Automation of (big) data processing for scientific research in heterogeneous distributed computing systems.

Lessons of BigPanDA project

Danila Oleynik,
JINR LIT

Thanks to!

- **BigPanDA team:** Kaushik De, Alexei Klimentov, Jack Wells, Shantenu Jha, Sergey Panitkin, Matteo Turilli, Ruslan Mashinistov, Pavlo Svirin, Sean Wilkinson
- **PanDA core team:** Tadashi Maeno, Fernando Barreiro Megino, Paul Nilson
- **JINR LIT Team:** Vladimir Korenkov, Tatiana Strizh, Andrey Dolbilov, Valery Mitsyn, Artem Petrosyan, Anton Balandin
- **NICA Team:** Konstantin Gerzenberger, Alexey Guskov, Yuri Minaev
- And many others!

What is this talk about?

- This talk is not about BigPanDA project in details (i don’t see the way to do it in 12 minutes ;-))
  - But with some important highlights
- This talk is not about PanDA WMS, but mentioned that PanDA WMS is the core of BigPanDA project
- This is about experiences collected during BigPanDA, and how this applies to the understanding of the needs and implementation of the distributed computing ecosystem for scientific communities.

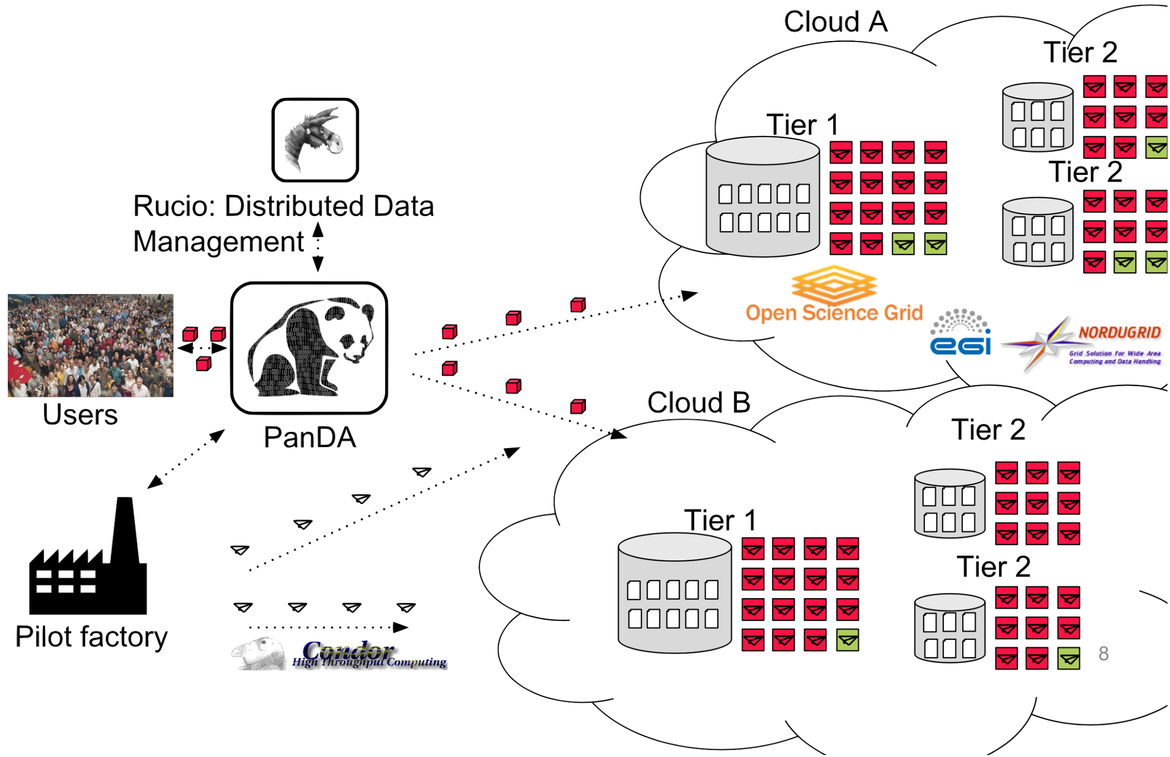
BigPanDA Project

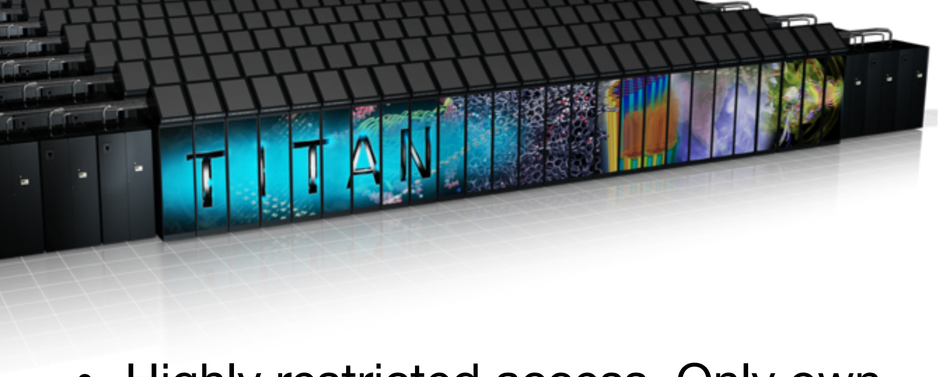
- BigPanDA was an RnD project between BNL, UTA, ORNL, Rutgers University and sponsored by DOE ASCR, with initial main aims:
  - study of possibility of integration of Leadership Computing Facilities into distributed high throughput computing infrastructure
  - enabling of usage of the same workload management system across different scientific communities
- Project was started in 2013 and successfully completed in 2019

PanDA WMS: the core of BigPanDA

- The PanDA Production ANd Distributed Analysis system has been developed by ATLAS since summer 2005 to meet ATLAS requirements for a data-driven workload management system for production and distributed analysis processing capable of operating at LHC data processing scale. ATLAS processing and analysis places challenging requirements on throughput, scalability, robustness, efficient resource utilization, minimal operations manpower, and tight integration of data management with processing workflow

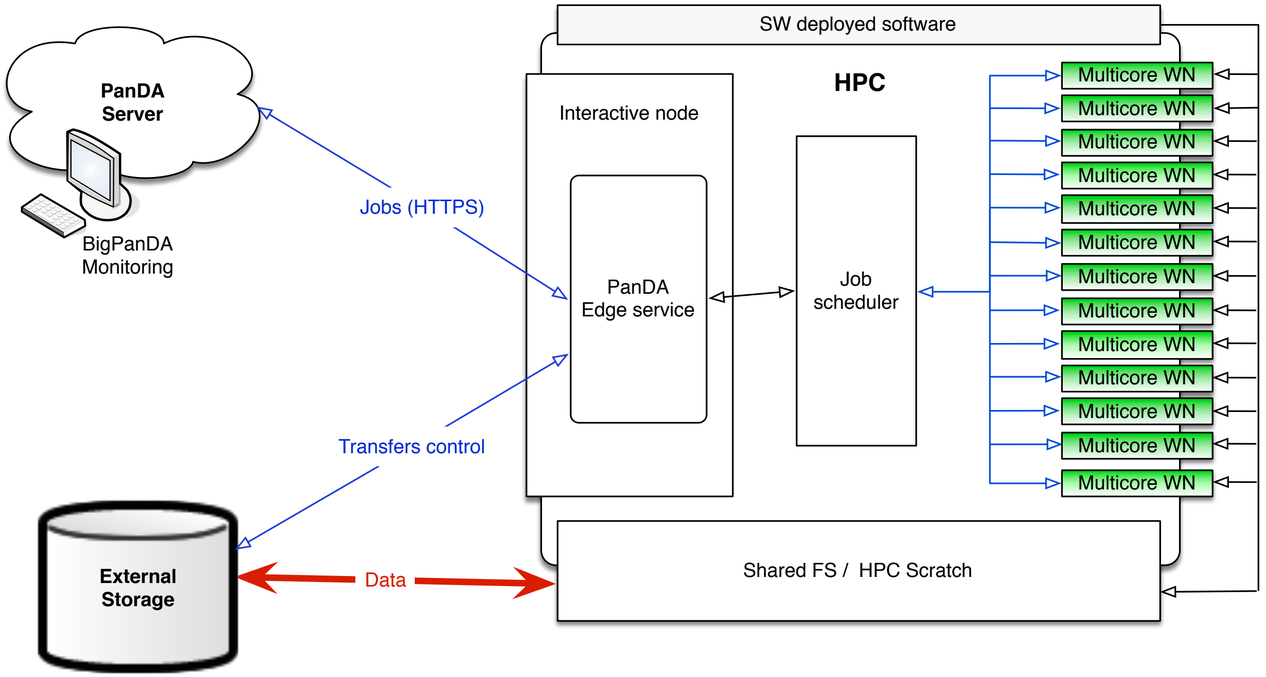
- PanDA Pilot - the execution environment (effectively a wrapper) for PanDA jobs. Pilots request and receive job payloads from the dispatcher, perform setup and cleanup work surrounding the job, and run the jobs themselves, regularly reporting status to PanDA during execution





LCF integration challenges

- Highly restricted access. Only own authentication machinery should be used (One-time password interactive authentication)
  - No network connectivity from worker nodes to the outside world. Pilot have no connectivity with external services (PanDA server etc.)
- Policy of usage: Limit on number of submitted jobs in batch queue per user and limit on number of running jobs per user
- Specialized OS (SUSE based CNL) and software stack
- Highly competitive time allocation. Geared toward leadership class projects and very big jobs



