Databases Report

A. Formica, S. Roe, E. Gallas, N. Ozturk, G. Dimitrov, P. Vasileva, I. Soloviev, A. Dumitru, A. De Salvo, J. Bahilo, L. Rinaldi

on behalf of the Databases team

ADC Weekly Meeting, May 15th 2018

Outline

- Run3 Conditions Developments
  - Prototype for REST Access to COOL Database
  - Gatling Stress Test
  - IOVDbSvc Updates
- Database Operations
  - Understanding DCS Data
  - Conditions Data Usage by Overlay and User Jobs
  - Frontier Backup Proxy
  - Frontier Kibana Monitoring Updates
  - Oracle Contract Renewal News
  - Database Monitoring
  - Online Databases
- Event White Board Developments

Introduction

- After the review of the “Conditions Data Infrastructure for Run-3” in December 2017 we started working on the two immediate action items (review report in CDS):
    - Develop a prototype for accessing the current COOL database using the **REST**ful (**R**epresentational **S**tate **T**ransfer) web services.
        - If any gains are possible in terms of caching
        - If a more simple http client can benefit in the core-software area
        - If it is possible to reduce some of the queries (gtag-tag resolution)
    - Understand the bottlenecks of the overlay production on the grid namely why the squid-Frontier caching system is not working.
- We’ll report on the above as well as on other developments (EWB) and operational topics.

