Integrating the PanDA Workload Management System with the Vera C. Rubin Observatory

Edward Karavakis[1,*], *Wen* Guan[1], *Zhaoyu* Yang[1], *Tadashi* Maeno[1], *Torre* Wenaus[1], *Jennifer* Adelman-McCarthy[2], *Fernando* Barreiro Megino[3], *Kaushik* De[3], *Richard* Dubois[4], *Michelle* Gower[5], *Tim* Jenness[6], *Alexei* Klimentov[1], *Tatiana* Korchuganova[7], *Mikolaj* Kowalik[5], *Fa-Hui* Lin[3], *Paul* Nilsson[1], *Sergey* Padolski[1], *Wei* Yang[4], and *Shuwei* Ye[1]

[1]Brookhaven National Laboratory, Upton, NY, USA
[2]Fermi National Accelerator Laboratory, Batavia, IL, USA
[3]University of Texas at Arlington, Arlington, TX, USA
[4]SLAC National Accelerator Laboratory, Menlo Park, CA, USA
[5]National Center for Supercomputing Applications, Urbana, IL, USA
[6]Vera C. Rubin Observatory, Tucson, AZ, USA
[7]University of Pittsburgh, Pittsburgh, PA, USA

Abstract. The Vera C. Rubin Observatory will produce an unprecedented astronomical data set for studies of the deep and dynamic universe. Its Legacy Survey of Space and Time (LSST) will image the entire southern sky every three to four days and produce tens of petabytes of raw image data and associated calibration data over the course of the experiment's run. More than 20 terabytes of data must be stored every night, and annual campaigns to reprocess the entire dataset since the beginning of the survey will be conducted over ten years. The Production and Distributed Analysis (PanDA) system was evaluated by the Rubin Observatory Data Management team and selected to serve the Observatory's needs due to its demonstrated scalability and flexibility over the years, for its Directed Acyclic Graph (DAG) support, its support for multi-site processing, and its highly scalable complex workflows via the intelligent Data Delivery Service (iDDS). PanDA is also being evaluated for prompt processing where data must be processed within 60 seconds after image capture. This paper will briefly describe the Rubin Data Management system and its Data Facilities (DFs). Finally, it will describe in depth the work performed in order to integrate the PanDA system with the Rubin Observatory to be able to run the Rubin Science Pipelines using PanDA.

1 Introduction

The Vera C. Rubin Observatory's Legacy Survey of Space and Time (LSST) [1] will conduct a ten-year survey of the Southern Hemisphere sky, producing an unparalleled dataset for studying the universe. LSST's imaging capabilities and nightly data collection will generate a significant volume of raw image data and associated calibration data, creating data processing and management challenges.

---

*e-mail: Edward.Karavakis@cern.ch

This paper provides an overview of the Rubin Observatory, including the LSST and its data generation and processing challenges, and it introduces the PanDA Workload Management System [2], highlighting its scalability and flexibility. Finally, it focuses on the integration of PanDA with the Rubin Observatory [3], including the mapping of a Rubin Directed Acyclic Graph (DAG) to PanDA workload, multi-site processing, PanDA deployment at SLAC Kubernetes, monitoring, near real-time log access, and prompt processing considerations.

2 Vera C. Rubin Observatory

2.1 The Legacy Survey of Space and Time (LSST)

Rubin Observatory's LSST is a project that aims to survey the entire Southern Hemisphere sky every three to four days, capturing a substantial volume of data. Equipped with a 3.2 gigapixel camera [4], the Observatory's telescope will image the entire visible sky, allowing for the detection of objects that have changed in brightness, such as supernovae, or in position, such as asteroids. The Observatory's light-collecting power and sensitive camera are expected to discover approximately 20 billion galaxies and an equal number of stars.

The data collection of LSST is substantial, with each night producing over 20 terabytes of raw image data and associated calibration data. Over the course of its ten-year mission, LSST is estimated to generate approximately 60 petabytes of raw image data and several hundreds of petabytes of science-ready images and a multi-petabyte relational database to host the astronomical catalog. The scale and complexity of this dataset present significant challenges in terms of data processing, storage, and management.

