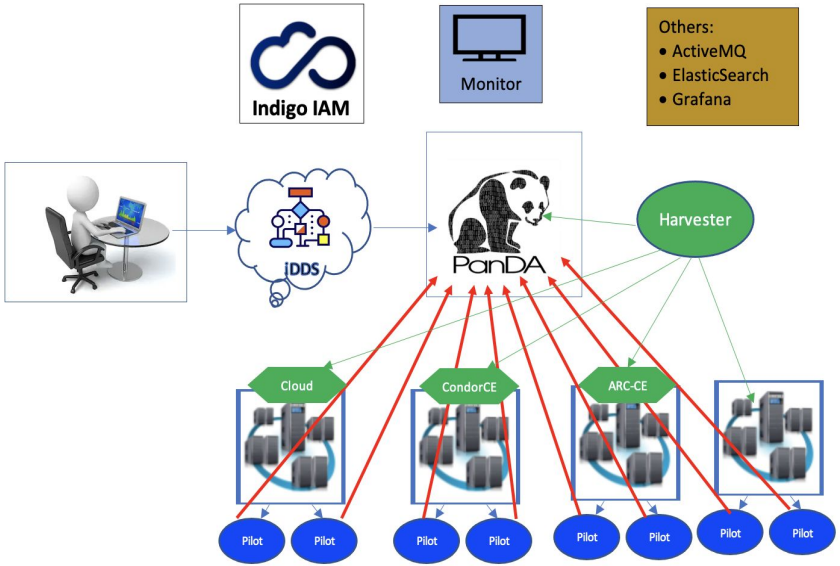
PanDA for Rubin update

Wen Guan, Edward Karavakis, Tadashi Maeno, Torre Wenaus, Zhaoyu Yang

on behalf of the PanDA team and Rubin DM team

BNL Cosmology Group Meeting
Feb 1st, 2023

PANDA/iDDS



- PanDA and Pilots (red lines) work together as a workload management system to integrate distributed computing resources
  - Pilot works as an agent to acquire the CPUs, validate the environments and pull jobs from PanDA. It hides differences of heterogeneous computing resources.
  - Pilot starts user jobs, monitors user jobs and heartbeats to PanDA server.
- Harvester provisions pilots on remote/local resources through various plugins based on their access methods.
- iDDS manages the workflows and DAG dependencies.
- Indigo IAM is employed for authentication.
- Monitors:
  - PanDA monitor: task view and job view
  - Additional monitors such as ElasticSearch and Grafana are integrated
- ActiveMQ:
  - Messaging service between PanDA components

The PanDA Rubin Team

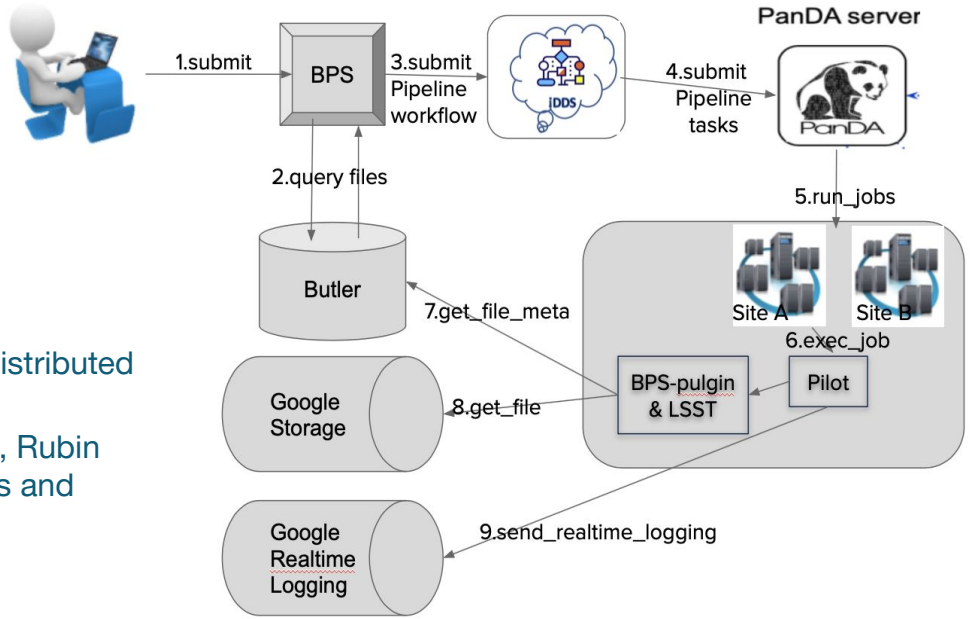
- Three new hires started almost the same time in mid 2022
  - Wen Guan: based at CERN, lead iDDS developer
  - Edward Karavakis: based at CERN
  - Zhaoyu Yang: at BNL
- Supporting team (from PanDA core team):
  - Tadashi Maeno: PanDA & iDDS projects lead (BNL)
  - Paul Nilsson: pilot developer (BNL)
  - Fernando Barreiro Megino (UT Arlington)
  - Fa-Hui Lin (UT Arlington)
  - Tatiana Korchuganova (Pittsburgh)
  - Alexei Klimentov (BNL): manages all these people as ATLAS workflow systems manager
  - Torre Wenaus: US ATLAS HL-LHC Computing Co-Manager, NPPS Group Leader
- Adapted the PanDA core team's operations to its more distributed nature
  - Core team meeting shifted to US compatible time, more use of chat tools for dev communication
- Special thanks to Shuwei Ye. He has worked with Wen since Sergey Padolski's departure, guaranteed the successful processing of DP0.2 campaign on the Rubin Google Interim Data Facility.
- Sergey Padolski: Rubin PanDA pioneer and trailblazer, laying foundations of the integration.

Progress since last update [Torre's Feb 2022 talk]