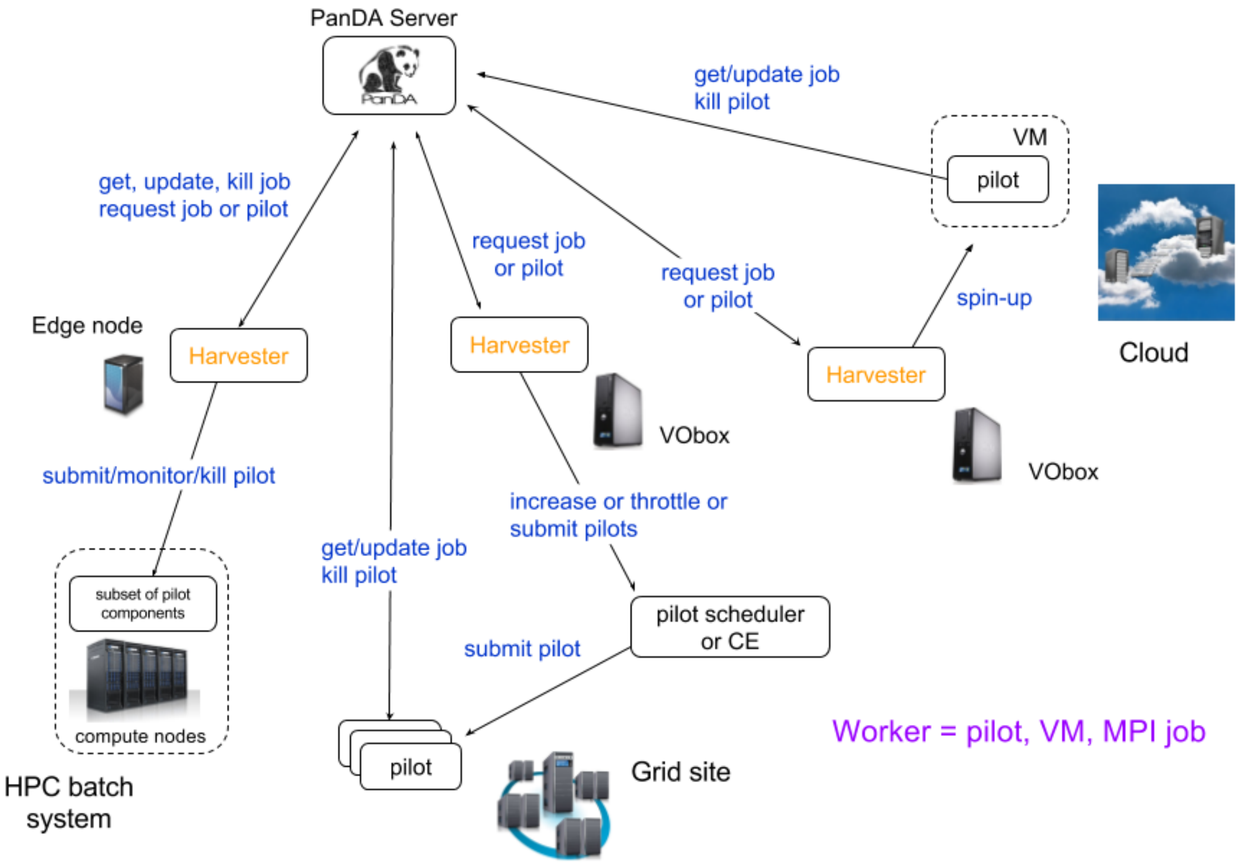
Simplified schema of integration

BigPanDA achievements

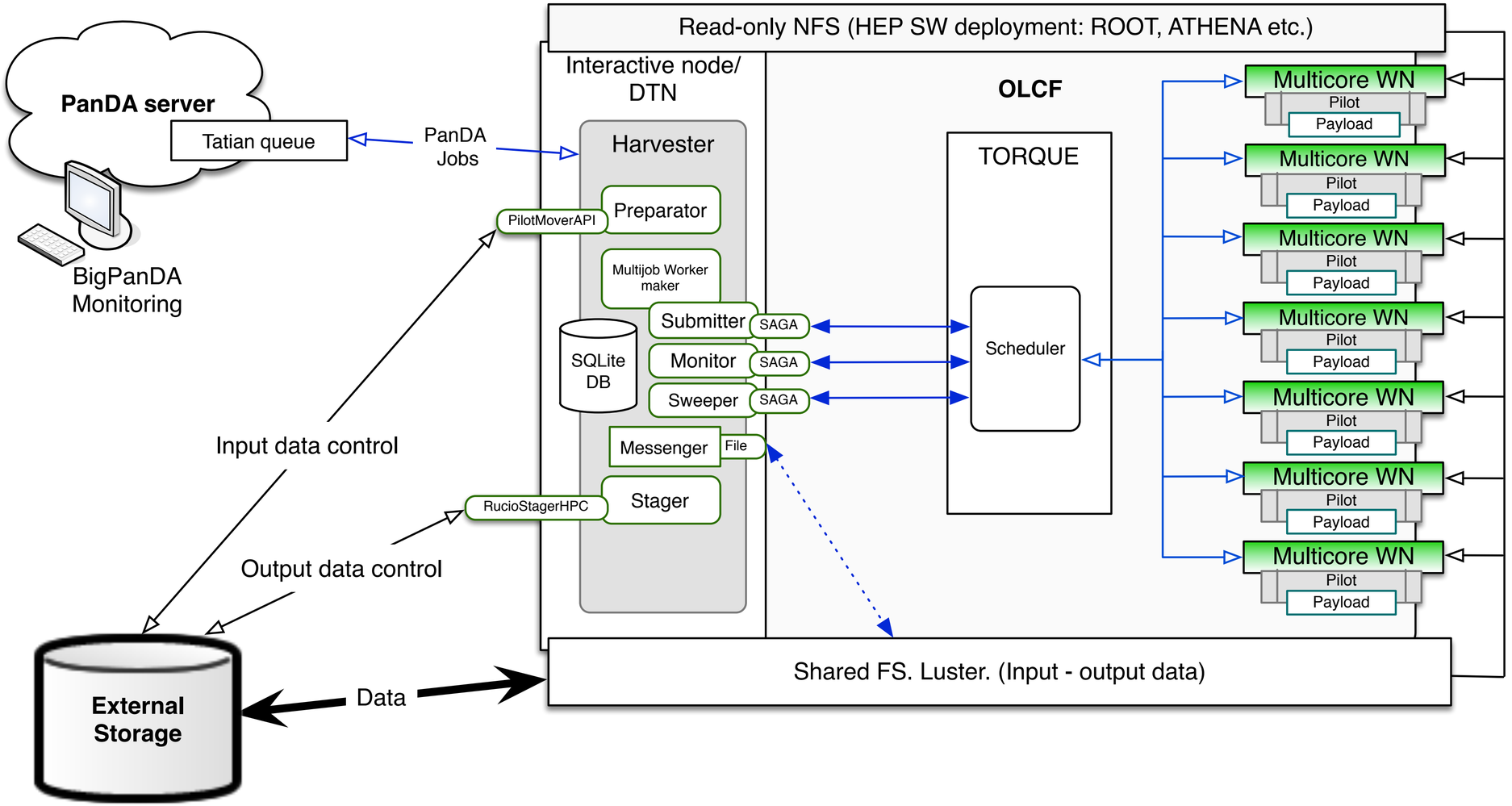
Inspiring of PanDA.

Re-thinking of treatment with computing resources

- **Harvester** is a resource-facing service between WFMS and the collection of pilots for resource provisioning and workload shaping. It is a lightweight stateless service running on a VObox or an edge node of HPC centers to provide a uniform view for various resources
- **Pilot 2** is a complete rewrite of the original PanDA Pilot which has been used in the ATLAS Experiment for over a decade. The new Pilot architecture follows a component-based approach which improves system flexibility, enables a clear workflow control, evolves the system according to modern functional use-cases to facilitate coming feature requests from new and old PanDA users.

