

# ATLAS Conditions Databases for T3 Frontier and Squid: Experience installing and working at T3

Frederick Luehring

[luehring@indiana.edu](mailto:luehring@indiana.edu)

Indiana University

US ATLAS Tier2/3 Workshop at UChicago

August 20, 2009

# Data Analysis Scenario

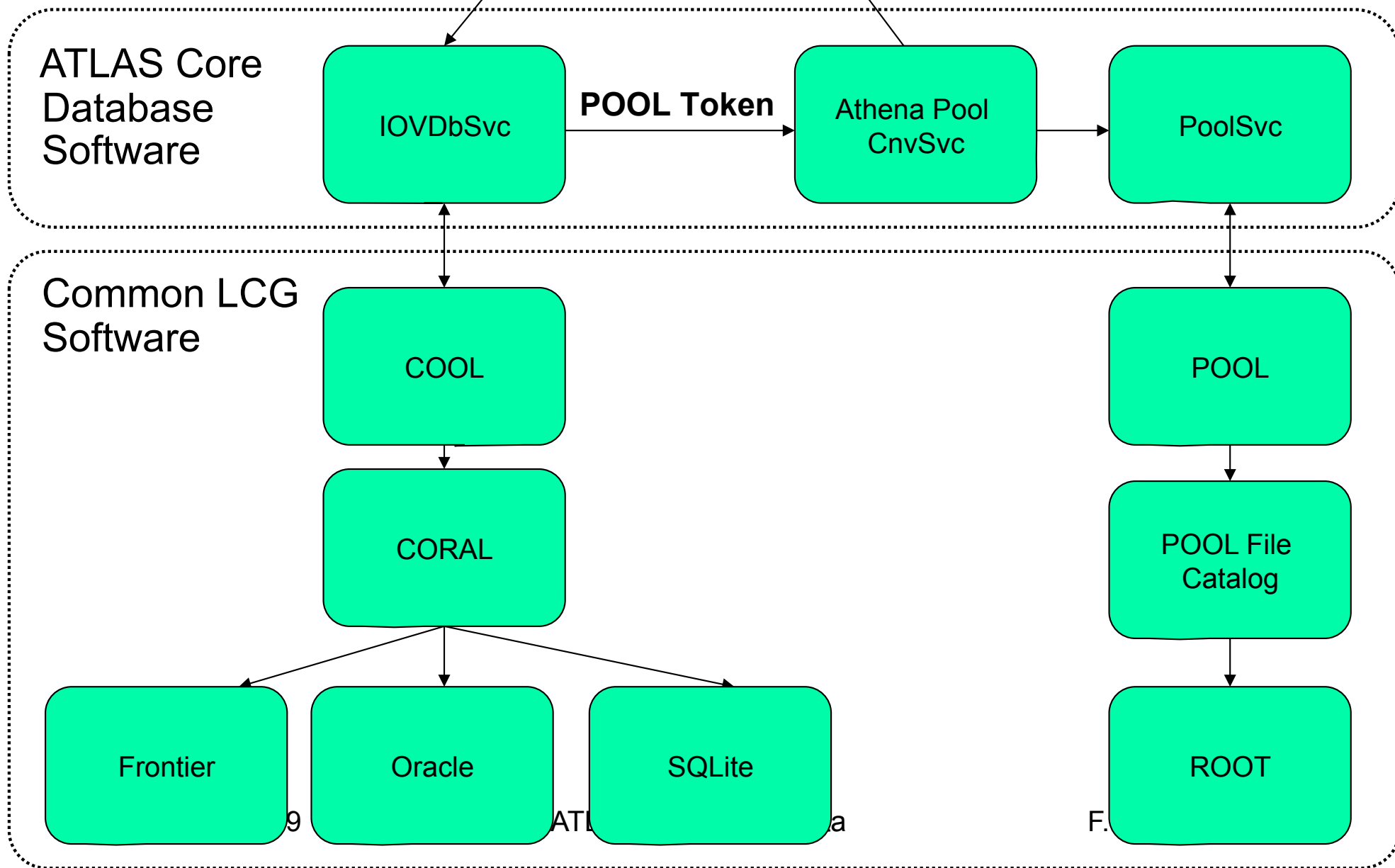
- It is possible to work efficiently at a Tier 3 or a laptop:
  - Subscribe an interesting dataset to the Tier 2
  - Use dq2-get to get the sample data onto the Tier 3
    - dq2-get from any US site is fast.
    - dq2-get from CERN is slow.
  - Develop/test your software package on the Tier 3
  - Run your Monte Carlo locally on the Tier 3 (if needed!)
  - Then use pathena to send large numbers of data analysis jobs to a Tier 2 or BNL (wherever the data is)
    - Have to be careful not to waste cycles which is real easy to do because pathena makes it simple to launch thousands of jobs.
    - It is possible to run against both RAW/ESD/AOD and MC/Event data.
  - Process DP3Ds and ntuples on the Tier 3 or laptop.
- Running against Event Data requires access to conditions data that is NOT included with the release.

