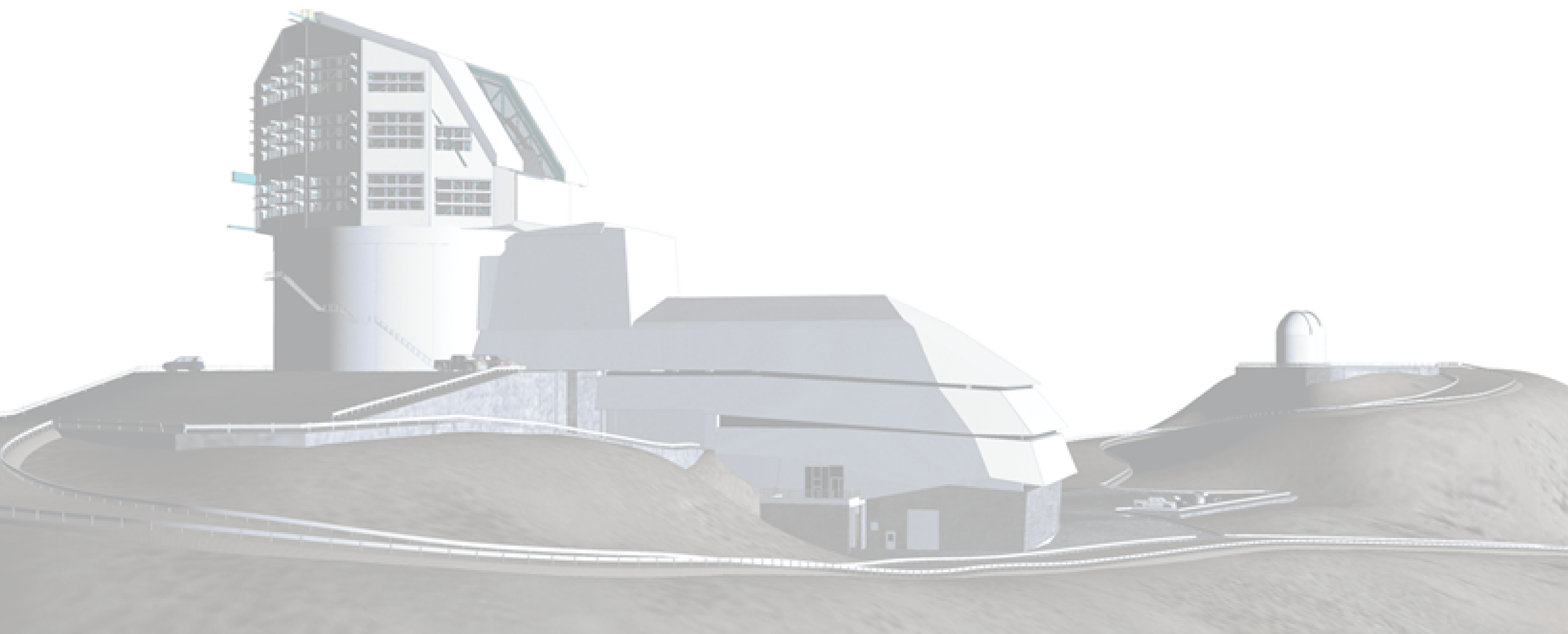
Vera C. Rubin Observatory
Rubin Observatory Operations

HSC PDR2 Reprocessing and Operations Rehearsal for DRP

Yusra AlSayyad, Jennifer Adelman-McCarthy, Brian Yanny, Orion Eiger, Eric Charles, Sierra Villarreal, Zhaoyu Yang, Wen Guan, Jim Bosch, Colin Slater, Michelle Gower, Tim Jenness, Lauren MacArthur, Eli Rykoff, Sophie Reed

RTN-063

Latest Revision: 2023-08-16



Abstract

In 2023, the campaign management team processed 100s of sq degrees of precursor data though a data release production on the new US data facility at SLAC. This activity demonstrated the production of a data release under simulated operational conditions.

Change Record

Version	Date	Description	Owner name
1	YYYY-MM-DD	Unreleased.	Yusra AlSayyad

Document source location: `https://github.com/lsst/rtn-063`

Contents

HSC PDR2 Reprocessing and Operations Rehearsal for DRP

1 Introduction

1.1 Test DRP on a 3% subset of real HSC data

In preparation for DRP (Data Release Processing), a set of some 17K exposures from the HSC PDR2 dataset were pushed through a current version of the DRP pipelines.

These exposures are then coadded into 710 tracts, including about 671 WIDE tracts of repeated depth a few in grizy bands, and 39 DEEP+UDEEP with up to 426 overlapping exposures in some bands in some UDEEP tracts.

These 17K exposures along with the coadd tracts and other outputs generated, represent approximately 3% of the first years' Rubin DR1 dataset.