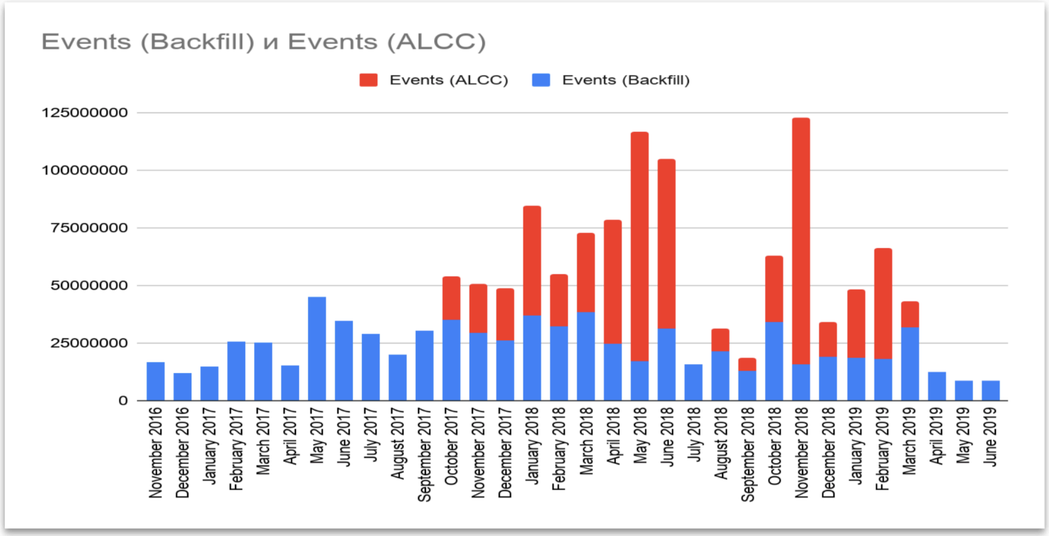


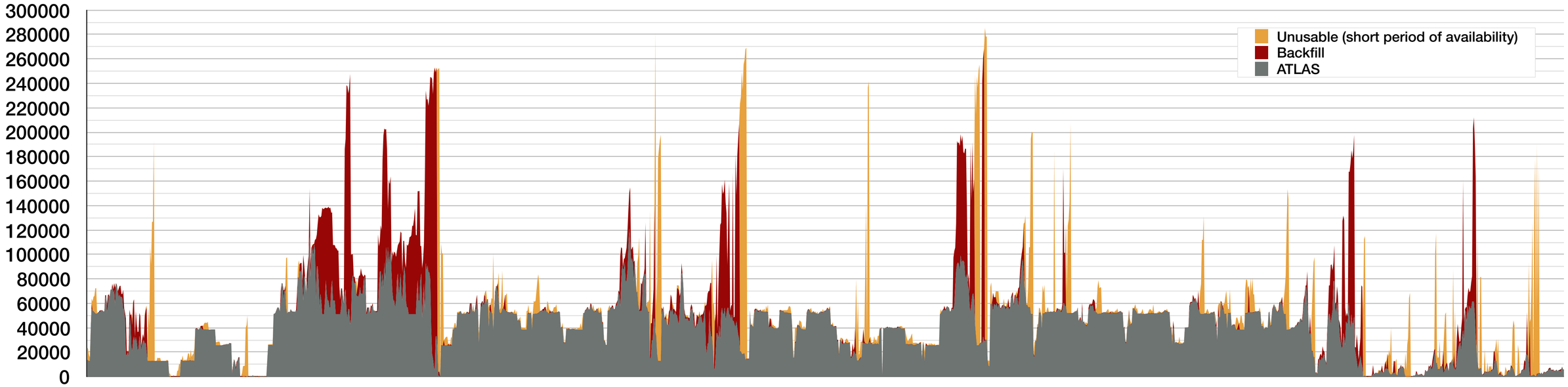
Harvester at OLCF



ATLAS at OLCF

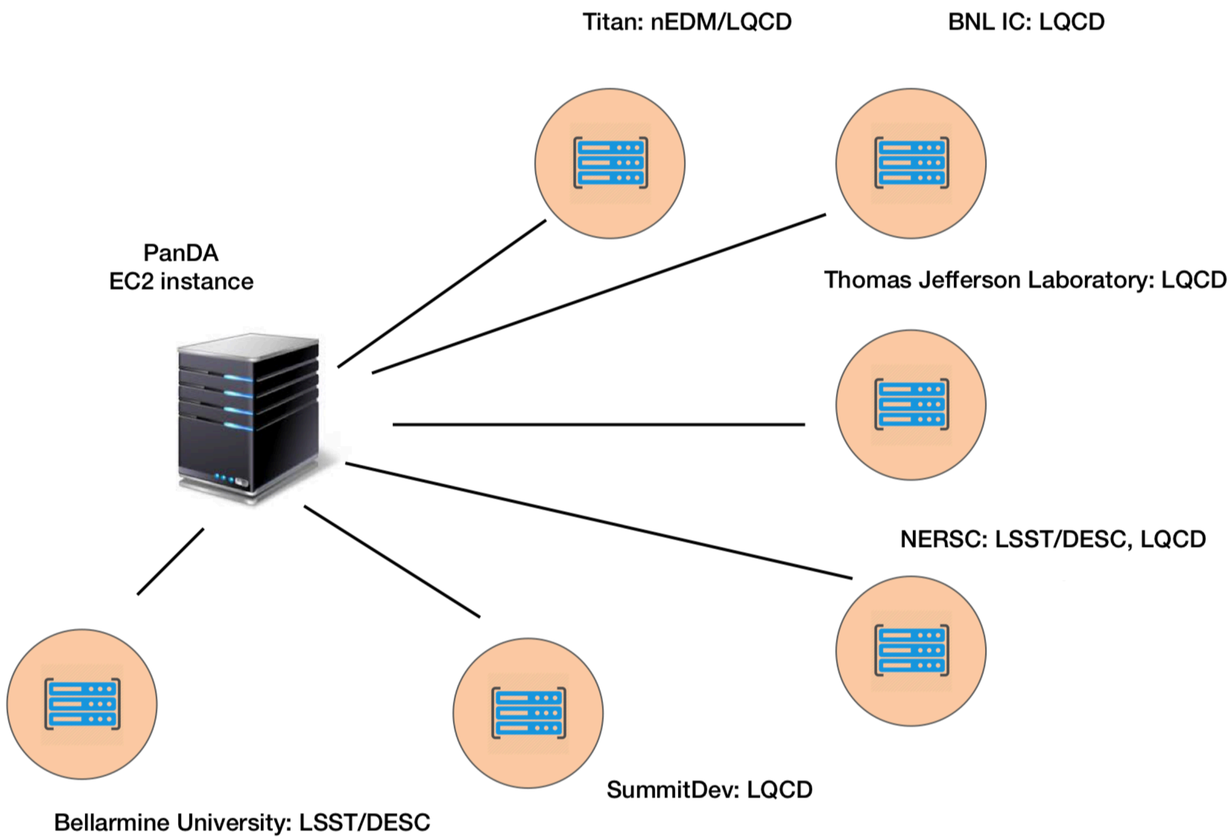
- Proof that HEP community may efficiently use of LCF's with bringing of valuable results for researchers. 1,4 billion of physics events were simulated at OLCF during production faze (2016-2019).
  - After initial success, ATLAS was granted with time allocations at OLCF, ALCF and NERSC in 2017-2019 through ALCC Program





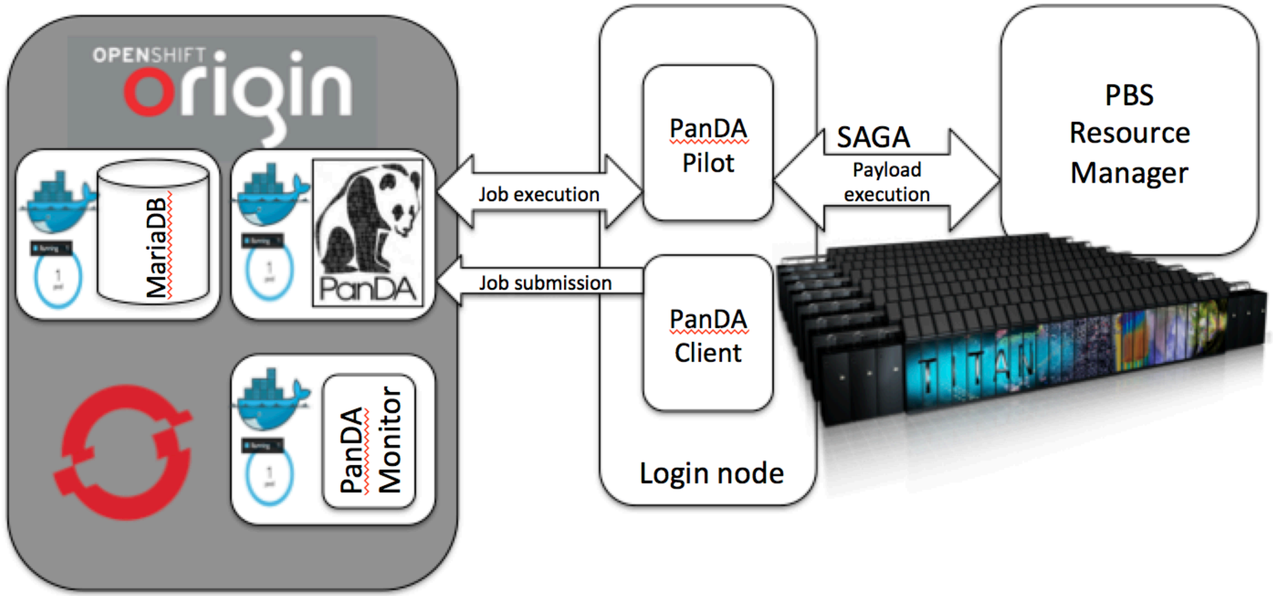
BigPanDA out of HEP
EC2 Instances

- PanDA instance was deployed on Amazon EC2 for:
  - debugging of deployment procedures and improvements for MySQL backend
  - As production instance for SciDAC-4 LQCD project, LSST/DESC and nEDM job submission to supercomputers, institutional clusters and Grids
  - This instance is used to support executions of workloads as on OLCF resources so on set of grid site



PanDA instance at OLCF

- In March 2017, PanDA WMS instance was deployed at OLCF within operating under Red Hat OpenShift Origin - a powerful container cluster management and orchestration system in order to serve various experiments at Titan supercomputer.

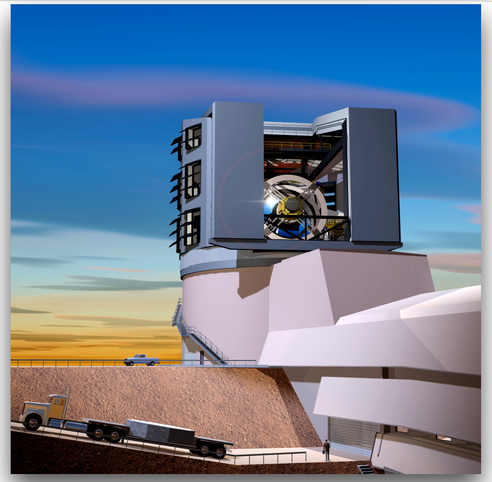


- A set of demonstrations serving diverse scientific workflows including particle physics experiments, biology studies of the genes and human brain, and molecular dynamics studies was implemented.

PanDA consumers out of HEP.

LSST - Large Synoptic Survey Telescope

- The goal of the Large Synoptic Survey Telescope (LSST) project is to conduct a 10-year survey of the sky that will deliver a 200 petabyte set of images and data products that will address some of the most pressing questions about the structure and evolution of the universe and the objects in it. The LSST survey is designed to address four science areas:
  - Understanding the Mysterious Dark Matter and Dark Energy
  - Hazardous Asteroids and the Remote Solar System
  - The Transient Optical Sky
  - The Formation and Structure of the Milky Way

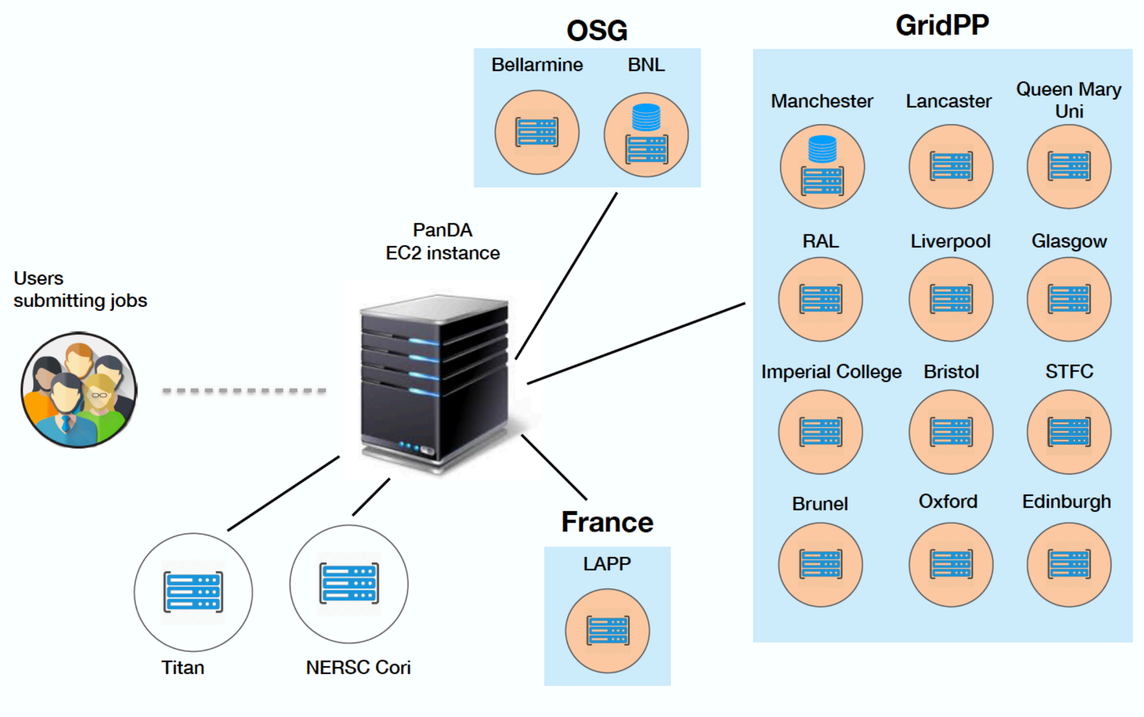


• 27-ft (8.4-m) mirror, the width of a singles tennis court
• 3200 megapixel camera
• Each image the size of 40 full moons
• 37 billion stars and galaxies
• 10 year survey of the sky
• 10 million alerts, 1000 pairs of exposures, 15 Terabytes of data .. every night!