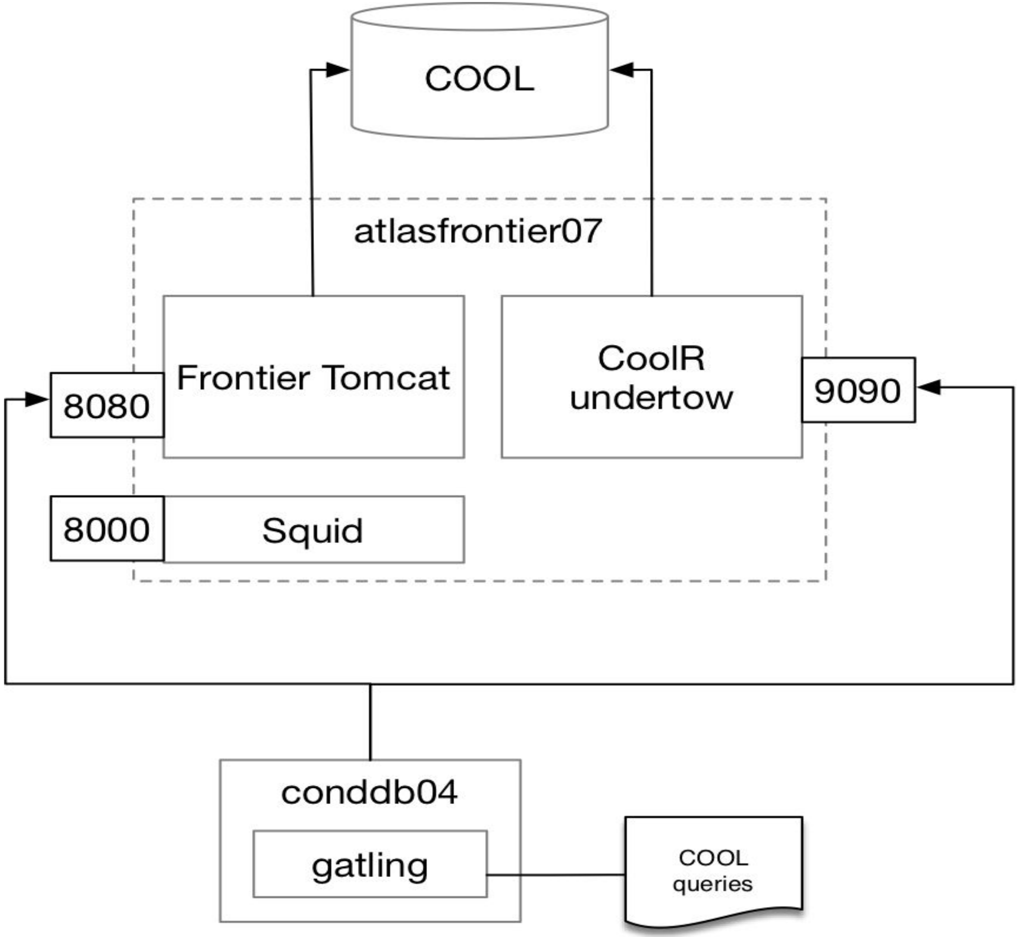
Prototype for REST Access to COOL

- CoolR server ready for first tests.
- Implements several endpoints: some of those use the parameters coming from URL as arguments to COOL-like queries which are “generated” inside a PL/SQL package in Oracle.
  Example:
  host:port/coolrapi/payloads/cool?schema=xxx&node=/some/node&db=CONDBR2&since=t0&until=tend&tag=ATAG
  Another URL instead utilize the same input as Frontier, i.e. the full COOL query.
  Example: ***host:port***/coolrapi/frontier/cool?q=a-zipped-query-like-the-one-of-frontier
- Endpoints will in general create JSON responses. The format of this JSON is similar to the one proposed for CREST.
- The prototype has been deployed in the same machine where an official installation of Frontier / Squid server was present (atlasfrontier07, thx to A.DS. and Chris).
- Both systems have been tested via gatling framework using the same set of 100 COOL queries (coming from Julio’s Frontier Kibana monitoring) for a given task: these queries have been selected among single version type queries.

Test Prototype Setup

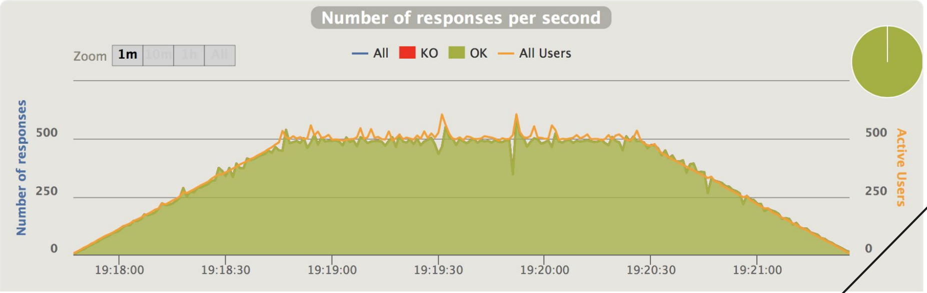
Description of the setup:

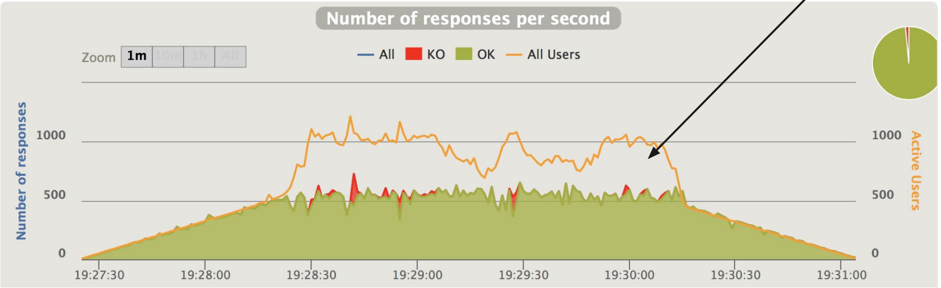
Use only direct ports to access Java servers; usage of Squid is for a later stage of testing, it needs further developments in CoolR