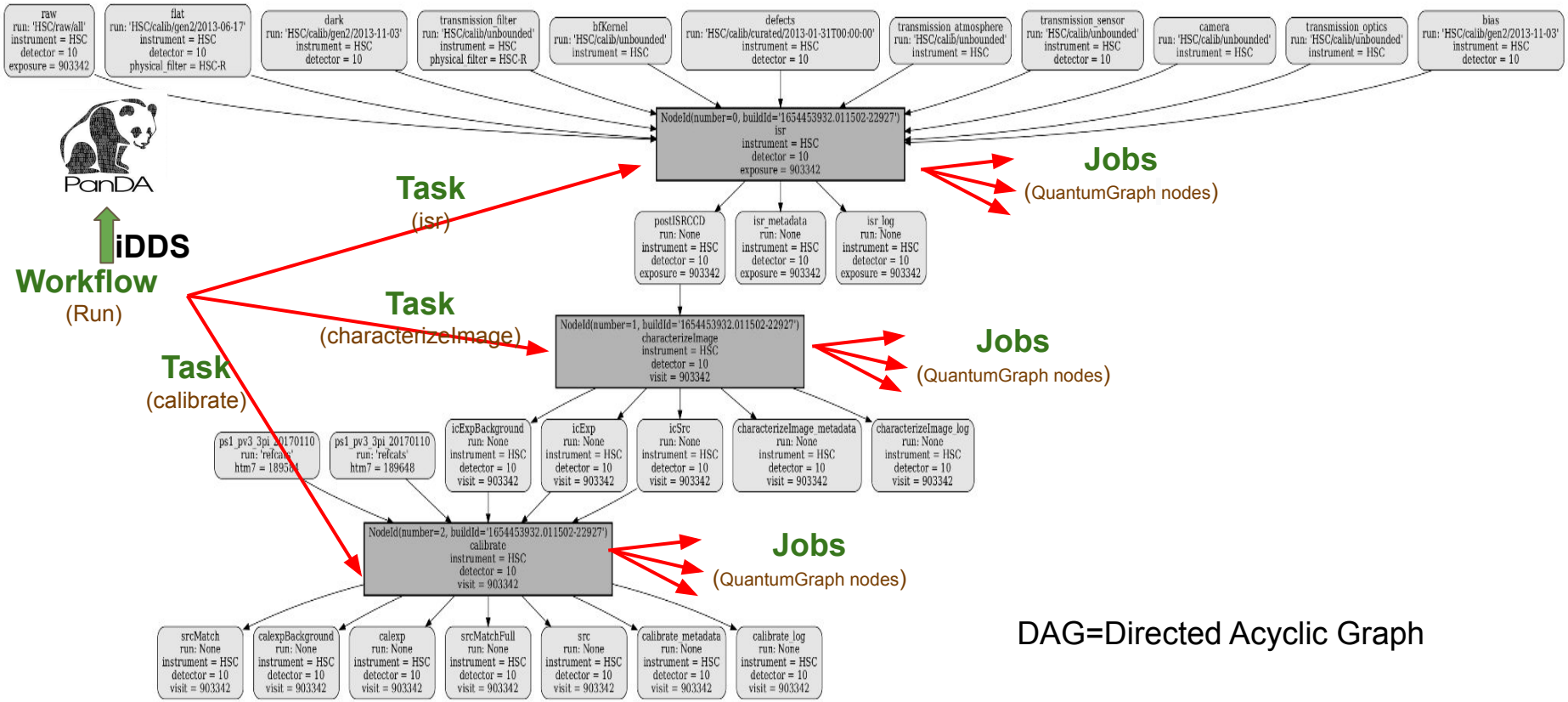
- **Production campaigns using PanDA**:
  - Successful DP0.2 processing in the first half of 2022
  - Monthly HSC reprocessing is ongoing since last July
  - PDR2 processing planned for 2023
- **Improved Rubin + PanDA integration**
  - PanDA/iDDS improvements for Rubin's large scale DAG workflows
  - Rubin production software stack integration with PanDA
  - Scaling to multiple sites: UK, France in addition to SLAC, Google
- **Kubernetes based deployment at SLAC now in place**
  - Containerization of PanDA components and helm based deployment
  - Performing scaling tests and preparing for pre-production
  - Will soon take over from CERN PanDA instance
- **Improved user experience**
  - Server performance, failure processing, latency reduction
  - Improved UI (command line + monitor)
  - Further tailoring the PanDA monitor for Rubin
  - Grafana based performance/health monitoring in progress
- **Campaign Management**
  - Working on higher level interface to manage production campaigns

Rubin + PanDA integration

- **LSST Science Pipelines (stack)**
  - Butler + pipeline framework
- **Butler**: Data access
  - Interface between data and pipeline tasks
- **BPS**: Batch Processing Service
  - Interface between Butler and PanDA
  - Integrate Rubin with PanDA/iDDS client
- **PanDA**: Workload management system
  - Manage and schedule Rubin workload to distributed computing resources
  - PanDA pilot integrates Rubin Butler access, Rubin workload execution, Google storage access and real-time logging
- **Google Cloud:**
  - Pilot logs storage and real-time logging
  - GKE clusters (for the Interim Data Facility)



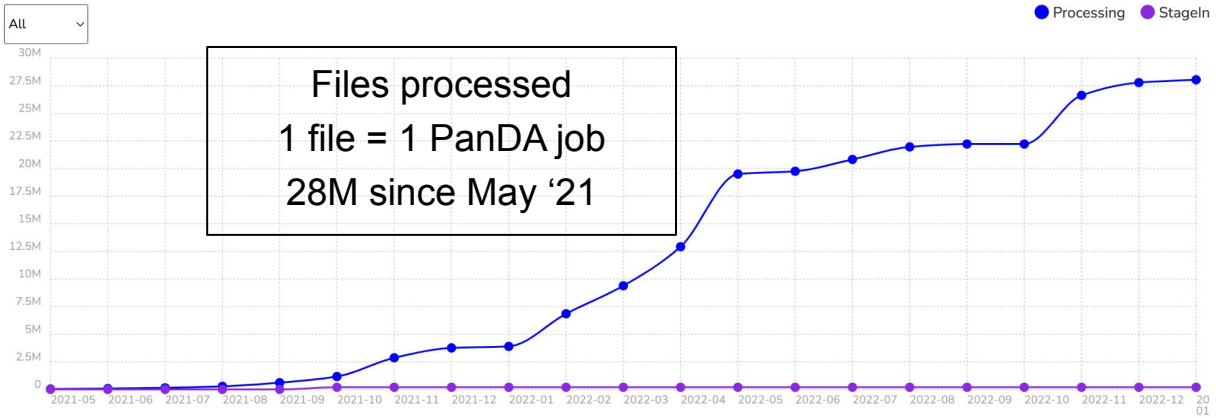
Mapping Rubin DAG to PanDA workload



DAG=Directed Acyclic Graph

DRP: Data Release Production

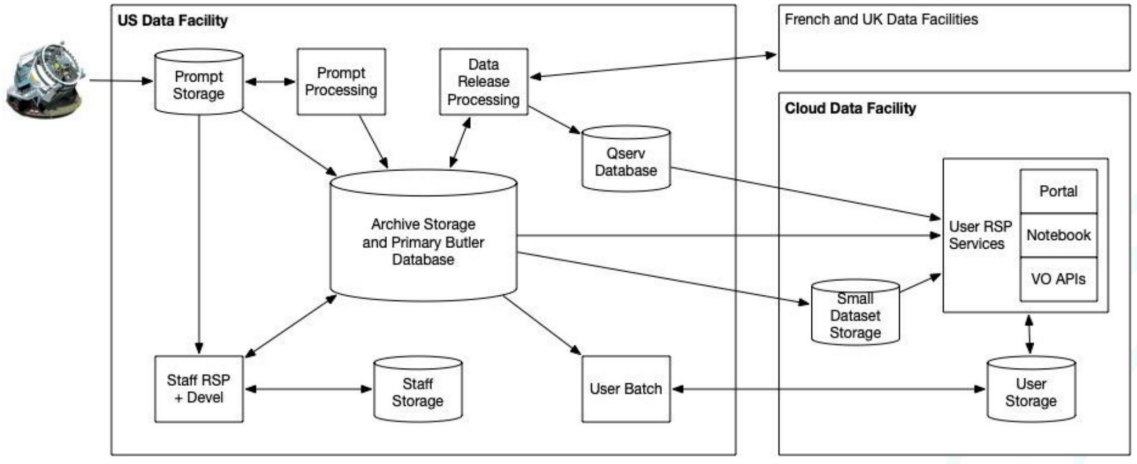
- **2022** production campaigns used PanDA
  - **DP0.2**: **16M** jobs@IDF
  - **HSC** reprocessing: 8M jobs@USDF