- Collaboration with LSST/DESC since 2013 in terms of BigPanDA project
- The LSST Science Pipelines can process data from several telescopes using LSST's algorithms
- Pipeline to PanDA WMS submission has been implemented to perform standard DESC simulation workflow

PanDA WMS for LSST Dark Energy Science Collaboration

- OSG: 2 sites (BNL and Bellarmine)
- GridPP
  - 31 Grid endpoints on 12 sites configured for LSST in UK,
  - 3 endpoints in France (LAPP Annecy)
- Storage for LSST now available:
  - 7 European sites (~10 TB of transient data available, data is transferred to NERSC and removed from storage)
  - 1 US (Astro storage @BNL: 200 TB)
- Ongoing process for establishing of workflow management system above PanDA WMS

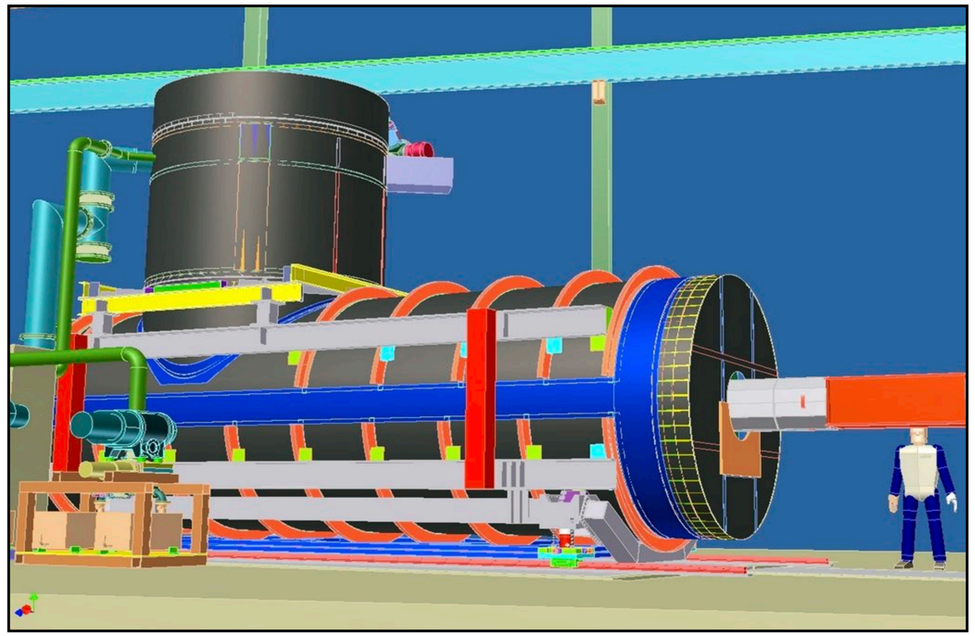


PanDA consumers out of HEP.

nEDM - Neutron Electric Dipole Moment

- Precision measurements of the properties of the neutron present an opportunity to search for violations of fundamental symmetries and to make critical tests of the validity of the Standard Model of electroweak interactions.
- The goal of the nEDM experiment at the Fundamental Neutron Physics Beamline at the Spallation Neutron Source (ORNL) is to further improve the precision of this measurement by a factor of 100. nEDM experiment requires detailed simulation of the detector.

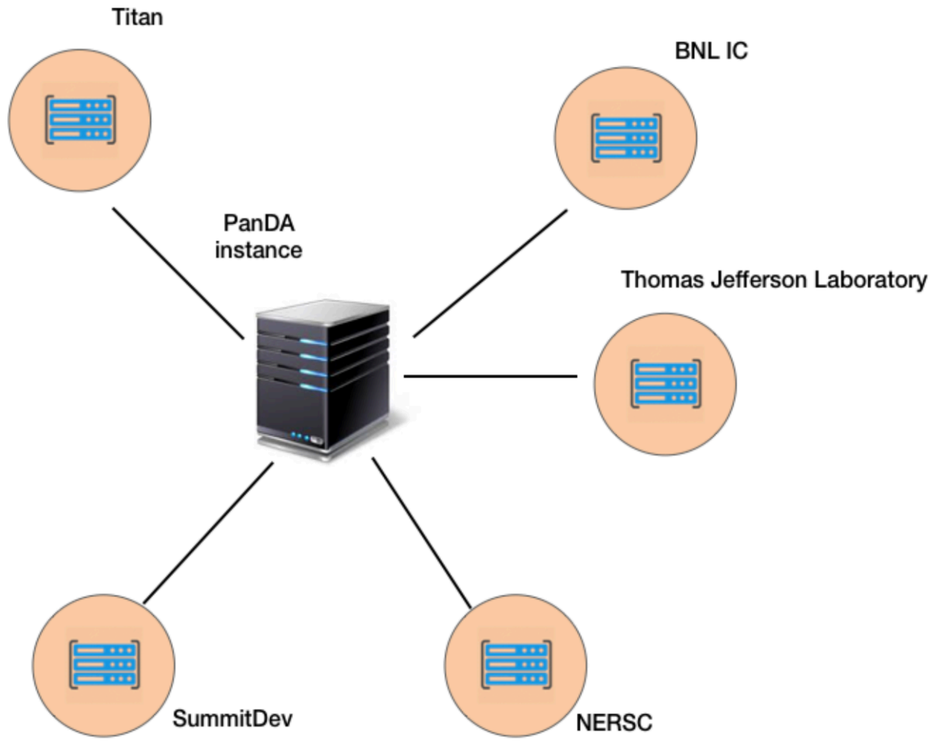
- Detailed nEDM detector simulations were executed on Titan via PanDA WMS
- nEDM PanDA migrated from EC2 instance to OLCF instance - since only OLCF resources are going to be used for processing
    - Short time of processing of events makes nEDM good candidate for backfill consumption at OLCF



PanDA WMS for Lattice QCD Computations

- In 2017, as a part of SciDAC-4 funded project, a collaboration was formed between several US LQCD groups and BigPanDA team with the goal to adopt PanDA WMS for the needs of the SciDAC-4 LQCD computational program

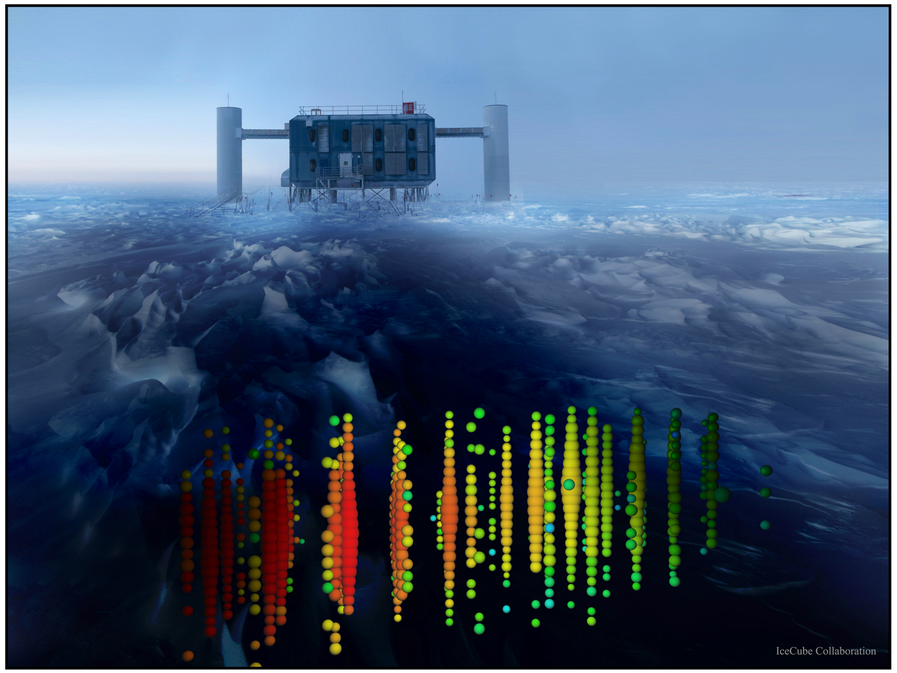
- A distributed environment for LQCD computations has been set up using PanDA Server instance deployed at the Amazon Cloud
- Variety of payloads, MPI and non-MPI:
  - HPC (Titan, Cori): GPU-based, multi-node, occupying ~8000 nodes per job, ~20 hours per each job, independent jobs
  - Institutional clusters BNL, TJL: GPU-based, single-node, ~12 hours each, with workflow management
  - New kinds of payloads will be available for Summit in the future



IceCube Neutrino Observatory

PanDA WMS for IceCube

- Together with experts from the experiment we were working on the demonstration of a real IceCube workflow Titan
  - Application: NuGen package - GPU application for atmospheric neutrinos simulations and analysis
  - Application packed in Singularity container.
  - Whole node, but not MPI application:
- ~45000 jobs in campaign (5000 input files)
  - Remote stage-in/-out the data from GridFTP storage with GSI authentication

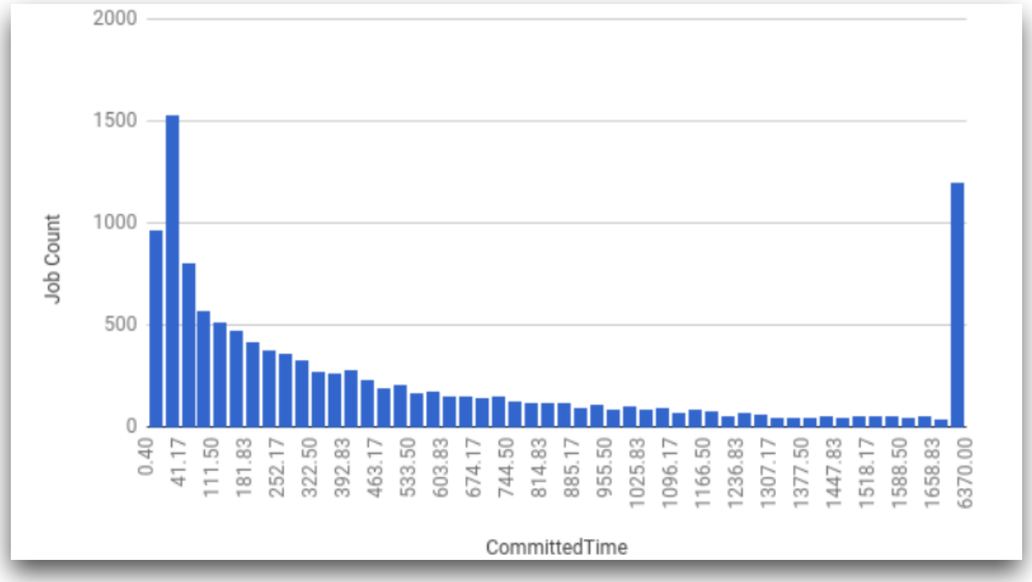


The IceCube Neutrino Observatory is the first detector of its kind, designed to observe the cosmos from deep within the South Pole ice. An international group of scientists responsible for the scientific research makes up the IceCube Collaboration

The IceCube detector is one cubic kilometre of ice —that would be enough water to fill one million swimming pools.

