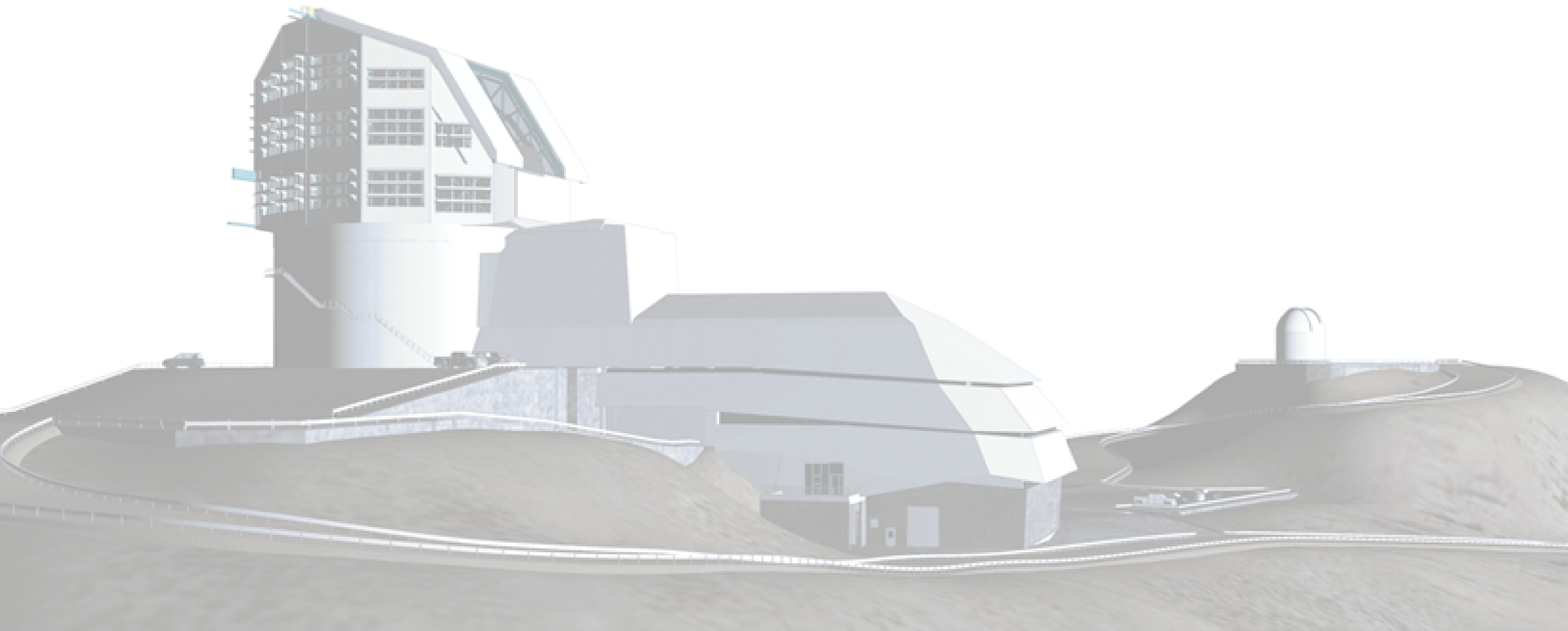
Vera C. Rubin Observatory
Data Management

Routine HSC/DC2 Processing at SLAC as early demonstrator

Hsin-Fang Chiang, Richard Dubois

RTN-024

Latest Revision: 2023-07-10



Abstract

Specifications to start routinely process HSC or DC2 data at SLAC/USDF in a fully automated way using PanDA

Change Record

Version	Date	Description	Owner name
1	YYYY-MM-DD	Unreleased.	Hsin-Fang Chiang

Document source location: `https://github.com/lsst/rtn-024`

Contents

Routine HSC/DC2 Processing at SLAC as early demonstrator

1 Introduction

The high-level goal is to routinely process a medium dataset at SLAC, in a fully automated way using PanDA. The scale of medium datasets follows the definitions in DMTN-091. The first target can be the HSC-RC2 or the DC2-test-med1 datasets that are processed monthly at NCSA currently.

Developers use these routine processings to regularly confirm the state of the DRP pipeline. A good early demonstration of this service at SLAC would bring a number of early Data Facility functions into play. To date, these processings have been done by hand; the goal would be to automate them while using PanDA to drive the workflow. Some early campaign tooling would be needed to orchestrate the set of data needing to be submitted to PanDA by `ctrl_bps`.

