



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 [1] requires fast and reliable access to what we call Auxiliary Data Files (ADF). These files, produced by Combined Performance, Trigger and Physics groups, contain conditions, calibrations, and other derived data used by the ATLAS software. 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. This paper will describe this implementation as well as the lessons learned in exploring and refining the new infrastructure with the potential for deployment during Run 2.

Usage of Auxiliary Data Files in ATLAS

The Auxiliary Data Files consist of a variety of calibrations, alignments, efficiencies, weights and other useful constants, which are produced by experts using the physics data and are essential for user analysis in the context of ATLAS distributed computing infrastructure. These files are currently stored in a simple file system structure under an *AFS* (Andrew File System) dedicated area, that we call *Calibration Area*, at CERN and then propagated into the *CVMFS* (CERN Virtual Machine File System) storage area, which is 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 of the calibration area is 2 GB of data. This represents $\approx 0.2\%$ of data accessed via the Conditions Database infrastructure. The calibration area files are not updated very frequently, and in general their validity span a large range in time. For the moment 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 even 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 can be used by the experts depending on their needs.

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 by the ASG (Analysis Software Group) manager
- 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

A new system would make the file storage more uniform among different packages: provide a central way for tagging and versioning the files, simplify the creation of a calibration area release, and facilitate on-line synchronisation. The ATLAS and CMS experiments are exploring a common solution to manage conditions data with the time scale of Run 3 (2020).