IceCube payloads on Titan

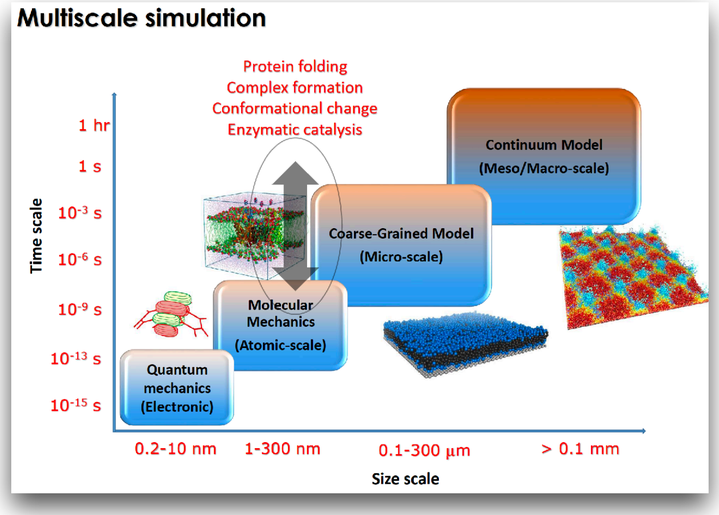
- IceCube payloads were not designed initially for support of MPP, so execution of this kind of payload in the traditional way of HPC will be inefficient (one node per job is not good for LCF's policies)
- PanDA on HPC allows to combine this kind of payload into assemblies and executes them as a simple MPI application (it was required to get efficient execution of ATLAS payloads)
  - To be effective this approach is required to join payloads with similar wall time into assemblies
  - Unfortunately, IceCube payloads do not have this type of characterization

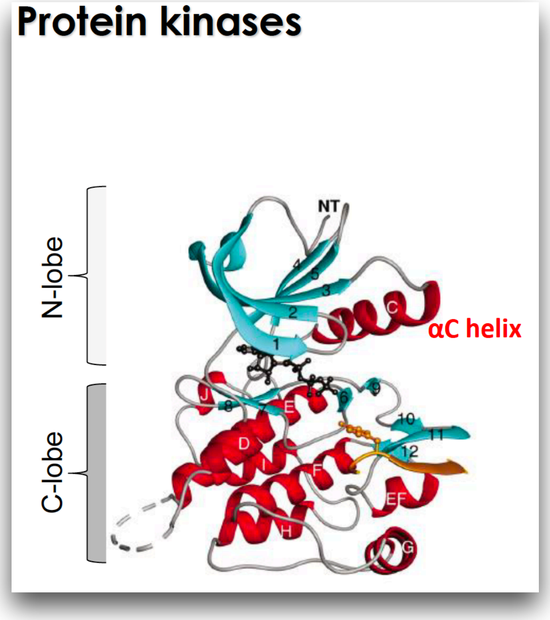


- To optimize allocation usage, the processing will be conducted in a few steps.
- On each step, we will run multiple job processes in parallel via MPI wrapper
- Starting with the walltime=20min on the first step, all failed jobs (jobs not completed because of the walltime limit) will be resubmitted on the next step with longer wall-time

PanDA WMS for Molecular Dynamics

- Simulating Enzyme Catalysis, Conformational Change, and Ligand Binding/Release. (Prof. Kwangho Nam (University of Texas at Arlington, USA)

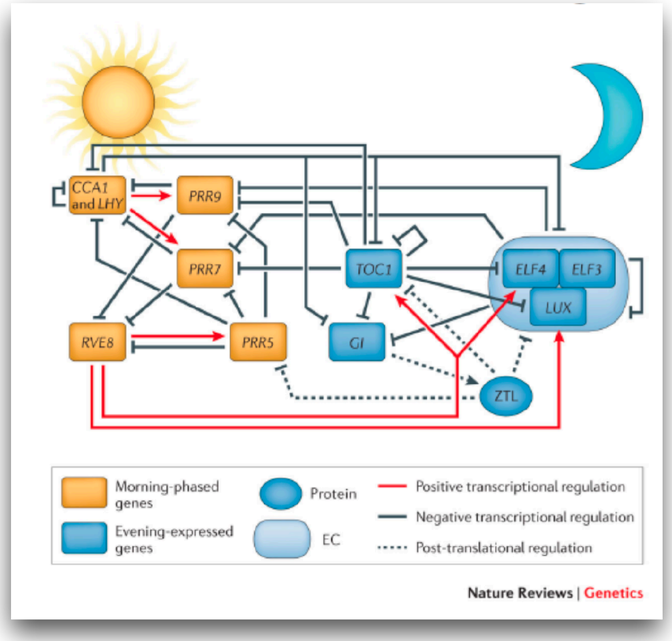




- CHARMM (Chemistry at HARvard Macromolecular Mechanics) payload (hybrid MPI/OpenMP/GPU) example built and executed on Titan
- Depending on the type of projects, payloads can expand beyond 500 nodes on Titan; currently, it uses 60-124 nodes for each project

PanDA WMS for Biology / Genomics

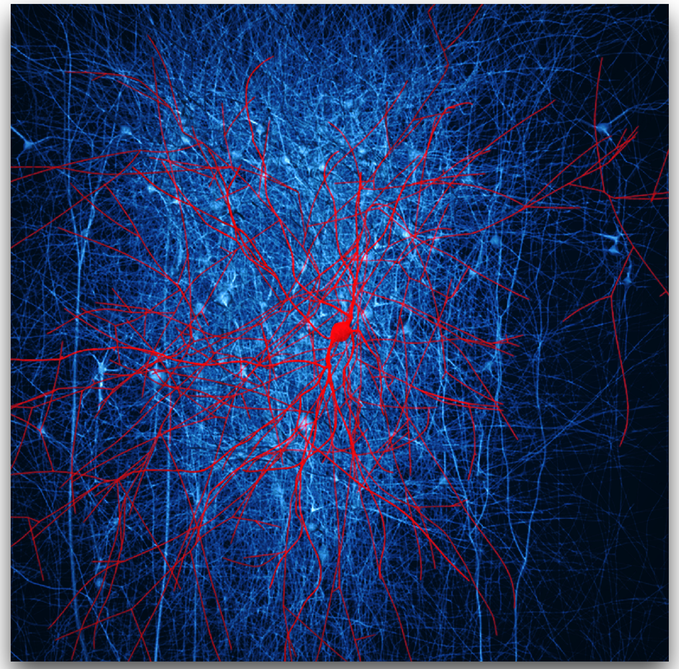
- In collaboration with Center for Bioenergy Innovation at ORNL, the PanDA based workflow for epistasis research was established. Epistasis is the phenomenon where the effect of one gene is dependent on the presence of one or more 'modifier genes', i.e. the genetic background.
- The GBOOST application, a GPU-based tool for detecting gene-gene interactions in genome-wide case control studies, was built and tested on Titan with PanDA as an example
- Input data were located in a set of eight input directories of 152 M each. Every PanDA job was configured to process single input directory in backfill mode on one node and walltime of 30 min. The output data of about 11M per job was located to the corresponding output dir



- Genetic regulatory networks
  - Phenotypes, or traits, are determined by genetic regulatory networks
  - These genetic networks are composed of genes that are organized to coordinate overt phenotypes
  - In quantitative genetic analyses, genes underlying the basis of traits are formalized

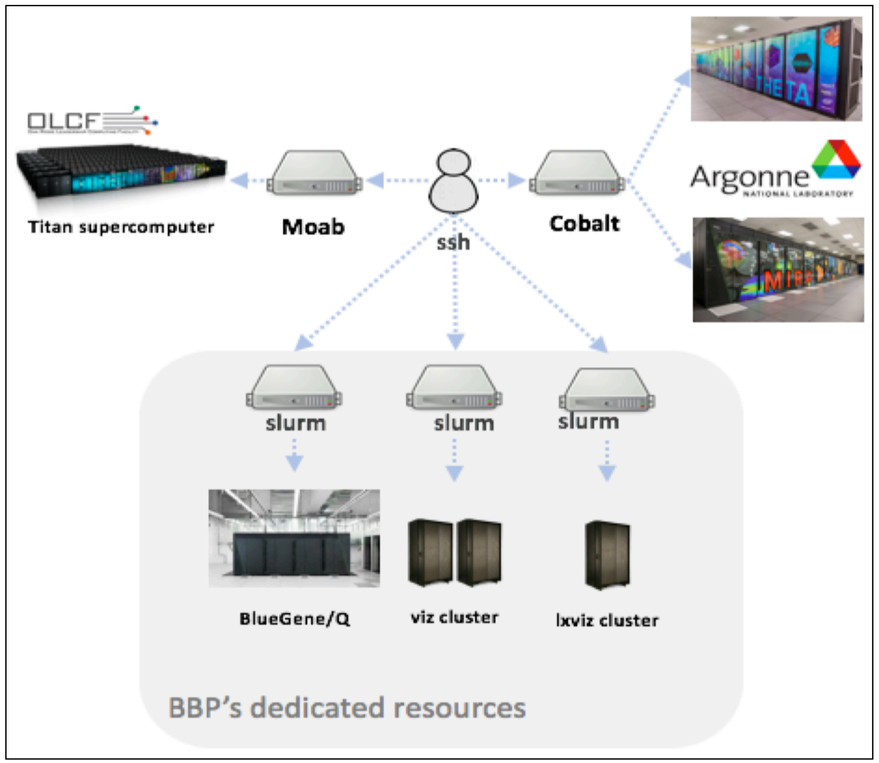
PanDA WMS for Blue Brain Project

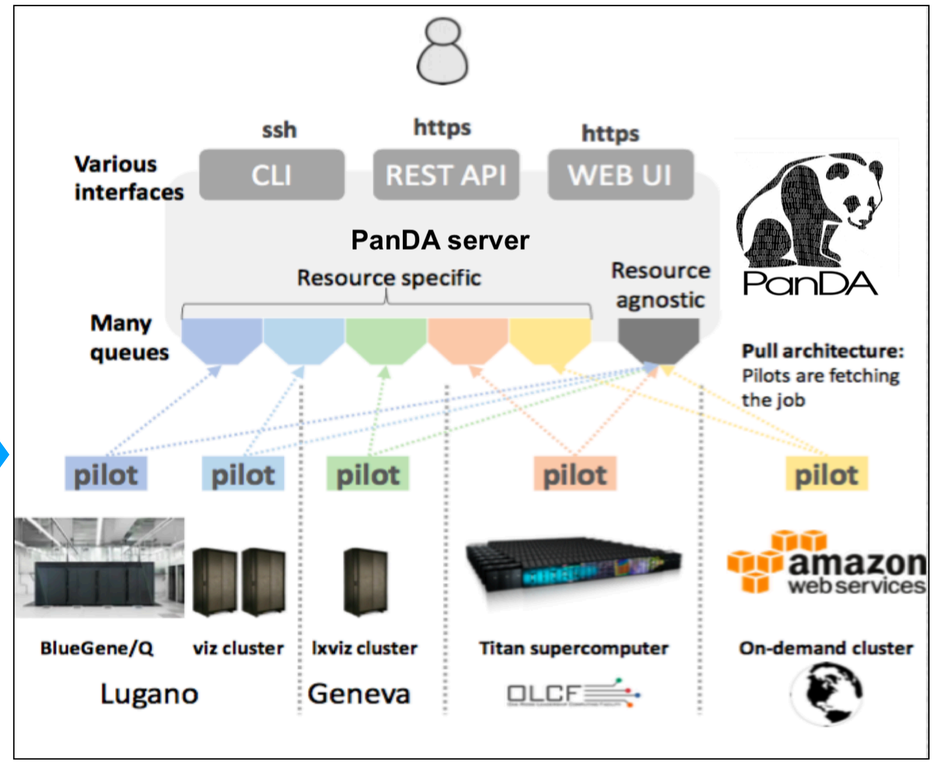
- The goal of the Blue Brain Project is to build biologically detailed digital reconstructions and simulations of the rodent, and ultimately the human, brain
- The supercomputer-based reconstructions and simulations built by the project offer a radically new approach for understanding the multilevel structure and function of the brain
- The novel research strategy allows the project to build computer models of the brain at an unprecedented level of biological detail
- Supercomputer-based simulation turns understanding the brain into a tractable problem, providing a new tool to study the complex interactions within different levels of brain organization and to investigate the cross-level links leading from genes to cognition



The presynaptic neurons of a layer 2/3 nest basket cell (in red) were stained (in blue) in a digital reconstruction of neocortical microcircuitry. Only immediate neighbouring presynaptic neurons are shown.