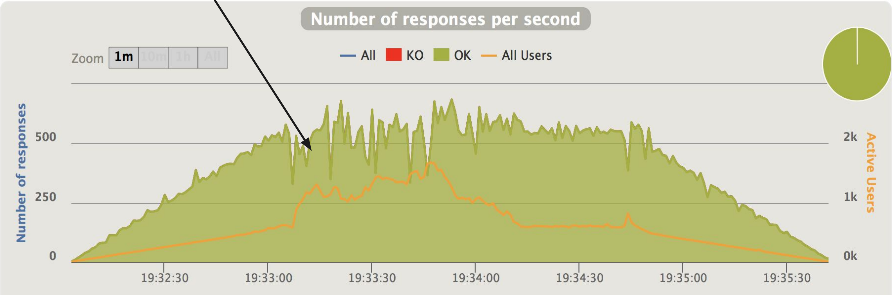


Gatling Stress Test

The main goal is to spot differences between Frontier (direct access, no caching) and CoolR when all others “parameters” are similar. Having the same machine and the same COOL query in input is thus our first stress test. We show here a “working” example for gatling tests (until about a pic point of 500 Req/s) and some other examples where we see the systems which are starting to suffer (in the pic range of 600 Req/s).







Frontier (left): server errors appear (config limit at 400 threads)
CoolR (right): number of active users increase because of queuing of req.

Next steps

- This exercise is preliminary to further testing:
  - Use query parameters via PL/SQL
- Introduce caching layer:
  - Use the same set of queries but providing ad hoc caching : e.g. any query providing a close interval and a “locked” tag is cachable by default for long time in principle.
- Caching layer should be identical for both systems:
  - Use the same SQUID on the atlasfrontier07 machine
- Important is to evaluate the length of the tests in order to probe a time span in which caching may be invalidated etc.

IoVDbSvc Updates

IOVDbSvc restructured to allow generation and reading of JSON files containing conditions data:

- Test job creates conditions data in local JSON-format files.
- These are uploaded to CREST.
- The job is reconfigured (one job option flag) to read and use the CREST db as a data source over http.
- Tags are created which mimic existing tags; tag resolution works from the CREST db.
    - Runs smoothly with all folder formats (COOL, Pool, Cool Vector Payload)
    - So far with only one IOV
    - **New work concentrates on extending this example to multiple IOVs**
      => recasting conditions data such that all channels are 'IOV-synchronised'
    - Along the way, IOVDbSvc restructured (caching in external class, string functions grouped in single file)

Understanding DCS Data

- We organized a few Conditions meeting dedicated to DCS and we continue to discuss progress since then (meeting again this Friday)
    - Many systems studied all their DCS folders and offline usage
    - Focus on reducing DCS volume … or in some cases, eliminate the folder
    - Working with Slava Khomutnikov (DCS2Cool expert) to improve smoothing
    - Notable recent work from Muhammad Alhroob (PIXEL)
        - PIXEL Week(Apr 9-12): Status and plan for conditions database
            - In depth study to rewrite folders and related software
- Collecting Oracle Stats weekly into a dedicated schema
    - Studies are ongoing
- Lorenzo has a new student:
    - Plan: algorithm on DCS data to reduce size but keep important features

Conditions Data Usage by Overlay Jobs

- While trying to understand the Frontier server loads caused by the overlay jobs we saw in the athena log files that there are many conditions data folders requested which are not used by the job in both digi and reco steps, each step has 3 stages to access conditions:
  - digi task, digi job - log.BSFilter, log.OverlayBS, log.EVNTtoHITS
  - reco task, reco job- log.ESDtoAOD, log.POOLMergeAthenaMPAOD0, log.RAWtoESD
  - 3rd stage of the reco step, RAWtoESD, has highest volumes and folders and the most folders read but the data not used:
    -
      ```
      Total payload read from COOL: 138899745 bytes in ((   6646.57 ))s
      WARNINGS Data requested but no data retrieved
      /LAR/BadChannelsOfl/EventVeto                           objs/chan/bytes 1/1/4 ((      47.14 ))s
      ```
      ... many similiar lines from LAR, PIXEL and SCT DCS, TILE, TRIGGER, and TRT
- Difficult to correlate these in the Frontier Kibana monitoring to check if the queries are cached or not with the current information available however eliminating these unnecessary accesses will certainly reduce the load.
- Overlay experts investigated and found a problem in the MC production with data overlay reco when the trigger processing is on, namely incorrect tracking configuration due to the online trigger specific version of SCT DCS folders being used in offline processing like in this overlay production, under discussion in ATLHI-188. Validation samples have been produced with blocking the unused folders and running the trigger in a separate step, validation is underway.