- With successful processing of DP0.2, PanDA was endorsed for DRP processing
- **2023** DRP is estimated to have ~**36M** jobs for PDR2, ~**8M** for HSC reprocessing

Rubin data facilities

- There are 3 main data facilities (USDF, FrDF, UKDF) and 1 cloud-based IDF (Google)
  - **USDF**: in S3DF at SLAC. All prompt processing, **25%** of data release processing, and data access services to the US and international community
  - **FrDF**: **50%** of data release processing, back up of raw data and published products
  - **UKDF**: **25%** of data release processing
  - **IDF**: Cloud-based Interim Data Facility, used for pre-ops activities



IDF: Interim Data Facility

- Configuration took into account the computing cost on Google Cloud
  - **6 PanDA queues** on the GKE clusters for different memory requests
  - **1 merge queue** to limit the database connections
  - The clusters can automatically scale their size according to the workloads.
- **DP0.2** processing from **Dec 2021-May 2022**
  - LSST DESC DC2 simulated sky survey
  - 20K visits, 150 images/visit, represents ~10 nights of data gathering
  - The production was grouped into 7 steps, ~**16M** PanDA jobs
  - Most jobs were processed on a cluster with ~**4000 cores,** up to **14GB/core** RAM
  - Total CPU usage: **2.5M core-hours**
  - Storage in data store: **2.55 PiB** after removal of intermediate data products

queue	maxMem(GB)	used by
DOMA_LSST_GOOGLE_TEST	14	default
DOMA_LSST_GOOGLE_MERGE	14	butler merge
DOMA_LSST_GOOGLE_HIMEM	40	
DOMA_LSST_GOOGLE_HIMEM_NON_PREEMPT	40	
DOMA_LSST_GOOGLE_EXTRA_HIMEM	236	
DOMA_LSST_GOOGLE_EXTRA_HIMEM_NON_PREEMPT	236	

US Data Facility (USDF)

- **Hybrid** model: Data at SLAC but users on the Cloud
- Science users: access via **Rubin Science Platform** (Jupyter based)
- **S3DF**: SLAC Shared Science Data Facility
  - Infrastructure launched last August
  - Most services deployed via kubernetes
  - Slurm batch system
  - Has been used for monthly HSC reprocessing. ~**8M** PanDA jobs have been processed

PanDA Queue	slurm queue	minRSS	maxRSS	Harvester mode	Brokerage
SLAC_Rubin	rubin	0GB	4GB	pull	on
SLAC_Rubin_Medium	rubin	4GB	8GB	pull	on
SLAC_Rubin_Himem	rubin_himem	8GB	18GB	pull	on
SLAC_Rubin_Extra_Himem	rubin_extra_himem	18GB	220GB	push	on
SLAC_Rubin_merge	rubin_merge	0GB	500GB	push	off
SLAC_Test	rubin	0GB	4GB	pull	off

Brookhaven National Laboratory

Multi-DF processing

- PanDA is for distributed workload management. However for the multi-DF processing of Rubin workflows, the main constraints are from the Butler access.
  - The quantum graph and execution Butler created at one DF are not portable to another DF.
  - After the processing of all pipeline tasks, one needs to merge the outputs and metadata back to the main Butler registry. The current Butler does not support this remotely either.
- The support for multi-DF processing needs developments both in DM middleware and PanDA+iDDS
- The existing PanDA command line tool **prun** is a temporary workaround to submit pipeline workflows to remote sites

Hello World job at FrDF (submitted remotely)

Job name: Hello World FrDF.31935512

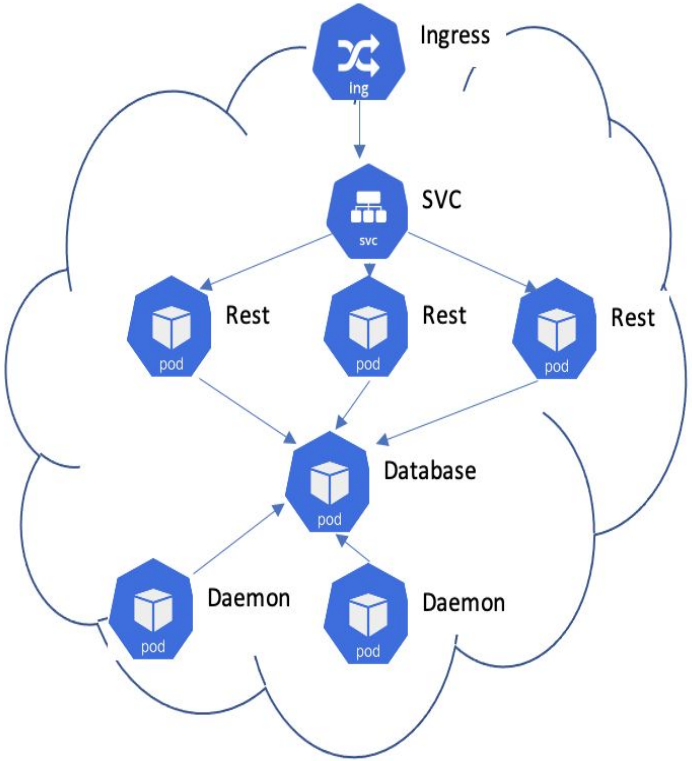
PanDA ID	Owner / VO	Request Task ID	Status	Transformation	Created Last modified	Time to start Duration [d:h:m:s]	Cloud Site	Cores	Priority	Attempt
31935512	Zhaoyu Yang / wlcg	140098 140098	finished	bash-c	2022-12-12 14:34:53 2022-12-12 14:38:26	0:0:02:30 0:0:00:05	EU CC-IN2P3_TEST	1	1000	1

Rubin pipeline jobs at UKDF (submitted remotely)