Blue Brain Project computing model evolution with PanDA WMS





- Dedicated PanDA instance on VM hosted by BBP
  - Improved PanDA User Interface: from CLI to Web interface
  - Integration with BB authentication system

BigPanDA out of HEP. Summary

- Seven subprojects with scientific groups from different sciences fields appeared during BigPanDA project
  - + great success with the automation of data processing for COMPASS experiment at CERN (https://indico.jinr.ru/event/738/session/6/contribution/187) and DNA study in Kurchatov institute (https://4science.ru/events/sfy2016/experts/07b1bfc63997439482b75d78baa29e06)
- Enough to make some conclusions:
  - There are no issues with the deployment of a workload management system to get unified access to distributed computing resources
  - No issues with the processing of thousands of jobs :-)
  - But, usually external collaborator does not aware about HTC paradigm, distributed computing etc., and some questions are raised:
    - How to organize dataflow driven processing?
    - How to automate processing steps?
    - How to manage data in a distributed environment?
    - Are there any solutions exists?

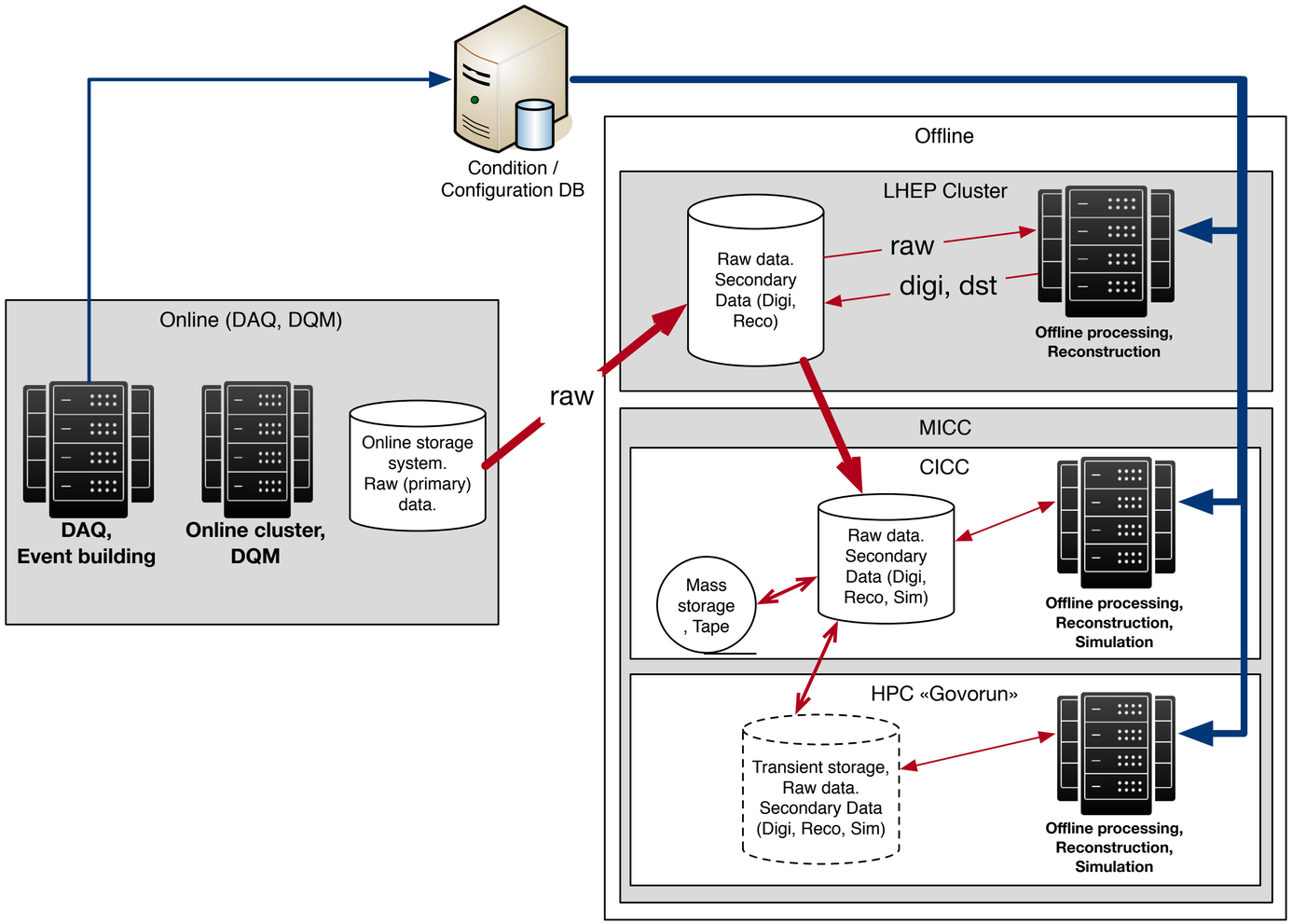
Computing infrastructure for current and future experiments at JINR

A lot of talks about computing infrastructure at JINR this week.

- To make long story short: we already have own heterogeneous distributed computing infrastructure which includes: dedicated on-line and off-line data processing facilities for NICA experiments, Central Information and Computer Complex, HPC, quite big data storage facility with multi-protocol access
  - We will need to have a subset of services which will allow to orchestrate of our facilities
- Variety of researches in JINR increases, and each of them already have or will have own requirements
  - Orchestration should be flexible enough to cover these requirements

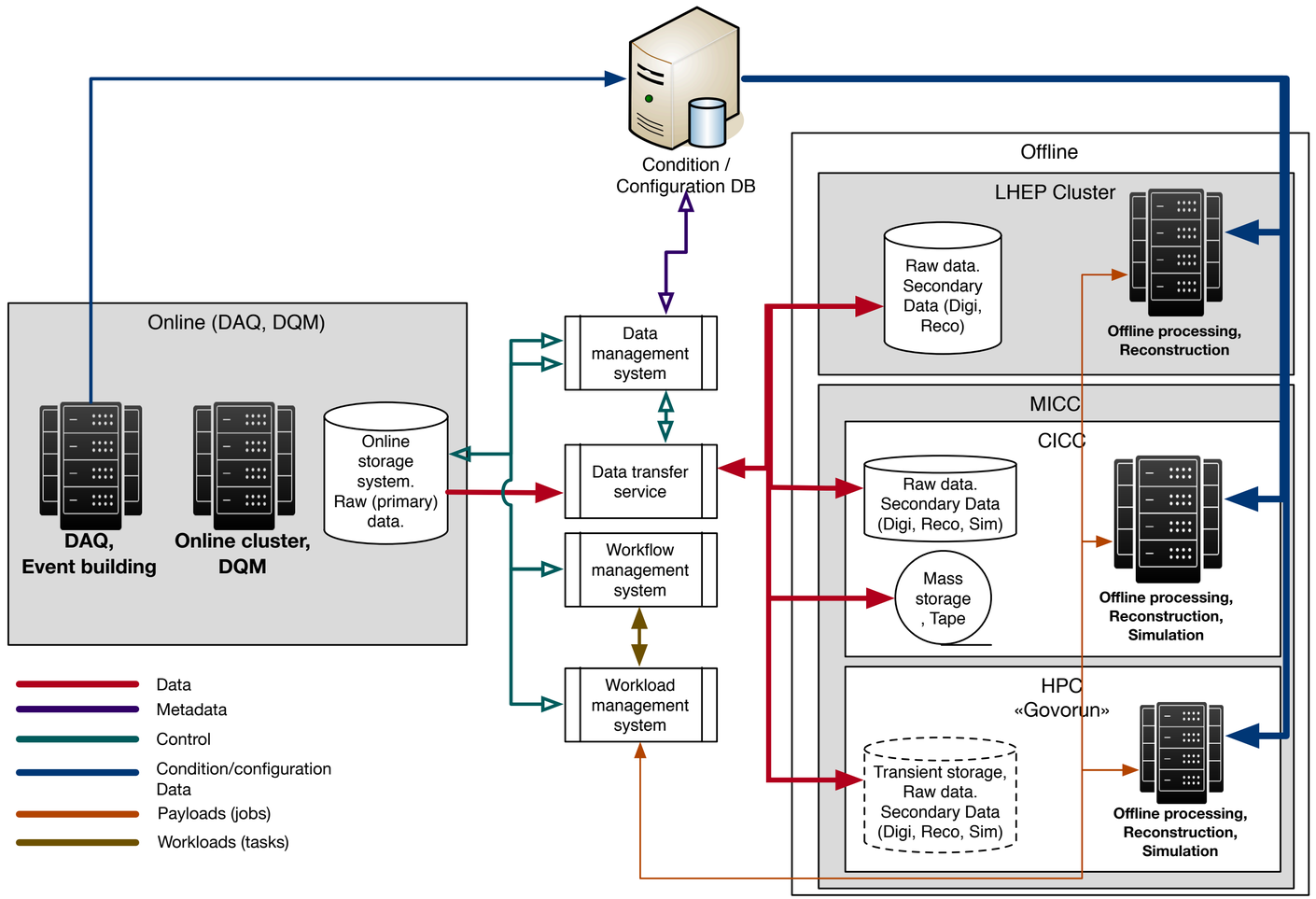
Use case: BM@N events reconstruction

- Raw data is produced by DAQ of the detector and stored on the online storage system
  - Initial processing of data (DQM) started on “on-line” resources (dedicated cluster)
- Relevant raw data should migrate to permanent storage and to storages which close to computing facilities
- Data should be processed and results stored for future analysis



