Andrés Felipe Alba Hernández
User Support for Distributed Computing

The following document compare POMS and Pegasus because both project are financed by the Department of Energy.

Pegasus is a workflow management system where a workflow is defined as a group of task that can be represented by a Direct Acyclic Graph (DAG). The nodes in the DAG represent task and the edges represent their relations. On another hand POMS was thinked as a Production Operation Management System, then it expect to handle productions campaigns which can be part of a workflow or can contain workflows itself.

Pegasus expected a DAX file as an entry, the file should contain an abstract description of the workflow (DAG) in XML language while POMS expect a configuration of the campaign and a launch template. In the case of a workflow inside the campaign (for POMS), the launch script is the one that define the DAG. Then, Pegasus and POMS can be seen as systems in different levels of abstraction. For instance, Pegasus uses the DAX file to generate an condor executable workflow and the user uses the pegasus-plan command for defining which workflow run, where to run (sites), the input directory and the output directory. In contrast, those kind of configuration are handled by jobsub tool at Fermilab and POMS is focus on handle production campaigns providing configuration options such as: submission node (important for handle different experiments), what submission script use, schedule to launch the job, and definition dataset among others. The different in the level of abstraction that both system are working can be see for example in the input dataset. While Pegasus use physical directory or replica catalogs mapping logical file ids/names to physical file names, POMS use dataset definition and let SAMWeb Project handle the files locations and metadata.

There are also differences in the way that user interact with POMS and Pegasus, while in the last one all the configuration is provided by scripts in (python, java or xml), POMS offered an interaction through the web interface. However, Pegasus provide a Web monitoring interface that provide different views. For instance, the workflows list view show the state, submission time, submission host and the workflow name, the Workflow details provide additional information as wall time, dagman file, and job status of the entire workflow. Moreover, at the job view you can have the information as stderr file, stdout file, site, host and exit code, and also information such as submission time, execute time, job termination time. Additionally, the web interface incorporate some general statistics about the workflows as: Type, succeed, failed, total, retries, workflow wall time, among others. In particular, this is the most similar aspect between POMS and Pegasus. Although, POMS has been designed to aggregate the information in certain ways that are useful for monitoring the production like split the information of a whole campaign day by day.

Finally, there are certain features that POMS handle and they aren't in Pegasus such as: job efficiency histogram, schedule the launch of the job campaigns, kill the jobs for certain campaign, sort jobs by exit code, show the file status (for example: consumed, delivered, kids declare), and handle the login with the FNAL credentials.

In conclusion, Pegasus is a good tool for users that were using HTcondor directly because it makes easy the submission and the monitoring of the workflow, and it is easy to scale. It also provide a graphical interface for monitoring the status of the jobs. However, Fermilab experiments has been using specific tools as jobsub, SAMWeb Projects, Grafana, among other that can not be directly integrated with Pegasus while POMS was designed to work integrated with those tools and provide a better experience for the users.