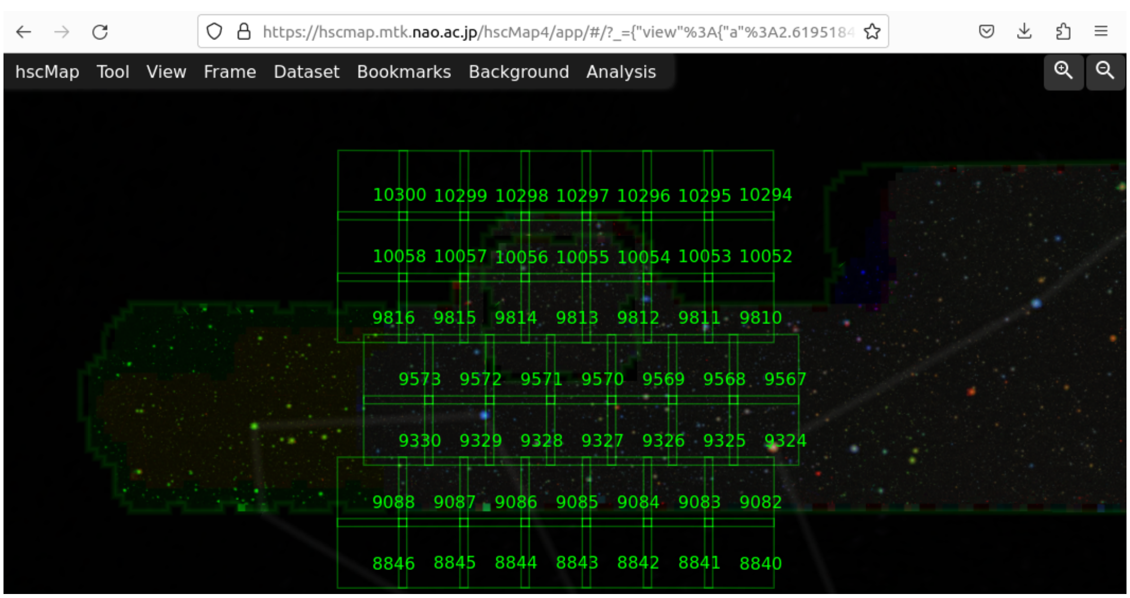


FIGURE 1: PDR2 close up with tract numbers.

1.2 PDR processing and instruments: HSC vs. LSSTCam

Each detector is a 2K x 4K pixel CCD image, of size about 57MB. There are 103 science quality CCDs/exposure. Detector 9 is masked due to known quality issues and ccds 104-112 are not on-sky science chips in the HSC mosaic.

The sky is divided into 'tracts' and subdivided into 'patches', as defined by skymap (hsc_rings_v1) which maps tract number to a square region in (RA,DEC) coordinates.

For HSC, each tract covers about 2.6 sq degrees on the sky and is subdivided into 81 patches on a 9×9 grid.

In contrast, the full LSSTCam has 4K x 4K pixel CCDs, there are 189 science CCDs each about 100 MB in size. Furthermore each tract on the sky is divided into a 7×7 grid patches.

For purposes of this exercise, the calibration exposures (flat fields in each band, biases, non-linearity maps, sky pupil per band and bad pixel gain and full well maps per detector) are all taken as given 'ancillary' inputs, pre-constructed and outside the scope of this campaign.

For DP1, DP2 and DR1 all good visits obtained to over a certain time interval, six months to one year in length are gathered at the start of the DRP. All are processed through step1 (Instrument Signature Reduction (ISR), image and PSF characterization, source detection and measurement), step2a (combine gather source tables), and step2b (perform astrometric calibration with a 2MASS or eventually Gaia reference catalog, extract calibration star catalogs) at multiple processing sites if available. Each visit may be processed independently and in parallel.

For the PDR2 HSC exercise, multi-site was not yet up, and so all processing was done at one site (the USDF).

Following step2b, the extracted, (and relatively small), calibration star catalogs are gathered together over the entire campaign footprint and are used in step2c to generate a single Forward Global Calibration Module (FGCM) photometric solution with zeropoints for each exposure processed.

These astrometric solutions and photometric zeropoints from steps 2b and 2c are applied in step2d and gathered together by visit in step2e.

There may be a 'visit-veto' rejection of poor quality (i.e. bad seeing) exposures at the end of step2e but prior to the start of step3. A subcollection is made containing only the remaining good visits, which are used for further processing. For the HSC PDR2 case, about 2.5K out of 17.3K visits were rejected leaving about 15K visits to be coadded.

Step3 of PDR processing performs coaddition, combining all calibrated visit images, first 'warping' them using the astrometric solution to put them onto common geometric footprint and then adding them weighted appropriately by photometric flux zeropoint and noise properties.

The astrometric algorithm used for PDR2 HSC is called 'jointcal'. This older algorithm may be replaced by 'gbdes' in future processings, as that algorithm accounts for 'tree rings', DCR and other subtle distortions.

Coadds for the HSC PDR2 footprint are currently done at the patch level, though future processings will also produce 'overlapping cell-based' coadds which further subdivides patches and tracts into regions of the sky where the PSF in each band is well-modeled as constant across an entire cell with no discontinuous jumps due to detector edges. These cell-based coadds in turn allows for the accurate measurement of weak lensing shears of all source stacked objects when combined with a PIFF PSF model.

Between May 2023 and (approx) Sep 2023 the HSC PDR2 was reprocessed with Rubin Science pipelines stack version v24.1.0, which was based on a weekly stack w_2023_7. Some modifications or fixes to the stack were applied during the processing, as described below.

In this document we discuss campaign management section 2, processing section 3 and quality assurance section 3 details of the HSC PDR2 reprocessing.

2 Campaign Management and communication

Here we cover the management structures in place for HSC PDR2 this includes the gro ups and meetings like the change control for the pipeline version.

2.1 Oversight

Two Operations departments involved were the production of PDR2;

2.2 Data Production

The membership of the team was:

- Yusra AlSayyad

2.3 Campaign Management

2.4 System Performance

The Rubin Observatory System Performance department is responsible for ensuring that the LSST as a whole is proceeding with the efficiency and fidelity needed to achieve its science requirements at the end of the 10-year survey. Aspects of this charge that were exercised extensively as part of PDR2 were a) QA and performance characterization analyses of the LSST data products by the Verification and Validation team and b) enabling the community to access and analyze the data and publish results by the Community Science team.

- Colin Slater (Lead Verification and Validation Scientist)

2.5 Coordination

During the production of HSC PDR2 regular coordination meetings were held every week.

2.6 Work Management

We used Jira to track work related to the Data Preview. Epics and milestones were created in the `DM` Jira Project. Story tickets were then attached to each epic For Data Management, to properly integrate the work into existing Data Management processes, any tickets that would result in code changes in pipelines software or middleware packages were created in the Data

Management Jira project. For System Performance, all work was carried out on tickets in the `PREOPS` Jira Project.