Conditions Data Usage by User Jobs

- Frontier Monitoring: Very useful (essential !)
    - Finding problematic time periods, tasks , queries, ...
- Have not seen issues with MC Overlay tasks recently (see previous slide)
- Recent incidents were with taskID = 0 (CERN local use -- not a Panda job)

  follow up is ongoing with particular users, groups
    - Rejected SQL (issue with the query itself)
        - Was able to find user name in this case
        - User writing erroneous COOL queries rather than using the COOL API
        - Action: Advise the user on 'best practices'
    - Disconnected queries (and associated proctime, dbtime, threads)
        - Same query submitted multiple times in 10 second intervals until it succeeds
        - Client Machine 128.142.153.73
        - dn=apache(48) Apache
        - Folder: COOLONL_TRIGGER/CONDBR2 /TRIGGER/LUMI/LVL1COUNTERS
        - Probaby from LumiCalc (known to be Conditions intensive)
        - Need to ask Chris Lee about Client Machine 128.142.153.73

Frontier Status - Backup Proxy

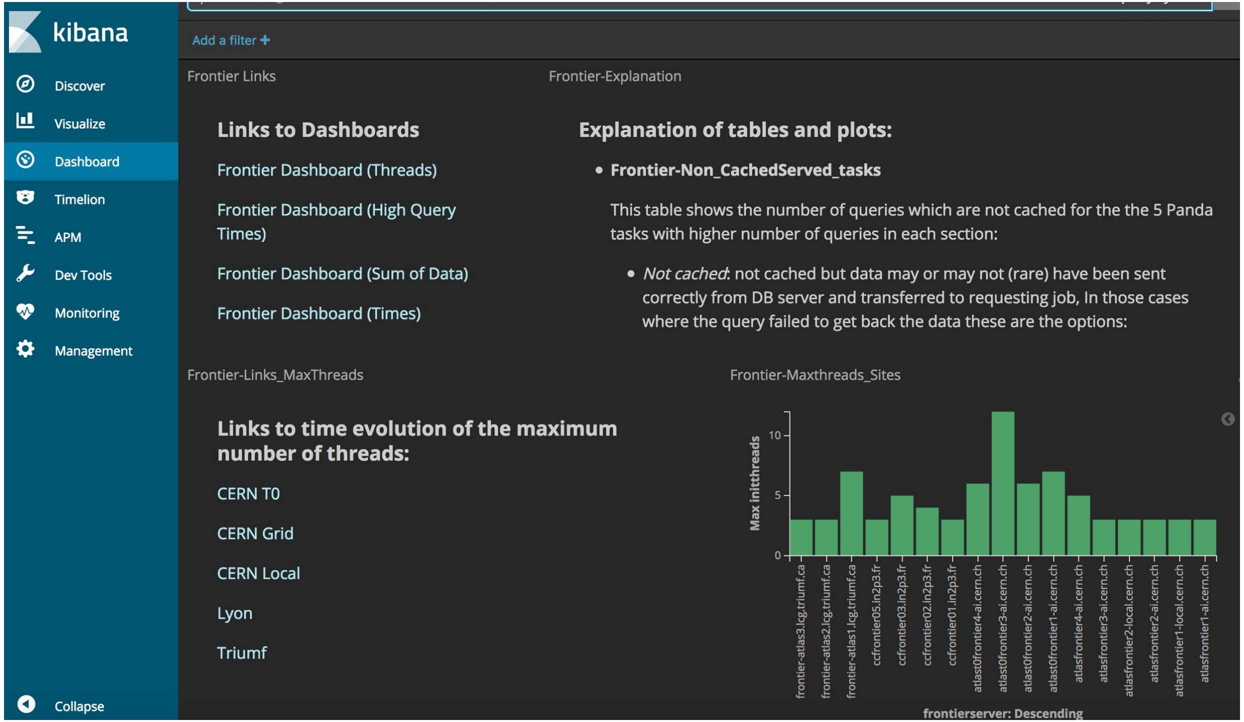
- Backup proxy instances ready
  - Co-hosted in the CMS existing proxies at CERN and FNAL
  - For each site there are 2 physical machines with 10 Gbps connectivity
  - The services of ATLAS and CMS are logically separated and running on different ports, so we do not risk to interfere each other
  - Aliases for each site are ready and ACLs set

