PanDA for the LSST Rubin experiment

2024 March 7

Brian Yanny - Fermilab (for the Rubin Campaign Management Team)

PanDA forum

Description of the Rubin experiment workflows

Basic daily,annual Rubin operations (currently scheduled to begin late 2025):

The 8m telescope on Cerro Pachon, Chile will obtain up to **2000 exposures (each with 200 CCD detector 100MB images of the sky) per night (40TB/night)**. During the **first year** of full operations a '**Data Release Processing (DRP)**' will combine 200 nights of exposures (**40 million files**) into some **1M coadd images and tables of detected objects with measured properties (flux, position, shape, size), along with (roughly) 100 million of 'Difference image' catalogs** and postage stamps of variable and moving objects on the sky. Also many intermediate files generated and tracked.

Processing will be done at 3 main sites: CC-IN2P3 (FRench DF), RAL, Lancaster (UKDF) and SLAC Data Facility (USDF) with some additional processing done on the mountain top and base sites in Chile.

Test DRP subset with PanDA and test dataset

PanDA has been selected for use as the Workflow management system for **Data Release Processing (DRP)** for at least the first full year of Rubin operations.

In 2023 we successfully used PanDA to process a ‘small’ (**3% of a DRP**) subsample of real data (Hyper Suprime Camera (HSC) mosaic images through arrangement with the NAO Japan) at the USDF and are now working to integrate our other processing sites into a multi-site processing system.

About 15 million individual jobs (‘quanta’) run with the PanDA-Doma system generating 2PB of output data products on 50TB of input data over more than 4 months wallclock time with a 3000 core processing cluster.

We’ll describe here some of our experiences with this test processing highlighting our PanDA usage.

Significant differences between HEP and Astro datasets:

Rubin jobs are different from many High Energy Physics (HEP) jobs in that:

1. Rubin are more **fine-grained** (i.e. looking for feedback on 40 million individual detector jobs). I.E. many more individual files to track.
2. Job run times range from a **few minutes** (was a few seconds until ‘clustering’) up to (currently) **24 hours** or more. Implications for queue sizes and timeouts.
3. Astronomers are uncomfortable with losing *any* data. ‘Every detection on **every detector matters**’. Implications for retries and can lead to ‘long tails waiting for a few straggler jobs to finish up before moving forward’.

