Sasquatch: Rubin Observatory metrics and telemetry service

Angelo Fausti Neto[1], Frossie Economou[1], Michael A. Reuter[1], Jonathan Sick[2 1], Russ Allbery[1], and Adam J. Thornton[1]

[1]Vera C. Rubin Observatory Project Office, 950 N. Cherry Ave., Tucson, AZ 85719, USA
[2]J.Sick Codes Inc., Penetanguishene, Ontario, Canada

ABSTRACT

At Vera C. Rubin Observatory, the need to manage metrics and telemetry data efficiently led to the creation of Sasquatch. Sasquatch consolidates our high-frequency telemetry harness, which captures the observatory engineering data, with the science performance metrics measured by the LSST Science Pipelines. Sasquatch utilizes InfluxDB, a time series database, to efficiently store and query time-series data. We combine InfluxDB Enterprise with Apache Kafka and deploy our solution on the Kubernetes platform. Our current setup at the US Data Facility enables real-time access to data mirrored from the Summit and leverages tools like Chronograf for time series data visualization, Kapacitor for alert management, and the Rubin Science Platform's notebook environment for data analysis using Python. Sasquatch is currently employed during Rubin Observatory's System Integration Testing and Commissioning phase and is an essential service as we transition into survey operations.

Keywords: Vera C. Rubin Observatory, LSST, Telemetry, Metrics, Time-series Databases

1. INTRODUCTION

The Vera C. Rubin Observatory's Legacy Survey of Space and Time (LSST) represents the most ambitious and comprehensive optical astronomy survey ever planned. The LSST will survey the visible sky every few nights, generating a data stream of 20 TB per night. The survey will produce a 500 PB dataset over its ten-year lifetime, enabling a wide range of scientific investigations, from studying dark matter and energy to exploring the Solar System.[1]

The LSST Data Management (DM) team is tasked with developing and maintaining the LSST Science Pipelines, which will process the data collected by the LSST. Additionally, the DM team oversees the performance of the LSST Science pipelines, ensuring that they meet the specifications outlined in the LSST Science Requirements Document (SRD).[2]

Rubin Observatory also generates a large amount of telemetry data, which is used to monitor the observatory systems' health and troubleshoot issues during operations.

This paper describes Sasquatch, a service responsible for recording metrics and telemetry data from various subsystems at Rubin Observatory. The paper is organized as follows: Section 2 provides an overview of the science performance metrics and telemetry data produced at Rubin Observatory and its representation as time series data. Section 3 describes the Sasquatch architecture. Section 4 presents the Sasquatch deployments at Rubin Observatory. Section 5 discusses the current challenges, and Section 6 concludes the paper.

2. TIME SERIES DATA

Rubin Observatory will conduct several data processing campaigns during survey operations, such as "first look" analyses immediately after the images are captured by the LSST Camera, Prompt Processing handling approximately a thousand visits each night, and annual Data Release Processing campaigns.[3] These campaigns will evaluate the quality of the data products through specific science performance metrics.[4,5] These metrics have a timestamp and are measured across multiple dimensions (including instrument, detector, visit, filter, patch, and tract) understood by the data Butler.[4–6]

Telemetry is also timestamped data and measured across multiple dimensions. For example, the forces applied on the actuators of the LSST Primary/Tertiary mirror (M1M3) Active Optics System are measured at 50 Hz

and identified by the actuator number. Similarly, for the LSST Camera, telemetry data is measured for each amplifier, detector, and raft, among other dimensions.

Time series databases (TSDBs) are optimized for efficiently storing and querying timestamped data, making them ideal for metrics and telemetry data.

The following section describes the Sasquatch architecture, which leverages InfluxDB[*], a popular TSDB.

3. SASQUATCH ARCHITECTURE

Sasquatch[†] design provides data durability and high availability by combining InfluxDB and Apache Kafka[‡] deployed on the Kubernetes platform.[7,8]

InfluxDB can handle high data throughput, making it suitable for writing and querying the high-frequency telemetry data produced by the Observatory Control System (OCS). In the current setup at USDF, data is replicated across two InfluxDB data nodes for redundancy, making it possible to write and query the database in case of a failure in one of the InfluxDB data nodes.

