PROCEEDINGS OF SCIENCE

Prototyping a Bulk Data Management System for CTAO with Rucio

Syed Hasan,[a,*] **Matisse Allaux,**[b] **Adrian Biland,**[a] **Frederic Gillardo,**[b] **Hancheng Li,**[c] **Maximilian Linhoff,**[d] **Etienne Lyard,**[c] **Marion Pierre,**[b] **Volodymyr Savchenko**[e] **and Roland Walter**[c] **for the CTAO Consortium**

[a] *ETH Zürich, Stefano-Franscini-Platz 5, 8093 Zürich, Switzerland*

[b] *Laboratoire d'Annecy De Physique Des Particules (LAPP), 9 Chem. de Bellevue, 74940 Annecy, France*

[c] *Université de Genève, Département d'Astronomie, Faculté de Science, Chemin Pegasi 51, CH-1290 Versoix, Switzerland*

[d] *CTAO, Platanenallee 6, 15738 Zeuthen, Germany*

[e] *EPFL, Rte Cantonale, 1015 Lausanne, Switzerland*

E-mail: shasan@phys.ethz.ch

Bulk Data Management, including the long-term archiving of massive datasets, is critical for advancing high-energy gamma-ray astrophysics research by ensuring data accessibility and scientific reproducibility. Within the Cherenkov Telescope Array Observatory (CTAO) [1], managing and preserving petabyte-scale data poses unique challenges. To address these challenges, we present our prototyping efforts for the Bulk Data Management System (BDMS), a key sub-system of CTAO's Data Processing and Preservation System (DPPS) designed for long-term preservation. BDMS leverages Rucio [2] — an open-source data management system developed at CERN. BDMS manages the ingestion of data products on-site, replication of data between CTAO Data centers, ensure their long-term preservation, and provide an interface to ingest, query, and retrieve. We provide details on the BDMS architecture and its main functional blocks, namely: Ingest (including replication), Data Management (track preservation, and monitoring), Archival Storage, File Query and Access, and BDMS Administration. Our prototyping contributions include containerised deployment using Helm charts and continuous integration tests on a Kubernetes (K8s) cluster provided by DESY Computing/Data center; metadata handling by implementing a setup to extract and store metadata from raw (DL0: Data level 0) data products, thereby enabling high-level dataset queries. Finally, we provide details on current status and outline our future plans.

39th International Cosmic Ray Conference (ICRC2025)
15–24 July 2025
Geneva, Switzerland

ICRC 2025
The Astroparticle Physics Conference
Geneva July 15-24, 2025

*Speaker

1. Introduction

The Cherenkov Telescope Array Observatory (CTAO) [1] is the next-generation very-high-energy (VHE) gamma-ray astronomical observatory under construction at two sites in La Palma (CTAO-N, Canary islands) in the northern and Paranal (CTAO-S, Chile) in the southern hemispheres. It will be sensitive to a range of energy accessible only to a few precursor Cherenkov Telescope experiments [3–5] and it will enable one to observe the non thermal universe with unprecedented sensitivity and angular resolution. The telescope arrays will be composed of several tens of telescopes, each generating multi-Gbps of data per second, for a total raw-data throughput in the order of 500 Gbps.

The Bulk Data Management System (BDMS) is a sub-system of Data Preservation and Preservation System (DPPS) [6] in CTAO. It manages the temporary and long-term storage of all data products from raw data (DL0) up to Science-ready (DL5) and for next-day and reprocessed analysis. DPPS includes the following sub-systems for pipelines: Calibration Production Pipeline (CalibPipe), Simulation Production Pipeline (SimPipe), Data Processing Pipeline (DataPipe), and Data Quality Pipeline (QualPipe). These pipelines provide software and workflows that execute on another DPPS sub-system, the Workload Management System (WMS) [7]. WMS retrieves data and metadata from BDMS and commits back the processing output there. Both BDMS and WMS are overseen by CTAO computing's Operations Management System (Ops).

