid up-front file replication latencies, redundant replication when work is rebrokered
- Leverage the WAN latency hiding of asynchronous fine grained data delivery
- Marshal/transform the data storage-side for efficient delivery and processing

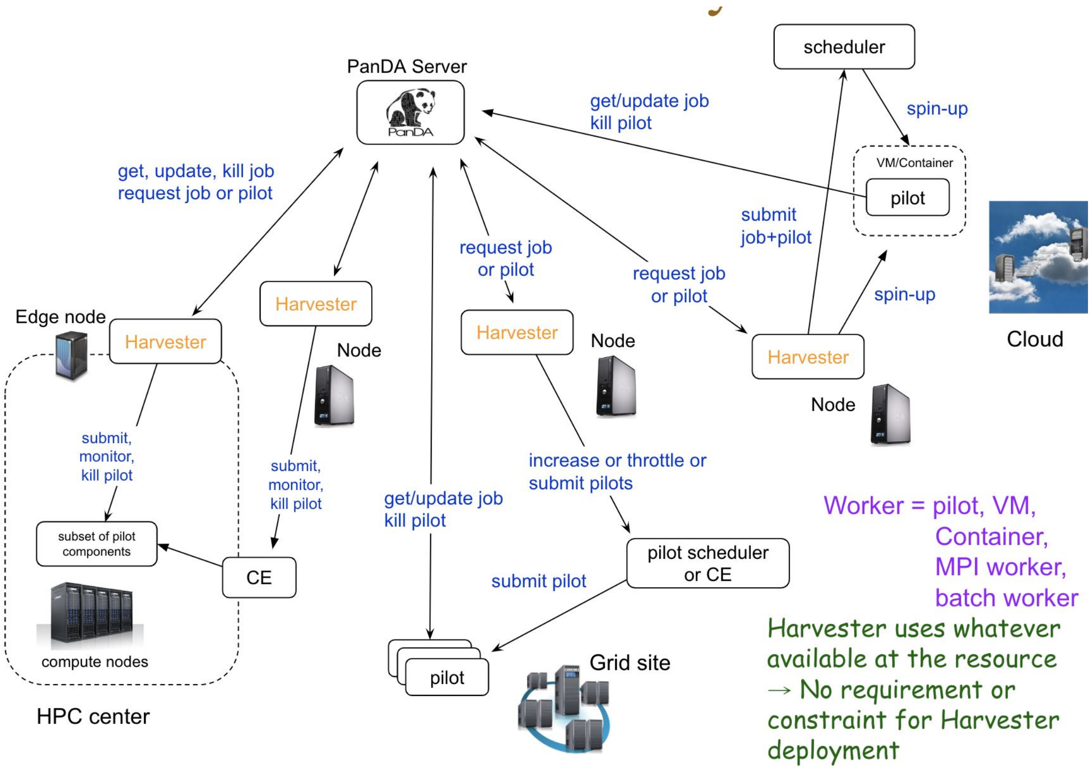
iDDS project is joint with IRIS-HEP, with interest from WLCG-DOMA, CMS

Brand new project in early prototyping, all welcome! Join up via HSF (contact me or Brian Bockelman)

Data carousel

- Orchestration of workload management, data management and storage to feed a production campaign's inputs from tape via a sliding window of disk-resident data
  - Disk footprint is only ~5-10% (perhaps) of input data set size
  - PanDA + Rucio provide the tight orchestration required
- Promising results from several rounds of trials
- Next step: explore optimizing data placement on tape for efficient retrieval by carousel
- Next year: use iDDS to dynamically stream data to remote consumers: efficiently and rapidly use distributed processing to minimize campaign completion time
- ATLAS R&D led by BNL NPPS and SDCC, with participation of ~all ATLAS Tier 1s and WLCG DOMA

Harvester



- Mediates interactions between a resource and workload/data management
- Encapsulates resource heterogeneity, presenting uniform interface to workload management and monitoring
- Incorporates via plugins the particular behaviour and functionality needed by a resource
- Particularly suited to complex HPCs but also used across the grid
- Experiment agnostic, with first usage beyond ATLAS recently reported (ASGC)

Leveraging exascale + accelerators

- **Training and optimizing deep learning neural networks** looks to be the only route for ATLAS to exploit exascale early (Summit now and new machines as they arrive).
  - **Fast simulation**, and particularly **fast chain** (fast all the way to analysis outputs)
  - **Analysis groups** already using ML
- We can expect ML applications will be well supported on future machines
  - Whatever their exotic architecture, they'll be ML engines
- Working on **scaling ML applications with PanDA** to utilize large scale resources in order to minimize turnaround time in development and tuning ML apps
  - **Distributed training and hyperparameter optimization** to achieve fast turnaround
    - Spiking workloads onto an HPC and/or across the grid for rapid turnaround
      - There's a productivity premium on rapid turnaround: physicists are waiting for the result
      - Distinctly different usage mode from steady state production, leveraging HPCs to empower developers of ML applications
    - PanDA provides large scale orchestration with management of data flows and metadata
  - Support this usage at user level, not just managed production
- In development on the grid and HPCs (Summit)
- Complement to this: PanDA now supports **concurrent CPU and GPU payloads**
  - When running GPU heavy apps like ML, complement with concurrent CPU apps to more fully utilize the resource