- First test completed on 14/05/2018
  - Switching two sites, one in EU (INFN-ROMA1) and one in US (AGLT2), to use the relevant backup proxy as primary, followed by the existing round-robin aliases of the site squids
  - The test lasted about 2 hours: we correctly saw the traffic flowing to the backup proxies, testing their functionality, all looks good
    - MTRG monitoring and awstats can be used to monitor the proxies
    - No failover monitoring yet for ATLAS, should be ready in June

- Final configuration proposed
  - Replace the current off-site squids with the backup proxy, or just add the backup proxy as last squid of the sites (please note, different proxies depending on the sites' region)
  - Remove the direct access to the launchpads, allowing for better protection of the central services
  - Failover monitoring, to be used by shifters and experts to identify malfunctioning sites
  - Plans and time of the migration to be discussed

Frontier Kibana Monitoring Updates

- Updates done; group the plots into 4 dashboards, increased size of the histograms so that axis are more readable, established a fixed period (1 hour) for the data shown on the plots, reworked tables to show all relevant information in a better way.
- It'll be moved to prod-version from dev-version. Ready to be incorporated with the ADC Overview page.



kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
Collapse
Add a filter +
Frontier Links
Frontier-Explanation
Frontier-Links_MaxThreads
Frontier-Maxthreads_Sites
10
5
0
Max initthreads
frontier-atlas3.lcg.triumf.ca
frontier-atlas2.lcg.triumf.ca
frontier-atlas1.lcg.triumf.ca
ccfrontier05.in2p3.fr
ccfrontier03.in2p3.fr
ccfrontier02.in2p3.fr
ccfrontier01.in2p3.fr
atlast0frontier4-ai.cern.ch
atlast0frontier3-ai.cern.ch
atlast0frontier2-ai.cern.ch
atlast0frontier1-ai.cern.ch
atlasfrontier4-ai.cern.ch
atlasfrontier3-ai.cern.ch
atlasfrontier2-local.cern.ch
atlasfrontier2-ai.cern.ch
atlasfrontier1-local.cern.ch
atlasfrontier1-ai.cern.ch
frontierserver: Descending

Links to Dashboards

Frontier Dashboard (Threads)

Frontier Dashboard (High Query Times)

Frontier Dashboard (Sum of Data)

Frontier Dashboard (Times)

Explanation of tables and plots:

- **Frontier-Non_CachedServed_tasks**

  This table shows the number of queries which are not cached for the the 5 Panda tasks with higher number of queries in each section:

  - *Not cached*: not cached but data may or may not (rare) have been sent correctly from DB server and transferred to requesting job, In those cases where the query failed to get back the data these are the options:

Links to time evolution of the maximum number of threads:

