Access to ATLAS Geometry and Conditions Databases

Vakho Tsulaia

University of Pittsburgh

ATLAS – South Caucasus
Software/Computing Workshop & Tutorial
Tbilisi, 2010-10-26

Outline

- Introduction to ATLAS Geometry Database

- Introduction to ATLAS Conditions Database

- Database Replication Technologies

- Strategy for Remote Access to ATLAS Databases

- Practical Info: Tips for Running User Analysis in Athena

Geometry DB

- Relational database of **Primary Numbers** for the ATLAS Detector Description
  - All data for building GeoModel description in single place
  - Contains pointers to external files
    - Identifier dictionaries
    - Magnetic field maps (becoming obsolete)
    - All such files are shipped with the s/w release, no extra steps needed for getting them

- Primary numbers stored inside **Data Tables**

- Data Tables logically organized into **Hierarchical Tree of Nodes**
  - Leaf Node: corresponds to Data Table
  - Branch Node: pure logical entity for building the tree
  - Each subsystem has its own branch
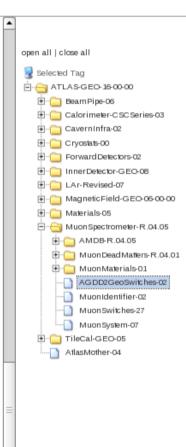 - ATLAS Branch Node: root of the node tree

3

Geometry DB

Tags

- **Leaf Node Tag**: subset of records in the Data Table

- **Branch Node Tag**: collection of tags of its children nodes

- Parent Tag to Children Tags relationships form **Tag Tree**

- Tag Tree identified by the **Tag of the root node**

- **Global tag**: tag of the ATLAS node
  - ATLAS geometry version. Ex.: ATLAS-GEO-16-00-00

- Tags can be **locked**
  - No changes are permitted in the data corresponding to a locked tag

4

Geometry DB Browser

http://atlas.web.cern.ch/Atlas/GROUPS/OPERATIONS/dataBases/DDDB

ATLAS-GEO-10-07-00, doc
ATLAS-GEO-10-08-00, doc
ATLAS-GEO-10-09-00, doc
ATLAS-GEO-10-10-00, doc
ATLAS-GEO-10-11-00, doc
ATLAS-GEO-10-12-00, doc
ATLAS-GEO-10-13-00, doc
ATLAS-GEO-11-00-00, doc
ATLAS-GEO-11-00-01, doc
ATLAS-GEO-11-01-00, doc
ATLAS-GEO-11-01-01, doc
ATLAS-GEO-11-02-00, doc
ATLAS-GEO-11-02-01, doc
ATLAS-GEO-11-03-00, doc
ATLAS-GEO-11-03-01, doc
ATLAS-GEO-11-04-00, doc
ATLAS-GEO-11-04-01, doc
ATLAS-GEO-12-00-00, doc
ATLAS-GEO-13-00-00, doc
ATLAS-GEO-13-00-01, doc
ATLAS-GEO-13-01-00, doc
ATLAS-GEO-14-00-00, doc
ATLAS-GEO-14-00-01, doc
ATLAS-GEO-14-01-00, doc
ATLAS-GEO-14-01-01, doc
ATLAS-GEO-14-02-00, doc
ATLAS-GEO-14-02-01, doc
ATLAS-GEO-15-00-00, doc
ATLAS-GEO-16-00-00, doc
ATLAS-GEOHF-08-00-02, doc
ATLAS-GEOHF-09-00-00, doc
ATLAS-GEOHF-10-00-00, doc
ATLAS-GEOHNSF-08-00-02, doc

open all | close all

Selected Tag
- ATLAS-GEO-16-00-00
  - BeamPipe-06
  - Calorimeter-CSCSeries-03
  - CavernInfra-02
  - Cryostats-00
  - ForwardDetectors-02
  - InnerDetector-GEO-08
  - LAr-Revised-07
  - MagneticField-GEO-06-00-00
  - Materials-05
  - MuonSpectrometer-R.04.05
    - AMDB-R.04.05
    - MuonDeadMatters-R.04.01
    - MuonMaterials-01
    - AGDD2GeoSwitches-02
    - MuonIdentifier-02
    - MuonSwitches-27
    - MuonSystem-07
  - TileCal-GEO-05
  - AtlasMother-04