The status of the epics and how they related to the relevant milestones was monitored as part of the weekly coordination, DPLT or SPLT meetings.

2.7 Change Control

PDR2 used v24.1.0 of the LSST Science Pipelines Software and that was derived from a weekly release from w23(?) Needed to use .rc2, .rc3 and then two hot fixes.

The Data Management Change Control Board (DMCCB), DPLT and SPLT delegated authority to a new Data Release Steering Committee that had the following membership:

- Yusra AlSayyad, representing the pipelines team.
- Colin Slater, representing the verification and validation team.

1. A request is made that a ticket should be applied to the release branch by applying a `backport-v24` tag to the Jira ticket.
2. The board would then discuss the relative merits of the back-port and if approved a `backport-approved` label would be added.
3. The work on the back-port would then be scheduled by the relevant T/CAM following instructions in the developer guide.[1]
4. Once the code is on the `v24.0.x` branch a `backport-done` label would be applied.

A Jira query was constructed to find all the tickets and track their porting status. There were XXX tickets approved for back-porting as part of the version 24 release process. If a ticket was rejected its label was removed, making it hard to determine counts for the number of tickets in that category. Three tickets were left in the requesting state in case they were needed, one is for a clean-up to the database schema that was discovered after we had finalized the processing; another was for an improvement to the graph-building efficiency but would have

[1] `https://developer.lsst.io/work/backports.html`

involved a very difficult back-port because there had been a package reorganization since the release branch had been created; and the final ticket was an improvement to the matched catalog filtering.

Once all the necessary back-porting has been completed for a specific step, the release manager would be instructed to start the process of creating a new patch release of the Science Pipelines. During PDR2 we made two formal releases of the version 24 software: v24.1.0.rc2 and v2 .1.0.rc3

This allowed us to state which release was used for each step, although we ensured that changes in later patch releases would not affect the processing from steps that were already completed using older patch releases.

There were additionally two 'hot fixes' (code changes in github which were checked out and setup and executed during processing, but not formally cut into a release stack) for dynamic sky estimation (meas_algorithms) and for healSparsePropertyMap fix to allow propertyMaps and Consolidated property maps to be made for areas which were missing some patches or some pieces of the footprint.

3 PDR2 processing on USDF cluster

The data processing was done using the Production and Distributed Analysis System (PanDA; DMTN-168) at the USDF at SLAC. The PanDA system handled workflow orchestration and job retries. Based on pre-production testing, five PanDA queues (`lowmem`, `mediummem`, `highmem`, `extra-highmem`, and `merge`) were deployed. Each queue had fixed numbers of slots, memory limits, wallclock runtime limits and pull-or-push mode as described in this table:

The production processing was organized into seven logical "steps". The high level workflow of workflows and the step organization is described in RTN-001 Sect. 2.1.4. Pipelines, V&V, and Processing teams all focused on one step at a time. Before the production processing started in each step, "pilot runs" with candidate software were carried out and signed off by V&V and Pipelines teams (Sect 2).

The workflow generation and submission were done via the PanDA BPS plugin a terminal window logged into the USDF. The BPS YAML configurations can be found in a branch of the

GitHub repo (https://github.com/lsst-dm/cm-prod).

Workflow progress was tracked via PanDA's iDDS monitoring page deployed at CERN and the JIRA-based campaign tooling (RTN-023).

On many occasions (20%) rescue workflows skipping successful jobs were run.

The processing took place between May 2023 to Sep 2023 and the total cpu usage over the course of PDR2 was approximately XXXX M core-hour; the compute resource usage is summarized below. Notable issues which came up during the production were summarized as follows.

- step1
  - The RSP's "large" memory option has 12 GB and imposed a limit on the size (number of quanta) of the workflows that could be submitted, as otherwise quantum graph generation failed due to an out-of-memory error. A "huge" memory option became available in late January 2022, doubling the memory to 24 GB and enabling larger workflows to be submitted for subsequent processing steps.
  - The quantum graph build time increased from about 1.5 hours for submissions at the beginning of step1 processing to about 8 hours at the end. This was due to the use of a single output chained collection, which needed to be examined in quantum graph generation, but which grew ever larger as production proceeded, thus increasing the graph generation time. To avoid this issue, starting in step3, we used separate output collections for the different workflow submissions in a given processing step, and then joined the separate collections together once all workflows were completed.
  - To improve efficiency, we clustered together 315 quanta (consisting of different step1 tasks for multiple detectors in the same exposure) into a single job on the PanDA system. However, if one quantum failed among these 315 quanta, the whole job stopped and led to unattempted quanta. The intent for this behavior was so that downstream jobs would know about upstream failures, but in this case, the detectors were independent of each other, and there were no downstream jobs in our step1 workflows. To resolve this issue of unattempted quanta, we ran rescue workflows where quanta for different detectors were no longer clustered with each other. Subsequently the default behavior was changed so that an individual

failure among clustered quanta would not stop the remaining quanta from being attempted.