Apache Kafka is a distributed data streaming platform for high throughput and fault tolerance. Sasquatch uses the Strimzi operator[§] to deploy Apache Kafka on Kubernetes. In Sasquatch, we replicate data across three Kafka brokers before writing to InfluxDB. When sending data to a Kafka topic representing a metric or a telemetry topic, Sasquatch requires an acknowledgment from at least two replicas for data durability.

Figure 1 shows the Sasquatch deployment at USDF, with InfluxDB, Kafka, and Kafka Connect being the main components. Kafka Connect runs MirrorMaker2, the connector responsible for mirroring the data from the Summit,[9] and the InfluxDB Sink connectors responsible for consuming data from the Kafka brokers and writing to InfluxDB.
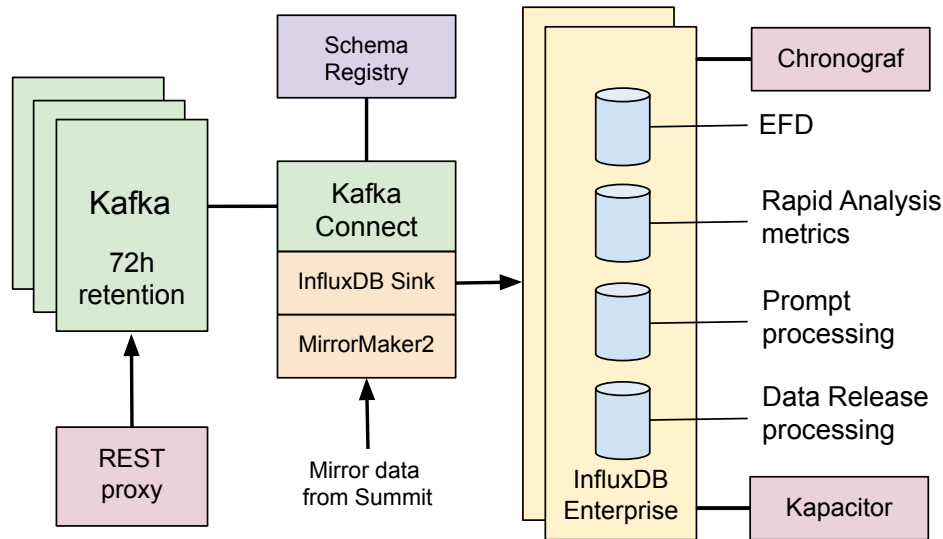

Figure 1. Sasquatch deployment at the US Data Facility.

Sasquatch manages multiple databases in InfluxDB to store data from various sources. For example, data produced by the OCS is routed to the Engineering and Facility Database (EFD) in InfluxDB. Similarly, metrics produced by the LSST Science Pipelines are routed to their corresponding databases, as illustrated in Figure 1. The routing mechanism is based on namespaces which are used to identify the data sources and the databases where the data should be stored.

---

[*]https://www.influxdata.com/time-series-database/

[†]https://sasquatch.lsst.io

[‡]https://kafka.apache.org

[§]https://strimzi.io

3.1 Sending data to Sasquatch

Sasquatch uses Avro for the data serialization format and the Confluent Schema Registry (SR)[¶] to store the Avro schemas and ensure they can evolve safely, without breaking the Sasquatch consumers.

Applications connecting to Sasquatch can use native Kafka and SR clients to send data to Kafka. For example, the OCS Kafka producers[‖] connect directly to Kafka and the SR to send data.

Alternatively, the REST Proxy[**] provides a REST interface to Kafka. It makes connecting to Sasquatch easy, and it is a good solution when data rates are low. For example, the LSST Analysis Tools package uses the REST Proxy to send metrics from the LSST Pipelines outputs to Sasquatch.

3.2 Data visualization, alerting, and analysis

Enabling users to create dashboards and alerts is essential for monitoring and troubleshooting the observatory systems, especially during Rubin Observatory's current System Integration Testing and Commissioning (SIT-COM) phase.