CERN T0

CERN Grid

CERN Local

Lyon

Triumf

Thanks to
Ilija Vukotic

Oracle Contract Renewal News

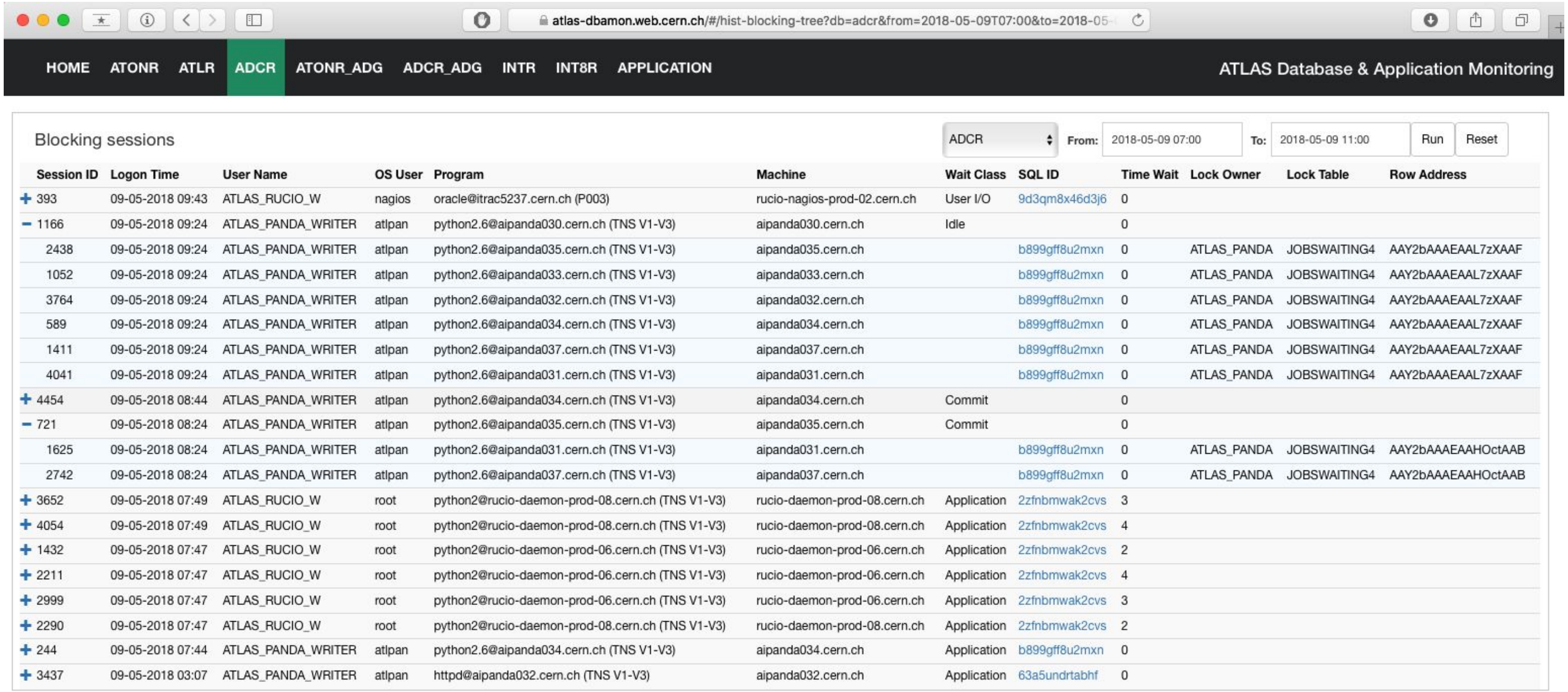
- Campus license model in the last 5 years (2013 - 2018), covering the Oracle usage for CERN and related activities at T1s and T2s (non-perpetual licenses).
- Today:
  - Renewed Campus license for the next 5 years (2018-2023): to cover the usage of CERN, T1s and T2s plus added some Oracle Cloud Services credit.
  - Started the move to perpetual processor licenses with a number licenses already acquired.
  - IT-DB slides
- Future activity would be DB consolidations: more databases on the same hosts (not the case of online or ADCR database).
- Plans for Oracle DB version upgrade in LS2. See here

Database Monitoring

New interface available: check for past blocking-blocked sessions on any database .