- step2
  - We expected that step2 would need a total of 3 workflows ($\approx$ 6600 visits each), based on quantum graph generation tests, but execution butler creation became the bottleneck instead, and we instead required 14 smaller workflows ($\approx$ 1500 visits each). For example, for the first of these smaller step2 workflows, quantum graph generation took 23 min, but execution butler creation took 7 hours, while job compute time was 2.5 hours (wall clock). Before step3 processing started, this issue was resolved in DM-33345, and execution butler generation time was dramatically improved, e.g., from 7 hours down to 20 minutes in this step2 example, or from 1 hour down to 6.5 minutes for a 1-tract step3 workflow.
- step3
  - Large numbers of simultaneous assembleCoadd (= image coaddition) jobs led to "too many request" errors reading from the object store (PREOPS-1034). Algorithmically, the rate spike was because the coaddition was done in small chunks and data reading happened for each chunk and each input warp. This could be alleviated by caching input files instead of requesting them from the object store each time. The caching configuration in the butler repo was changed so not to expire anything in coaddition. The first 7 step3 workflows needed to be redone due to this problem. This also led to the investigation of the "hidden" errors in the coadds, where coadd images were reported ok but actually had problems (DM-33786). More details are discussed in Sect. 4). Later we found out that the initial fix of the caching was too aggressive and led to out-of-disk-space errors for a couple of large faro tasks (matchCatalogsPatchMultiBand and matchCatalogsTract), but subsequently caching was better optimized to be sufficient for coadds, but not so much to cause disk space problems for the faro tasks.
  - Long-running forcedPhotCoadd jobs were failing/re-attempting at a much higher rate than before, caused by a change in the PanDA pilot version, which reset the job heartbeat timeout limit back to its default of 2 hours, whereas it had been set to 20 hours previously. This issue also uncovered the need for more frequent heartbeat message logging for long-running forcedPhotCoadd (DM-33854) and faro tasks (DM-33820). The timeout limit was set again to 20 hours to allow step3 production to continue efficiently.

- About once every two workflows or so, a deblend job or two failed due to out-of-memory error (DM-33690), because of very bright/large objects on the coadd image. These failures were fixed in rescue workflows, but required long run times (>10 hrs) and large memory ($\approx$ 40 GB) using the `extra-highmem` queue, though ultimately with deblending results that were likely unreliable. The most extreme example (tract 4648, patch 29) took 12 unsuccessful attempts before finally succeeding in the `extra-highmem-non-preempt` queue (DM-33947), after 2 days 22.7 hours run time, 190 GB memory, and extra efforts from the PanDA team to bypass heartbeat logging issues. This was the last image processing job to be completed in step3. DM-33690 implemented deblending configuration changes to skip these problematic deblends, so this should not be an issue subsequent to DP0.2.
- The large faro tasks matchCatalogsPatchMultiBand and especially matchCatalogsTract took long run times and large memory ($\approx$ 130 GB for the latter). They needed to be run on the `extra-highmem` queue and were also prone to preemption and multiple re-attempts due to heartbeat logging issues (DM-33820).
- Some delays arose when jobs became stuck as they were unexpectedly scheduled to a non-IDF queue, which had its "region" set erroneously to "LSST", same as for the IDF queues. This issue was corrected by the PanDA support team.
- There were also some delays due to unexpected consequences from PanDA server and client updates, again resolved by the PanDA support team.

• step4

- An attempt was made to use the Google Artifact Registry (GAR) instead of Docker Hub to download the LSST processing stack. However, this resulted in intermittent authentication problems that led to 20-50% job failures in three step4 workflows. Despite an attempt to reconfigure the IDF computing cluster involved, the authentication problems continued, and we reverted to using Docker Hub and reran the affected workflows.
- We also saw a drop in the maximum number of running jobs in the IDF queue involved, from nearly 4000 down to as few as about 2000, possibly related to the above authentication issues. PanDA support increased the maximum number of new worker nodes created from 50 to 80 to help stabilize the number of running jobs back to nearly 4000.

• step5

  - We encountered a very slow start to jobs running on the PanDA system due to the large number of jobs per task (>100,000) in the first step5 workflow. This was solved by reducing the maximum number of jobs per task from the default value of 70,000 down to 30,000, so that a single large task was divided into smaller "chunks" (each with <30,000 jobs) that ran much more efficiently in the PanDA system. We do need an iDDS server with more resources.

- step6, step7
  - Minor or no issues.

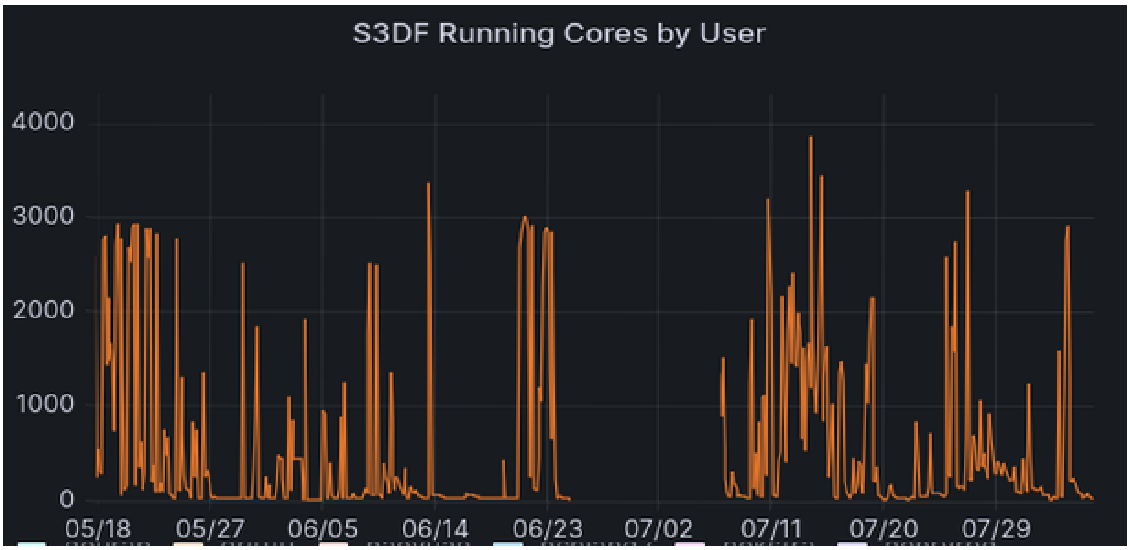


FIGURE 2: Cores in use in PDR2 step1,step2,step3 campaign.