Sasquatch uses Chronograf for data visualization and Kapacitor for alert management. Dashboards and alert rules are defined using InfluxQL, a SQL-like query language with a rich set of functions specific to time series data analysis. Alert notifications are sent to particular channels on Slack to notify users about a metric excursion or the observatory system's health.

Sasquatch is also integrated with the Rubin Science Platform (RSP).[10,11] On the RSP at USDF, users can access data in real-time and historical data captured by Sasquatch. This integration enables users to analyze data in a JupyterLab environment with Python and leverage tools such as Times Square[12] for publishing parameterized Jupyter Notebooks.

4. SASQUATCH DEPLOYMENTS

Table 1 illustrates the various environments where we deploy Sasquatch, each with a different retention period, storage capacity, and target audience. Sasquatch deployments are managed by Phalanx[††], our Kubernetes-based internal developer platform for managing deployment configuration.

Table 1. Sasquatch environments at Rubin Observatory

Environment	Retention Period	Current Capacity	Audience
Summit	30 days	5 TB	Staff at the Summit
USDF	Project lifetime	60 TB	Project at large
IDF (Google Cloud Platorm)	1 year	30 TB	Project at large (online backup)
Teststands (Tucson & Base Facility)	10 days	1 TB	Developers

At the Summit, we store data for 30 days, and the primary audience for this deployment is the observatory staff. At the USDF, we are required to keep the data for the project's lifetime and make it available to the project at large. We also deploy Sasquatch on the Google Cloud Platform, where we keep the last year of data as an online backup in case users cannot access Sasquatch at the USDF. Finally, we have independent test stand deployments in Tucson, US, and at the Base Facility in La Serena, Chile. At the test stands, we record simulated data, and the primary audience for these deployments is our developers.

[¶]https://docs.confluent.io/platform/current/schema-registry/

[‖]https://ts-salkafka.lsst.io

[**]https://docs.confluent.io/platform/current/kafka-rest/index.html

[††]https://phalanx.lsst.io

5. CHALLENGES

During the Rubin Observatory construction phase, we have been collecting metrics to characterize the performance of the LSST Science Pipelines algorithms running on precursor datasets. [5,13] At the observatory, we started collecting telemetry data in 2019 from the LSST Auxilary Telescope (AuxTel) runs and more recently during the SITCOM phase of the Main Telescope systems, notably M1M3,[14] Telescope Mount Assembly (TMA)[15] and Secondary Mirror (M2) systems.[16]

Even before the start of survey operations, we recorded 20 TB of metrics and telemetry data in Sasquatch, with data rates peaking at 565 GB/week.[17] With the integration of the LSST Camera into the OCS, which is planned for this year, we expect the data rates to increase even further.

Table 2 shows the projected storage growth for the EFD database. From the data rates observed so far, we estimate that the size of the database will increase by 30 TB yearly during survey operations. By the end of the 10-year survey, we expect around 1 PB of data in Sasquatch.

Table 2. Projected storage size for the EFD at USDF by year considering a replication factor of two.

End of Year	EFD Size	RF	Total Storage	Main Drivers
2022	2 TB	2	-	AuxTel
2023	15 TB + SM	2	60 TB	M1M3, TMA, M2 + AuxTel
2024	35 TB + SM	2	100 TB	MainTel + AuxTel
2025	65 TB + SM	2	160 TB	Survey operations
2035	0.3-0.5 PB + SM	2	0.7-1 PB	Survey operations

Managing a 1 PB database on premises is challenging. We are responsible for maintaining the underlying hardware infrastructure, solving performance and scalability issues, ensuring data consistency between the Summit and USDF databases, performing database operations such as backups and restores, and providing data access tools to our users. In Sasquatch, we are working on several fronts to address these challenges.

We recently transitioned from InfluxDB OSS to an InfluxDB Enterprise cluster at USDF to improve database performance, scalability, and support. Currently, at the USDF, the InfluxDB Enterprise cluster has 2x8-core data nodes with 30 TB of local SSD storage each and a replication factor (RF) of two for redundancy. To increase storage capacity, we can add more nodes to the cluster. We are constantly reviewing the projected storage size for the EFD at USDF (table 2) and expanding the cluster capacity as required. We are working closely with InfluxData support to understand the best practices for scaling InfluxDB and leverage the new features of the InfluxDB 3.0 clustered product, such as storage tiering and the fact that the latest version is Kubernetes native.

