Requirements for conditions DB area for Run3 and beyond

Andrea Formica
on behalf of Atlas DB coordination

Outline

- Conditions DB usage today:
  - Use cases: offline and online
  - Infrastructure and data volume
- Estimation for future Conditions DB needs:
  - Database technologies
  - Software libraries

Conditions Data in ATLAS

- Conditions are stored in Oracle via COOL API…
  - we usually say that conditions are stored in COOL
- What do we have in COOL today
  1. Configuration
  2. DCS (subset of data from PVSS Oracle Archive)
  3. Monitoring (MONP200)
  4. **Calibrations, Alignments, ….**

- In general how do conditions scale ?
  - linearly with time and number of channels in the detector
  - not really with “physics” data…luminosity does not really matter at the first order for conditions data

Offline and Online…

- Conditions data are produced and consumed in:
  - data taking data-flows: online, express stream processing, bulk processing
  - MC productions
  - Reprocessing
  - Calibration files from physics groups (today file system storage in calibration area)
- Access to conditions
  - Direct access: COOL API / **Coral**
    - both libraries are supported today by IT division
    - Coral is used in online environment to access Oracle tables behind the conditions use-case (see slide 6 for details)
  - Distributed access:
    - Frontier (Tomcat) / SQUIDs
    - Coral Server (Atlas online)

Oracle infrastructure for conditions

- ATONR (online cluster), ATLR (offline cluster)
  - many schemas and data to administer (thanks to Gancho and Petya)
  - we provide below some numbers on the present data volumes (measured in 2016 and 2017, these numbers integrate all conditions content from Run1 and Run2, for data and MC and also monitoring tables)

DB/metric	ATONR		ATLR	
#DB schemas	17		18	
#Tables	***2016***: 9524 ***2017***: 9732		5563 5696	
Data volume	***2016***: 1.7 TB ***2017***: 2.2 TB	*Trigger and TDAQ dominate the volume*	1.0TB 1.6TB	*DCS dominates the volume*
#rows (Giga)	***2016***: 1.9 ***2017***: 2.4		2.7 4.3	*new DCS folder may increase a lot the number of rows*
Largest # of rows (Giga)	***2016***: 1.2 (trigger) ***2017***: 1.5 (trigger)		2.5 (dcs) 4.0 (dcs)	

Online specific use cases

- In addition to COOL conditions data, online systems register several informations
  - Run number database
  - OKS configuration archive
  - physics data files DB packages (SFO)
  - messages logging archive (ERS)
  - monitoring data archival packages (CoCa and MDA)
  - trigger configuration packages
- Most of the use CORAL for direct DB access
  - allow to switch from Oracle to SQLite in a transparent way in case of connection problems
  - no evolution foreseen in long term Run3 and Run4

Oracle infrastructure for online

- ATONR (online cluster)
  - few schemas essentially dedicated to data taking and trigger
  - we provide below some numbers (thanks to Gancho) on the present data volumes

DB/metric	ATONR
#DB schemas	8
#Tables	220 (dominated by trigger)
Data volume	~600 GB (half coming from logging)
#rows (Giga)	4.6
Largest # of rows (Giga)	3.5 (MDA)

Future plans: overview

- Database technologies
  - Relational choice not under discussion: we think to stick to Oracle for future runs
- Changes in data model
  - under study for offline use case the possibility to simplify the storage by strongly reducing the number of tables and schemas (see next slide)
  - this change has in any case no big impact on data volumes at DB level if all payloads are stored in Oracle (may have an impact on indexes)
- Access to conditions
  - Keep direct access libraries : COOL API / Coral
    - we ask for IT division support on the used libraries
  - Distributed access: simplify the distribution model
    - single DB instance at CERN with front-end web servers

On going studies

- Conditions DB
  - simplification of data model (at the DB level): use only Blobs, basically use a data model similar to CMS; on going project for which interest has been shown by several experiments (see Paul Laycock talk for details)
  - enhance the separation between backend and client, enforcing a pure REST architecture for DB access
    - possibility to support several relational backends
  - ~O(10) tables instead of ~O(10000)
- DCS data
  - can we adopt better time-series DB for monitoring ?
  - still, we should prepare digested data to store in Conditions DB
  - Influx DB and PBeast among the possibilities to explore alternative solutions for DCS data (monitoring use case only)
- Monitoring of conditions usage
  - on going development to gather information from data processing jobs in order to better understand the conditions access and usage on the client side (all the system in based on relational tables)
  - Development of dashboards to monitor distributed conditions access (Frontier logs and Athena logs) using ElasticSearch and Kibana

Estimates for data volume

- Extrapolation to Run3 and beyond
  - Conditions data : from past experience we can gather about 1 TB per year
  - Metadata and bookkeeping information (improve monitoring on conditions usage) : scale with processing activities
  - Online specific relational usage: no special differences respect to the past

Summary

- Relational database
  - Oracle (or relational in general): we can only foresee an increase of usage in general
- non Relational database
  - in the conditions area we are interested in exploring time-series solutions for several monitoring needs
  - ElasticSearch and similar tools are started being used in the monitoring of conditions usage and Frontier servers
- Software libraries
  - COOL and Coral : we still need support for both libraries even though we are making efforts to migration to a simpler solution for conditions.
  - Coral: large usage in online environment for specific monitoring and configuration needs

…some personal comments

- Impressive amount of work done by IT on DB on demand and other services…
  but
  - in order to reach a production level for a given use case in an experiment we still need a lot of time and knowledge
  - this is why we developed a special confidence in Oracle: thanks to IT support and to our (Atlas) DBAs we were capable to overcome many obstacles in production services
  - we need to exchange on new technologies with IT and learn together before being capable to adopt them on production services