4 Data Product Quality Assurance

During PDR2, the Verification and Validation Team was responsible for identifying problems and bugs in the pipelines and data products.

Here are some sample Quality Analysis Figure either selected from the standard V&V Analysis plots, or generated from the PDR2 step3 (coadd) output images and tables.

1. 3

2. 4

3. 8

4. 5

5. 6

6. 7

There were there main phases of V&V work:

- A period of analysis using a "pilot run" before the start of production, which ran a single tract through all steps of the pipeline, using the codebase planned for the release.
- Two "gates", one at the end of single frame processing and another after coadd construction, where production was halted for V&V to confirm that all the data products were ready before moving on to the next step of processing.
- Spot checks during processing, and follow-up of unexpected errors or failed tasks.

During these main phases, the V&V team made extensive use of the plotting capabilities in analysis_drp along with adding new diagnostic plots. Much of the analysis was performed by writing notebooks to test out new diagnostics for data products that were recently added to the pipelines. The team also drew on experience from many prior processings (particularly of Hypersuprime-cam) to quickly distinguish "known" problems from new problems.

A notable success occurred during coadd construction, when as part of the spot checks during processing the team noticed some regions inside successfully-processed patches had no coadd sources detected. One of the plots that lead to this discovery is shown below. This was particularly unexpected because entire patches are expected to succeed or fail entirely, it was highly unusual for portions to fail silently.

The eventual explanation was that the coaddition code operated on sub-patch-sized regions sequentially, in order to limit peak memory usage, and so it would read from disk different portions of the input warp images as it progressed. On a typical POSIX filesystem these reads typically either all succeed or all fail, but in the cloud environment the object store would sometimes deny individual requests as a form of rate-limiting. The coaddition code could have caught this, but since that type of failure was never encountered in prior usage, it mistakenly proceeded without raising an exception. Because this issue was identified early during coaddition, only a few days worth of processing had to be redone.

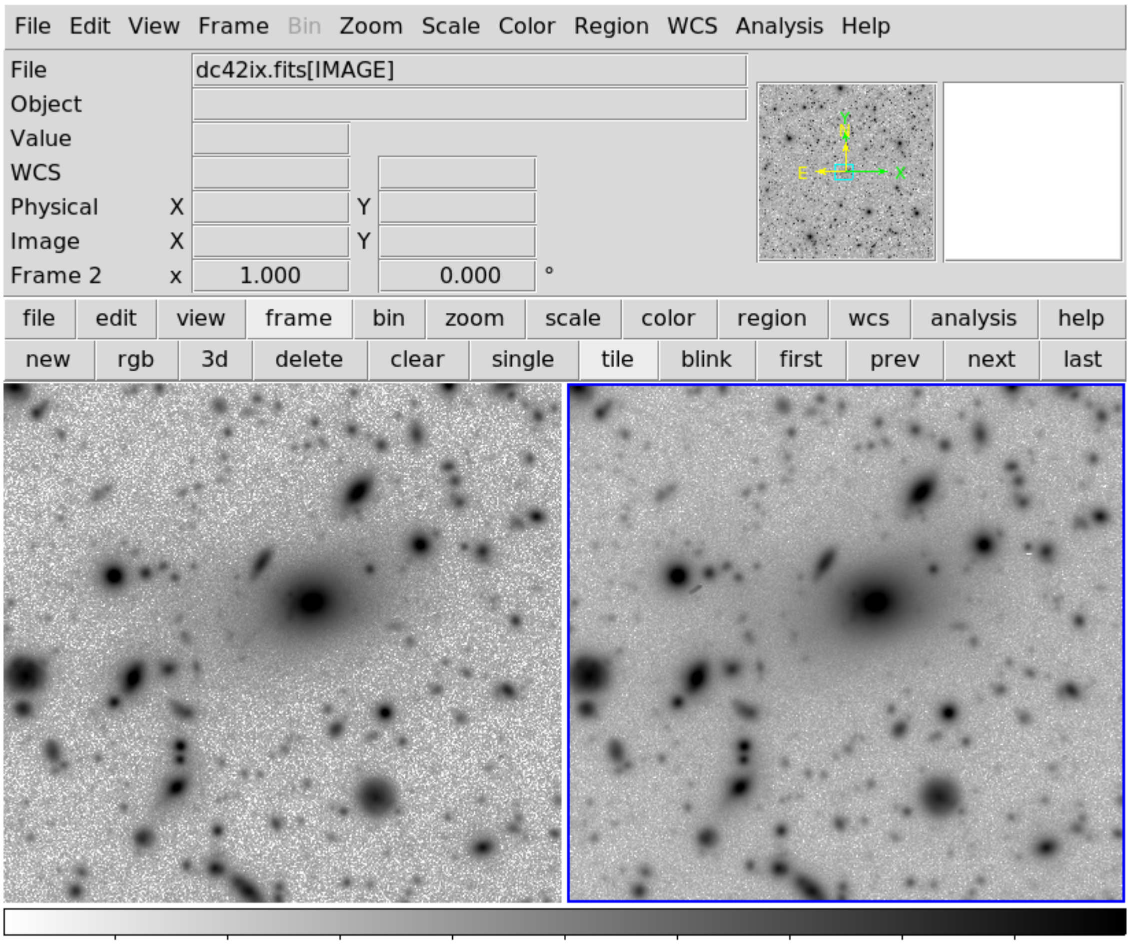


FIGURE 3: N=33 patch coadd (left) and N=300 patch coad (right). While the visible objects are similar, the coadd to the right has less sky noise due to the deeper coadd, thus the S/N of detected objects is higher.

A change!

The data previews are also a chance to learn from the issues that we didn't catch during V&V; most notably were two cases where invalid flux calibrations were being applied to certain measurement algorithms, resulting in NaNs in the output. This case shows the value of having a wide breadth of testing coverage. For future releases we will ensure that we have some form of testing for every column in every user-facing data product. Even if the tests are relatively simple, they may identify significant issues.