PanDA ID Attempt# of maxAttempts#	Owner / VO Group	Request Task ID	Transformation	Status	Created	Time to start d:h:m:s	Duration d:h:m:s	Mod	Site	Priority	N input events (N input files)	Max PSS/core, GB	Job info
33809393 Attempt 1 of 1	iddssv1 / wlcg	3199 144325	bash-c-enc	finished	2023-01-28 21:50:01	0:1:04:23	0:0:01:41	2023-01-28 22:56:23	LANCS_TEST brokeroff Set brokeroff for one year	1000	0 (0)	0.37	
	Job name: u_lsst001_UKDF_w52_remote_20230128T213356Z_isr_3199_25518.33809393 #1												
	Datasets: Out: PandaJob_#[pandaid]/												
33809392 Attempt 1 of 1	iddssv1 / wlcg	3199 144325	bash-c-enc	finished	2023-01-28 21:50:01	0:1:04:23	0:0:01:35	2023-01-28 22:56:23	LANCS_TEST brokeroff Set brokeroff for one year	1000	0 (0)	0.26	
	Job name: u_lsst001_UKDF_w52_remote_20230128T213356Z_isr_3199_25518.33809392 #1												
	Datasets: Out: PandaJob_#[pandaid]/												
33809391 Attempt 1 of 1	iddssv1 / wlcg	3199 144325	bash-c-enc	finished	2023-01-28 21:50:01	0:1:04:08	0:0:01:53	2023-01-28 22:56:23	LANCS_TEST brokeroff Set brokeroff for one year	1000	0 (0)	0.25	
	Job name: u_lsst001_UKDF_w52_remote_20230128T213356Z_isr_3199_25518.33809391 #1												
	Datasets: Out: PandaJob_#[pandaid]/												
33809390 Attempt 1 of 1	iddssv1 / wlcg	3199 144325	bash-c-enc	finished	2023-01-28 21:50:01	0:1:04:02	0:0:02:03	2023-01-28 22:56:23	LANCS_TEST brokeroff Set brokeroff for one year	1000	0 (0)	0.32	
	Job name: u_lsst001_UKDF_w52_remote_20230128T213356Z_isr_3199_25518.33809390 #1												
	Datasets: Out: PandaJob_#[pandaid]/												
33809389 Attempt 1 of 1	iddssv1 / wlcg	3199 144325	bash-c-enc	finished	2023-01-28 21:50:01	0:0:57:14	0:0:02:38	2023-01-28 22:50:33	LANCS_TEST brokeroff Set brokeroff for one year	1000	0 (0)	0.38	
	Job name: u_lsst001_UKDF_w52_remote_20230128T213356Z_isr_3199_25518.33809389 #1												
	Datasets: Out: PandaJob_#[pandaid]/												
33809388 Attempt 1 of 1	iddssv1 / wlcg	3199 144325	bash-c-enc	finished	2023-01-28 21:50:01	0:0:54:57	0:0:01:48	2023-01-28 22:46:54	LANCS_TEST brokeroff Set brokeroff for one year	1000	0 (0)	0.38	
	Job name: u_lsst001_UKDF_w52_remote_20230128T213356Z_isr_3199_25518.33809388 #1												

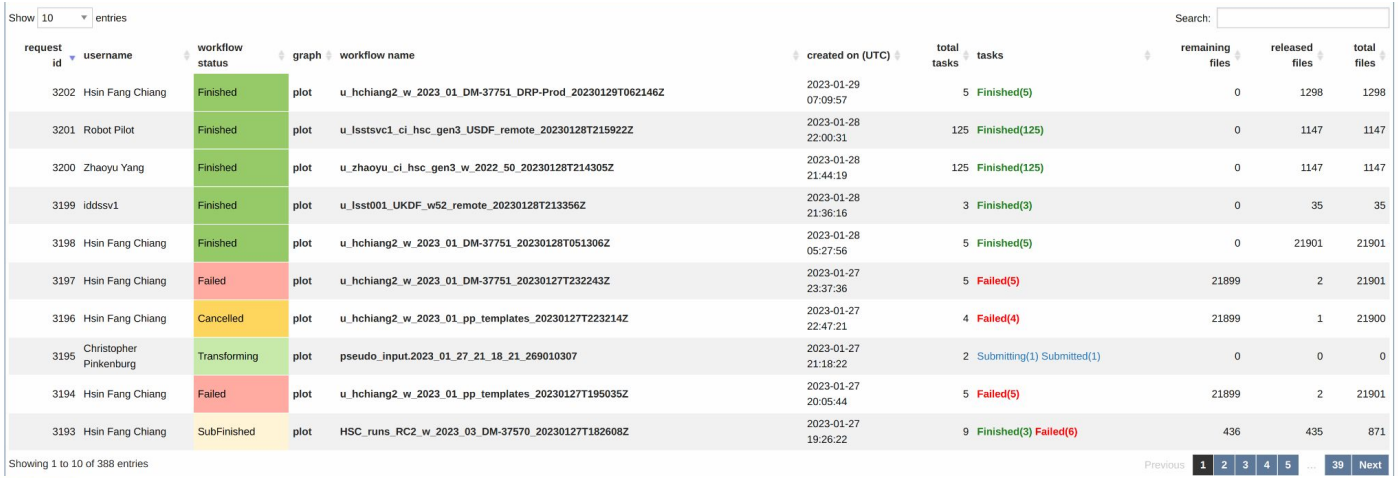
PanDA deployment at SLAC K8s

- Main components:
  - PanDA Server and JEDI, Indigo IAM authentication, Harvester, iDDS, PanDA monitor, ActiveMQ
- PostgreSQL
  - Based on a test PostgreSQL database
  - In process to get a production-ready DB with CNPG (CloudNativePostgreSQL)
- Long standing issues with network in/out access at SLAC
  - No outbound access to FrDF and UKDF
  - Will a single ingress balancer scale with production load?
- PanDA monitor (with DEBUG on) available
  - https://rubin-panda-bigmon-dev.slac.stanford.edu:8443 with IAM to support login with institute's credentials



PanDA monitor development

- The DOMA instance of the PanDA monitor was developed for Rubin job monitoring
  - DOMA is a CERN/LHC R&D project that offers a playground for non-ATLAS experiments to try PanDA, iDDS
- Many features have been added for the Rubin workflow monitoring
  - Hierarchical navigation at different levels: workflow->tasks->jobs->logs
  - The job view shows the payload errors (if any) reported by pilot
  - Memory usage monitoring using prmon (open source tool originally from ATLAS)
  - Display consistent task status (tasks/jobs have dependencies)
  - Display the tasks by workflow ID on the monitor
- The same monitor is used by all non-ATLAS experiments, e.g. sPHENIX