Ensuring consistency between the databases at the Summit and USDF is a critical task. To achieve this, we have adopted MirrorMaker2. This component at USDF is responsible for mirroring data from the Summit. It can automatically recover data from network outages shorter than the Kafka retention period, which is currently set to 72 hours. If data is present in Kafka but missing in InfluxDB at USDF, we can deploy repairer connectors to replay the missing data to InfluxDB. This process is currently manual and triggered by data consistency checks between the Summit and USDF databases. In the event that the data is no longer present in Kafka, data recovery can still be performed by restoring database backups from the Summit to USDF. We are actively working on automating the data recovery process to significantly reduce the time required to recover data and to minimize the impact on users.

To improve data access to users, we are working on tools to aggregate telemetry data[18] and export the data to formats like Apache Parquet and VO TAP (Table Access Protocol)[19] to leverage existing data access mechanisms in the RSP.

Finally, we must continuously adapt as Sasquatch and OCS software components evolve. At the Summit, the OCS is adopting Kafka for the middleware implementation.[20,21] That means the OCS will record commands, events, and telemetry data directly in Kafka. In principle, Sasquatch and the OCS could share the same Kafka cluster, simplifying their integration and avoiding the duplication of resources and data. However, this will also bring new challenges we must address in the Kafka configuration. While the OCS needs to ensure low latency for

commands and events, Sasquatch must prioritize data durability for recording telemetry data. We are working with the OCS team to understand the requirements and constraints of the new Kafka deployment and adapt Sasquatch to the new environment.

6. CONCLUSION

This paper provides a comprehensive overview of Sasquatch. Over the years, Sasquatch's design has evolved significantly, providing high availability and data durability for metrics and telemetry data. Sasquatch combines InfluxDB Enterprise and Apache Kafka to efficiently store and query time series data. Sasquatch is deployed in multiple environments at Rubin Observatory, each with different retention periods, storage capacity, and target audience.

Sasquatch enables real-time access to data mirrored from the Summit and historical data at the USDF. Sasquatch also leverages tools such as Chronograf for data visualization and Kapacitor for alert management and is integrated with the RSP notebook aspect for data analysis with Python.

While we are still improving the Sasquatch deployment at the USDF and Summit environments, InfluxDB Enterprise and Apache Kafka offer a solid foundation for Sasquatch. Sasquatch is currently employed during Rubin Observatory's SITCOM phase and is an essential service as we transition into survey operations.