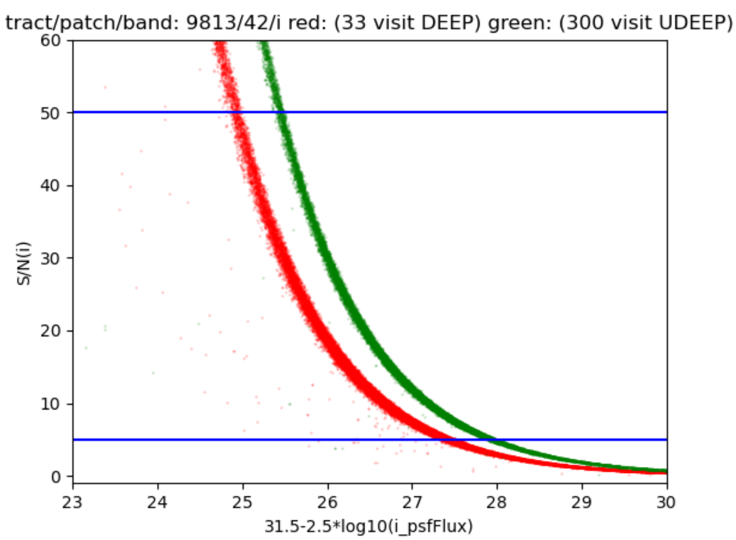


FIGURE 4: N=33 patch coadd S/N vs. PSF mag (red) and N=300 patch coadd (green). With the deeper coadd, the S/N for objects of the same magnitude is about 2x higher.

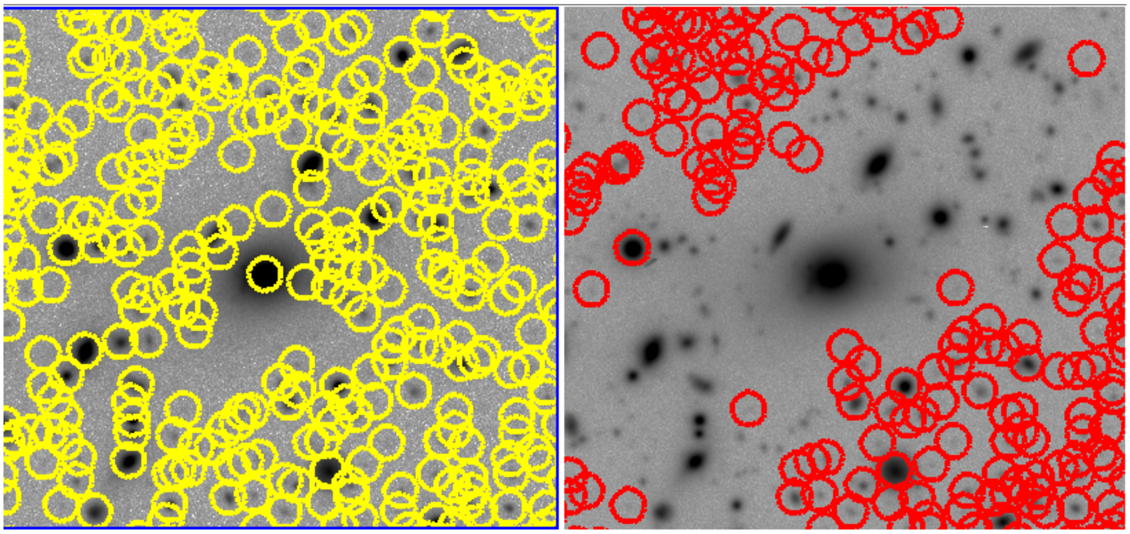


FIGURE 5: N=33 patch coadd (left) and N=300 patch coad (right). Here the deblending or detection of objects in the very deep coadd misses some objects in the wings of bright central galaxies or bright stars.

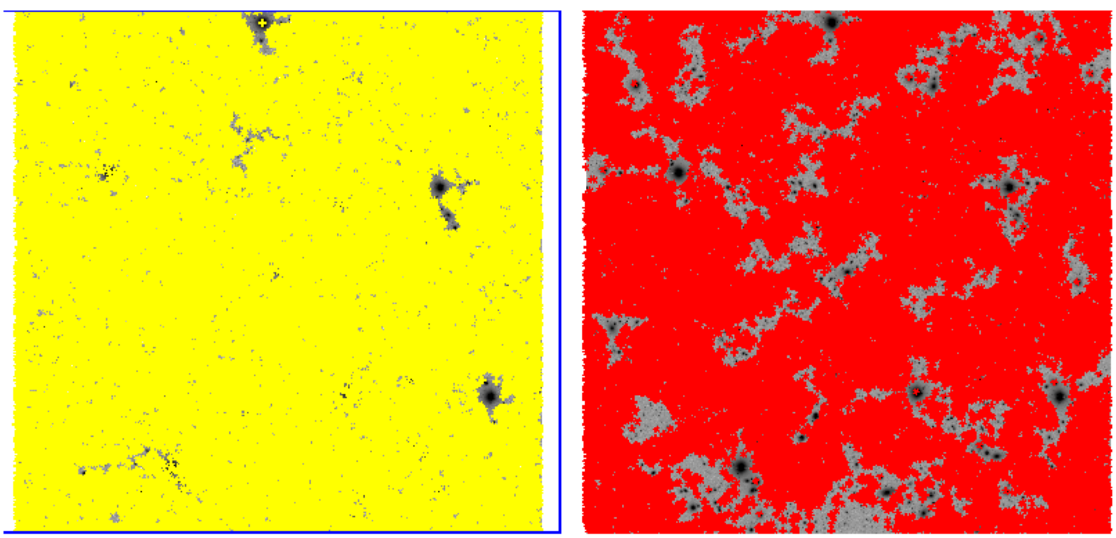


FIGURE 6: N=33 patch coadd (left) and N=300 patch coad (right). Here the deblending or detection of objects in the very deep coadd misses some objects in the wings of bright central galaxies or bright stars.