2.2 Data Butler

The Data Butler [5] serves as the interface between the data and the pipeline tasks, facilitating seamless access to the substantial volume of data generated by the Observatory. Data Butler is responsible for data management, providing functionalities for data ingestion, organization, metadata management, and retrieval.

At its core, Data Butler acts as a data repository, storing and organizing the datasets acquired by the Observatory. It ensures the integrity and traceability of the data, allowing for efficient and reliable data processing workflows. Data Butler's metadata management capabilities enable efficient querying and filtering of data based on various criteria, such as time, location, or specific scientific requirements.

In addition to data management, there is an execution framework that uses the facilities of the Data Butler to simplify the interface between algorithmic code and job execution. It provides the necessary inputs and parameters to the tasks, allowing them to operate on the relevant data subsets. Data Butler's ability to handle complex dependencies and manage the flow of data between tasks ensures the efficient execution of the data processing workflows.

Furthermore, Data Butler enables data provenance and reproducibility by keeping track of the processing history and maintaining a record of all modifications and operations performed on the data. This provenance tracking is vital for scientific reproducibility and quality control, allowing researchers to trace back to the original data sources and understand the steps involved in generating the final scientific products.

2.3 Batch Production Service (BPS)

BPS [6] is the component of the Rubin Observatory pipeline execution system responsible for providing an interface to several Workload Management Systems (WMSs). It acts as a bridge

between the data management capabilities of Data Butler and the execution capabilities of various WMSs, such as PanDA, HTCondor, Parsl, and Pegasus, utilizing plugins to facilitate the seamless integration and coordination between these components. It orchestrates and manages the batch processing of data, ensuring seamless coordination between Data Butler and the WMS. It facilitates the retrieval of required data subsets from Data Butler, delivers them to the WMS for processing, and handles essential tasks such as data movement, job generation, and task scheduling.

2.4 Data Facilities

To address the data processing and management challenges of LSST, the Observatory has established multiple data facilities [7]. These facilities include the United States Data Facility (USDF), the French Data Facility (FrDF), the UK Data Facility (UKDF), and the Interim Data Facility (IDF) hosted on the Google Cloud Platform (GCP).

USDF, located at the SLAC Shared Scientific Data Facility (S3DF) [8] at the SLAC National Accelerator Laboratory, plays a crucial role in LSST's data processing and storage. It handles all prompt processing activities, 35% of the data release processing, and provides data access services to the US and international research community. FrDF, hosted at the CC-IN2P3 in Lyon, France, contributes to 40% of the data release processing and serves as a backup for raw data and select published products. UKDF, hosted by IRIS and GridPP in the UK, handles 25% of the data release processing.

IDF serves as a cloud-based interim facility used for pre-operations activities. It provides computational resources and storage for early data processing, algorithm development, and testing. IDF provided a platform to integrate the Rubin and PanDA software systems before the compute hardware was installed at SLAC. It was used to demonstrate the viability of the system by processing all the data for the Data Preview 0 exercise [9].

3 Production and Distributed Analysis (PanDA) Workload Management System

PanDA is a highly scalable and flexible workload management system [10] that was initially developed to address the computational challenges of the ATLAS experiment [11] at CERN. PanDA has demonstrated its capability to manage 24x365 processing on approximately 800,000 concurrent cores globally for ATLAS, encompassing all types of workflows, resource types, and a large user base. Its horizontal scaling capabilities have been further enhanced with support for Kubernetes (K8s) [12] deployment.

PanDA offers an easy-to-use submit client over the command line interface or as a Python API, allowing users to submit their scientific workflows through a Python script or a Jupyter notebook. With PanDA's ability to smoothly scale horizontally, it is well-suited to handle the anticipated demands of the Rubin Observatory, with about 100,000 concurrent jobs in 3 geographically distributed data facilities.

Scalability is a crucial aspect of PanDA's design, allowing it to effectively manage large-scale scientific workloads. The system seamlessly integrates with various computing resources, including clusters, grids, and clouds, and can dynamically allocate resources based on workload demands. This flexibility ensures efficient resource utilization, minimizing idle time and maximizing productivity.

