# Accessing OLCF Resources Using Pegasus WMS

George Papadimitriou[1], Pavlo Svirin[2], Karan Vahi[1], Mats Rynge[1], Rafael Ferreira da Silva[1], Jason Kincl[4], Vickie Lynch[4], Ewa Deelman[1], Anirban Mandal[3], Jeffrey Vetter[4], Valentine Anantharaj[4], Jack Wells[4], Alexei Klimentov[5], Kaushik De[6]

[1]University of Southern California – Information Sciences Institute [2]CERN [3]Renaissance Computing Institute [4]Oak Ridge National Laboratory [5]Brookhaven National Laboratory
[6]University of Texas at Arlington

## Pegasus Workflows on OLCF

*Problem definition - Motivation*

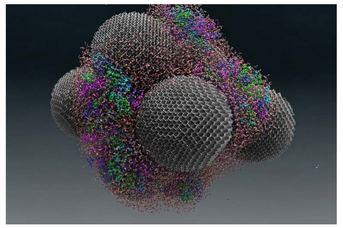
Accessing OLCF resources with Pegasus WMS has been difficult for DOE scientists in the past, because of either the need to install and configure Pegasus' software stack (Pegasus and High Throughput Condor) for different types of head nodes or handle issues arising from 2-factor authentication when they wanted to orchestrate remote submissions. Previous solutions included approaches like the rvGAHP, but when a new machine arrives (eg. Summit), all the steps of setting up the workflow submit environment have to be done once again.
Creating a workflow submission environment shouldn't discourage users from using a workflow management system. It should be an easy and well defined process that motivates them to take advantage of all the benefits a workflow management system has to offer, such as portability, automated data management, better application tracking, or making complex workflows easier to capture.
With this work we present two new approaches to support local submissions and remote submissions of Pegasus workflows, on OLCF's computing resources.

## Pegasus Impact on DOE Science

*Diamonds that deliver!*

Panorama and Pegasus enabled cutting-edge domain science research and development that has the potential to solve some of the challenges associated with **drug discovery and delivery**:

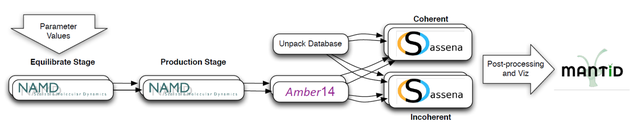
- The motions of a tRNA (or transfer RNA) model system can be enhanced when coupled with nanodiamonds, or diamond nanoparticles approximately 5 to 10 nanometers in size



*Water is seen as small red and white molecules on large nanodiamonds spheres. The colored tRNA can be seen on the nanodiamond surface. Image: Michael Mattheson, ORNL (https://www.ornl.gov/news/diamonds-deliver).*

- We have developed an SNS Pegasus workflow to confirm that nanodiamonds enhance the dynamics of tRNA when in the presence of water. The workflow calculates the epsilon which best matches experimental data. These calculations used almost **400,000 CPU hours on a Cray XE6at NERSC**.

- The workflow runs NAMD parallel simulations, which varies the epsilon between -0.01 and -0.19 for each temperature specified (it requires 800 cores: equilibrium runs take ~1.5hs and production runs 12-16hs). AMBER's cpptraj removes global translation and rotation, and SASSENA calculates neutron scattering intensities from the trajectories (400 cores, 3-6hs). This workflow was used to computer 4 temperatures between 260K and 300K, which generated **~3TB of data**.




## LEARN MORE

**Pegasus:** https://pegasus.isi.edu/

**Pegasus GitHub:** https://github.com/pegasus-isi/pegasus

**Panorama360:** https://panorama360.github.io

**BigPanda:** http://news.pandawms.org

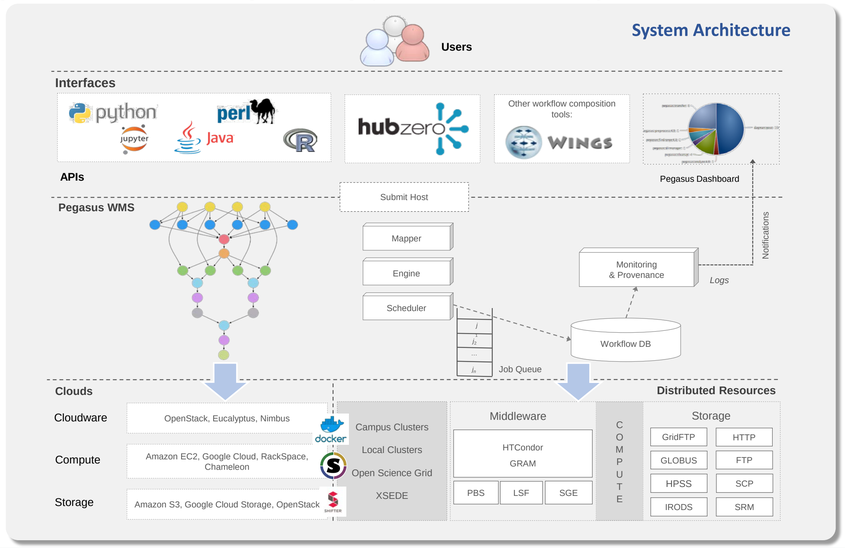
*Pegasus is funded by the National Science Foundation under the OAC SI2-SSI program, Grant #1664162*

*Panorama 360 is funded by the US Department of Energy under Grant #DE-SC0012636M*