# Need For Conditions Data

- My work with conditions data is an unintentional side effect of testing the TRT online monitoring code using cosmic ray RAW and ESD datasets.
- In order to read the event data needed to fill the TRT monitoring histograms, conditions data about the event data must also be read.
- My goal was to do analysis but at the MWT2 and at the Indiana University Tier 3 and not at the Tier 1 (that worked fine!).

# Conditions Data Access Design

Thanks to Sasha Vanachine for this diagram. Peter Van Gemmeren & Richard Hawking also contributed.



# Conditions Database Issues

- The use of the Conditions Databases in ATLAS has proved to be complicated when processing real cosmic data (NOT MC). In the end it turns out that:
  - Some conditions data needs to be read from Oracle at BNL
    - To read at a reasonable speed you need to use a Squid cache to access the FroNTier server at BNL.
    - Thanks to lots of work by the ATLAS database group there has been a significant improvement in the reading speed by remote clients as various improvements have been put into athena.
  - Conditions data in Oracle sometimes references external POOL files
    - So you need local copies of hundreds of .pool.root files and to construct an xml file saying where these files can be found locally.
- Based on tests of filling TRT monitoring histograms from the RAW & ESD files, solutions exist that allow the efficient reading for both of these classes of data.

# Use of Oracle at BNL

- To read the conditions data from Oracle at BNL (or for that matter from the CERN Tier 0 or any Tier 1) you need to put the following lines in your jobOptions:

```
#-----  
# Load Conditions Database from nearest tier 1 site  
#-----  
  
from DBReplicaSvc.DBReplicaSvcConf import DBReplicaSvc  
DBReplicaSvc = DBReplicaSvc(UseCOOLSQLite=False)  
svcMgr += DBReplicaSvc
```

- This will (as the comment states) get the conditions data from the nearest tier 1 site (BNL in the US) BUT there are two issues...
  - With the recent versions of athena (15.4.0 and higher), it is possible in some cases to set environmental variables instead using the above lines.

# Issues with Reading Conditions

- The two issues are:
  1. Reading from the BNL Oracle in this way results in a 15 minute to 45 minute delay starting a job.
    - Also if everyone in US ATLAS tries to simultaneously read the BNL Oracle there is possibility that the BNL Oracle will become overloaded.
  2. Not all the conditions data is in Oracle. Most calibrations data are in POOL files which are referenced from within Oracle and must be locally on the site running the job.
- The rest of this talk will discuss the solution to these two issues.
  - Note: there is probably not a “one size fits all” solution to this. Different sites may choose to proceed differently.
  - This talk will list the possibilities that I am aware of and discuss their pluses and minuses.

# Issue 1 Solution: FroNTier/Squid

- An experimental FroNTier server and Squid cache has been setup at BNL by Carlos Gamboa and John De Stefano.
- Any Tier 2, Tier 3, or end-user computer can access it using a local Squid cache (best) or the BNL Squid cache (no local setup required and still much faster than remote access directly to Oracle).
  - Carlos, John, and I have been working with the Tier 2 sites to setup local Squid caches and they all have done so.
- Instructions on installing a local Squid cache:  
<http://www.usatlas.bnl.gov/twiki/bin/view/Admins/SquidTier2>
- Always specify both the FroNTier server (BNL for now) and also the Squid cache (local or BNL) using a configuration file and/or environmental variables.



# Comments on FroNTier/Squid

- Please note that the FroNTier client is used to access a Squid cache (local or at BNL). The client libraries must be in the path of the athena job
  - The following line must be added to the jobOptions to use FroNTier Squid for release 15.3.X (other releases should be fine):

```
DBReplicaSvc.UseCOOLFrontier=True
```

# Defining the Locations in Frtr/Squid

- For releases 15.4.0 (and higher), if you set (or arrange to have set) `$FRONTIER_SERVER`, you will automatically use Frontier without the need for further configuration at the job options level.
- Prior to release 15.4.0, set `$FRONTIER_SERVERi` and `$FRONTIER_PROXYi`, if using environmental variables to specify the FroNTier server and Squid cache. Otherwise you must provide a modified `dblookup.xml` file in `$CORAL_DBLOOKUP_PATH`. See the next slide for an example of the syntax needed needed to specify the use of FroNTier/Squid in `dblookup.xml`.
- There is an idea under discussion involving AtlasSite for setting these variables for each site.
  - Tricky because ATLAS/CMS use different FroNTier servers.