Show 10 entries — Search:

request id	username	workflow status	graph	workflow name	created on (UTC)	total tasks	tasks	remaining files	released files	total files
3202	Hsin Fang Chiang	Finished	plot	u_hchiang2_w_2023_01_DM-37751_DRP-Prod_20230129T062146Z	2023-01-29 07:09:57	5	Finished(5)	0	1298	1298
3201	Robot Pilot	Finished	plot	u_lsstsvc1_ci_hsc_gen3_USDF_remote_20230128T215922Z	2023-01-28 22:00:31	125	Finished(125)	0	1147	1147
3200	Zhaoyu Yang	Finished	plot	u_zhaoyu_ci_hsc_gen3_w_2022_50_20230128T214305Z	2023-01-28 21:44:19	125	Finished(125)	0	1147	1147
3199	iddssv1	Finished	plot	u_lsst001_UKDF_w52_remote_20230128T213356Z	2023-01-28 21:36:16	3	Finished(3)	0	35	35
3198	Hsin Fang Chiang	Finished	plot	u_hchiang2_w_2023_01_DM-37751_20230128T051306Z	2023-01-28 05:27:56	5	Finished(5)	0	21901	21901
3197	Hsin Fang Chiang	Failed	plot	u_hchiang2_w_2023_01_DM-37751_20230127T232243Z	2023-01-27 23:37:36	5	Failed(5)	21899	2	21901
3196	Hsin Fang Chiang	Cancelled	plot	u_hchiang2_w_2023_01_pp_templates_20230127T223214Z	2023-01-27 22:47:21	4	Failed(4)	21899	1	21900
3195	Christopher Pinkenburg	Transforming	plot	pseudo_input.2023_01_27_21_18_21_269010307	2023-01-27 21:18:22	2	Submitting(1) Submitted(1)	0	0	0
3194	Hsin Fang Chiang	Failed	plot	u_hchiang2_w_2023_01_pp_templates_20230127T195035Z	2023-01-27 20:05:44	5	Failed(5)	21899	2	21901
3193	Hsin Fang Chiang	SubFinished	plot	HSC_runs_RC2_w_2023_03_DM-37570_20230127T182608Z	2023-01-27 19:26:22	9	Finished(3) Failed(6)	436	435	871

Showing 1 to 10 of 388 entries — Previous 1 2 3 4 5 … 39 Next

Failed jobs view

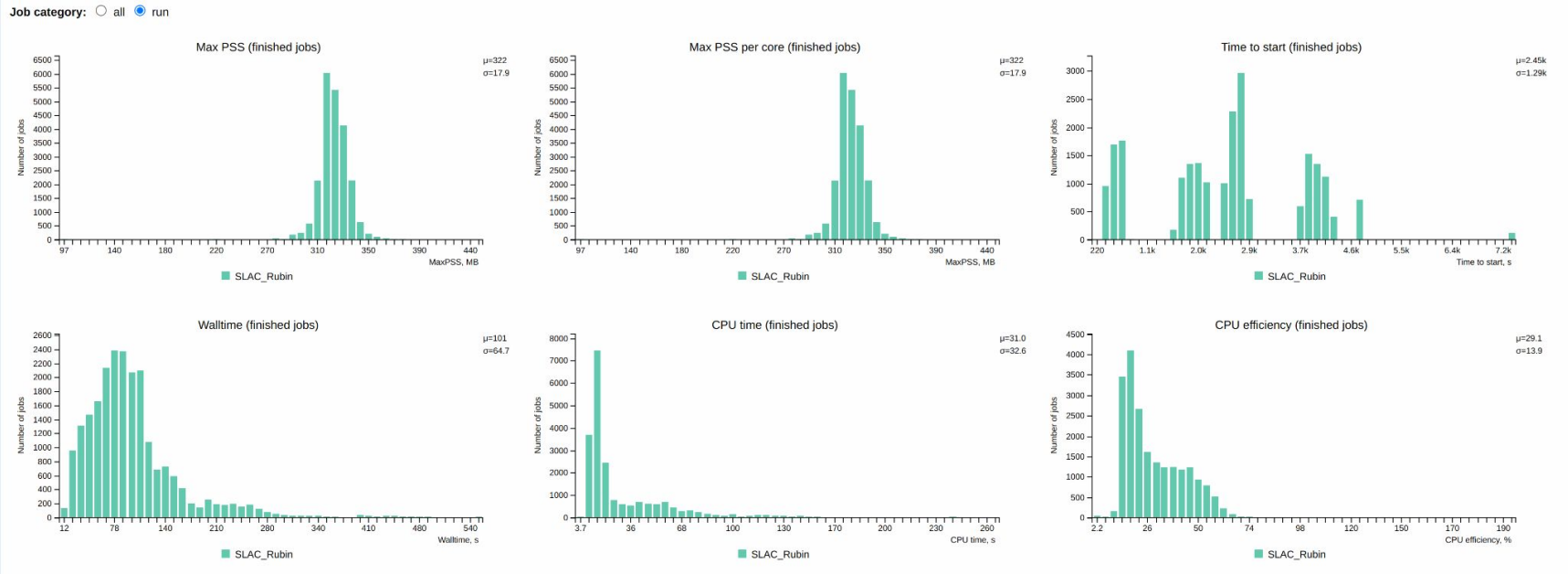
- After bug fix in pilot, the monitor shows the correct errors for the failed jobs
- The Campaign Management tools collect the errors for decision-making

Major error codes reported by pilot

PanDA ID Attempt# of maxAttempts#	Owner / VO Group	Request Task ID	Transformation	Status	Created	Time to start d:h:m:s	Duration d:h:m:s	Mod	Site	Priority	N input events (N input files)	Max PSS/core, GB	Job info
33539239 Attempt 3 of 3	Orion Eiger / wlcg	3123 143067	bash-c-enc	failed	2023-01-18 22:03:27	0:0:00:09	0:0:03:45	2023-01-18 22:07:23	SLAC_Rubin online no active blacklisting rules defined	500	0 (0)	1.15	**pilot, 1305:** Failed to execute payload:Unable to measure aperture correction for required algorithm 'modelfit_CModel': only 0 sources, but require at least 2. **trans, 1:** Unspecified error, consult log file
	Job name: u_eiger_cm_HSC_test_w2248_step1_group4_w01_001_characterizeImage_3123_24363.33539056 #3												
	Datasets: **Out:** PandaJob_#{pandaid}/												
33539238 Attempt 3 of 3	Orion Eiger / wlcg	3123 143067	bash-c-enc	failed	2023-01-18 22:02:29	0:0:00:14	0:0:03:57	2023-01-18 22:07:12	SLAC_Rubin online no active blacklisting rules defined	500	0 (0)	1.01	**pilot, 1305:** Failed to execute payload:Unable to measure aperture correction for required algorithm 'modelfit_CModel': only 1 sources, but require at least 2. **trans, 1:** Unspecified error, consult log file
	Job name: u_eiger_cm_HSC_test_w2248_step1_group4_w01_001_characterizeImage_3123_24363.33539055 #3												
	Datasets: **Out:** PandaJob_#{pandaid}/												
33539237 Attempt 3 of 3	Orion Eiger / wlcg	3123 143067	bash-c-enc	failed	2023-01-18 22:01:59	0:0:00:44	0:0:03:29	2023-01-18 22:06:21	SLAC_Rubin online no active blacklisting rules defined	500	0 (0)	1.02	**pilot, 1305:** Failed to execute payload:Unable to measure aperture correction for required algorithm 'modelfit_CModel': only 0 sources, but require at least 2. **trans, 1:** Unspecified error, consult log file
	Job name: u_eiger_cm_HSC_test_w2248_step1_group4_w01_001_characterizeImage_3123_24363.33539057 #3												
	Datasets: **Out:** PandaJob_#{pandaid}/												