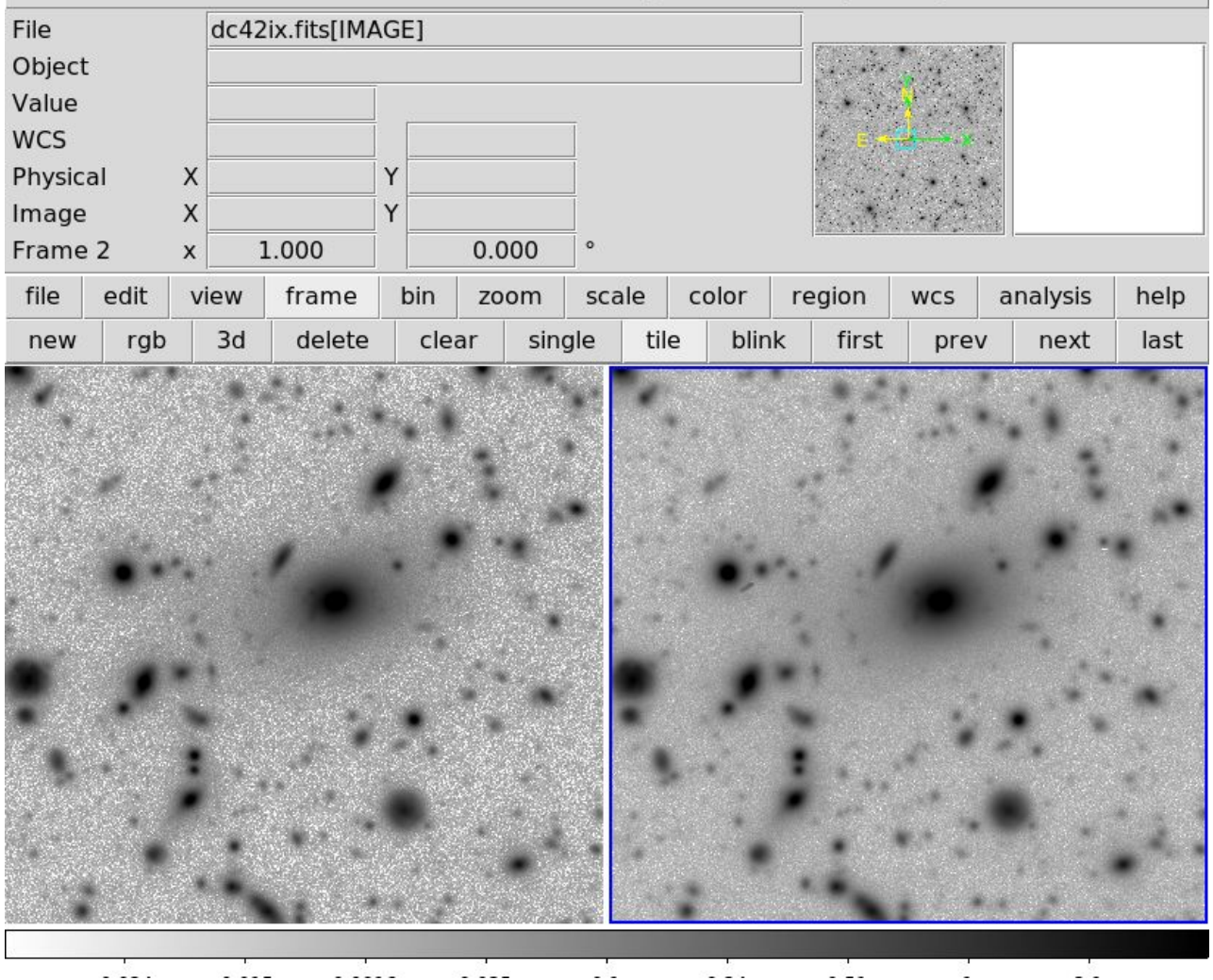
Success:

Shallow (left, N=30)

and deep (right,N=300)