Furthermore, PanDA's scalability extends beyond its ability to handle increasing computational demands. It also supports the growth of user communities, accommodating diverse research groups and institutions. With over 1,500 users and approximately 300 million jobs

per year within the ATLAS collaboration, PanDA has proven its ability to support large-scale scientific collaborations effectively.

4 Integration of PanDA with the Vera C. Rubin Observatory

Figure 1 illustrates the integration of PanDA within the data processing workflow of the Rubin Observatory.[1] The core components of this integration include the LSST Science Pipelines [13], which consist of the algorithmic payloads and the Data Butler and pipeline framework. Data Butler serves as the interface between the data and the pipeline tasks, providing data access and facilitating seamless interaction between the two.

BPS acts as an intermediary between Rubin's pipeline execution system and PanDA, serving as the interface for integrating Rubin with PanDA and the iDDS client. PanDA, as the workload management system, manages and schedules the Rubin workload across distributed resources. The PanDA pilot, integrated with the Rubin Data Butler access, facilitates the execution of the Rubin workload while also providing access to Google storage and enabling real-time logging. The storage of pilot logs in GCP, as required by Rubin, enables efficient storage and real-time monitoring of processing tasks. It's worth noting that PanDA can also route logs to other logging platforms if necessary.

Community engagement and collaboration was fostered through regular weekly meetings between all relevant teams, ensuring effective coordination for essential development work.
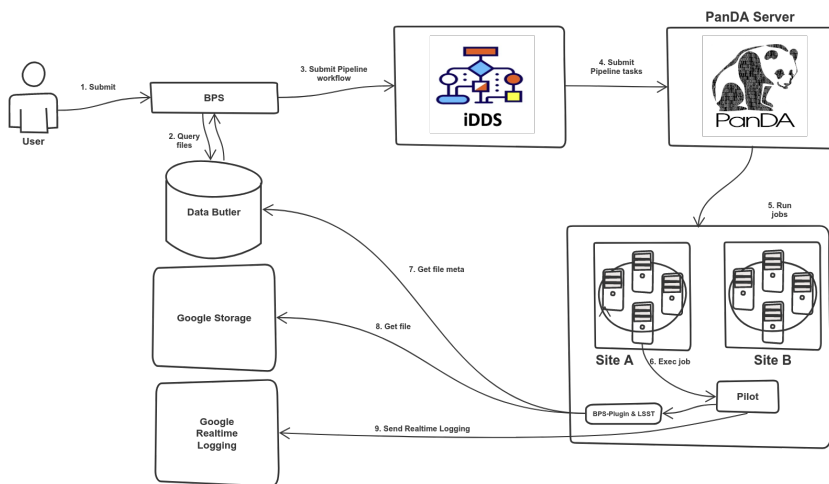
Figure 1. PanDA integration with Rubin Observatory.

4.1 Mapping Rubin DAG to PanDA Workload

The Rubin Science Pipelines, which form the core data processing framework for the Rubin Observatory, are composed of various pipeline tasks organized in a DAG structure. These DAGs describe the dependencies between tasks and the flow of data processing. In the context of the Rubin Observatory, LSST's processing workflows are modeled as quantum graphs,

---

[1]A user guide is available at https://panda.lsst.io.

where each node represents the execution of a science algorithm with a well known set of inputs and outputs. The integration of PanDA with the Observatory involves mapping these DAGs to PanDA workflows, enabling efficient and optimized execution of the Observatory's processing tasks, taking advantage of PanDA's scalability and workload management capabilities.

The intelligent Data Delivery Service (iDDS) [14] plays a crucial role in supporting complex, fine-grained workflows defined through DAGs or workflow description languages. iDDS has been successfully utilized in the ATLAS experiment for the data carousel activity [15] and various machine learning and analysis workflows [16].

4.2 Multi-Site Processing

Rubin Observatory's data processing requirements span multiple data facilities. PanDA's support for multi-site processing is particularly valuable in this context, allowing efficient coordination and workload distribution across different sites.

To enable multi-site processing for Rubin [17], both the Rubin Data Management middleware and PanDA/iDDS required development in order to ensure that the quantum graph and execution Data Butler, created at one data facility, can be seamlessly used at another facility. Additionally, mechanisms for merging outputs and metadata back to the main Data Butler registry with the help of Rucio [18] were developed to support efficient processing across multiple sites.