Memory Usage Monitoring

- Memory monitoring is important for debugging job failures
  - Most quanta(jobs) in a group use similar amount of memory, but a small percentage of quanta can keep failing at different points, indicating higher memory is needed.
- Memory plots have been added in the PanDA monitor for Rubin



Workflow monitor with command line

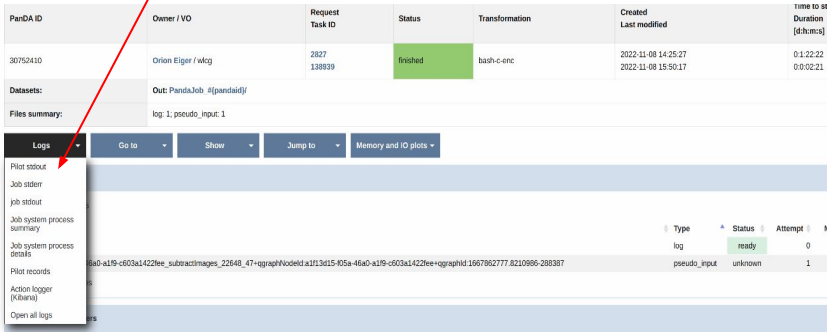
- To query job status fast without visiting the UI, a command line tool was developed in the BPS-PanDA plugin
- The CLI was only added recently. More functionality can be added on request.

```
 X    STATE    %S  ID    OPERATOR  PROJECT CAMPAIGN PAYLOAD                    RUN
--- --------- --- ---- ---------- ------- -------- ------- ----------------------------------------
    SUCCEEDED  92 3123 Orion Eiger                         u_eiger_cm_HSC_test_w2248_step1_group4_w01_001


Path: None
Global job id: None


                       UNKNOWN MISFIT UNREADY READY PENDING RUNNING DELETED HELD SUCCEEDED FAILED PRUNED EXPECTED
---------------------- ------- ------ ------- ----- ------- ------- ------- ---- --------- ------ ------ --------
TOTAL                        0      0       0     0       0       0       0    0       517     13     29      559
---------------------- ------- ------ ------- ----- ------- ------- ------- ---- --------- ------ ------ --------
pipetaskInit                 0      0       0     0       0       0       0    0         1      0      0        1
isr                          0      0       0     0       0       0       0    0       103      0      0      103
characterizeImage            0      0       0     0       0       0       0    0       103      3      0      106
calibrate                    0      0       0     0       0       0       0    0       103     10      3      116
writePreSourceTable          0      0       0     0       0       0       0    0       103      0     13      116
transformPreSourceTable      0      0       0     0       0       0       0    0       103      0     13      116
mergeExecutionButler         0      0       0     0       0       0       0    0         1      0      0        1
```

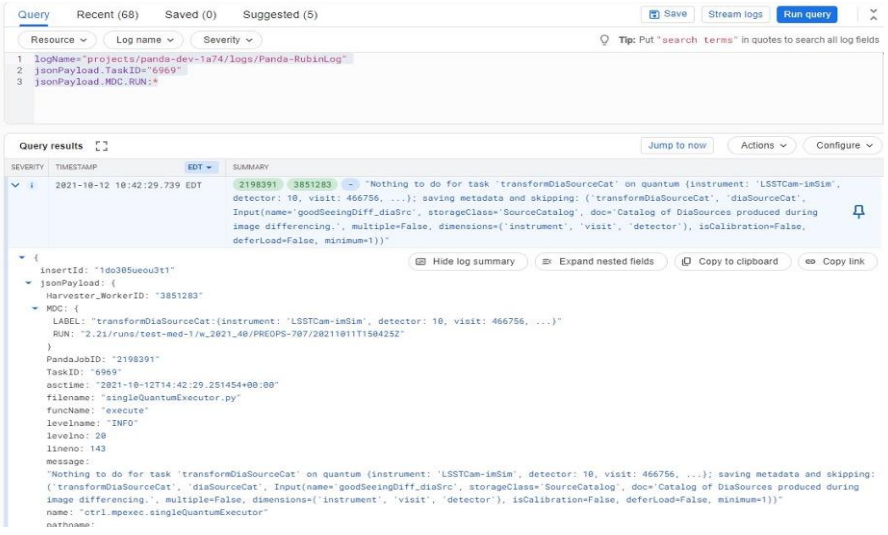
Real-time logging

- Conventional log access:
  - At the end of a job execution, pilot uploads the logs including the full pilot log, payload stdout and payload stderr dump to the Google cloud (GCS) bucket
- New (**near) real-time log access:**
  - In Rubin, pilot captures the payload log and sends as json to **Google Cloud Logging**
  - In addition to the payload logs, recent development allows pilot to send its own logs to Google Cloud Logging
- The real-time logs provide complementary information for monitoring and debugging.
- Strong interest from ATLAS. It has been refactored to be **experiment agnostic**

pilot logs uploaded to GCS
(N/A if job is killed)

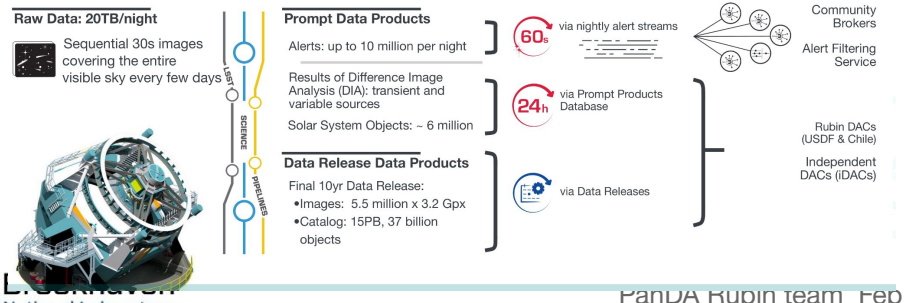


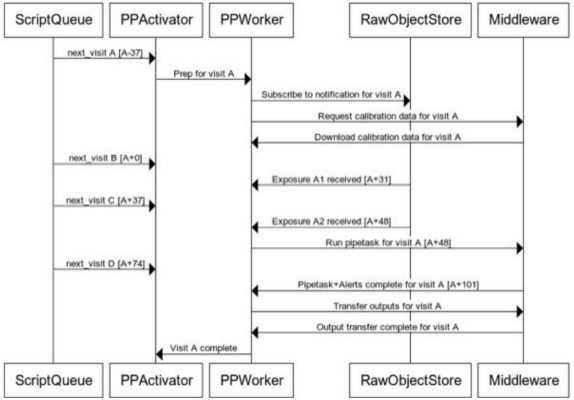
Google Cloud Logging



Prompt processing

- Prompt processing in Rubin:
  - To be able to initiate processing in a few seconds
  - On dedicated resources at SLAC
  - Reuse of WN for each visit to skip downloading calibration data in the processing
- Developments for rapid workload provisioning and processing
  - Semi-persistent pilot up and running on WN
  - Task resurrection via notification to skip overhead before
  - Generating jobs
  - Job pushed to the pilot via ActiveMQ
  - Direct communication channel between JEDI and PanDA server
- The mechanism is ready for Rubin to try
- The developments are also useful to minimize latencies and support pseudo-interactive analysis in ATLAS