ACKNOWLEDGMENTS

REFERENCES

[1] Ivezić, Ž., Kahn, S. M., Tyson, J. A., Abel, B., Acosta, E., Allsman, R., Alonso, D., AlSayyad, Y., Anderson, S. F., Andrew, J., Angel, J. R. P., Angeli, G. Z., Ansari, R., Antilogus, P., Araujo, C., Armstrong, R., Arndt, K. T., Astier, P., Aubourg, É., Auza, N., Axelrod, T. S., Bard, D. J., Barr, J. D., Barrau, A., Bartlett, J. G., Bauer, A. E., Bauman, B. J., Baumont, S., Bechtol, E., Bechtol, K., Becker, A. C., Becla, J., Beldica, C., Bellavia, S., Bianco, F. B., Biswas, R., Blanc, G., Blazek, J., Blandford, R. D., Bloom, J. S., Bogart, J., Bond, T. W., Booth, M. T., Borgland, A. W., Borne, K., Bosch, J. F., Boutigny, D., Brackett, C. A., Bradshaw, A., Brandt, W. N., Brown, M. E., Bullock, J. S., Burchat, P., Burke, D. L., Cagnoli, G., Calabrese, D., Callahan, S., Callen, A. L., Carlin, J. L., Carlson, E. L., Chandrasekharan, S., Charles-Emerson, G., Chesley, S., Cheu, E. C., Chiang, H.-F., Chiang, J., Chirino, C., Chow, D., Ciardi, D. R., Claver, C. F., Cohen-Tanugi, J., Cockrum, J. J., Coles, R., Connolly, A. J., Cook, K. H., Cooray, A., Covey, K. R., Cribbs, C., Cui, W., Cutri, R., Daly, P. N., Daniel, S. F., Daruich, F., Daubard, G., Daues, G., Dawson, W., Delgado, F., Dellapenna, A., de Peyster, R., de Val-Borro, M., Digel, S. W., Doherty, P., Dubois, R., Dubois-Felsmann, G. P., Durech, J., Economou, F., Eifler, T., Eracleous, M., Emmons, B. L., Fausti Neto, A., Ferguson, H., Figueroa, E., Fisher-Levine, M., Focke, W., Foss, M. D., Frank, J., Freemon, M. D., Gangler, E., Gawiser, E., Geary, J. C., Gee, P., Geha, M., Gessner, C. J. B., Gibson, R. R., Gilmore, D. K., Glanzman, T., Glick, W., Goldina, T., Goldstein, D. A., Goodenow, I., Graham, M. L., Gressler, W. J., Gris, P., Guy, L. P., Guyonnet, A., Haller, G., Harris, R., Hascall, P. A., Haupt, J., Hernandez, F., Herrmann, S., Hileman, E., Hoblitt, J., Hodgson, J. A., Hogan, C., Howard, J. D., Huang, D., Huffer, M. E., Ingraham, P., Innes, W. R., Jacoby, S. H., Jain, B., Jammes, F., Jee, M. J., Jenness, T., Jernigan, G., Jevremović, D., Johns, K., Johnson, A. S., Johnson, M. W. G., Jones, R. L., Juramy-Gilles, C., Jurić, M., Kalirai, J. S., Kallivayalil, N. J., Kalmbach, B., Kantor, J. P., Karst, P., Kasliwal, M. M., Kelly, H., Kessler, R., Kinnison, V., Kirkby, D., Knox, L., Kotov, I. V., Krabbendam, V. L., Krughoff,

K. S., Kubánek, P., Kuczewski, J., Kulkarni, S., Ku, J., Kurita, N. R., Lage, C. S., Lambert, R., Lange, T., Langton, J. B., Le Guillou, L., Levine, D., Liang, M., Lim, K.-T., Lintott, C. J., Long, K. E., Lopez, M., Lotz, P. J., Lupton, R. H., Lust, N. B., MacArthur, L. A., Mahabal, A., Mandelbaum, R., Markiewicz, T. W., Marsh, D. S., Marshall, P. J., Marshall, S., May, M., McKercher, R., McQueen, M., Meyers, J., Migliore, M., Miller, M., Mills, D. J., Miraval, C., Moeyens, J., Moolekamp, F. E., Monet, D. G., Moniez, M., Monkewitz, S., Montgomery, C., Morrison, C. B., Mueller, F., Muller, G. P., Muñoz Arancibia, F., Neill, D. R., Newbry, S. P., Nief, J.-Y., Nomerotski, A., Nordby, M., O'Connor, P., Oliver, J., Olivier, S. S., Olsen, K., O'Mullane, W., Ortiz, S., Osier, S., Owen, R. E., Pain, R., Palecek, P. E., Parejko, J. K., Parsons, J. B., Pease, N. M., Peterson, J. M., Peterson, J. R., Petravick, D. L., Libby Petrick, M. E., Petry, C. E., Pierfederici, F., Pietrowicz, S., Pike, R., Pinto, P. A., Plante, R., Plate, S., Plutchak, J. P., Price, P. A., Prouza, M., Radeka, V., Rajagopal, J., Rasmussen, A. P., Regnault, N., Reil, K. A., Reiss, D. J., Reuter, M. A., Ridgway, S. T., Riot, V. J., Ritz, S., Robinson, S., Roby, W., Roodman, A., Rosing, W., Roucelle, C., Rumore, M. R., Russo, S., Saha, A., Sassolas, B., Schalk, T. L., Schellart, P., Schindler, R. H., Schmidt, S., Schneider, D. P., Schneider, M. D., Schoening, W., Schumacher, G., Schwamb, M. E., Sebag, J., Selvy, B., Sembroski, G. H., Seppala, L. G., Serio, A., Serrano, E., Shaw, R. A., Shipsey, I., Sick, J., Silvestri, N., Slater, C. T., Smith, J. A., Smith, R. C., Sobhani, S., Soldahl, C., Storrie-Lombardi, L., Stover, E., Strauss, M. A., Street, R. A., Stubbs, C. W., Sullivan, I. S., Sweeney, D., Swinbank, J. D., Szalay, A., Takacs, P., Tether, S. A., Thaler, J. J., Thayer, J. G., Thomas, S., Thornton, A. J., Thukral, V., Tice, J., Trilling, D. E., Turri, M., Van Berg, R., Vanden Berk, D., Vetter, K., Virieux, F., Vucina, T., Wahl, W., Walkowicz, L., Walsh, B., Walter, C. W., Wang, D. L., Wang, S.-Y., Warner, M., Wiecha, O., Willman, B., Winters, S. E., Wittman, D., Wolff, S. C., Wood-Vasey, W. M., Wu, X., Xin, B., Yoachim, P., and Zhan, H., "LSST: From Science Drivers to Reference Design and Anticipated Data Products," *ApJ* **873**, 111 (Mar 2019). DOI: https://doi.org/10.3847/1538-4357/ab042c.