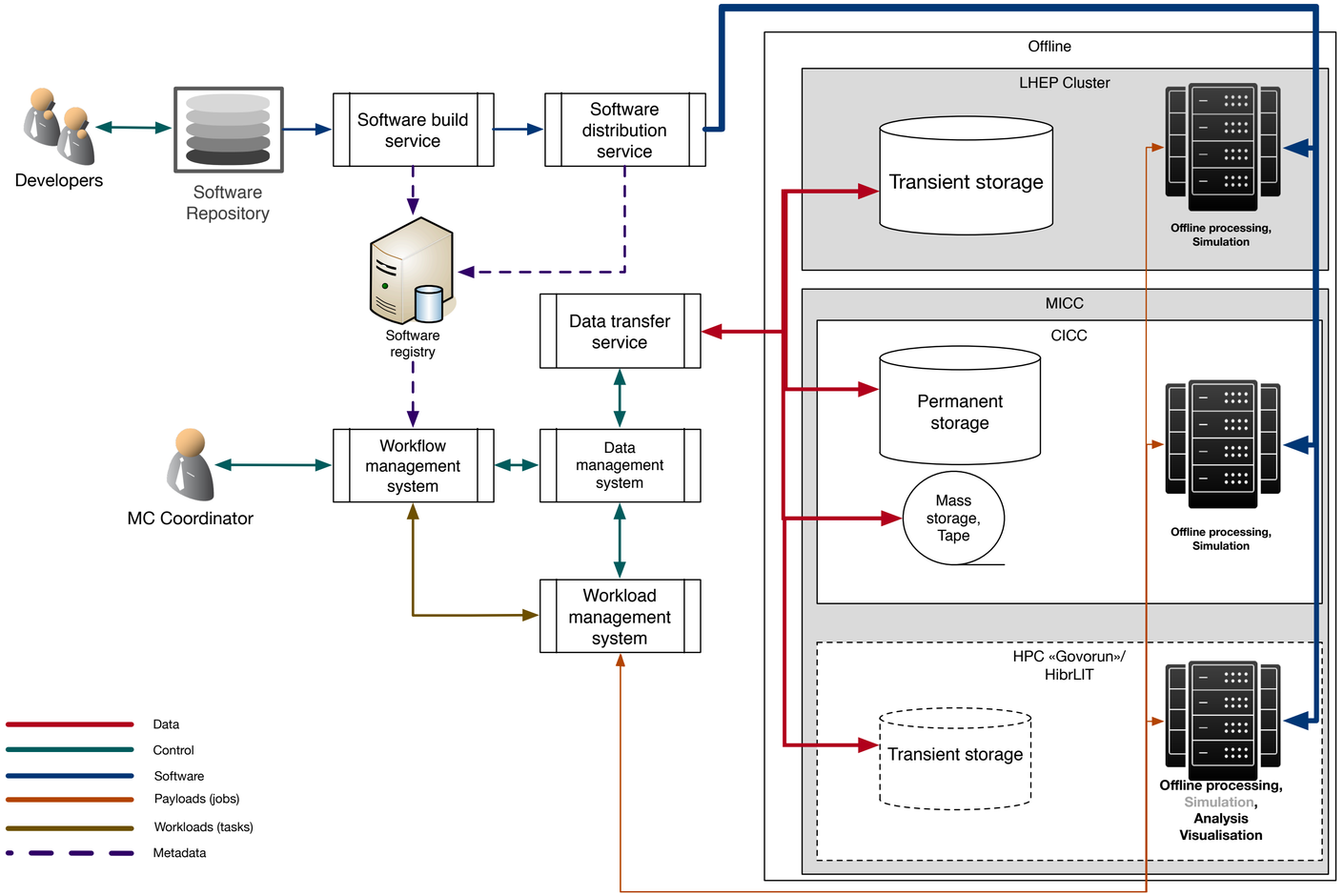
Automation of BM@N reconstruction workflow

- Automation of data processing means the sequence of transformations of source data to the data in the format which is used for final analysis
- Key components required for automation:
  - **Workflow management system** - control the process of processing of data on each step of processing. Produce tasks, which required for processing of certain amount of data, manages of tasks execution.
  - **Workload management system** - processes tasks execution by the splitting of the task to the small jobs, where each job process a small amount of data. Manage the distribution of jobs across the set of computing resources. Takes care about generation of a proper number of jobs till task will not be completed (or failed)
  - **Data management system** - responsible for distribution of all data across computing facilities, managing of data (storing, replicating, deleting etc.)
  - **Data transfer service:** takes care about major data transfers. Allow asynchronous bulk data transfers.



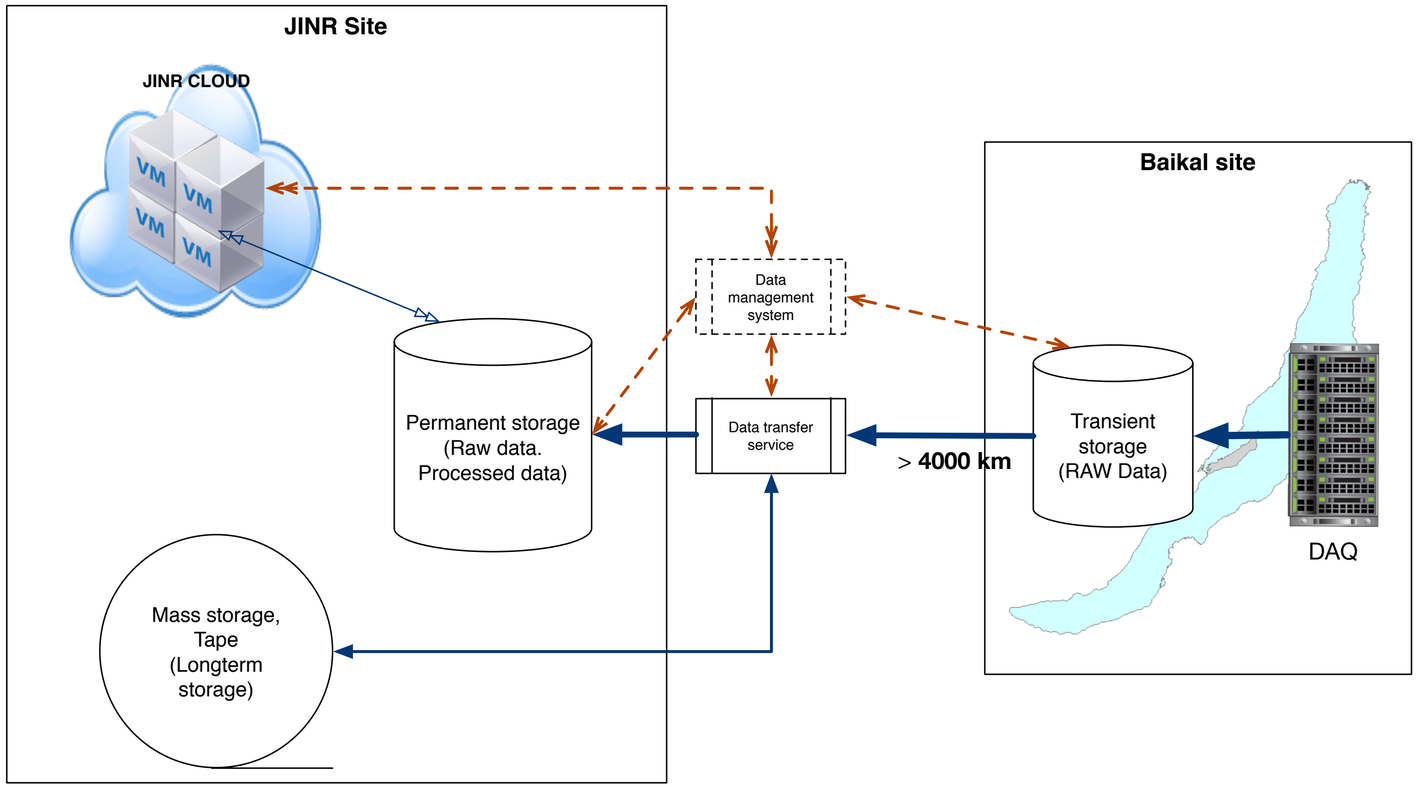
Use case: SPD simulation

- Simulation - another huge consumer of computing resources.
- Can be (should be) started before facility will be ready to collect data
- Accompanied by intensive software development
- Key components:
  - WFMS
  - WMS
  - DMS & DTS
  - Software build service - required for automation of building of new releases of SW
  - Software distribution service - service which allow automatic deployment of new versions of SW in heterogeneous environment



