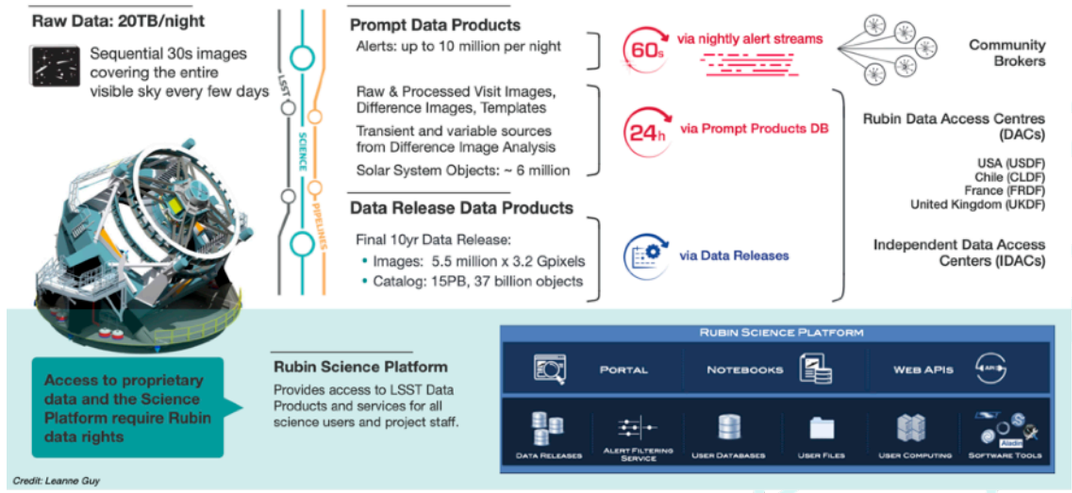

VERA C. RUBIN
OBSERVATORY
Rubin Observatory
Cloud Experience
Kian-Tat Lim, DM Software Architect
NSF
AURA
U.S. DEPARTMENT OF
ENERGY
SLAC
CHARLES AND LISA SIMONYI FUND
LSST
CORPORATION

# Most Clouds Are Bad for Astronomers

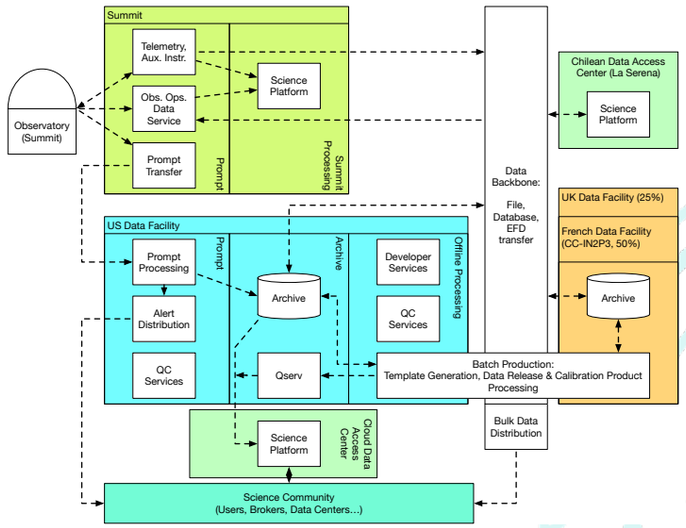
# Legacy Survey of Space and Time






Rubin data processing has three major use cases
• Alert Production processes images from the telescope within 60 seconds
• Data Release Production reprocesses all images taken to date each year
• Science users on the Rubin Science Platform do analysis and reprocessing
Also developers need to do both ad hoc and production-style processing

# Simplified Architecture




View of the Data Management system with telescope as an input device
Far left: near-realtime Prompt Processing
Right: High Throughput Computing across multiple sites
Bottom and upper right: Dedicated resources for science user access and analysis

# Proof-of-Concept Engagements

- Three engagements with two providers
- DMTN-078/DMTN-125
  - Leveraged to get cloud-native experience and improve deployment models
  - Learned about potential bottlenecks in high bandwidth-delay product networks
  - Learned about interacting with vendors
- DMTN-114/DMTN-137
  - Tested workflow execution middleware at modest scale (up to 1200 vCPU)
  - Made use of "spot" or preemptible instances
- DMTN-150/DMTN-157
  - Tested improved workflow execution middleware at similar scale (up to 1600 vCPU)
  - Learned about desirability of HTTP/2 persistent connections for long-haul network