Coadd images

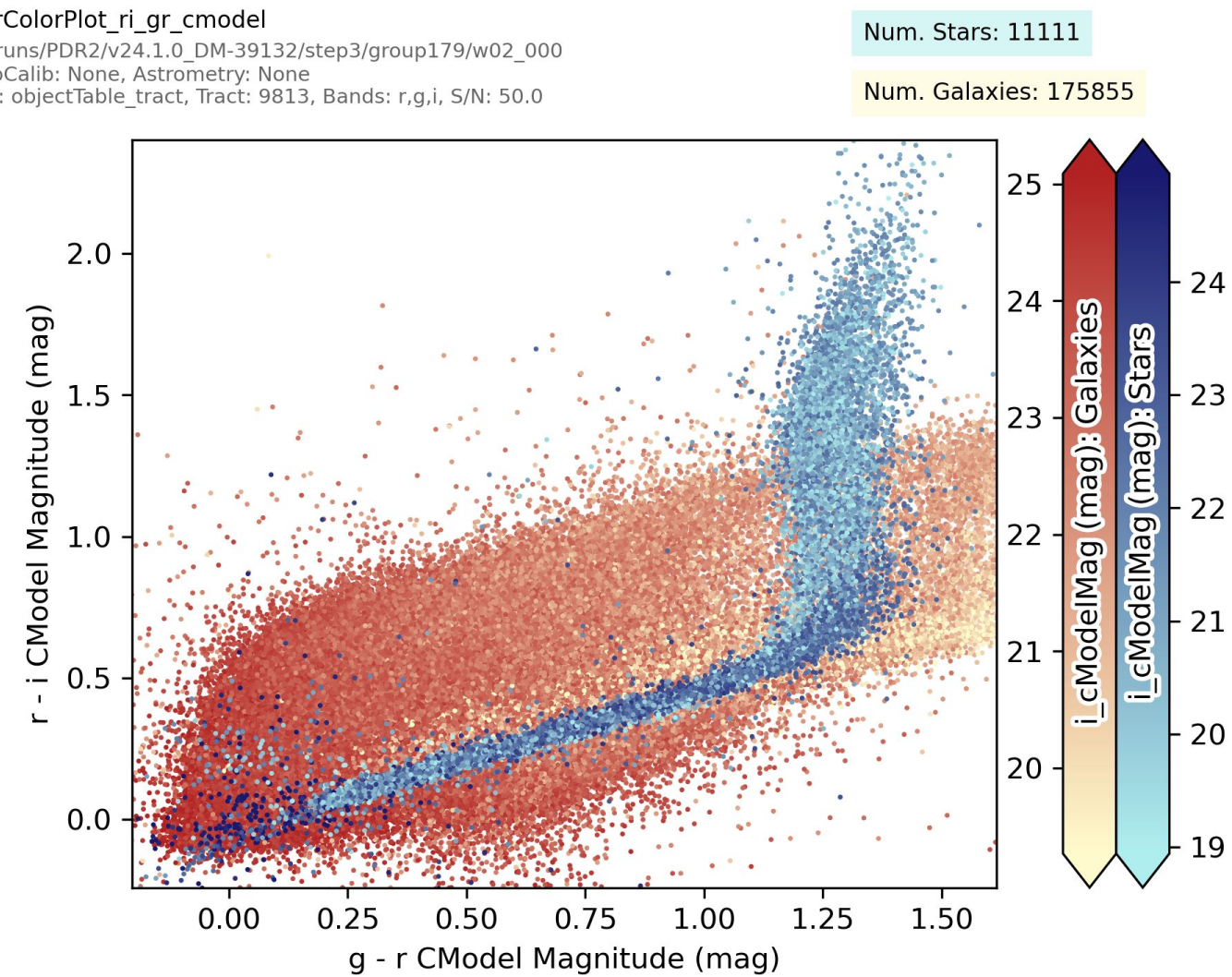
of the sky



Success:

Coadd Object catalogs,

With properties such as color and shape available to separate star (blue) from galaxy (red) populations.



Successful interaction with PanDA team::

PanDA developers (Wen Guan, Eddie K) (and earlier Sergey P, Zhaoyu Y) were and are very responsive to the many issues that we have come up with and made updates to configurations to help with issues.

Some notes on issues (most resolved) were:

1. Switching over from Oracle CERN based PanDA-Doma system to USDF Postgres backed local (to Rubin) PanDA system. Increasing local harvester power.
2. Adding longer wallclock time outs (72 hours to 96 hours on higher memory queues)
3. Modifying queue sizes from a few hundred up to several thousand slots (still tweaking)
4. Increasing timeouts when components of the system we located far apart.
5. Setting a ‘throttle’ when 100K jobs are in the queues to prevent trying to run too many simultaneously, overwhelming IDDS and other components.
6. Many small configuration and connection issues between PanDA SLURM and other systems.

Work in progress:

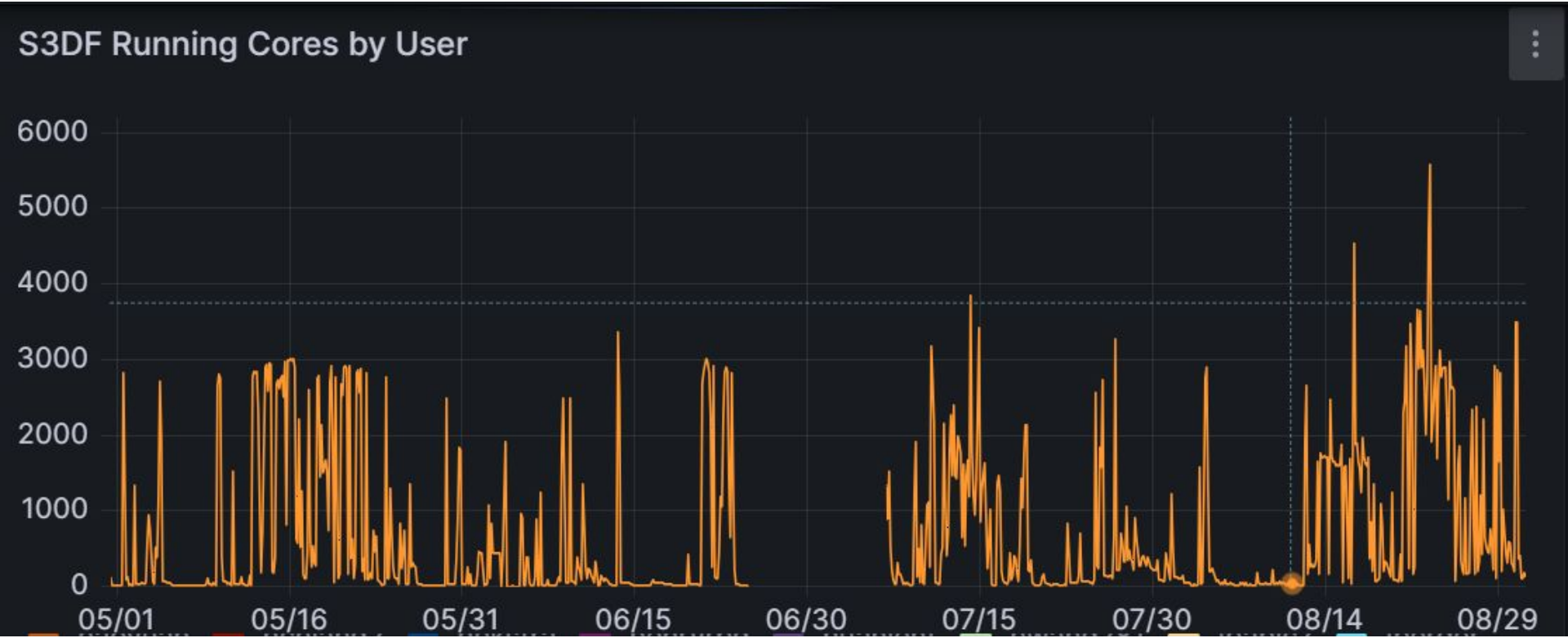
Several issues came up during the several month processing campaign, some of which were related to the nature of the Rubin data.

1. We had trouble keeping the available cpu cores evenly and steadily loaded.

   Since this campaign (in 2023), we've worked with the PanDA development team to set queue sizes and wallclock timeouts, improve our detection of issues such as timeouts and memory overflows to automate retries and increased resources available to PanDA harvester. This has improved the situation greatly – though the short and long mixed nature of some Rubin jobs is still an issue in some cases.

   Followup: We have had successful 'stress tests' of PanDA where we're able to keep serveral thousand cores busy for many hours 'solid', but still have rampdown-rampup issues moving from step-to-step where there are gaps in filling the queues. May be Rubin workflow bottleneck not PanDA.

Work in Progress: Desire steady use of many K cores



Work in progress:

2. Tackling jobs which are running ‘too long’ sometimes.

This may not be a PanDA issue… we have occasional jobs which run for up to 60 hours that, normally take only 15 hours. One thing that may be happening is multiple jobs are getting ‘pinned’ to run on the same core/cpu and thus each running 4x too slow. It may be happening for many many ‘short running jobs’ but we only notice for the very long 15 hour jobs since it extends our campaign run time many hours past nominal. This may be a SLURM or other compute cluster config issue, but if PanDA has any advice or sees similar things we’d be interested in talking about it. We are working on more monitoring to catch these things as they happen.

Work in progress:

3. Running on remote sites sometimes leads to timeouts

We now have a central PanDA at USDF, but are using it to submit jobs to remote sites at FRDF, UKDF. Most things run the same, but sometimes there are timeouts or extra retries at these remote sites that seem to be related to the large distances involved. It would be good to have a better understanding of how communication between the different pieces of the system (panDA server, harvester, iids, database?) works to understand where the various timeouts might be happening. We're sure this is all known and documented but we just haven't been able to absorb the complexity yet amongst the operators.

Summary:

- PanDA is a very powerful, scalable workflow system with excellent monitoring and logging.
- It is also quite complex and we rely quite a lot on the PanDA experts to help us when things go wrong (often not a PanDA issue, but sometimes hard to tell).
- We find communication a key, and appreciate the attempts by PanDA developers/experts to plug into and use the Rubin JIRA (problem report) system and to attend meetings and communicate on Rubin SLACK.
- Though we have a ways to go to make things ‘push button’ –work is progressing steadily in terms of scaling up the processing and the efficient use of resources so we are ready for our first multisite data release processing in late 2025 to early 2026.