[2] Ivezić, Ž. and The LSST Science Collaboration, "LSST Science Requirements Document," (January 2018). Vera C. Rubin Observatory LPM-17, https://ls.st/LPM-17.

[3] Jurić, M., Axelrod, T., Becker, A., Becla, J., Bellm, E., Bosch, J., Ciardi, D., Connolly, A., Dubois-Felsmann, G., Economou, F., Freemon, M., Gelman, M., Gill, R., Graham, M., Guy, L., Ivezić, Ž., Jenness, T., Kantor, J., Krughoff, K., Lim, K.-T., Lupton, R., Mueller, F., Nidever, D., O'Mullane, W., Patterson, M., Petravick, D., Shaw, D., Slater, C., Strauss, M., Swinbank, J., Tyson, J., Wood-Vasey, M., and Wu, X., "Data Products Definition Document," (July 2023). Vera C. Rubin Observatory LSE-163, https://lse-163.lsst.io/.

[4] Bosch, J., AlSayyad, Y., Armstrong, R., Bellm, E., Chiang, H.-F., Eggl, S., Findeisen, K., Fisher-Levine, M., Guy, L. P., Guyonnet, A., Ivezić, Ž., Jenness, T., Kovács, G., Krughoff, K. S., Lupton, R. H., Lust, N. B., MacArthur, L. A., Meyers, J., Moolekamp, F., Morrison, C. B., Morton, T. D., O'Mullane, W., Parejko, J. K., Plazas, A. A., Price, P. A., Rawls, M. L., Reed, S. L., Schellart, P., Slater, C. T., Sullivan, I., Swinbank, J. D., Taranu, D., Waters, C. Z., and Wood-Vasey, W. M., "An Overview of the LSST Image Processing Pipelines," in [*Astronomical Data Analysis Software and Systems XXVII*], Teuben, P. J., Pound, M. W., Thomas, B. A., and Warner, E. M., eds., *Astronomical Society of the Pacific Conference Series* **523**, 521 (2019). DOI: https://doi.org/10.48550/arXiv.1812.03248.

[5] Guy, L. P., Bechtol, K., Carlin, J. L., Dennihy, E., Ferguson, P. S., Krughoff, K. S., Lupton, R. H., Slater, C. T., Findeisen, K., Kannawadi, A., Kelvin, L. S., Lust, N. B., MacArthur, L. A., Martinez, M. N., Reed, S. L., Taranu, D. S., and Wood-Vasey, W. M., "Faro: a framework for measuring the scientific performance of petascale Rubin Observatory data products," in [*Software and Cyberinfrastructure for Astronomy VII*], *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **12189**, 121890M (Aug. 2022). DOI: https://doi.org/10.1117/12.2628887.