2 Specifications

2.1 Raw data size

- HSC-RC2: 432 exposures; about 3TB.
- DC2's test-med-1 dataset: 345 exposures; about 1TB.
- all DC2 run 2.2i is about 10 TB.

2.2 Butler repository

A Postgres database will be used as the Butler Registry. The Butler datastore can be either an object store or a POSIX shared file system; the Butler datastore should be assessible from the exeuciton nodes.

2.3 CPU cycles to the processing

<1000 node-hours, old data https://confluence.lsstcorp.org/display/ sthrush/Node+Utilization+for++HSC-RC2+Reprocessing+Jobs

More current data: how much CPU time was used for the more recent NCSA runs?

2.4 Medium-sized DRP workflow

A standard DRP HSC-RC2 or DC2-test-med-1 workflow includes multiple (around 6?) BPS submissions run as a sequence. Definitions of the included Science Pipelines tasks are part of the stack release e.g. `https://github.com/lsst/obs_subaru/blob/master/pipelines/DRP.yaml`. Currently, the submission orchestration is handled manually by humans; the new campaign tooling or tentative campaign prototype should do this to allow automatic submissions.

2.5 Rubin software stack

Weekly stack releases need to be provided.

For execution, PanDA should soon be able to use Rubin's official stack docker releases. (Not today... but soon hopefully)

For developers, an installation on the filesystem may be needed to allow version mix-and-match. (developer processing is a stretch goal)

2.6 Workflow management system

This will use the PanDA system and the PanDA plugin as implemented in the `ctrl_bps` package. The frontend is part of the Rubin `lsst_distrib` software stack.

The PanDA system provides a monitoring page, but customization is needed if this is the only monitoring page.

2.7 Submission management

Automatic tooling does not exist as of today.

Mechanism to manage bps submissions. This includes tracking tools of submissions. The new campaign tooling from DP0.2 may be used here. First prototype at SLAC can be a hackaround for a known dataset and workflow without the new campaign tooling.

For automatic submission, the triggering mechanism may be cron jobs on a submission machine, kubernetes cron jobs, Apache Airflow, jenkins-like triggering, GH Action, and so on. This may be set up on a regular schedule to generate the parameters and submit to PanDA via BPS.

Need to be able to easily and accurately track the status of submissions.

2.8 Policy for output data storage

In the short term all are saved. Later, a policy to remove old runs is wanted. Steve Pietrowicz is developing Butler expiration tooling for the OODS; it might be applicable here.

2.9 PanDA DOMA authentication

it may be better for DRP to run as a service account that operators can sudo to or otherwise get credentials for, rather than have individual users with their own database and object store accounts/credentials. This will avoid having releases (which may have parts run by different operators, or no operator at all if triggered by cron) be mishmashes from different users.

PanDA DOMA authorization lasts 24 hours. Also the access permissions for the area in the repo where the outputs are written. The service account's umask isn't quite right to allow others in the same group to access those directories. The current SLAC setup needs adjustments.

2.10 Profiling (bonus)

Jim Chiang's profiling tools, created for DESC DC2 at NERSC

2.11 PanDA as a developer service (bonus)

Next step, to allow running workflows with a non-released stack. This might need a shared stack on POSIX shared filesystem where all execution nodes can assess. Or, use a standard CVMFS `lsst_distrib` distribution plus developer's own ticket branch on disk.

3 Needs

- One or two maintainers of the Butler repos. They put together the Butler repos and ensure the readiness and correctness of the Butler repos. Occasional updates to the repos may be needed. Data Curation group?
- PanDA setup: SLAC as a PanDA client site. Submission environment. PanDA needs to be able to use Rubin's standard weekly stack releases without Sergey building images each time.
- submission mechanism. (1) need to be able to submit automatically and programmatically (2) authentication for PanDA
- submission tracking tooling, including status monitoring of a sequence of submissions. The PanDA monitoring page as-is today is not sufficient; developers need to know the relevant overall run status and failures a lot more easily than the current page and developers aren't expected to know PanDA specifics.

4 Smaller test

automatic weekly pipeline_check or ci_hsc Needs a fresh registry every time, and create the repo from scratch.

Currently at NCSA the ci_hsc is run weekly manually.

A References

[DMTN-091], Wood-Vasey, M., Bellm, E., Bosch, J., et al., 2023, *Test Datasets for Scientific Performance Monitoring*, DMTN-091, URL `https://dmtn-091.lsst.io/`,
Vera C. Rubin Observatory Data Management Technical Note

B Acronyms