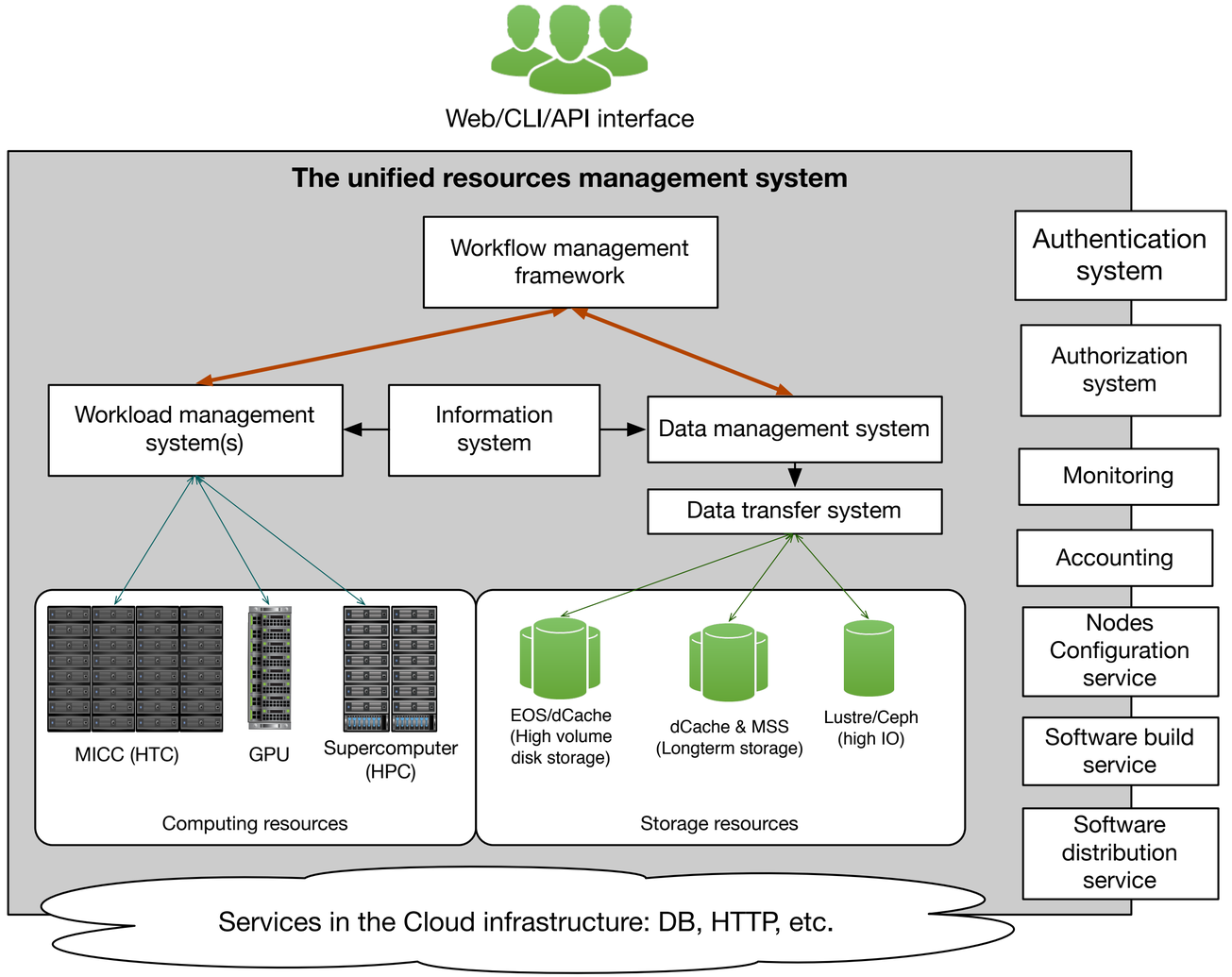
Use case: Baikal-GVD

- The detector placed quite far from the data processing facility: >4000 km.
  - weak network,
  - limited support
- Data volume is not too big but spat across a huge amount of files
- Data management system will provide the ability to catalogue the data and manage transfers
- Reliable data transfer service is needed
- Next step is the automation of processing

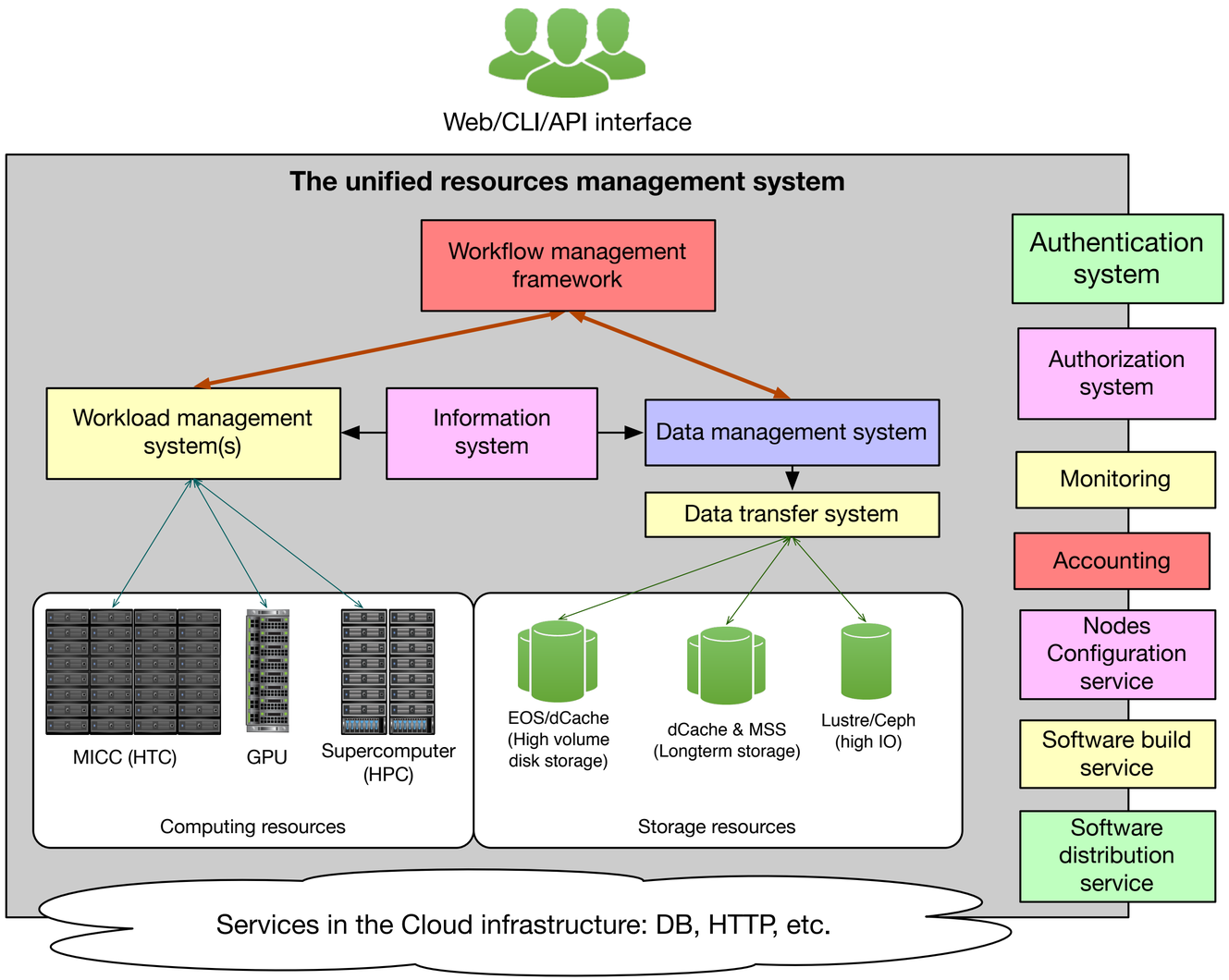


Unified Resource Management System

- The Unified Resource Management System is a IT ecosystem composed from the set of subsystem and services which should:
  - Unify of access to the data and compute resources in a heterogeneous distributed environment
  - Automate most of the operations related to massive data processing
  - Avoid duplication of basic functionality, through sharing of systems across different users (if it possible)
  - As a result - reduce operational cost, increase the efficiency of usage of resources,
  - Transparent accounting of usage of resources

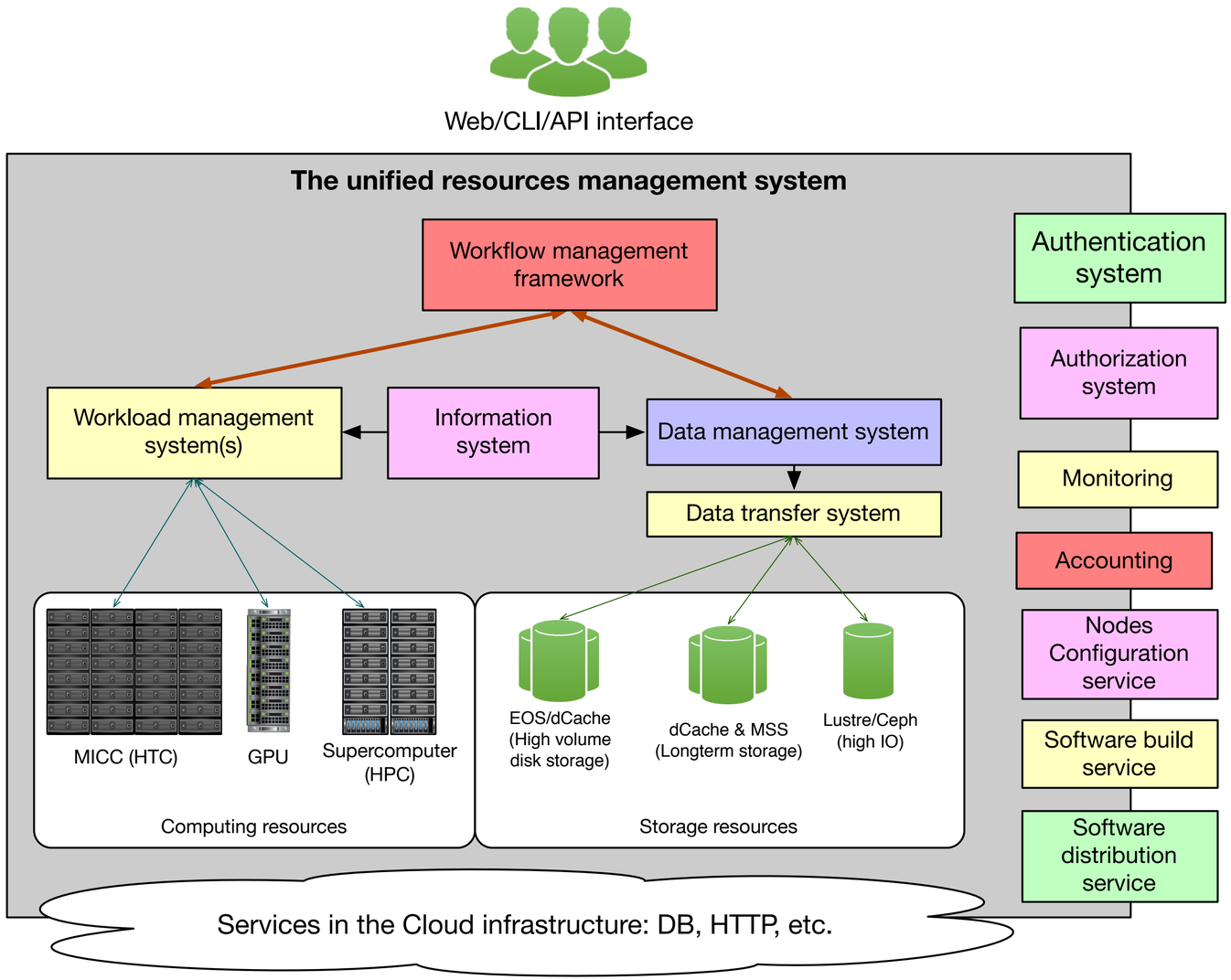


URMS: status (first steps)



- Some core subsystem already exist in JINR
  - Authentication system (Kerberos based, with SSO supporting for Web applications)
  - CVMFS as Software distribution service
- In progress:
  - deployment of FTS as the core of Data transfer system
  - We already have some infrastructure monitoring
  - A lot of research in WFMS and WMS fields, we may declare a list of requirements:
    - We should avoid limitations by scale as much as possible.
    - Advanced monitoring system
    - WMS with MultiVO support
    - Priority and share management
    - Task-based job management
  - Looks like that Rucio will be natural choice as cross experiment Data Management System
  - Software build service -prototype already exist in Cloud infrastructure - more test and proper documentation required

URMS: next steps



- We should agree on the Authorization System which will be used to manage user access to resources. The closest candidate is VOMS - but, we need to be coherent with Authentication System
- Accounting - we have nothing for the moment. It causes problems when we try to analyse our facilities.
- Nodes configuration - does not look like a service for the moment. Homemade scripts and human-oriented instructions
- Information system store and provide a description of computing and storage resources, including availability (shutdowns) of resources.