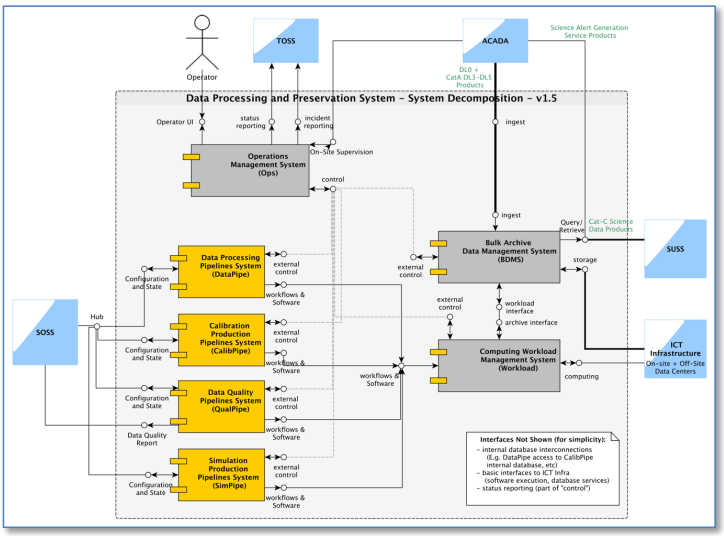

Figure 1: BDMS Functional diagram with external interfaces to ACADA, SUSS, off-site ICT data centers, and internal interfaces to WMS.

Figure 1 shows the functional diagram of BDMS with its internal interface to WMS and external interaces to ACADA (Array Control and Data Acquisition) supervisory system at each CTAO site, SUSS (Science User Support System) for high-level science operations workflow, Ops (Operations Management System), and off-site ICT datacenters (DCs) for storage. BDMS is a software system composed of several services, agents and databases (DBs), that manages the replication of data between remote CTAO sites to the DCs and between DCs and ensuring that they reach a preserved state where each data product exists at two or more off-site DCs. CTAO has

defined and selected six DCs: four off-site DCs (CSCS, PIC, DESY, Frascati) and two on-site DCs (CTAO-N:La Palma, CTAO-S: Paranal). BDMS also ensures that all data products are findable by metadata and retrievable and manages their movement to and from cold (e.g., tape) storage when necessary. In the DPPS data flow, BDMS manages DL0 raw data and preserves it long-term in the Bulk Archive. The DL1 and DL2 data produced in the intermediate processing stage of DPPS is managed by BDMS for short-term storage and in this scenario no ingest is needed and BDMS is just a storage system.

2. BDMS Architecture

BDMS follows the Open Archival Information Systems (OAIS) [1] standards design principle, which itself is largely based on the experience from high energy astronomy archives. This architecture has different functional units, each meant to handle different aspects of the data products lifecycle. The INTEGRAL archive[2] was the first one to be implemented in the framework of OAIS. Figure 2 shows a high-level BDMS architecture design following the OAIS functional model. First the ingest process makes sure that the data is valid, and has all the required associated metadata available, extracting it from the data itself. Both data and associated metadata are archived by the ingest component, while a data management component, active throughout the lifetime of the archive monitors the state of the data products and takes the necessary steps to ensure their preservation. The query component allows users and processes to explore the archive, based on high-level metadata queries. Eventually a file access component allows to retrieve actual data from the archive, either based on metadata queries or direct logical filenames access. The Dissemination information package (DIP) constitutes the data product that is archived and preserved with associated metadata and is available to be queried and retrieved by the BDMS user and it also constitutes the data product to be accessed by the WMS for conversion into higher-level data products.

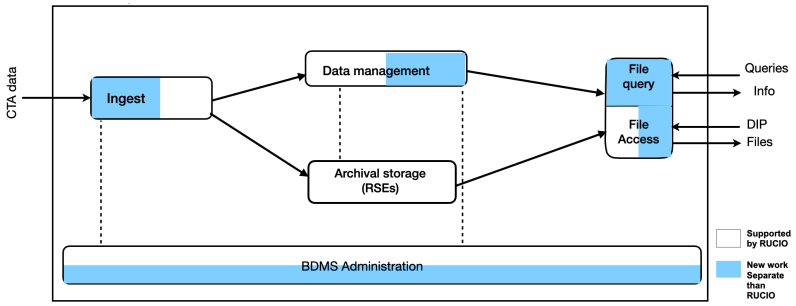

Figure 2: BDMS architecture showing data flow from CTA ingest through data management and archival storage to user queries.