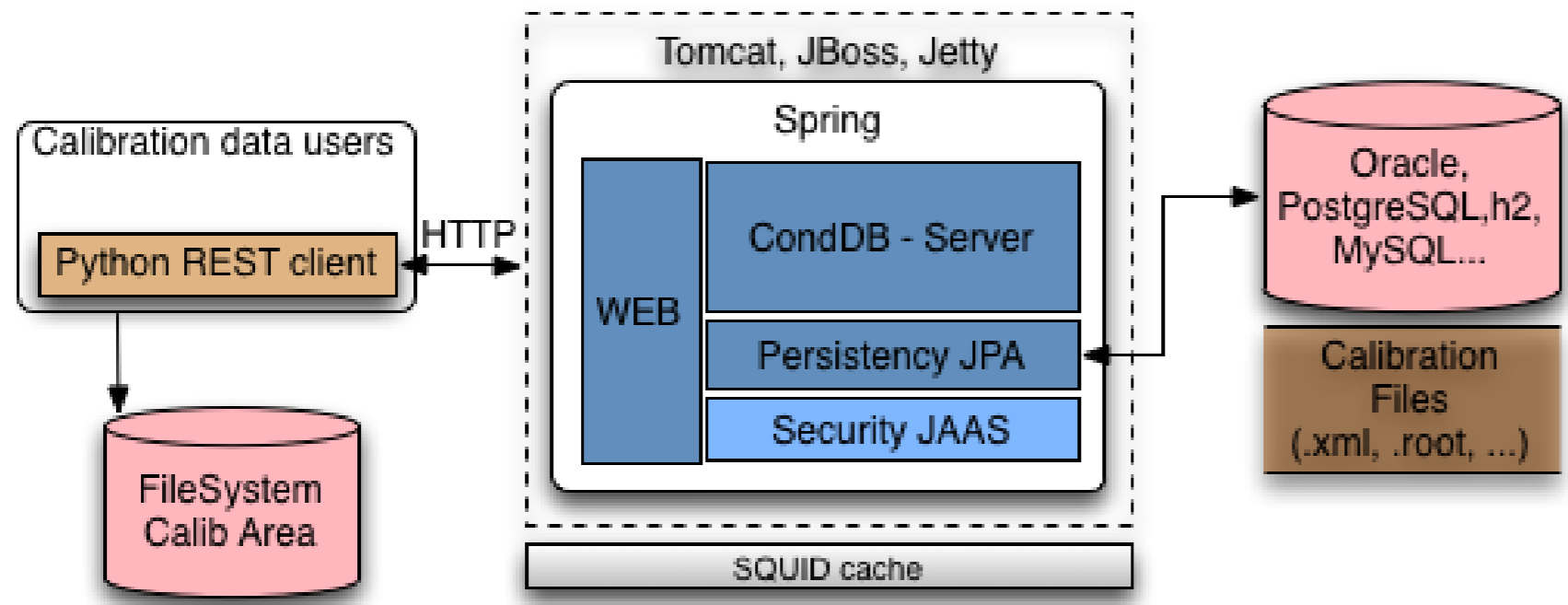
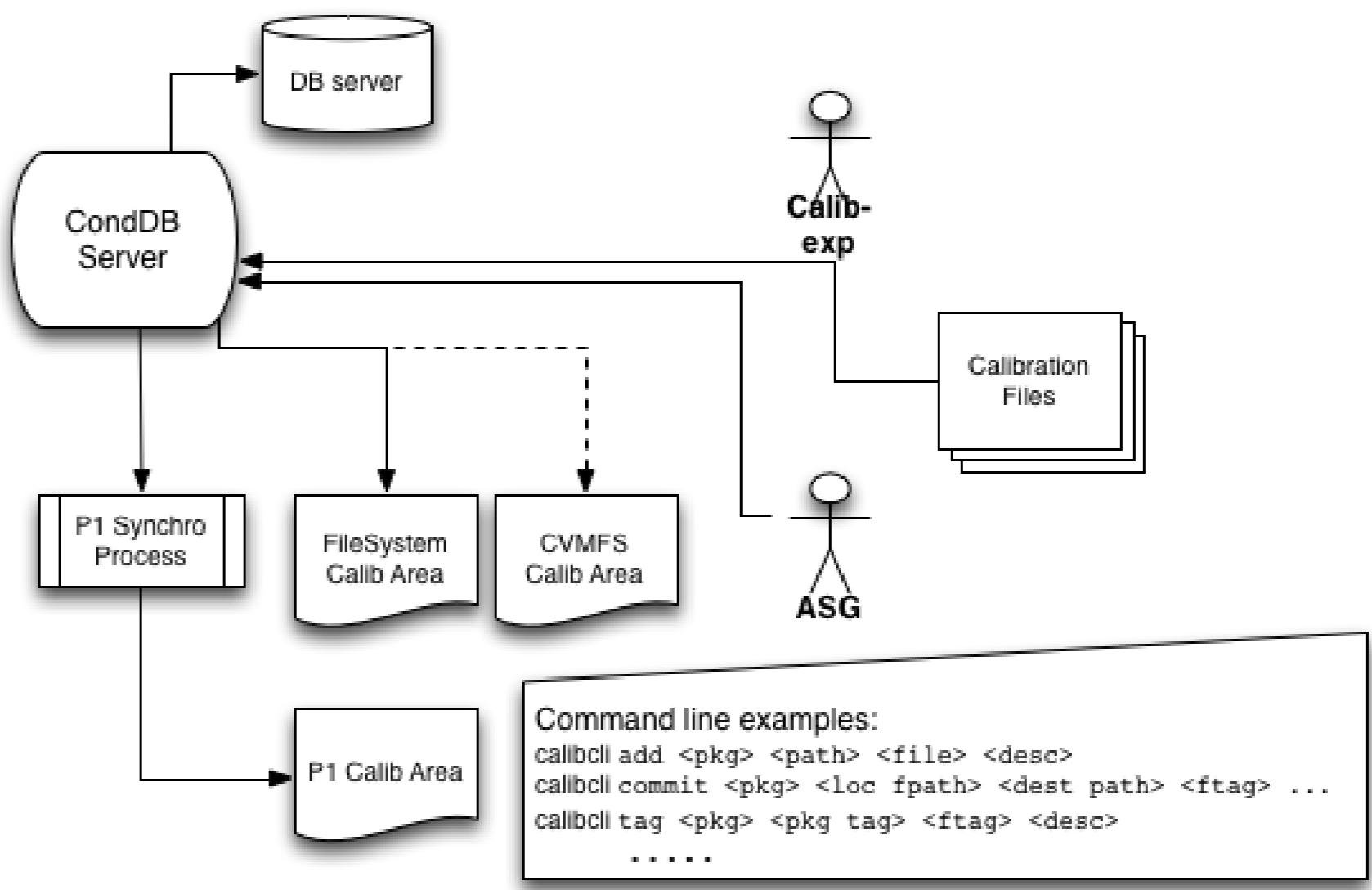


Figure: Run 3 conditions infrastructure.