Gives insights on the locks that happened previously and ability to investigate the reasons at a later stage

https://atlas-dbamon.web.cern.ch/#/hist-blocking-tree



atlas-dbamon.web.cern.ch/#/hist-blocking-tree?db=adcr&from=2018-05-09T07:00&to=2018-05-
HOME
ATONR
ATLR
ADCR
ATONR_ADG
ADCR_ADG
INTR
INT8R
APPLICATION
ATLAS Database & Application Monitoring

Blocking sessions

ADCR **From:** 2018-05-09 07:00 **To:** 2018-05-09 11:00 Run Reset

Session ID	Logon Time	User Name	OS User	Program	Machine	Wait Class	SQL ID	Time Wait	Lock Owner	Lock Table	Row Address
+ 393	09-05-2018 09:43	ATLAS_RUCIO_W	nagios	oracle@itrac5237.cern.ch (P003)	rucio-nagios-prod-02.cern.ch	User I/O	9d3qm8x46d3j6	0			
− 1166	09-05-2018 09:24	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda030.cern.ch (TNS V1-V3)	aipanda030.cern.ch	Idle		0			
2438	09-05-2018 09:24	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda035.cern.ch (TNS V1-V3)	aipanda035.cern.ch		b899gff8u2mxn	0	ATLAS_PANDA	JOBSWAITING4	AAY2bAAAEAAL7zXAAF
1052	09-05-2018 09:24	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda033.cern.ch (TNS V1-V3)	aipanda033.cern.ch		b899gff8u2mxn	0	ATLAS_PANDA	JOBSWAITING4	AAY2bAAAEAAL7zXAAF
3764	09-05-2018 09:24	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda032.cern.ch (TNS V1-V3)	aipanda032.cern.ch		b899gff8u2mxn	0	ATLAS_PANDA	JOBSWAITING4	AAY2bAAAEAAL7zXAAF
589	09-05-2018 09:24	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda034.cern.ch (TNS V1-V3)	aipanda034.cern.ch		b899gff8u2mxn	0	ATLAS_PANDA	JOBSWAITING4	AAY2bAAAEAAL7zXAAF
1411	09-05-2018 09:24	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda037.cern.ch (TNS V1-V3)	aipanda037.cern.ch		b899gff8u2mxn	0	ATLAS_PANDA	JOBSWAITING4	AAY2bAAAEAAL7zXAAF
4041	09-05-2018 09:24	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda031.cern.ch (TNS V1-V3)	aipanda031.cern.ch		b899gff8u2mxn	0	ATLAS_PANDA	JOBSWAITING4	AAY2bAAAEAAL7zXAAF
+ 4454	09-05-2018 08:44	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda034.cern.ch (TNS V1-V3)	aipanda034.cern.ch	Commit		0			
− 721	09-05-2018 08:24	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda035.cern.ch (TNS V1-V3)	aipanda035.cern.ch	Commit		0			
1625	09-05-2018 08:24	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda031.cern.ch (TNS V1-V3)	aipanda031.cern.ch		b899gff8u2mxn	0	ATLAS_PANDA	JOBSWAITING4	AAY2bAAAEAAHOctAAB
2742	09-05-2018 08:24	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda037.cern.ch (TNS V1-V3)	aipanda037.cern.ch		b899gff8u2mxn	0	ATLAS_PANDA	JOBSWAITING4	AAY2bAAAEAAHOctAAB
+ 3652	09-05-2018 07:49	ATLAS_RUCIO_W	root	python2@rucio-daemon-prod-08.cern.ch (TNS V1-V3)	rucio-daemon-prod-08.cern.ch	Application	2zfnbmwak2cvs	3			
+ 4054	09-05-2018 07:49	ATLAS_RUCIO_W	root	python2@rucio-daemon-prod-08.cern.ch (TNS V1-V3)	rucio-daemon-prod-08.cern.ch	Application	2zfnbmwak2cvs	4			
+ 1432	09-05-2018 07:47	ATLAS_RUCIO_W	root	python2@rucio-daemon-prod-06.cern.ch (TNS V1-V3)	rucio-daemon-prod-06.cern.ch	Application	2zfnbmwak2cvs	2			
+ 2211	09-05-2018 07:47	ATLAS_RUCIO_W	root	python2@rucio-daemon-prod-06.cern.ch (TNS V1-V3)	rucio-daemon-prod-06.cern.ch	Application	2zfnbmwak2cvs	4			
+ 2999	09-05-2018 07:47	ATLAS_RUCIO_W	root	python2@rucio-daemon-prod-06.cern.ch (TNS V1-V3)	rucio-daemon-prod-06.cern.ch	Application	2zfnbmwak2cvs	3			
+ 2290	09-05-2018 07:47	ATLAS_RUCIO_W	root	python2@rucio-daemon-prod-08.cern.ch (TNS V1-V3)	rucio-daemon-prod-08.cern.ch	Application	2zfnbmwak2cvs	2			
+ 244	09-05-2018 07:44	ATLAS_PANDA_WRITER	atlpan	python2.6@aipanda034.cern.ch (TNS V1-V3)	aipanda034.cern.ch	Application	b899gff8u2mxn	0			
+ 3437	09-05-2018 03:07	ATLAS_PANDA_WRITER	atlpan	httpd@aipanda032.cern.ch (TNS V1-V3)	aipanda032.cern.ch	Application	63a5undrtabhf	0			