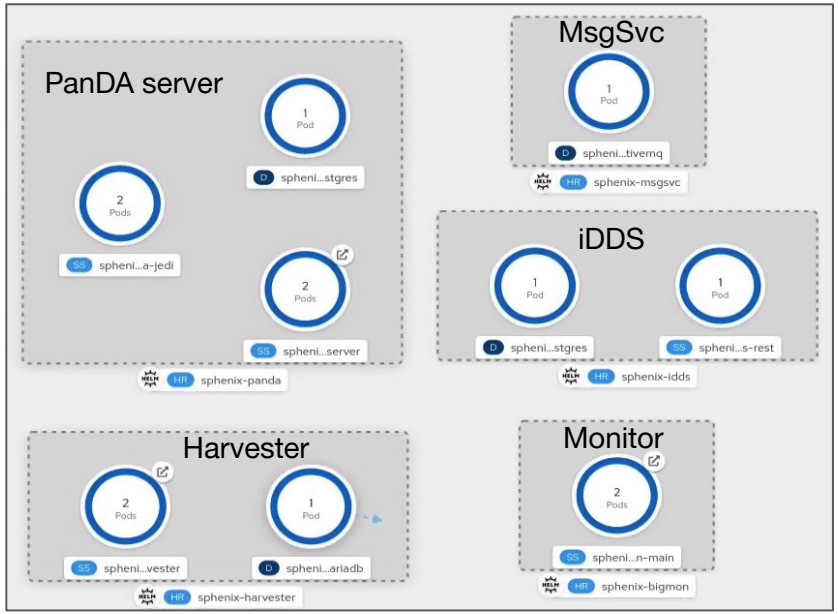


New developments

- Clustering
  - 1 QuantumGraph node is normally a very short job, typically a couple of minutes
  - Working on bulk processing jobs in clusters to improve the efficiency
- Integration with Campaign Management (CM)
  - Rubin CM centrally manages the production processing of Rubin data
  - Adding new Agents to send task/job detail information messages to CM
  - Working to provide simplified http-based APIs for CM to retrieve task/job detail information (as a backup of messages)
- Computing resource description - avoiding the CRIC dependency
  - CRIC (Computing Resource Information Catalog) is used by ATLAS PanDA to manage its hundreds of computing resources
    - Overkill for Rubin with a small number of sites
  - PanDA now supports simple json-file based computing resource description
- Improvements on iDDS
  - Message-driven structure to improve the efficiency of agents
  - Make use of database triggers
  - Redis to cache information

NPPS synergies with Rubin work
(beyond the obvious ATLAS one)

- ATLAS Google project now entering its second phase
  - See supplementary slide
- sPHENIX adopted PanDA and chose the same new approach to PanDA service deployment
  - PostgreSQL database (instead of Oracle)
  - Kubernetes (or open source variant OKD) based deployment of services (instead of VMs)
  - Efficient synergy between the efforts
  - PanDA installed in SDCC and under test
    - Rucio also, integration in progress
  - PanDA/iDDS/Rucio based sPHENIX production software in development by NPPS
- David's DESC work (you know all about it)
  - Its basis in parsl resonates with several NPPS activities
  - parsl used by Rubin, DUNE
    - exploring PanDA parsl integration for Rubin
  - Hope to start soon on parsl/funcX integration with PanDA to support DOE supercomputer (LCF) access
    - collaboration with CSI
    - funcX is a parsl based 'function as a service' for HPCs



BNL PanDA OKD components

Summary

- PanDA has been endorsed for Rubin Data Release Production processing. The production processing loads will increase steadily.
- The current production uses the DOMA PanDA system@CERN. Deployment of PanDA at SLAC K8s is close to completion.
  - PanDA@SLAC configuration very similar to PanDA@BNL, a good thing
- Many new implementations for Rubin:
  - The real-time logging sends both the payload logs and pilot logs to Google Cloud Logging
  - The PanDA monitor has been further improved to meet Rubin needs
  - Containerization of PanDA components and helm based deployments
  - Prompt processing mechanism is ready for Rubin
- Many new developments are ongoing: higher level interface for Campaign Management tooling, simplified resource description, clustering of pipeline tasks, support for multi-DF processing
- End-user usage of PanDA is what we support and offer helps to

Resources

- Rubin PanDA manual: https://panda.lsst.io
- PanDA monitoring for Rubin: http://panda-doma.cern.ch/
- Slack channels
  - Rubin users support
  - Rubin PanDA development
- PanDA Rubin weekly highlights summarizing BNL team's work

- PanDA: https://panda-wms.readthedocs.io/en/latest/
- iDDS: https://idds.readthedocs.io/en/latest/
- Harvester: https://github.com/HSF/harvester/wiki
- Pilot: https://github.com/PanDAWMS/pilot3/wiki

Supplementary slides

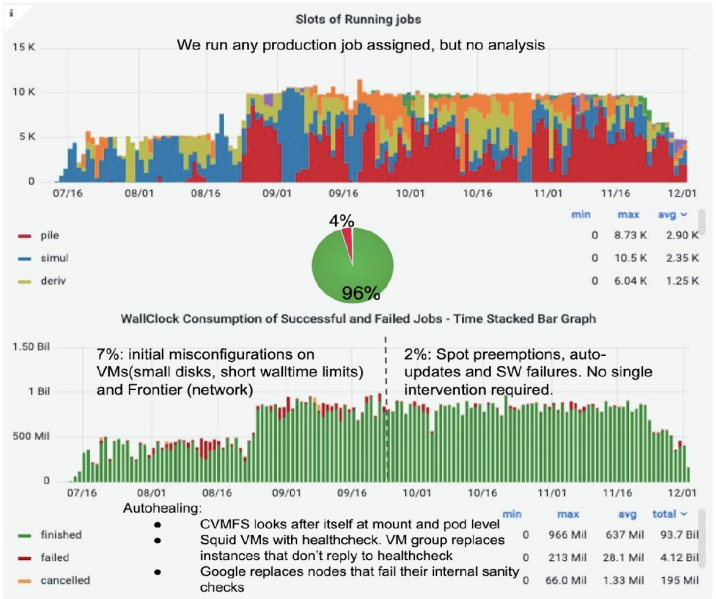
ATLAS Google R&D Project

Google Cloud

- New BNL-led R&D project with Google is starting
- **Strong leverage with PanDA Rubin effort**
  - with of course the important Google role in Rubin
- Building on a successful prior round
  - PanDA grid site equivalent with processing and Rucio storage
  - All production workflows work fine with very high efficiency
  - See supplementary slide for Phase 1 projects
- Particular focus on using and evaluating the Google cloud for analysis
  - Leverage tools and capabilities of Google Cloud Platform (GCP)
    - ARM, GPUs, FPGA, large memory/CPU, large databases like BigQuery
    - Kubernetes engine with on-demand scaling
    - Jupyter notebooks with DASK backend and Cloud storage
    - Cloud storage with S3
  - Ideal for bursty work and requests, leverage the elasticity
- Also make a fair comparison to traditional resources
  - Evaluate Google as an ATLAS site able to run all workflows
  - Provide a measure of total cost of ownership
- Resource scale 10k virtual CPUs, duration 18 mo