# Working with Vendors

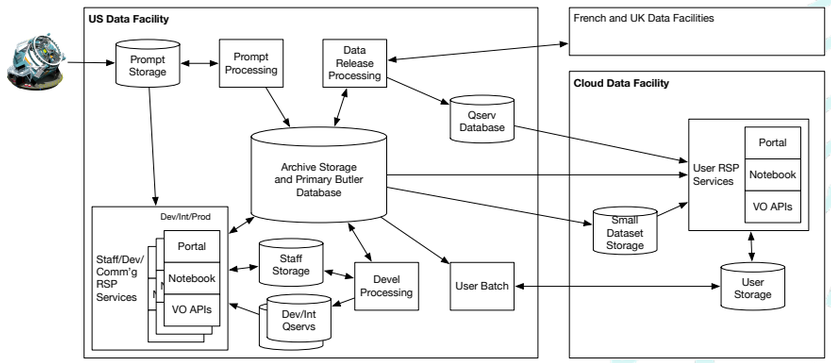
- Low bureaucracy, high flexibility, willingness to assist are highly desirable
- Deep engagement with the vendor's engineering teams helps a lot
- Consultants can be very useful; unlike vendor engineers, they can often work independently on customer code to resolve problems

# Cost Modeling

- Complex spreadsheets from us outlining needs and from vendor matching to technologies and prices
- Compute costs (4.1E8 core-hours in year 10) are reasonable
- Storage costs for frequently-accessed data (100s of PB of both in-process and released results) are large
  - Paying for more durability and performance than needed
- Egress costs can be a problem as well, but mitigations exist
  - Keep most data inbound to the cloud or inside the cloud
  - Only have summarized data or visualizations exit
  - Credits based on total spend
  - Can consider dedicated interconnect to get discounts

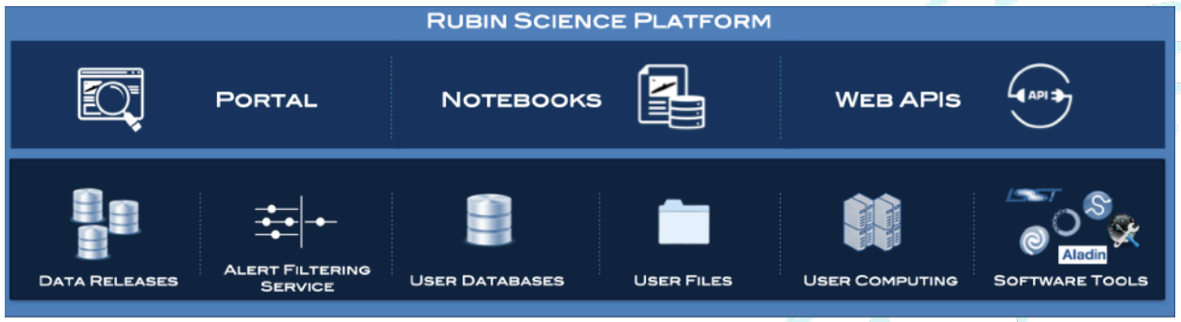
# Decision: Hybrid Model

- Most storage and large-scale compute is on-prem
  - While cloud compute is not expensive, moving the data is not advisable
- Users are supported in the cloud



# Rubin Science Platform

- Users doing science on large datasets using web-based applications:
  - Portal (access and visualization)
  - Notebooks (analysis)
  - APIs (remote access and processing)



# Uses of Cloud Services (1)

- Rubin Science Platform
  - Security (separately managed identities, limited interfaces with on-prem facilities)
  - Elasticity (especially immediately after an annual Data Release)
  - Bring-your-own (combining project resources with externally-granted resources)
  - Technology (GPU/TPU, easy production-quality deployments of sophisticated infrastructure services)
- Compute
  - Executed DP0.2, demonstrated scaling to larger numbers of nodes (4000 vCPU)
  - Not expected to execute main survey Data Release Production
  - Cost of storing or egressing large data products is excessive
  - User batch could be done in the cloud, but it could have the same drawbacks as DRP
  - But it would have the advantages of the RSP as well

# Uses of Cloud Services (2)

- Development testing
  - Elasticity
  - Technology
  - Rapid prototyping with advanced services (serverless)
- Possible future for large-scale database
  - Custom database still has advantages
  - Spherical geometry is becoming less of a differentiator
  - Shared scan is still a win, but with NVMe IOPS, also less
  - Special/spatial indexes
  - Storage costs could still be an issue
- Archival (tape-replacement) storage
  - TCO if not retrieved may be comparable

# Cloud Reliability

- Kubernetes upgrades roll through semi-arbitrarily (some scheduling controls)
  - Should design services to deal with rolling outages
- Durability of storage is extremely high (maybe more than necessary)
- Service outages are rare and usually short, compared with on-prem
- 24x7 support for basic infrastructure and even for higher-level services may be better than on-prem

# Conclusions, Status, Plans

- Hybrid model is suitable for Rubin use cases
- Practicing today with Interim Data Facility on Google Cloud Platform hosting simulated data
- Building out back-end on-prem infrastructure to practice integration and tune caching parameters
- Continuing to track developments in Cloud services and pricing