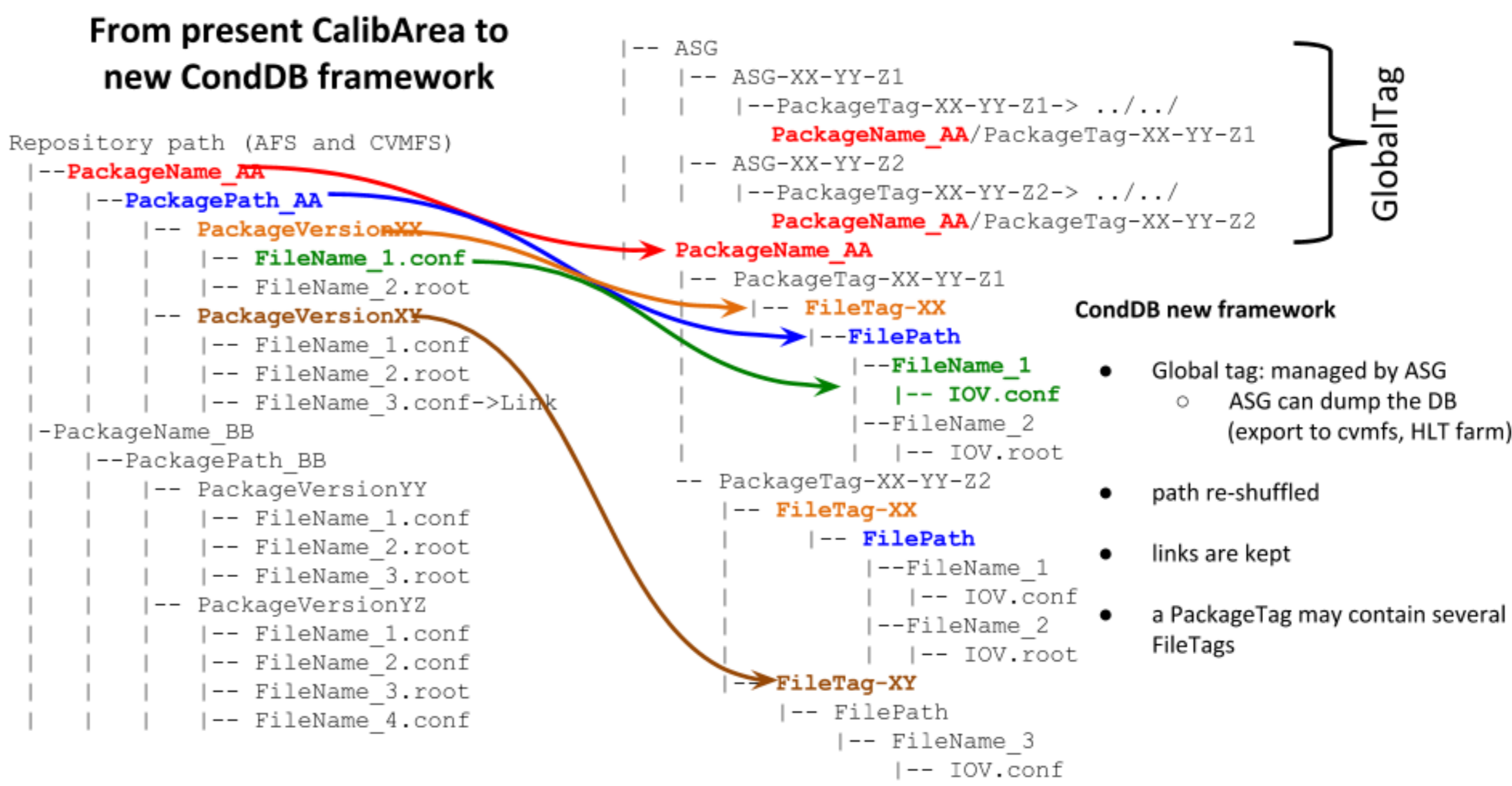
This solution foresees 3 main components: a back-end to persist the conditions data objects and the related meta-data like interval ranges, tags and global tags (for the moment based on Oracle), an intermediate web server providing access to the storage layer and delivering REST functions to clients, and finally a set of client libraries (Python and C++) in order to use the REST API from the data processing frameworks. The data model at the level of Oracle consists today of a simple set of tables to handle the meta-data, and all conditions data objects are stored as files (BLOBs in the database language). A more detailed description can be found in the CHEP2015 article [2]. This data model seems well adapted to contain the ADF. The proposed solution is to store the analysis ADF in the conditions database as any other conditions, and then to 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

For this prototype we have been gathering a subset of the ADF to 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. Some additional tables were added in order to store in the database the complete URI of the files uploaded. The strategy adopted in this first prototype was to map every file for a given calibration package in a TAG inside Oracle. Other possible mappings will be explored in future, to finally decide what is the optimal way of storing the data with the minimal amount of changes to be performed at the level of the software producing and or accessing these data.



- 1) **calibcliadd** < *package* > < *path* > < *file* > < *description* >
Add a file for a given software package and in a given path. The path will be used later to migrate the file under AFS. This action does not upload data, but it is used only to create the relevant meta-data for later management of the specified auxiliary data file.
- 2) **calibclicommit** < *package* > < *localfilepath* > < *destpath* > < *filetagextension* > < *destfile* >
Commit a local file into the conditions database. The file will be uploaded in this case, and stored in a tag using by default iov 0.
- 3) **calibclitag** < *package* > < *packagetag* > < *filetagextension* > < *description* >
This will allow to associate a tag to a given package. This tag will reference all files uploaded for that package.



Summary

We have presented a proposal for investigating the management of ADF using the new Conditions Database architecture. A command line client to import the data into the new system was developed and query commands to navigate the data files will be added. This conditions area proved well suited for the new architecture. We are looking for other conditions areas that may immediately benefit from this architecture that we hope to implement with the CMS experiment for the LHC Run 3, planned for year 2020.

References

1. ATLAS experiment <http://cern.ch/atlas>
2. D Barberis et al. 2015 Designing a future Conditions Database based on LHC experience *J. Phys.: Conf. Series* **664** 042015