[1] Reference Model for an Open Archival Information System, NASA, https://pds-engineering.jpl.nasa.gov/wp-content/uploads/documents/pds2010/study/data_model/oais_reference_model_blue_book.pdf

[2] INTEGRAL archive, INTEGRAL Science Data Centre (ISDC), University of Geneva, https://www.astro.unige.ch/integral/archive

2.1 Rucio

Rucio [2] is an open-source data management system originally developed by CERN to meet the data handling needs of large-scale scientific experiments such as those at the Large Hadron Collider (LHC), starting with ATLAS and later on with CMS and LHCb. It is designed to manage massive volumes of scientific data across geographically distributed storage infrastructures. Rucio enables users to organize, replicate, and access data efficiently, supporting policy-driven data placement, automated data transfers, and data recovery. It uses the concept of abstract storage element to represent storage endpoints and applies replication rules to ensure data availability and preservation. Rucio supports a wide range of data transfer protocols—including xrootd, webdav, GridFTP, and S3, making it flexible for integration with different storage technologies.

Rucio's architecture is modular and extensible, allowing for custom policy changes, metadata handling, and authentication integration. Rucio is increasingly employed in other scientific domains such as Neutrino (DUNE), Astronomy (SKAO, LSST) that require scalable and reliable distributed data management. Rucio has been selected as the core technology by CTAO for BDMS to support large-scale data ingestion, replication, bulk archive, and data management. CTAO aims to store its telescope data and preserve it for bulk archiving for at least 30 years once the data taking starts. At least 6 PB of compressed archive data will be generated annually.

2.2 Ingest

The `Ingest` component leverage Rucio to manage the ingestion of data products produced at on-site and off-site locations. The lowest-level raw data (DL0) will be produced on-site by ACADA. Higher-level data will be produced on and off-site by WMS via pipeline workflows. In this paper, we present the ACADA on-site ingestion design scenario. The `Ingest` component handles ingestion and replication of CTAO's DL0 observational data products (in ZFITS format). During a night of observations, ACADA continuously writes data products to a specific file system area on-site. Once ACADA is done with a particular data product, it changes ownership to DPPS and creates a symlink with the same name as that of the file plus an additional '.trigger' extension. Now, the file is ready for ingestion by BDMS.

The `Ingest` transforms local file paths into BDMS logical file names (LFNs) that follow a hierarchical namespace structure organized by virtual organization (VO) and RUCIO scope. Before registering files as replicas in Rucio, there is a data integrity check that verifies checksum (FITS DATASUM and CHECKSUM keywords). The files passing the check have their physics metadata extracted depending on the file type (Trigger, Subarray, and Telescope). The next step registers the data product as a replica in Rucio. Finally, the extracted physics metadata such as observation identifiers (obs_id, sb_id), telescope configurations (tel_ids, subarray_id), and temporal information (start_time, end_time) is added to the newly registered replicas according to their respective file type.

`Ingest` also manages the replication part of data management. The replication rules are applied to the newly ingested data product to distribute the physical file to off-site RSEs. The replication strategy is the following: the first rule creates exactly one replica at one of the off-site RSEs to prevent parallel transfers from the on-site location, while a second rule (when multiple copies are requested) sources additional replicas from off-site RSEs. In this way, the bottleneck

link from on-site to off-site RSE is used only once. Figure 3 presents the control sequence of ingest and replication of DL0 raw data products by BDMS.

The `Ingest` also has an ingestion daemon feature to orchestrate automated incoming data products using a multi-processing architecture supporting parallel ingestion. Upon detecting new data files via trigger file monitoring, the daemon distributes tasks across a configurable pool of worker processes, each executing the complete ingestion workflow—from file verification and metadata extraction through Rucio replica registration and replication rule creation. It also maintains task tracking with real-time metrics including success/failure counts, queue depths, and file processing rates, while making sure there is process-level isolation to ensure individual task failures don't impact overall system stability. A dedicated result handling thread asynchronously processes completed tasks and updates metrics to provide visibility into the performance of the Ingestion. The daemon ensures single-instance operation through file-based locking during the its entire run and implements graceful shutdown procedures that allow all pending tasks to complete.

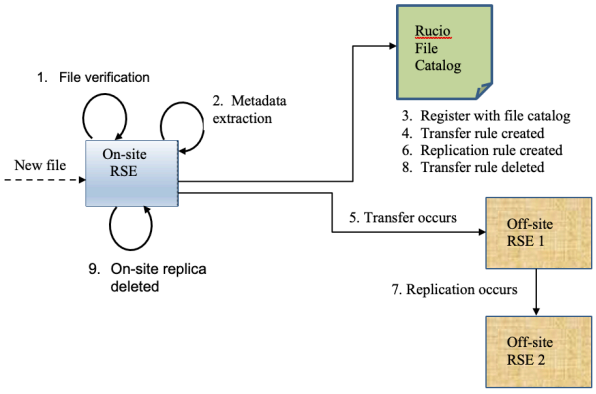

Figure 3: Control flow of Ingest and Replication of DL0 data products produced on-site from ACADA.