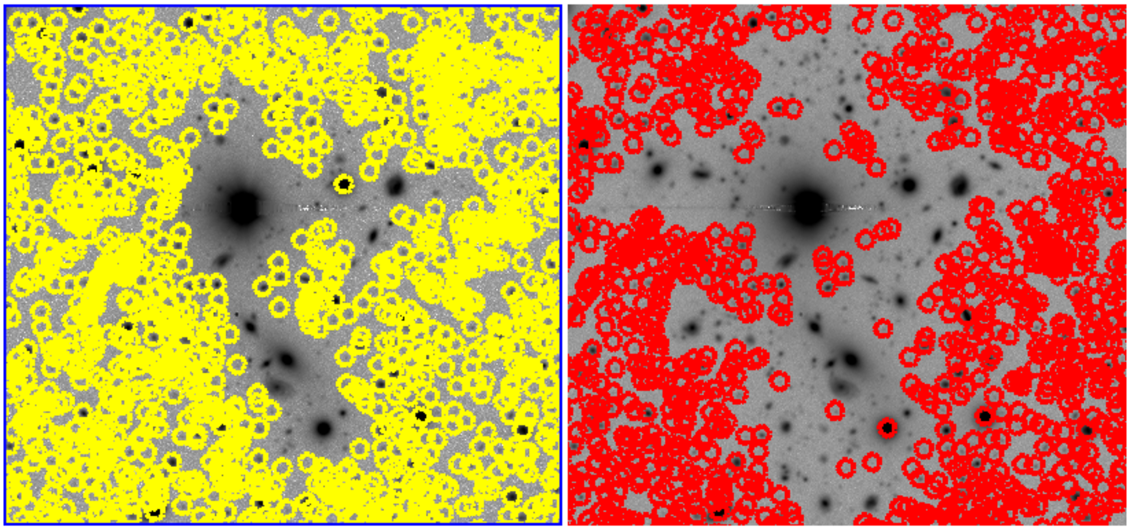


FIGURE 7: N=33 patch coadd (left) and N=300 patch coad (right). Here the deblending or detection of objects in the very deep coadd misses some objects in the wings of bright central galaxies or bright stars.

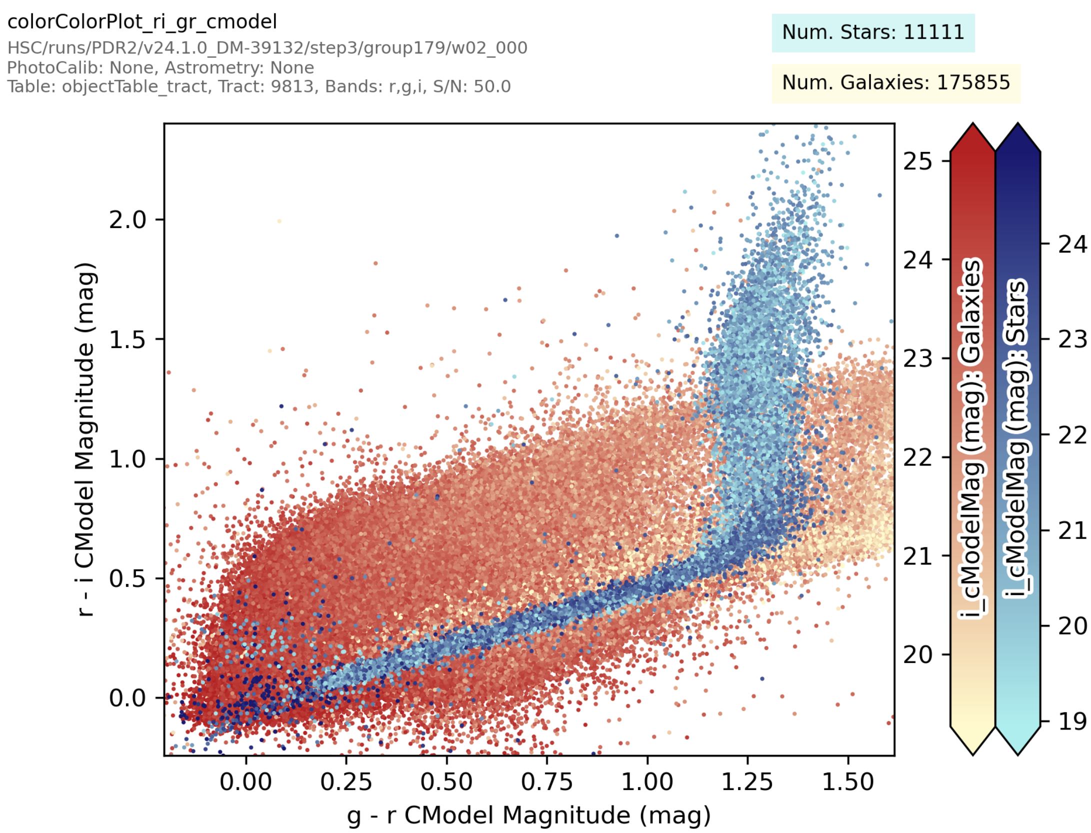


FIGURE 8: Color color plot of objects masured in UDEEP COSMOS tract 9813 with star/galaxy separation (N=300).