# Using the BNL Squid

- Even if a site does not install a local Squid cache, a significant improvement in the conditions data read time can be made by using the Squid at BNL to access FroNTier at BNL.
  - As an added bonus using the BNL Squid will also reduce the load on the BNL Oracle.
- To use the BNL Squid (releases before 15.4.0), provide lines like below in the `dblookup.xml` (or use part of it and set environment variables):

```
<service name="frontier://BNL/(proxyurl=http://  
squid-frontier.usatlas.bnl.gov:23128)  
(serverurl=http://squid-frontier.usatlas.bnl.gov:  
28080/frontieratbnl) (retrieve-ziplevel=5)/  
ATLAS_COOLONL_INDET" accessMode="read"  
authentication="password" />
```

# Issue 2: POOL Conditions Files

- Many necessary conditions POOL files are not installed with the software release (athena).
  - Athena uses an XML file to find the location of the POOL Conditions files on the local system.
    - If the files are missing or the xml file is missing, the job fails.
  - These files mainly contain calibration information that is “too large” to fit into the Oracle database.
    - Last year (2008) produced approximately 100 GB of such information.
    - Each year ATLAS will produce another ~500 GB of information.
    - This is far too large to be included in the Athena distribution kit.
- Sites need to download the needed (or all) of the POOL conditions files and generate the XML file specifying their locations using dq2-ls.
  - Thanks to work by Mario Lassnig, it is now possible to generate an XML file even at a “small” standalone site.

# Getting the Conditions DB COOL Files

- Getting the POOL conditions files for 2008:
  - For a site in the ToA with storage tokens known to DQ2, subscribe to 5 datasets using the to the dataset patterns using the `dq2-register-subscription`:
    - `cmccond.*`
    - `comcond.*`
    - `oflcond.*`
    - `cond08_mc.*`
    - `cond08_data.*`
  - For a Tier 3 or laptop which is not in the ToA and not known to DQ2, use `dq2-get -L ROAMING` to download all or the needed part of the datasets matching the above 5 patterns.
- From now on there will be only the two obvious patterns per year (`condYY_mc.*` and `condYY_data.*`).

# XML Catalog of POOL Conditions Files

- An XML catalog of the POOL conditions files is needed.
  - One entry is required per POOL file.
- For a site registered in DQ2, use “`dq2-ls -L <storage token> -P`” repeatedly to generate a `PoolFileCatalog.xml` file for the datasets matching the 5 patterns of the previous page (eg):

```
dq2-ls -L MWT2_DATADISK -P cmccond.*
dq2-ls -L MWT2_DATADISK -P comcond.*
...
```
- For a standalone site, there is a new `dq2-ls` switch (`-T`) that builds `PoolFileCatalog.xml` from a local directory tree (eg):

```
dq2-ls -L ROAMING -P -T /d02/conditions
```

# Using the PoolFileCatalog.xml File

- Athena expects two input files:

`PoolFileCat_comcond.xml`

`PoolFileCat_oflcond.xml`

These files should be in a subdirectory called `poolcond/` under the `run/` directory of the job.

It is simplest to use `PoolFileCatalog.xml` as the content of both files rather than try to edit it to split the file into Monte Carlo and data files. Trust me here!

- It may be necessary to modify the xml file content to correctly reference the local file system. For MWT2, I had to switch from a srm specification to a dCache:

`srm://uct2-dc1.uchicago.edu:8443/srm/managerv2?SFN=/pnfs/uchicago.edu/atlasdatadisk/...`

To:

`dcache:/pnfs/uchicago.edu/atlasdatadisk/...`

# Extra Issue/Idea: pcache

- Charles Waldman has written a script (pcache) that caches files are frequently fetched by Panda jobs.
  - The files are cached in a fixed size scratch area on the worker servers. When the scratch directory is full, a FIFO algorithm is used to clean up.
  - When Charles studied the MWT2 site in February, he found that various versions of the ATLAS sqlite conditions database were the most used files.
- The same sort of efficiency can be achieved when processing a run with thousands of datafiles, using calibrations conditions a small number of POOL files.
- Using pcache or something similar seems to be required to prevent a “hot files” problem with the conditions files.



# Links

- Setting up a Squid cache at your site:

<http://www.usatlas.bnl.gov/twiki/bin/view/Admins/SquidTier2>

- Discussion of remote access to conditions data:

<https://twiki.cern.ch/twiki/bin/view/Atlas/RemoteConditionsDataAccess>

# Experts on Conditions Data

- The FroNTier experts are Dave Dykstra (FNAL/CMS), Carlos Gamboa (BNL), & John De Stefano (BNL).
  - Additional work on cache consistency by David Front (Weizmann) and Sasha Vanichine (ANL)
  - FroNTier/Squid integration into Athena by Elizabeth Gallas (Oxford) and Richard Hawkings (CERN)
  - dq2-ls -T option by Mario Lassnig (Innsbruck)
  - pcache provided by Charles “the bicyclist” Waldman (Chicago)
- All of the above people have been a significant help to me and provided pieces of the puzzle.