ATLAS DD Database

Node **AGDD2GeoSwitches** (show column descriptions)

Tag :   **AGDD2GeoSwitches-02**, created: *(date unknown)*
Status: **LOCKED**, *(date unknown)*
Comment: **For R.04.01, 21/05/2010**

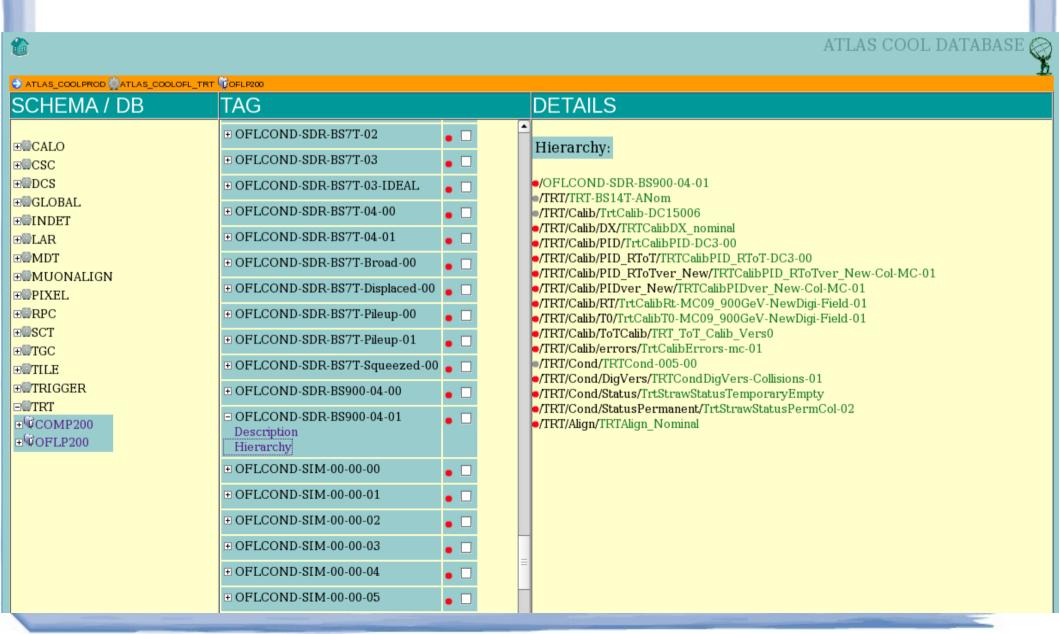| AGDD2GEOSWITCHES_DATA_ID	KEYNAME	KEYVALUE
long	string	int
200	ECT_Toroids	0
201	BAR_Toroid	1
202	Feet	1
203	RailAssembly	1
204	JFSH_Shield	0
205	JDSH_Shield	0
206	JTSH_Shield	0
207	pp2	1
208	MBAP_AccessPlatform	1
209	MBAP_Sector13	0
210	MBWH_BigWheels	1
211	SADL_CalorimeterSaddle	1
212	TBWH_BigWheels	1
213	TGC3_BigWheels	1
214	TGC1_BigWheels	1
215	MDTRail	0
216	servicesAtZ0	1
217	HFTruckRail	1

Conditions DB

- Large relational database containing information about **Detector Status**, **Data-Taking Conditions**, **Calibrations**, **Alignment ...**

- ATLAS Conditions DB is a **COOL Database**
  - COOL: one of 3 components of the LCG Persistency Framework (other two: POOL, CORAL)

- Conditions data maps to **transient C++ objects**, which are accessible to Athena Algorithms at run time through **Transient Store**

- Two types of object data storage:
  - Inline Payload: data stored in the relational tables itself
  - Referential Payload: data stored in POOL files outside the relational database
    - Tables hold POOL tokens

6

Conditions DB

- Data is organized into **Folders** containing objects of the same type

- Folders are organized into **Foldersets** to form a **Hierarchy**

- Many independent trees of folders (**schema**)
    - INDET, LAR, MDT, DCS, GLOBAL, …

- Separate schema for **Online** and **Offline** folders
    - COOLONL_XXX vs COOLOFL_XXX

- Two separate conditions databases for **Data** (COMP200) and **MC** (OFLP200)
    - Logical division of tables inside each schema