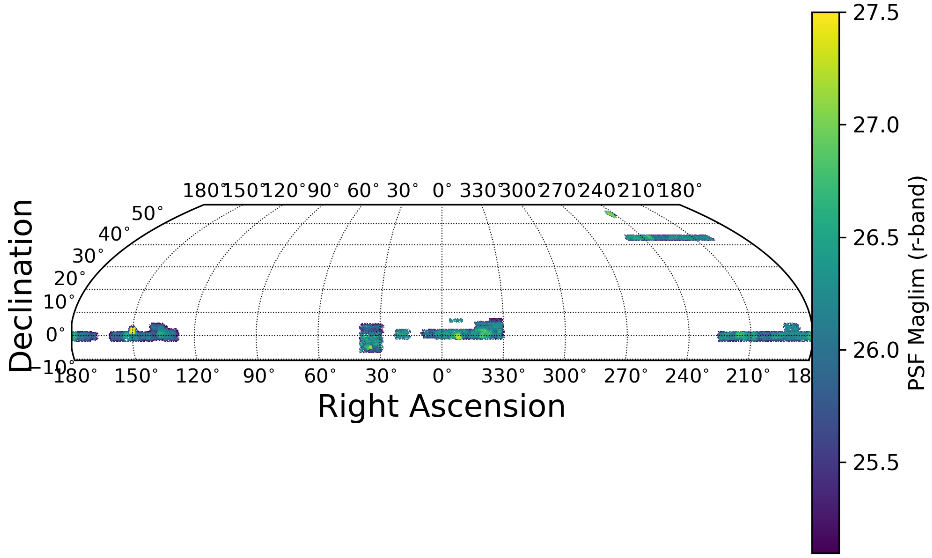


FIGURE 9: Footprint of PDR2 dataset.

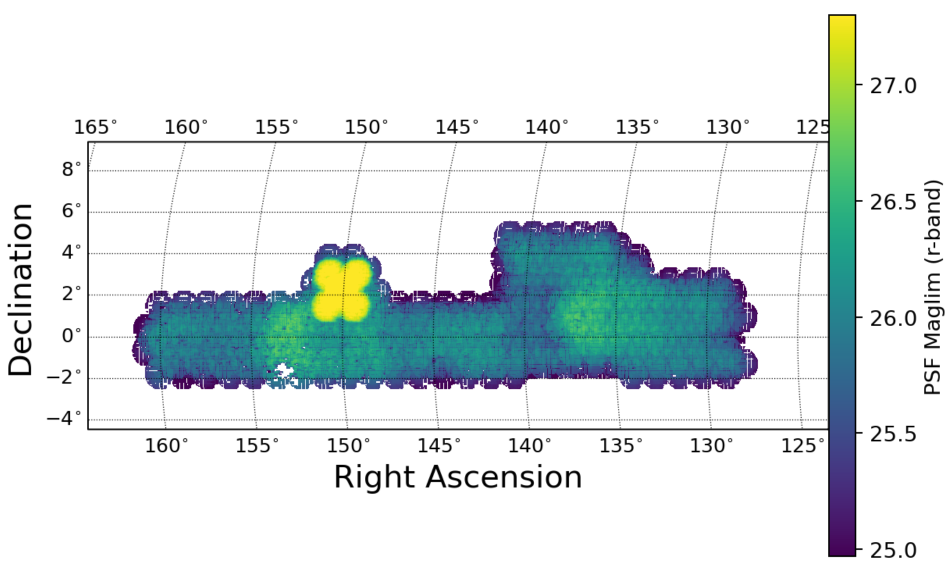


FIGURE 10: Closeup of PDR2 footprint depth near the COSMOS UDEEP tract.

As one further example of potential issues encountered during production, a few days after the start of single frame processing the pipelines began to show hundreds of failures with the error message `Exception ValueError: No reference objects supplied`. This was not seen in the pilot run, so production was paused while we investigated. By plotting the locations of these failures on the sky, we determined that these sensors fell outside the footprint of the stellar input catalog, and hence there were no stars available for calibrating these images. These were thus “unprocessable” and no corrective action was required, but it illustrated the value of realtime error collection and monitoring during production.

A References

[RTN-001], O'Mullane, W., 2021, *Data Preview 0: Definition and planning.*, RTN-001, URL `https://rtn-001.lsst.io/`,
Vera C. Rubin Observatory Technical Note

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

[RTN-023], Yanny, B., Slater, C., Padolski, S., et al., 2021, *Campaign Tooling – tools for generating, monitoring and tracking data processing campaigns*, RTN-023, URL `https://rtn-023.lsst.io/`,
Vera C. Rubin Observatory Technical Note

B Acronyms

Acronym	Description
2MASS	Two-Micron All Sky Survey
BPS	Batch Production Service
CAM	CAMera
CCD	Charge-Coupled Device
CERN	European Organization for Nuclear Research
DCR	Differential Chromatic Refraction

DEC	Declination
DEEP	Deep Extragalactic Evolutionary Probe
DM	Data Management
DMCCB	DM Change Control Board
DMTN	DM Technical Note
DP0	Data Preview 0
DP1	Data Preview 1
DP2	Data Preview 2
DPLT	DP Leadership Team
DR1	Data Release 1
DRP	Data Release Production
FGCM	Forward Global Calibration Model
GAR	Google Archive Registry
GB	Gigabyte
HSC	Hyper Suprime-Cam
IDF	Interim Data Facility
ISR	Instrument Signal Removal
LSST	Legacy Survey of Space and Time (formerly Large Synoptic Survey Telescope)
MB	MegaByte
OPS	Operations
PDR	Preliminary Design Review
PDR2	Public Data Release 2 (HSC)
POSIX	Portable Operating System Interface
PSF	Point Spread Function
PanDA	Production ANd Distributed Analysis system
QA	Quality Assurance
RA	Right Ascension
RSP	Rubin Science Platform
RTN	Rubin Technical Note
SLAC	SLAC National Accelerator Laboratory
T/CAM	Technical/Control (or Cost) Account Manager
US	United States
USDF	United States Data Facility

YAML	Yet Another Markup Language