2.3 Data Management

Track Preservation State of Data Products Each data product after completion of its replication enters the preservation state meaning it has been fully archived. There will be a separate shepherd process that runs continuously to keep a track on each data product's preservation state. Considering a scenario of a disk failure or corruption leading to loss of files/datasets at a data center storing the archived data, data recovery and availability are the immediate goals. The data recovery and preservation is possible by a daemon process that quickly assesses the loss of files/datasets by first identifying the lost data and its location and then recovers them by copying the same files/datasets from a replica stored in a healthy data center.

Generate storage reports by monitoring An important part of the BDMS is the monitoring of Rucio storage elements (RSEs) which are deployed at the off-site and on-site data centers. Tools such as Grafana are already integrated with Rucio to generate such reports. This can also be extended to monitor the FTS service and Rucio database. The output is in the form of GUI reports about disk usage arising from data transfers at each RSE for different types of DL0 data products – events, monitoring, logs, report.

2.4 Archival Storage (RSEs)

RSEs serve as storage end points to store data products (DL0 FITS and ZFITS raw data, high-level raw data > DL0, and simulated data) while keeping them in preservation state after replication for data management needs. There are different storage elements available - XRootD, dCache, or even webdav. An RSE is a container for physical files. It has a unique identifier and a set of meta-attributes. Its attributes include the supported protocols, its host address and port, the quality of service it provides, and its physical location.

2.5 File Query

BDMS enables users to search for files using either metadata-based queries or direct logical file name (LFN) lookups. For metadata queries, users can search using metadata filters (e.g., data_type="monitoring", obs_id=200000002) to retrieve a list of matching file identifiers (DIDs). For direct access, users can query specific files by their LFN to retrieve associated metadata including checksums, file size, and custom attributes.

2.5.1 Generate Dissemination Information Package (DIP)

The DIP consists of the list of files returned from a metadata query or a single file from an LFN query. When users search using metadata filters, Rucio (through `DIDClient`) returns all matching DIDs. For LFN queries, users directly access a specific file's complete metadata set. In both cases, users can retrieve full metadata for any file in the DIP, accessing all attributes beyond just the query criteria used for filtering.

2.6 File Access and Retrieval

After obtaining a DIP (list of files from a metadata query or a specific LFN), users can retrieve the actual file content. BDMS supports downloading individual files by LFN or entire datasets from metadata query results. For retrieving files from the archive, there are various options: (1) direct download by LFN with automatic checksum validation, (2) parallel downloads of multiple files from a DIP, (3) selective download of specific files from a dataset, (4) RSE-specific downloads. In regard to requests coming from WMS to access files from BDMS, DIRAC integration enables programmatic access through either Rucio clients or more commonly via the Rucio file catalog (RFC) plugin interface.

2.7 BDMS GUI and Administration

We created a prototype based on W3Browse[3] from NASA HEASARC allowing users to make high-level queries with metadata and get back corresponding data as explained in the next section. The BDMS Manager is responsible for end-to-end BDMS operations including ingest, data management, storage elements (RSEs) set-up at data centers and their management, and query and file access. The authentication and authorization for users to access bulk archive data is granted by CTAO either through a Token-based (Indigo IAM) or X.509 certificate mechanism.

[3] W3Browse, NASA HEASARC, https://heasarc.gsfc.nasa.gov/cgi-bin/W3Browse/w3browse.pl

3. BDMS prototyping contributions

Containerised deployment with docker compose We implemented containerised deployment for BDMS leveraging Rucio development images and docker compose files. We use two types of certificate generation and proxy. The first one is a real user grid-based certificate and CA certificate from CERN and use them to generate a VOMS (Virtual Organization Management Server) proxy certificate from a VOMS server where we have an account and membership. The second type uses self-signed certificates and proxy. With both these certificate generation methods authenticating Rucio client to storages and Rucio server was successful. This allowed us to test Rucio core functionality using its command line interface - uploading and downloading of files, replication using rules for on-site to off-site data transfers, querying for list of file replicas and list of RSEs, attaching files to datasets, setting up accounts with root and non-root permissions, etc.

We also worked with W3Browse to test BDMS bulk archive functionality for query and file retrieval. The metadata extraction from ACADA DL0 files by looking at both event and auxiliary file constitutes the first step. The metadata file generation from the extracted metadata serves as input for populating the database in W3Browse and allowing the parameter search, and query interface to get info on files for retrieval.