## Pegasus Workflow Management System

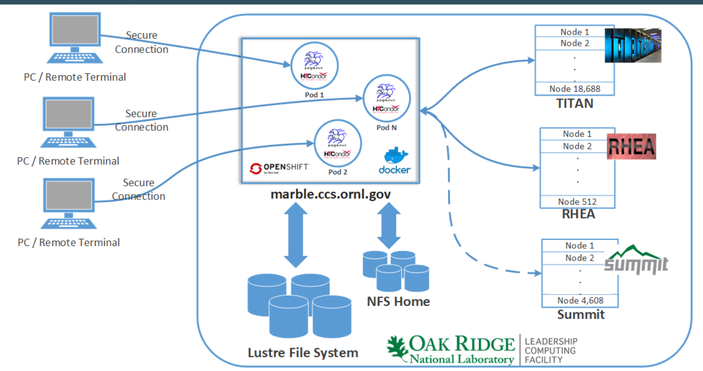
*Overview of the Pegasus WMS*

- Pegasus (https://pegasus.isi.edu) is a system for mapping and executing abstract **application workflows** over a range of execution environments
- The same abstract workflow can, at different times, be mapped to **different execution environments** such as OLCF(Titan, Rhea, Summit), XSEDE, OSG, commercial and academic clouds, campus grids, and clusters
- Pegasus can easily scale both the size of the workflow, and the resources that the workflow is distributed over, ranging from just a few computational tasks **up to 1 million**
- Stores static and runtime **metadata** associated with workflow, files and tasks.
- Pegasus-MPI-Cluster enables fine-grained task graphs to be executed **efficiently on HPC** resources




## Local Submissions of Pegasus Workflows at OLCF

*Workflow submit host as a service*




- In order to support local submissions and make the creation of the environment simpler we are leveraging the container orchestration support on OLCF, based on Openshift, using Docker containers.
- The CCS Marble cluster provides access to the Lustre FS, the NFS Home, Titan, Rhea and Summit via cross submission.
- We have prepared recipes that are ready to submit on Titan and Rhea, and in the future Summit as well (https://github.com/pegasus-isi/pegasus-olcf-kubernetes).
- Users can authenticate themselves on the marble cluster, build and spawn new pods, preconfigured as workflow submit nodes. After connecting to a Pegasus submit pod, the experience of submitting jobs is similar to that of a dedicated login node of each system.

## Panda Workload Management System

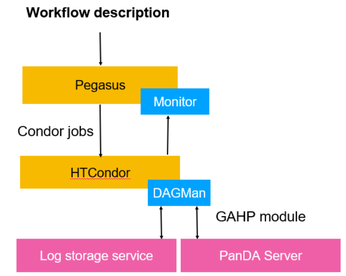
*Overview of the Panda WMS*

- The PanDA workload management system was developed for the ATLAS experiment at the Large Hardon Collider as a **new approach to distributed computing**.
- Some of its core ideas are:
  - make hundreds of **distributed sites** appear as **local**,
  - by providing a central queue to the users
  - **reduce** site related **errors** and reduce **latency**
  - **hide middleware** while supporting diversity and evolution
  - **hide** infrastructure **variations**

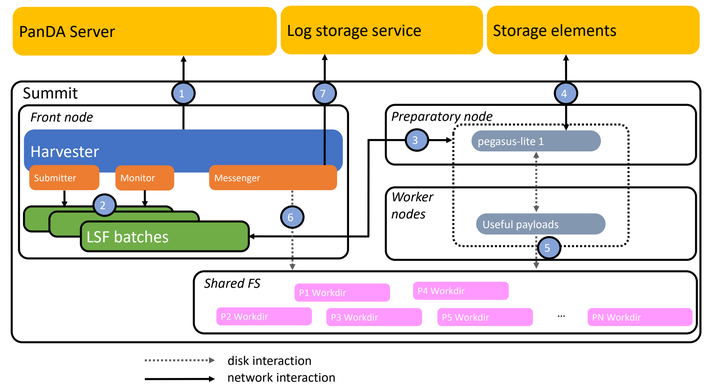
## Remote Submissions of Pegasus Workflows at OLCF

*Submit workflows using Panda*

- PanDA, by leveraging local automation features and exposing a modern REST API, can enable Pegasus' remote submissions.
- In order to support execution of Pegasus workflows via PanDA we had to extend Condor's GAHP module.
- The extended module allows HTCondor to interface with the Panda Server, submit and track jobs to the exposed resources.




- Harvester is a new-generation edge service for PanDA which resides on a front node of an HPC resource and interacts between the PanDA Server, where the payloads are stored, and the compute nodes.




- When job description(s) are fetched (1) the Submitter module of Harvester creates batches (2) which are then submitted to the LSF batch system of Summit (3). The monitor module keeps watching (2) the life cycle of a batch until it finishes.
- Every Pegasus/PanDA payload for Summit has a pegasus-lite wrapper which has the following steps:
  - Data stage-in: executed on preparatory nodes (4)
  - Execution of a useful payload: wrapped with *jsrun* and executed on worker nodes
  - Data stage-out: executed on preparatory nodes
- Output data and logs for a payload are written to a working directory (5) on the shared filesystem. When batch completes Harverster's Messenger module identifies (6) whether the payload was successful or no and uploads the logs to Log Storage Service where they can be fetched by HTCondor's modules and then analyzed by Pegasus.