7

Conditions DB

- Objects in Folders are stored with an **I**nterval **O**f **V**alidity (start-stop), channel number and optionally a tag

- Tag of a Folderset corresponds to tags of its daughter Foldersets and Folders

- Tag tree uniquely identified by the root tag – **Global Conditions Tag**

  – Data global tags: COMCOND-XXX

  – MC global tags: OFLCOND-XXX

Conditions DB Tag Browser
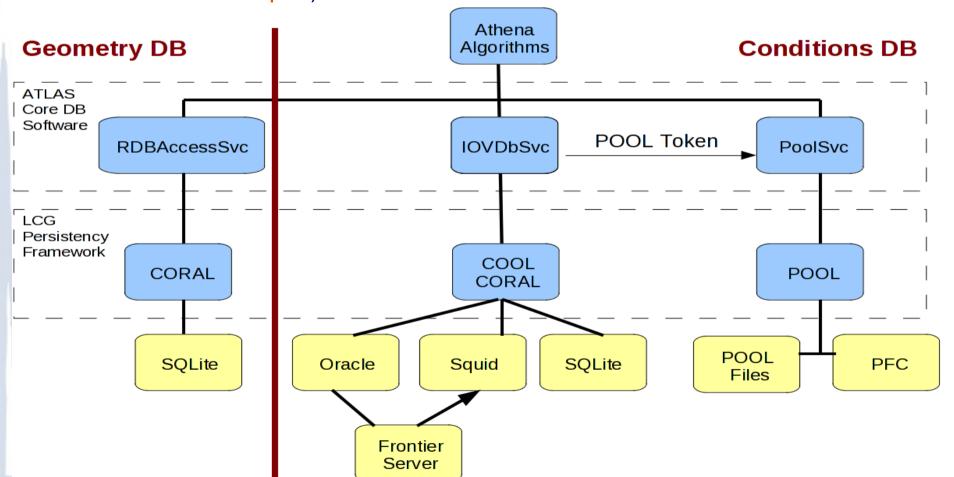
https://atlas-coolbrowser.web.cern.ch/atlas-coolbrowser

Some statistics...

… for demonstrating data volumes in each database

- Statistics obtained in Feb-2010

	Conditions DB	Geometry DB
Number of schema	29	1
Number of tables	8K	0.9K
Total number of rows	762M	0.5M
Data volume Oracle	0.5TB	0.1GB
Data volume POOL	0.2TB	--

DB Access in Athena

- Athena applications access conditions and geometry databases using common LCG software libraries **POOL**, **COOL** and **CORAL**

    - Allows for transparent usage of various technologies (Oracle, SQLite, FroNTier/Squid)

Replication

Geometry DB

- Master copy: **Oracle** server at CERN

  - For data loading and management

  - For new code development and testing

  - Not used by any production!

- **SQLite** replica

  - Entire database dumped into a SQLite file

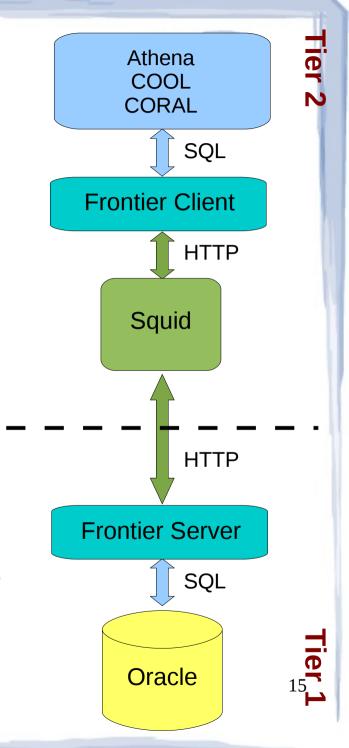 - Delivered to remote sites using Database Release technology

  - Used in production

Replication

Conditions DB

- Master copy: **Oracle** servers at CERN
  - Online Conditions DB (ATONR)
  - Offline Conditions DB (ATLR)

- ATLR database replicated to 10 **Oracle servers at T1**-s using **Oracle Streams** technology

- Direct Oracle access or via **FroNTier/Squid**
  - Faster access by client application
  - Alleviate load on Oracle servers

