



First Use of LHC Run 3 Conditions Database Infrastructure for Auxiliary Data Files in ATLAS

L Aperio Bella¹, D Barberis², W Buttinger¹, A Formica³, E J Gallas⁴, G Rybkin⁵, L Rinaldi⁶
on behalf of the ATLAS collaboration

¹CERN, ²Dipartimento di Fisica, Università di Genova, and INFN Sezione di Genova, Via Dodecaneso 33, I-16146 Genova, Italy, ³CEA/Saclay IRFU/SEDI, 91191 Gif-sur-Yvette, France, ⁴Department of Physics, Oxford University, Denys Wilkinson Building, Keble Road, Oxford OX1 3RH, United Kingdom, ⁵Laboratoire de l'Accélérateur Linéaire, Université Paris-Sud, CNRS/IN2P3, Orsay, France, ⁶Dipartimento di Fisica, Università di Bologna, and INFN Sezione di Bologna, Via Irnerio 46, I-40126 Bologna, Italy

Abstract

Processing of the large amount of data produced by the ATLAS experiment requires fast and reliable access to what we call Auxiliary Data Files (ADF). In ATLAS this data has, thus far for historical reasons, been collected and accessed outside the ATLAS Conditions Database infrastructure and related software. For this reason, along with the fact that ADF are effectively read by the software as binary objects, this class of data appears ideal for testing the proposed Run 3 Conditions data infrastructure now in development.

Usage of Auxiliary Data Files in ATLAS

- Consist of a variety of calibrations, alignments, efficiencies, weights and other useful constants, produced by experts using the physics data, essential for user analysis
- Currently stored in a simple file system structure under an *AFS* dedicated area, called *Calibration Area*, at CERN and then propagated into the *CVMFS* storage area, accessible from external sites via Squid Proxy, and into the High Level Trigger (*HLT*) farm, located near the ATLAS experiment, for on-line processing

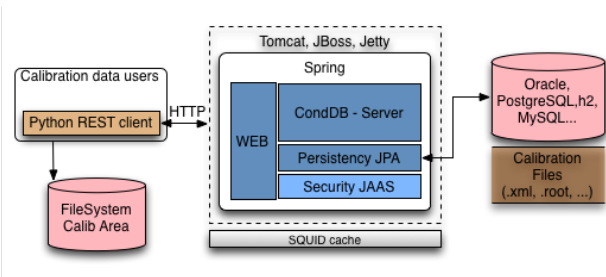
The Calibration Area on AFS The total volume is 2 GB of data, $\approx 0.2\%$ of data within the Conditions Database infrastructure.

- Files are not updated very frequently, and in general their validity span a large range in time, this IOV structure is taken into account adding the time information in the name of the sub-directories inside a given package when needed, or at the level of the file names
- Every system has defined its own way to handle the internal file dependencies for a given package. The file type used is extremely heterogeneous: ASCII files (XML or TXT) and ROOT files

Present Limitations The Calibration Area today:

- ▶ Directory structure determined essentially by the package experts
- ▶ Difficult to handle centrally and time-consuming to create a given calibration area release
- ▶ Every system has adopted some model for tagging and versioning
- ▶ Needs to be exported to Point 1 — ATLAS detector and trigger computing farm location

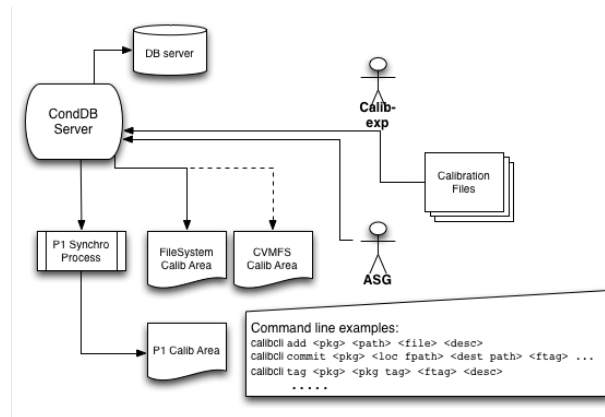
Managing the Auxiliary Data Files using the New LHC Run 3 Conditions Server



- Store the analysis ADF in the conditions database as any other conditions
- Dump a directory structure in CVMFS in an automatized way in order to keep the file access as it is today

Prototype to Manage Auxiliary Data Files in the Conditions Database

Verify the possibility to map the existing directory structure in the calibration area into the TAG and IOV based structure of the conditions data model — every file for a given calibration package in a TAG in Oracle DB



Python Client for Analysis ADF Developed CLI tools to manage the data via the REST API

- 1) Add a file for a given software package and in a given path
- 2) Commit a local file into the conditions database
- 3) Associate a tag to a given package, referencing all files uploaded for that package

From present CalibArea to new CondDB framework

From present CalibArea to new CondDB framework

Repository path (AFS and CVMFS)

```

|-- PackageName_AA
|-- PackagePath_AA
|-- PackageVersionXX
|-- FileName_1.conf
|-- FileName_2.root
|-- PackageVersionXY
|-- FileName_1.conf
|-- FileName_2.root
|-- FileName_3.conf->Link
|-- PackageName_BB
|-- PackagePath_BB
|-- PackageVersionYY
|-- FileName_1.conf
|-- FileName_2.root
|-- FileName_3.root
|-- PackageVersionYZ
|-- FileName_1.conf
|-- FileName_2.conf
|-- FileName_3.root
|-- FileName_4.conf

```

CondDB new framework

```

|-- ASG
|-- ASG-XX-YY-Z1
|-- PackageTag-XX-YY-Z1-> ../../
PackageName_AA/PackageTag-XX-YY-Z1
|-- ASG-XX-YY-Z2
|-- PackageTag-XX-YY-Z2-> ../../
PackageName_AA/PackageTag-XX-YY-Z2
|-- PackageTag-XX-YY-Z1
|-- FileTag-XX
|-- FileName_1
|-- IOV.conf
|-- FileName_2
|-- IOV.root
|-- PackageTag-XX-YY-Z2
|-- FileTag-XX
|-- FileName_1
|-- IOV.conf
|-- FileName_2
|-- IOV.root
|-- FileTag-XY
|-- FilePath
|-- FileTag-YY
|-- FileTag-YY
|-- IOV.conf

```

GlobalTag

- Global tag: managed by ASG
 - ASG can dump the DB (export to cvmfs, HLT farm)
- path re-shuffled
- links are kept
- a PackageTag may contain several FileTags