Online Databases

- Smooth operations since beginning 2018.
- Keep software release used during 2017 data taking (tdaq-07-01-00).
- Few patches for newly discovered issues.
- **Oracle databases** (trigger and partially detector configuration, conditions):
    - CORALCOOL-3009 and ATDSUPPORT-245- partial fix for CORAL server and proxies reconnect after Oracle service glitch.
    - ATONLBS-19 - fix simultaneous requests to COOL blocked CTP and luminosity block updates.
- **OKS Configuration DB** (DAQ configuration)
    - ADTCC-169 - avoid duplicated objects on OKS server repository via inheritance hierarchy
- **P-BEAST** (raw operational monitoring archive)
    - Used more and move widely (~60 dashboards from most ATLAS detectors and systems).
    - Move to Grafana 4.
    - Implemented extra functionality for data selection, modification and aggregation.
    - Work on data correlations, annotations and read query optimisations.

Event White Board (EWB) Developments

- Ongoing tests to get to the most efficient DB schema based on Oracle 12c
  - Goal: simplicity, reliability, flexibility, performance in write and read (quite a challenge).
  - Current prototype: 36K collections with 185M event records having JSON data blocks.
- Exploration of the DB built-in JSON functions and search capabilities
  - Currently partitioned JSON search index is not possible. Oracle acknowledged a related bug. Service request for a patch is opened and waiting for a resolution.
- Oracle REST Data Services (ORDS) setup for REST interface to the relational EWB objects.
- Focus on the ESS (Event Streaming Service) requirements towards the EWB system. Joint work with ATLAS DBAs and Nicolo Magini.

Conclusions

- Good progress with prototyping of REST access to COOL data however need manpower to work on several areas, **please join if interested:**
  - Gatling stress tests
  - Development of the conversion tools to convert COOL data into CREST data, testing of methods to understand what data can migrate from COOL to CREST
  - Analytics/monitoring of the data collected from squid/Frontier to understand better the caching issues
- Good progress also with Event White Board developments
- Operationally databases work fine and stable