VERA C. RUBIN OBSERVATORY

Working at USDF

Kian-Tat Lim
DM Software Architect

Commissioning Science Validation Bootcamp
22-24 May 2023

NSF AURA U.S. DEPARTMENT OF ENERGY SLAC CHARLES AND LISA SIMONYI FUND FOR ARTS AND SCIENCES LSST CORPORATION

What Is USDF?

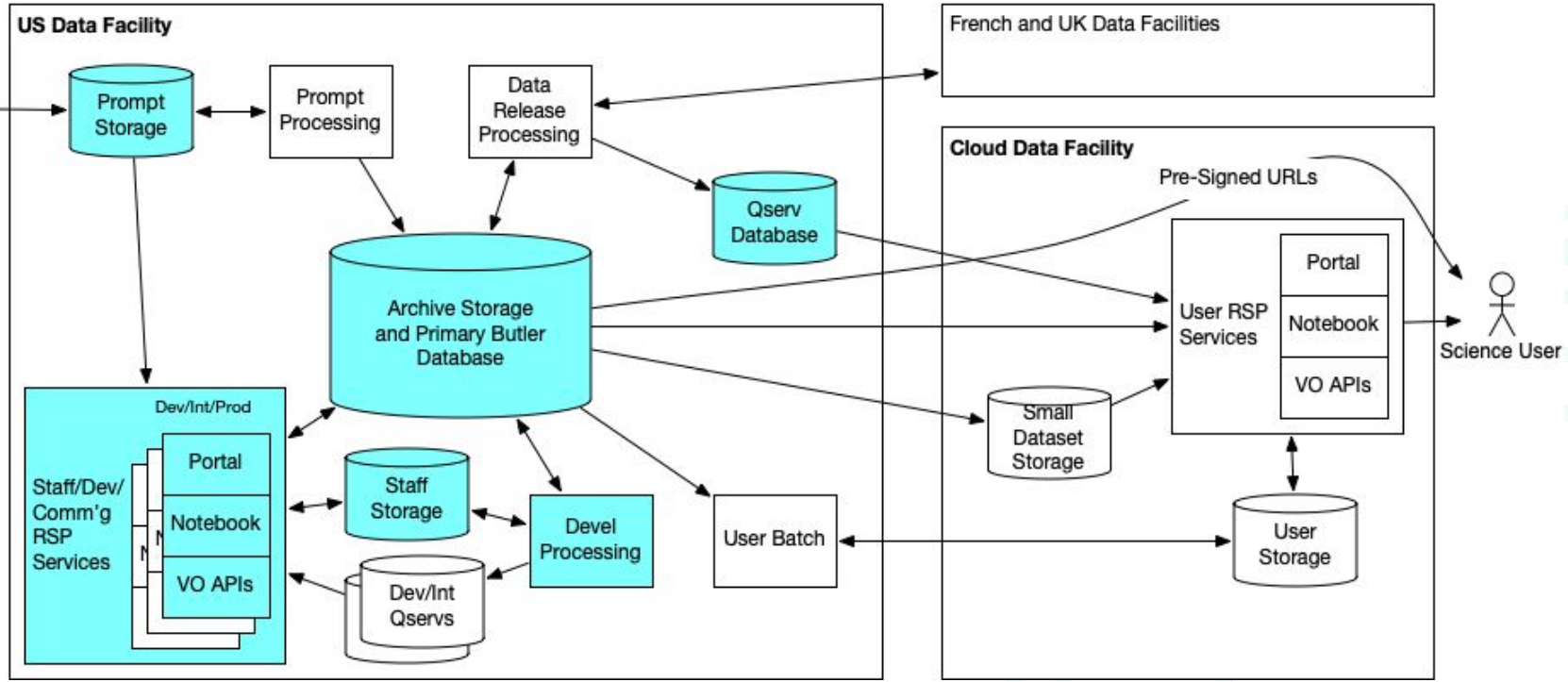
Permanent storage for raw data and released data products at SLAC National Accelerator Laboratory, served via a hybrid model through the US Data Access Center in Google Cloud.

Host for pre-release data and analysis for staff, including developers, Commissioning Team, and other in-kind contributors.

- All access authenticated and authorized through SLAC accounts.

USDF's host building will be shut down for a power upgrade from June 26 to July 3, 2023.

USDF Components for SIT-Com



Data Security

- Embargoed data (30 days during Commissioning, 80 hours during Operations) **must not** leave USDF.
  - Can view in browser but **not** save to a local machine or transport to an institutional resource.
  - Applies to all data in the `/repo/embargo` Butler repo.
- Post-embargo data must be kept private until released.
  - Can move to other machines but not share with non-staff.
  - All such data is in the `/repo/main` Butler repo.
  - Make sure someone doesn't take a picture of your screen and scoop our Communications team.
  - Still working out details of how daily and annual Data Releases will be identified, likely via collection.
- All data products are subject to the data rights policy (RDO-13)

Getting Help

- Developer Guide
- See if the problem is known: #ops-usdf-announce and #ops-usdf
- See if the problem is generic or USDF-specific
  - For example, if you are having `git push` problems, check githubstatus.com
  - Try things at data.lsst.cloud if you are a DP0 delegate or on a local machine
  - If the problem is generic use Slack support channels, e.g.:
    - Rubin Science Platform: #dm-rsp-support
    - Installing Science Pipelines packages: #dm-build-problems
    - Using the Middleware: #dm-middleware-support
    - Conda: #dm-conda
    - Shell/programming: #software-dev
- If the problem is only at the USDF: #ops-usdf
- For things that require SLAC system administrators, E-mail usdf-help@slac.stanford.edu

SLAC Accounts

- Everyone should have one; procedure is documented in the Dev Guide.
- Password change is required twice a year and can be done via web form.
- Forgotten or expired passwords: contact usdf-help@slac.stanford.edu.
- Check your account status at this page.