- Replicated to **SQLite**
  - Delivered to remote sites using Database Release technology

Replication

Conditions DB: Data traffic

Replication

Conditions DB. FroNTier/Squid

- Frontier: distributed database access system
  - Includes data caching
- Main components
  - Frontier server
    - Communicates directly with Oracle server
    - Provides data to Squids
  - Squid
    - Communicates with Frontier server over Http
    - Caches retrieved data locally for its clients
- ATLAS put Frontier system into operation late in 2009
  - Frontier servers at T1 sites
  - ~60 Squids all over the world
    - Mostly T2, some T3 too

Tier 2

Athena
COOL
CORAL

SQL

Frontier Client

HTTP

Squid

HTTP

Frontier Server

SQL

Oracle

Tier 1

15

Replication

Conditions DB. POOL files

- Conditions payload POOL files organized into **datasets**, which are distributed using Atlas **DDM** (Distributed Data Management) system

- Distributed to **all T1** and **T2** sites
  - Some T3-s too

- Need to be included into **POOL File Catalog** (PFC) in order to be read by Athena jobs
  - Automated procedure of generating local PFC

Database releases

- **Mechanism of distributing 'frozen' snapshots of the Conditions, Geometry and Trigger databases outside CERN**

- A **tarball** with
  - SQLite replicas

  - POOL Files and POOL File Catalogs

  - Configuration files for setting up database connections

- Allows for running Athena in '*traveling laptop*' mode
  - No need to have network access …

  - … provided you have **right** DB Release version

17

Database releases

Data vs MC

- **MC** DB Release
  - Aka 'baseline' DB Release. 3-digit versions (ex. **DBRelease-12.8.2**)
  - Contains all data necessary to run any MC job:
    - Simulation/Digitization/Reconstruction
  - Each new DB Release is a superset of its predecessor.
  - You can always use the most recent DB Release version

- **Conditions** DB Release for real data
  - Built on top of a baseline DB Release. 4-digit versions (ex. **DBRelease-12.8.2.1**)
  - Includes complete replica of the geometry database
  - Includes conditions (SQLite & POOL) only for given set of runs

Strategies for remote database access

- **Simulation production.** Baseline DB Release

- **T0 processing.** Direct Oracle access for Conditions. Special SQLite replica for Geometry

- **Reprocessing.** Conditions DB Release

- **Calibration/alignment/etc.**
  - Conditions:Direct Oracle access at CAF&T1, FroNTier/Squid at T2
  - Geometry: SQLite (DBRelease)

- **User analysis**
  - Conditions: FroNTier/Squid
  - Geometry: DB Release

19

Tips for users

What Global Conditions and Geometry tags to use?

- **Auto-configure** your job
  - Make the job read global tags from its input file (ESD, AOD)

- In job options:

```
from RecExConfig.RecFlags import rec
rec.AutoConfiguration=['everything']
```

- In job transforms:

Command line parameter 'autoConfiguration=everything'

https://twiki.cern.ch/twiki/bin/view/Atlas/RecExCommonAutoConfiguration

Tips for users

How to configure my environment to access

- FroNTier/Squid?
- Conditions payload POOL files?
- DB Release for geometry (and MC conditions if needed)?

All that is done for you automatically...

… just sit back and enjoy the ride!

Tips for users

If things go wrong …
and it seems to be related to database access

- Useful information on Wiki
  - Athena DB Access:
    https://twiki.cern.ch/twiki/bin/view/Atlas/AthenaDBAccess

  - COOL Troubles:
    https://twiki.cern.ch/twiki/bin/viewauth/Atlas/CoolTroubles

  - Atlas DB Release:
    https://twiki.cern.ch/twiki/bin/viewauth/Atlas/AtlasDBRelease

- These wiki documentations should be able to help you in narrowing down the problem and then you'll be in position of
  - Either ask your site admin
  - Or send e-mail to Database Operations<hn-atlas-DBOps@cern.ch>

22

Conclusions

- Access to ATLAS non-event databases (Conditions and Geometry) is fairly complex

- Need to deal with various access patterns, support different technologies

- The great effort has been made to make DB access for user analysis as transparent as possible …

- Hopefully you find the presented material helpful in understanding details of this complex environment and in diagnosing database related problems

23

Coffee break …