[6] Jenness, T., Bosch, J. F., Salnikov, A., Lust, N. B., Pease, N. M., Gower, M., Kowalik, M., Dubois-Felsmann, G. P., Mueller, F., and Schellart, P., "The Vera C. Rubin Observatory Data Butler and pipeline execution system," in [*Software and Cyberinfrastructure for Astronomy VII*], *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **12189**, 1218911 (Aug. 2022). DOI: https://doi.org/10.1117/12.2629569.

[7] Economou, F., Krughoff, S., Sick, J., Fausti, A., Thornton, A., and Hoblitt, J., "DM-EFD prototype implementation," (April 2019). Vera C. Rubin Observatory SQuaRE Technical Note SQR-029, https://sqr-029.lsst.io/.

[8] Fausti, A., "Sasquatch: beyond the EFD," (March 2023). Vera C. Rubin Observatory SQuaRE Technical Note SQR-068, https://sqr-068.lsst.io/.

[9] Fausti, A., "The EFD replication service," (January 2021). Vera C. Rubin Observatory SQuaRE Technical Note SQR-050, https://sqr-050.lsst.io/.

[10] Krughoff, S. and Economou, F., "On accessing EFD data in the Science Platform," (September 2018). Vera C. Rubin Observatory Data Management Technical Note DMTN-082, https://dmtn-082.lsst.io/.

[11] Economou, F., "The Rubin Science Platform," (July 2023). Vera C. Rubin Observatory Data Management Technical Note DMTN-212, https://dmtn-212.lsst.io/.

[12] Sick, J., "The Times Square service for publishing parameterized Jupyter Notebooks in the Rubin Science platform," (November 2021). Vera C. Rubin Observatory SQuaRE Technical Note SQR-062, https://sqr-062.lsst.io/.

[13] Wood-Vasey, M., Bellm, E., Bosch, J., Carlin, J., Guy, L., Ivezic, Z., MacArthur, L., and Slater, C., "Test Datasets for Scientific Performance Monitoring," (March 2024). Vera C. Rubin Observatory Data Management Technical Note DMTN-091, https://dmtn-091.lsst.io/.

[14] Ferguson, P. S., Sevilla, I., and Hernández, J., "M1M3 Tests," (May 2024). Vera C. Rubin Observatory Commissioning Technical Note SITCOMTN-088, https://sitcomtn-088.lsst.io/.

[15] Kelkar, K., "Summary of TMA tests," (April 2024). Vera C. Rubin Observatory Commissioning Technical Note SITCOMTN-121, https://sitcomtn-121.lsst.io/.

[16] Hernández, J., "M2 Tests Index," (March 2024). Vera C. Rubin Observatory Commissioning Technical Note SITCOMTN-120, https://sitcomtn-120.lsst.io/.

[17] Fausti, A., "USDF EFD storage requirements," (February 2024). Vera C. Rubin Observatory SQuaRE Technical Note SQR-085, https://sqr-085.lsst.io/.

[18] Fausti, A., "The EFD Transformation Service," (August 2021). Vera C. Rubin Observatory SQuaRE Technical Note SQR-058, https://sqr-058.lsst.io/.

[19] Dowler, P., Rixon, G., Tody, D., and Demleitner, M., "Table Access Protocol Version 1.1." IVOA Recommendation 27 September 2019 (Sept. 2019). DOI: https://doi.org/10.5479/ADS/bib/2019ivoa.spec.0927D.

[20] Ribeiro, T., Mills, D., and Clements, A., "Replacing DDS with Kafka as middleware technology for the Rubin Observatory Control System," in [*Software and Cyberinfrastructure for Astronomy VIII*], Ibsen, J. and Chiozzi, G., eds., *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series* **13101**, 13101–59, in press (2024). https://dx.doi.org/doi#here.

[21] Owen, R., "Exploring Kafka for Telescope Control," (April 2022). Vera C. Rubin Observatory TSTN-033, https://tstn-033.lsst.io/.