4.3 PanDA Deployment on Kubernetes

To support the integration of PanDA with the Rubin Observatory, a deployment of PanDA at the SLAC National Accelerator Laboratory was carried out, utilizing Kubernetes (K8s) as the underlying container orchestration platform. The deployment included all the necessary components of PanDA, such as the PanDA Server, JEDI, Indigo IAM authentication, Harvester, iDDS, PanDA Monitor, and ActiveMQ.

In addition, a highly available PostgreSQL database cluster, based on the CloudNative-PostgreSQL (CNPG) [19] framework, was deployed to ensure the reliability and scalability of the database backend for PanDA. The use of Kubernetes and CNPG provides essential cloud-native capabilities, such as self-healing, high availability, rolling updates, and resource management, ensuring the stability and performance of the PanDA deployment at SLAC.

4.4 Monitoring and Near Real-Time Log Access

Effective monitoring is crucial for ensuring the smooth operation of the Rubin data processing workflows through PanDA. To enhance the monitoring capabilities, PanDA Monitor [20] was extended to support Rubin job monitoring.

As seen in Figure 2, PanDA Monitor provides hierarchical navigation at different levels, allowing users to track the progress of workflows, tasks, jobs, and logs. Additionally, memory usage monitoring using the prmon [21] tool, originally developed by ATLAS, was integrated into PanDA Monitor. This enhancement provides valuable insights into memory consumption during job execution.

Furthermore, the integration of near real-time log access into the PanDA environment offers a significant advantage for monitoring and debugging purposes. The Pilot captures payload logs and sends them as JSON to the Google Cloud Logging service [22], allowing direct access to logs from PanDA Monitor. This near real-time log access complements conventional log access, providing additional information for troubleshooting and performance analysis.

Show 10 entries   Search:

request id	username	workflow status	graph	workflow name	created on (UTC)	total tasks	tasks	remaining files	released files	total files
3202	Hsin Fang Chiang	Finished	plot	u_hchiang2_w_2023_01_DM-37751_DRP-Prod_20230129T062146Z	2023-01-29 07:09:57	5	Finished(5)	0	1298	1298
3201	Robot Pilot	Finished	plot	u_lsstsvc1_ci_hsc_gen3_USDF_remote_20230128T215922Z	2023-01-28 22:00:31	125	Finished(125)	0	1147	1147
3200	Zhaoyu Yang	Finished	plot	u_zhaoyu_ci_hsc_gen3_w_2022_50_20230128T214305Z	2023-01-28 21:44:19	125	Finished(125)	0	1147	1147
3199	iddssv1	Finished	plot	u_lsst001_UKDF_w52_remote_20230128T213356Z	2023-01-28 21:36:16	3	Finished(3)	0	35	35
3198	Hsin Fang Chiang	Finished	plot	u_hchiang2_w_2023_01_DM-37751_20230128T051306Z	2023-01-28 05:27:56	5	Finished(5)	0	21901	21901
3197	Hsin Fang Chiang	Failed	plot	u_hchiang2_w_2023_01_DM-37751_20230127T232243Z	2023-01-27 23:37:36	5	Failed(5)	21899	2	21901
3196	Hsin Fang Chiang	Cancelled	plot	u_hchiang2_w_2023_01_pp_templates_20230127T223214Z	2023-01-27 22:47:21	4	Failed(4)	21899	1	21900
3195	Christopher Pinkenburg	Transforming	plot	pseudo_input.2023_01_27_21_18_21_269010307	2023-01-27 21:18:22	2	Submitting(1) Submitted(1)	0	0	0
3194	Hsin Fang Chiang	Failed	plot	u_hchiang2_w_2023_01_pp_templates_20230127T195035Z	2023-01-27 20:05:44	5	Failed(5)	21899	2	21901
3193	Hsin Fang Chiang	SubFinished	plot	HSC_runs_RC2_w_2023_03_DM-37570_20230127T182608Z	2023-01-27 19:26:22	9	Finished(3) Failed(6)	436	435	871

Showing 1 to 10 of 388 entries   Previous 1 2 3 4 5 … 39 Next

Figure 2. Workflow monitoring in PanDA Monitor allowing hierarchical navigation.