Interactive Usage

- ssh
  - Do not do significant work on `s3dflogin` nodes (many filesystems are unavailable anyway).
  - If you are using persistent servers (e.g. `tmux`, `VSCode`), choose an interactive node at random (currently sdfrome001 or sdfrome002).
- RSP
  - Home directory is shared with `ssh`, so changes on one side will affect the other.

Software

- Rubin Science Platform
  - https://usdf-rsp.slac.stanford.edu
- CernVM File System (official releases and weeklies)
  - `/cvmfs/sw.lsst.eu/linux-x86_64/`
- “Shared Stack” (all releases including nightlies)
  - `/sdf/group/rubin/sw`

Butler Repositories

- Repositories can be specified by a path to a configuration file, a path to the directory containing a configuration file named `butler.yaml`, or an alias (which, for historical reasons, often looks like a shortened path).
- Common aliases:
  - `/repo/main` (all non-embargo data pertaining to the real sky)
  - `/repo/embargo` (currently includes all Summit images after 2022-08-31 but will change soon)
  - `/repo/dc2` (Dark Energy Science Collaboration Data Challenge 2 simulated data)
  - `/repo/tts` (Tucson Test Stand)
  - `/repo/ir2` (SLAC Camera testing at Interaction Region 2)
  - `/repo/bts` (Base Test Stand, coming soon)
  - `LATISS`, `LSSTComCam`, `LSSTCam` point to `/repo/main` (unlike the Summit and test stands)
- `+sasquatch_dev`: Repositories with this alias have an in-development feature allowing certain data products to be output directly to the metrics database

Engineering and Facilities Database

The Sasquatch documentation gives URLs and labels for accessing the EFD.

The `lsst_efd_client` Python package is typically used (available in RSP and shared stack).

There is a separate talk about accessing the EFD visually via Chronograf.

Large File Annex

- URL translation: "`https://s3.cp.lsst.org/`" → "`s3://rubin:`"
- Set `LSST_DISABLE_BUCKET_VALIDATION=1` with `w_2023_16` or later.
- Use `lsst.resources` or `mc` or `boto`. Cannot use AWS CLI.
  - With `boto`, you need this code:

```
from botocore.handlers import validate_bucket_name
client.meta.events.unregister("before-parameter-build.s3", validate_bucket_name)
```

- Selected LFA datasets will be ingested into the Butler when available.

Batch Resources

Batch Production Service

- YAML defines large-scale workflow including pipeline to execute and execution parameters.
- `submit`, `report`, `cancel`, `restart` subcommands.
- Plugins for different workflow management systems:
  - HTCondor: create `~/.lsst/condor-info.py`, then run `allocateNodes.py s3df`
  - PanDA: intended for large-scale production, not ad hoc analysis
  - Parsl: available if desired

Slurm

- Underlying S3DF batch system
- Two clusters: milano (newer) and roma, 12K cores overall, 6K “owned” by Rubin
- Some useful commands: `sinfo`, `squeue`, `sstat`, `scancel`

Storage

Still some complexity with symlinks as things get organized; will be removed when backward compatibility is not an issue.

- `sdfhome` (Weka native)
  - `/sdf/home` (28GB quota)
  - `/sdf/group/rubin` (shared data only, 138TB)
    - `/sdf/group/rubin/shared/data` (automatically maintained clones of test and validation data)
    - `/sdf/group/rubin/shared` (other large data of common interest)
  - `/sdf/scratch` (132TB, purged)
- sdfdata (Weka NFS)
  - `/sdf/data/rubin` (5.4PB, 1TB quota): `repo`, `shared`, `user`
- Temporary space:
  - `/lscratch` (>300GB, shared among all jobs on the node)
  - `/tmp` (~20GB, shared among all jobs on the node)
- Lustre (/fs/ddn): going away

Service Developers

- All services should be deployed on Kubernetes.
  - USDF uses vcluster virtual cluster technology to give developers administrative control over Kubernetes while limiting the potential damage from mistakes.
  - `{clustername}.k8s.slac.stanford.edu` is used to get credentials.
- `vault.slac.stanford.edu` is used to store service secrets, including database and storage credentials. We aim to give all staff access to everything they need by default, but certain specialized use cases may require getting credentials from vault.

Monitoring

https://grafana.slac.stanford.edu

- Rubin > Campaign Management
- Rubin > USDF User Quota
- S3DF > Server Metrics (filtered to just sdfrome001 and sdfrome002)
- General > Weka Cluster Overview