3.1 Current status

3.1.1 Helm chart based containerised deployment at DESY

We moved to Helm chart deployment at the DESY Kubernetes Test Cluster using *kind* (kubernetes in docker) set-up as this benefits us to set-up local *kind* cluster for BDMS developments and also have the same *kind* set-up with additional stages and checks on the CI pipeline run. For BDMS, there is a dedicated single chart that has sub-charts for all dependencies (Rucio server, Postgres, Rucio daemons, certificate generation, File Transfer Service - FTS (including ActiveMQ broker). In addition, there are sub-charts to deploy test storages (RSEs), BDMS test client, bootstrapping Rucio, configure test Rucio, and also chart for DPPS AIV toolkit deployment to generate test report. At the DPPS level, there is also a single 'dpps' chart that has the chart for each subsystem and it helps the AIV (Assembly Integration Verification) team to have integration tests for the overall system.

3.1.2 BDMS application code and Testing with continuous integration (CI) pipelines

The BDMS code development started with few unit tests and integration tests to test Rucio core API functionality: upload, download, query by LFN and metadata using filter, file localization to list replicas, test dataset retrieval by metadata attributes. We then made progress towards development of `Ingest` component first by implementing use-cases related to (i) Ingestion on-site (with metadata added) with data produced by ACADA and transferred to DPPS, (ii) replication to off-site RSEs from on-site RSE for data preservation, and (iii) ingestion queue daemon to ingest data products concurrently as soon as it is discovered at the on-site top directory (or shared mount path with the BDMS client).

For the application code, there are unit tests to test individual functions and also integration tests to test the complete ingest workflow with daemon included. The code is integrated with

Gitlab CI/CD setup that performs static checks of the code, Helm lint/kube lint, builds the project including documentation, run the unit tests and integration tests, uploads the coverage report and trigger the analysis of the quality gate via Sonarqube, building of docker containers, and pushing documentation to Gitlab pages.

4. Current Status and Next Steps

The first release of BDMS - 0.1.0 delivered important milestones: Deployment of Rucio 35.4 using Helm charts, BDMS client package with custom CTAO-specific Rucio policy package, and integration tests for DPPS Release 0.0 use-cases. The second release of BDMS - 0.2.0 focused on delivering unit and integration tests for DPPS Release 0.1 use-cases: ACADA case to ingest files on-site, Replication of Data Products, Extraction of metadata from ACADA-LST1 (Large Size Telescope) DL0 FITS files. The current release of BDMS - 0.3.0 adds Ingestion Queue daemon feature to discover files on-site and submit them concurrently for parallel ingestion. The next step is to work on upcoming milestones such as ACADA DL0 ingestion of files on-site at La Palma and replication to PIC and other CTAO datacenters, integration of Indigo IAM authentication for users, and ingestion of data produced by WMS after running pipelines workflows for short-term storage.

5. Acknowledgements

We gratefully acknowledge financial support from the agencies and organizations listed here: https://www.ctao.org/for-scientists/library/acknowledgments/

References

[1] Zanin, R., et al. (CTA Observatory, CTA Consortium, LST). (2022). CTA – the world's largest ground-based gamma-ray observatory. *Proceedings of Science*, *ICRC2021*, https://doi.org/10.22323/1.395.0005

[2] Barisits, M., et al. (2019). Rucio: Scientific data management. *Computing and Software for Big Science*. Springer International Publishing. https://doi.org/10.1007/s41781-019-0026-3

[3] De Lotto, B., et al. (MAGIC Collaboration). (2011). The MAGIC telescopes: Performance, results, and future perspectives. *TAUP 2011 Proceedings*.

[4] Prokoph, H., et al. (H.E.S.S. Collaboration). (2019). The H.E.S.S. experiment: Current status and future prospects. *ICRC 2019 Proceedings*.

[5] Weekes T.C. et al. VERITAS: the Very Energetic Radiation Imaging Telescope Array System, *Astroparticle Physics 17, 2002*.

[6] Kosack, K. (2024, November). Data processing and preservation for CTAO. Astronomical Data Analysis Software & Systems XXXIV, Malta. https://pretalx.com/adass2024/talk/EAEHVC/

[7] Arrabito, L., Bregeon, J., Faure, A., Gueta, O., Pigoux, N., & Tsaregorodtsev, A. (2024). The Cherenkov Telescope Array Observatory workflow management system. *EPJ Web of Conferences*, *295*, 04044. https://doi.org/10.1051/epjconf/202429504044