4.5 PanDA for Prompt Processing

Prompt processing is a critical aspect of the Rubin Observatory's operations, involving the rapid processing of data and issuing alerts within a short timeframe of the order of a minute. PanDA is currently being evaluated for prompt processing within the Observatory, with the aim of enabling the initiation of processing within seconds on dedicated resources at SLAC.

To achieve prompt processing, several developments were implemented. These include the use of semi-persistent pilots running on worker nodes, task resurrection through notifications to skip unnecessary overhead, direct job communication via ActiveMQ, and a communication channel between the JEDI component and the PanDA server. These advancements significantly reduce latency and support near-real-time data processing, making prompt processing feasible for the Rubin Observatory.
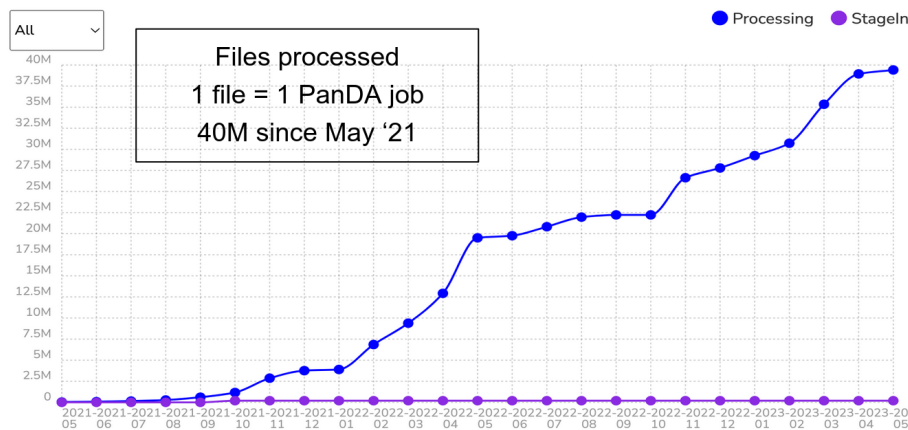
5 Results

Figure 3. Accumulation of the number of Rubin Observatory jobs managed by PanDA since May 2021.

During Phase 2 of the Rubin Observatory's Data Preview 0 (DP0.2) in 2021 [9], PanDA demonstrated the capability to run 16 million jobs at the Google-based IDF [23]. Most jobs

were processed on a cluster with approximately 4000 cores, up to 14GB/core RAM with a total CPU usage of 2.5M core-hours. Eight million jobs were also processed for the Hyper-SuprimeCam (HSC) reprocessing at USDF in SLAC.

The successful processing of DP0.2 drove the decision to endorse PanDA for the Data Release Production (DRP) campaigns [24] and since 2022, Rubin Observatory has been using PanDA for the DRP campaigns. Figure 3 shows the number of Rubin jobs processed with PanDA since May 2021. For the 2023 DRP campaigns, it is estimated to have around 36 million jobs for the HSC Public Data Release 2 (HSC-PDR2) and around 8M for the HSC reprocessing.

6 Conclusion

The integration of the PanDA WMS with the Rubin Observatory has proven to be successful in enabling efficient and scalable data processing for LSST. Through successful deployments and campaigns, such as DP0.2 and HSC reprocessing, PanDA has demonstrated its reliability and performance in meeting the processing requirements of the Rubin Observatory. The integration efforts have paved the way for seamless execution of Rubin Science Pipelines, efficient multi-site processing, improved monitoring and log access, and prompt processing capabilities in order to be able to handle the significant amount of data generated by the Observatory.

The integration of PanDA with the Rubin Observatory serves as a testament to the value of collaborative efforts between scientific facilities and distributed computing projects. It showcases the successful integration of advanced workload management systems with complex astronomical data processing workflows. Most importantly, the PanDA development work undertaken for Rubin Observatory is experiment agnostic and is currently used by other experiments as well, benefiting the wider scientific community.