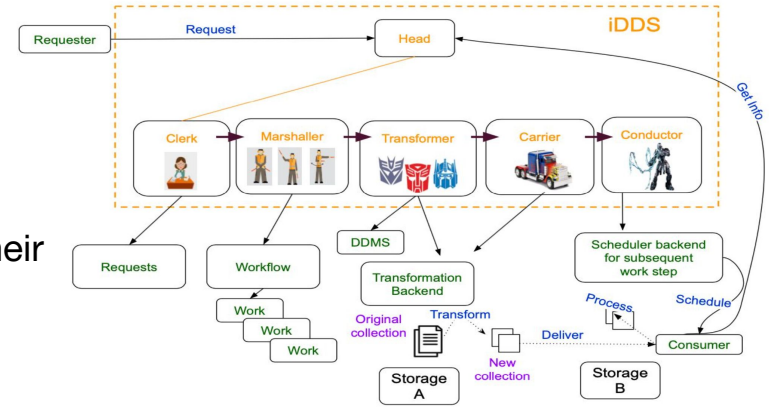
Stability

GOOGLE

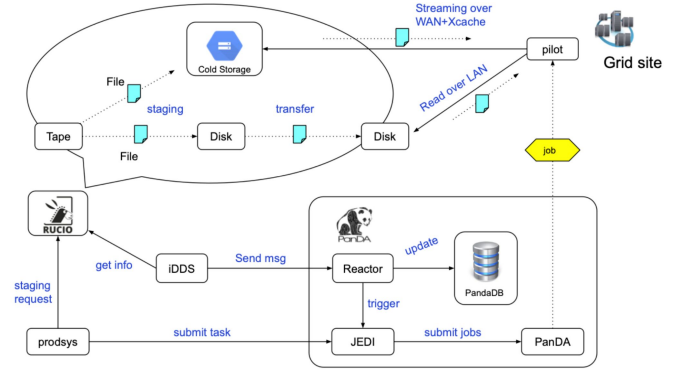


Intelligent Data Delivery Service (iDDS)

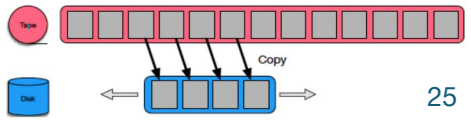
- iDDS is an experiment-agnostic add-on to PanDA (or other workload manager) supporting granular data delivery and **orchestration of complex workflows** that are efficient in their use of storage, network and processing resources
  - A joint project with IRIS-HEP (NSF), project hosted by HSF
  - Used by ATLAS, Rubin, sPHENIX
- **Essential for the support of Rubin's DAG based workflows**
- **iDDS applications had a busy year of development, benefiting from Rubin work and vice versa**
  - **ATLAS Data Carousel** processes tape-resident data using a small disk storage footprint via a sliding window orchestrated by PanDA, iDDS and Rucio
    - In production for all ATLAS production workflows
  - **Highly scalable ML services**
    - Enable analysts to run processing-intensive AI/ML applications on large distributed scale
    - Shorten optimization and training latencies by orders of magnitude
  - **Active learning services** drawing on the ML work used in ATLAS dark sector analysis, pubs being written



Intelligent Data Delivery Service (iDDS)

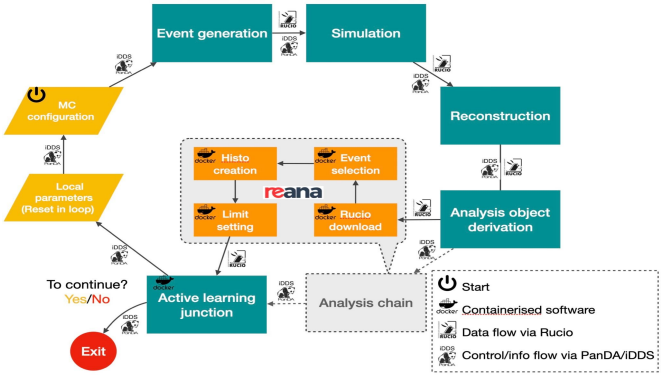


ATLAS Data Carousel using iDDS

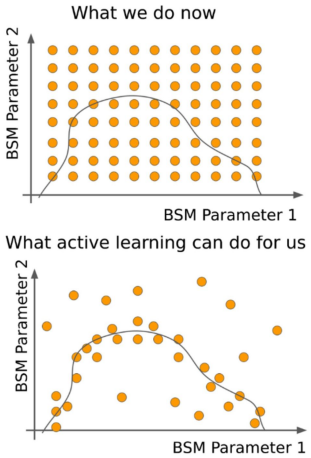


Large Scale Complex Workflows with PanDA/iDDS

- Ready access to diverse large scale resources can greatly accelerate developing processing-intensive applications
  - Shorten turnaround times by orders of magnitude
  - **Expanded scope for scientific creativity in developing applications**
- We are developing such services with PanDA/iDDS
  - **Hyperparameter optimization service** in production use for ATLAS fast simulation (GAN based calorimeter simulation FastCaloGAN)
  - An adaptation of this service with the same iterative refinement structure uses iterative regression to efficiently calculate a limit surface by rapidly 'learning' where the surface is, a.k.a. '**active learning**'
  - NPPS and Omega groups at BNL are applying the active learning service in the $H \rightarrow ZZ_d \rightarrow 4\ell$ dark sector analysis, pub notes in progress
    - Greater efficiency, scalability, automation enables a wider parameter search (instead of 1D, 2D or even 4D on large scale resources) and improved physics
- Working to generalize the services
  - **Use ATLAS work as a springboard for developing tools useful to the broader community**
- Exploring a new EIC use case: AI assisted detector design using Bayesian Optimization
- Plan collaboration with CSI on using funcX send work to LCFs
  - Such workflows can effectively leverage these GPU based machines



Active learning with PanDA/iDDS + REANA



Active Learning via iterative regression on a limit surface