7 Acknowledgments

References

[1] Z. Ivezic et al., *LSST: From Science Drivers To Reference Design And Anticipated Data Products*, , Astrophys. J., **873**, 111 (2019)

[2] T. Maeno et al., *PanDA: Production and Distributed Analysis System*, Computing and Software for Big Science (to appear)

[3] S. Padolski, S. Ye, E. Karavakis, *Running Science Pipelines using PanDA* (2022), Vera C. Rubin Observatory Data Management Technical Note DMTN-168, `https://dmtn-168.lsst.io`

[4] S.M. Kahn et al., *Design and development of the 3.2 gigapixel camera for the Large Synoptic Survey Telescope*, in *Ground-based and Airborne Instrumentation for Astronomy III*, edited by I.S. McLean, S.K. Ramsay, H. Takami (2010), Vol. 7735 of *Proc. SPIE*, p. 77350J

[5] T. Jenness et al., *The Vera C. Rubin Observatory Data Butler and pipeline execution system*, in *Software and Cyberinfrastructure for Astronomy VII* (2022), Vol. 12189 of *Proc. SPIE*, p. 1218911, `arXiv:2206.14941`

[6] M. Gower et al., *Adding Workflow Management Flexibility to LSST Pipelines Execution*, in *Astronomical Data Analysis Software and Systems XXXII* (2023), Vol. in press of *ASP Conf. Ser.*, `arXiv:2211.15795`

[7] F. Hernandez et al., *Overview of the distributed image processing infrastructure to produce the Legacy Survey of Space and Time (LSST)*, Proc. of CHEP 2023 (to appear)

[8] *SLAC Shared Scientific Data Facility*, `https://sdf.slac.stanford.edu/`

[9] W. O'Mullane et al., *Data Preivew 0.2 and Operations Rehearsal for DRP* (2023), Vera C. Rubin Observatory Technical Note RTN-041, `https://rtn-041.lsst.io`

[10] T. Maeno et al., *Utilizing Distributed Heterogeneous Computing with PanDA in ATLAS*, Proc. of CHEP 2023 (to appear)

[11] ATLAS Collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider*, J. Inst. **3**, S08003 (2008)

[12] *Kubernetes: Production-grade container orchestration*, `https://kubernetes.io/`

[13] J. Bosch et al., *An Overview of the LSST Image Processing Pipelines*, in *Astronomical Data Analysis Software and Systems XXVII*, edited by P.J. Teuben, M.W. Pound, B.A. Thomas, E.M. Warner (2019), Vol. 523 of *ASP Conf. Ser.*, p. 521, `arXiv:1812.03248`

[14] W. Guan et al., *Distributed Machine Learning with PanDA and iDDS in LHC ATLAS*, Proc. of CHEP 2023 (to appear)

[15] M. Barisits et al., *ATLAS Data Carousel*, in *EPJ Web of Conferences* (EDP Sciences, 2020), Vol. 245, p. 04035

[16] C. Weber et al., *An Active Learning application in a dark matter search with ATLAS PanDA and iDDS*, Proc. of CHEP 2023 (to appear)

[17] K.T. Lim, *Multi-Site Data Release Processing Using PanDA and Rucio* (2022), Vera C. Rubin Observatory Data Management Technical Note 212, `https://dmtn-213.lsst.io`

[18] M. Barisits et al., *Rucio: Scientific Data Management*, Comput Softw Big Sci **3**, 11 (2019)

[19] *CloudNativePG - PostgreSQL Operator for Kubernetes*, `https://cloudnative-pg.io/`

[20] T. Korchuganova et al., *BigPanDA monitoring system evolution in the ATLAS experiment*, Proc. of CHEP 2023 (to appear)

[21] *Process monitor (prmon) github repository*, `https://github.com/HSF/prmon/`

[22] *Google cloud logging*, `https://cloud.google.com/logging`

[23] B. Yanny et al., *Compute Resource Usage of DP0.2 production run* (2023), Vera C. Rubin Observatory Technical Note 39, `https://rtn-039.lsst.io`

[24] W. O'Mullane et al., *Near term worfklow for pre-operations with PanDA* (2022), Vera C. Rubin Observatory Technical Note RTN-013, `https://rtn-013.lsst.io`