Tile-in-ONE: A web platform which integrates Tile Calorimeter data quality and calibration assessment

A. Sivolella¹, F. Ferreira¹, C. Maidantchik¹, C. Solans², A. Solodkov², B. Burghgrave³ & Y. Smirnov³

on Behalf of ATLAS Tile Calorimeter System

¹Univ. Federal do Rio de Janeiro, ²CERN, ³Northern Illinois University

E-mail: tile-in-one@cern.ch

Abstract. The ATLAS Tile Calorimeter collaboration assesses the quality of calibration data in order to ensure its proper operation. A number of tasks is then performed by executing several tools and accessing web systems, which were independently developed to meet distinct collaboration’s requirements and do not necessarily are connected with each other. Thus, to attend the collaboration needs, several programs are usually implemented without a global perspective of the detector, requiring basic software features. In addition, functionalities may overlap in their objectives and frequently replicate resources retrieval mechanisms.

Tile-in-ONE is a designed and implemented platform that assembles various web systems used by the calorimeter community through a single framework and a standard technology. It provides an infrastructure to support the code implementation, avoiding duplication of work while integrating with an overall view of the detector status. Database connectors smooth the process of information access since developers do not need to be aware of where records are placed and how to extract them. Within the environment, a dashboard stands for a particular Tile operation aspect and gets together plug-ins, i.e. software components that add specific features to an existing application. A server contains the platform core, which represents the basic environment to deal with the configuration, manage user settings and load plug-ins at runtime. A web middleware assists users to develop their own plug-ins, perform tests and integrate them into the platform as a whole. Backends are employed to allow that any type of application is interpreted and displayed in a uniform way. This paper describes Tile-in-ONE web platform.

1. The ATLAS Tile Calorimeter

ATLAS is a general purpose detector at one of four collision sites at the Large Hadron Collider (LHC) at CERN [1]. The Tile calorimeter (TileCal) is the hadronic calorimeter in the central region of the detector, a sampling calorimeter made of alternating pieces of plastic scintillator and iron. Photo-multipliers (PMTs) are used as photo-detectors[2]. TileCal has one long barrel (split in two sides, A and C) and two extended barrels hadron parts, totalizing four partitions (as illustrated in Figure 1). Each detector cylinder is built of 64 independent wedges, also called modules. There is up to 45 electronic channels (each channel associated to one PMT) in each module, which are fed by high voltage sources (HV). Almost 10,000 channels need to be monitored in TileCal analyses. Identifying problematic PMTs during data quality analysis and keeping the power sources that feed the electronic channels stable are two requirements examples on the calorimeter operation, otherwise it can interfere negatively on data...
acquisition [3].

![Figure 1](image)

**Figure 1.** Layout of the ATLAS Tile Calorimeter (in green, with its four read-out partitions) surrounding the liquid argon electromagnetic calorimeter. The ATLAS muon chambers and the toroidal magnetic systems surrounding TileCal, are not shown.

ATLAS experiment stores all detector conditions data in a unique repository, the ATLAS Conditional database (COOL DB) [4]. TileCal conditions should be constantly updated in the COOL DB in order to keep the collaboration informed about the calorimeter status. Besides that, all COOL DB stored records are used to configure data reconstruction and to perform analyses. The database maintenance requires technical knowledge by the researchers, such as SQL language, SQLite files, COOL DB specific commands and structure.

TileCal community also uses a software tool for calibration purposes, named TUCS (TileCal Universal Calibration Software) [5], which intends to access COOL DB data and build plots in a standard way. TUCS consists of a package of scripts and also requires knowledge about computing and specific settings by the researchers. In addition, the huge amount of data spread in different locations makes the data quality analyses difficult not only to new collaborators.

The members of the Tile calorimeter system are responsible for the operation and the quality of the data by means of several tools and web interfaces that have been developed throughout the different phases of the experiment. Due to the diversity of the groups involved in each task, most of these tools were developed independently, without following the same guideline. Most of these tools lack basic features and are implemented without a global perspective of the detector. In many cases, they overlap in objectives and resources with another one. Another common situation is that in order to perform a single task, the collaborator must navigate through different tools, databases and software, which is time consuming and susceptible to mistakes. This brings an evident necessity of an infrastructure to allow the implementation of any functionality without having to duplicate the effort while being possible to integrate with an overall view of the detector status. Tile-in-ONE web platform is intended to meet these needs, by providing a unified interface, which integrates all the web systems and tools used by the Tile calorimeter, with a standard development technology and documentation, allowing the researchers to focus on the analysis.

### 2. From several web systems to the Tile-in-ONE integrated web platform

Several web systems were developed during different detector phases, each one complying with a specific goal [6]. The data acquisition and storage is performed by ATLAS Trigger and Data AcQuisition (TDAQ) systems [7] and reconstructed with the Athena ATLAS offline software [8]. As soon as data is
reconstructed, the Data Quality Monitoring Framework (DQMF) [9] is applied and generates automatic status flags that certify the quality of the data, reducing the number of channels that should be analyzed. At this stage, plots that are built during the data reconstruction help the Data Quality (DQ) offline analysis. A web system was developed (Figure 2) to integrate graphical results into a single interface, guiding the data quality group to perform the offline analysis. This illustrates three systems linked by a single web system: figure 2(a) lists reconstructed runs and available produced plots; figure 2(b) provides detailed information about a single module from a given run, accessed from previous illustration. From this interface, it is also possible to insert comments concerning detector performance; and 2(c) illustrates some plots that were built during reconstruction and are available on Dashboard Web System.

Figure 2. Dashboard Web System, developed in 2010 to integrate all the generated TileCal DQ assessments into a single web system.

By historical reference, a list of known damaged channels is also called by the collaboration as the Bad Channels List, which is also stored in the COOL DB. The MCWS (Monitoring Calibration Web System) and DCS (Detector Control System) systems support the DQ group analyzing up to 10,000 TileCal PMTs. MCWS was created to support the channels analysis, providing the Bad Channels List and calibration constant values through the web. The DCS Web System was developed to monitor the high voltage that feed the PMTs, allowing the researchers to access quickly the detector control parameters.

It is clear in the presented scenario that several (web) tools support different aspects of the TileCal operation. The software applications were implemented in various stages of the detector test beam, commissioning and collisions data taking, involved inby distinct collaborators, who not necessarily are still involved with calorimeter activities. Therefore, the tools maintenance is not fully assured by the
same developers. It would be helpful if all available tools were unified and placed into a singular web site, making use of the same technology to facilitate reuse, eventual repairs and support extensions and enhancements. These requirements are fulfilled by the Tile-in-ONE web platform, described in the following sections.

3. Tile-in-ONE Dataflow
Tile-in-ONE web platform was designed to integrate several data analyses performed by TileCal community. It unifies different tools into a single interface, providing a framework where users can develop source code straight on the web. This guarantees the usage of the same technologies, which makes the platform maintenance easier. Moreover, it supports users to reuse source code by making all implementations available for every single user.

Figure 3 illustrates the platform data flow. Tile-in-ONE platform provides a code development functionality, known as Playground where the user can write down source codes. It is also possible to load previous codes or codes produced by other users. Once the program is finished, the user can submit it to be handled on the server side. The server renders the source code to other machines, so called slave machines in this context, depending on the data source identified in the code by the server. Each slave machine covers all needed settings to access specific repositories that are used by TileCal, in a way that, users can avoid knowing the required configurations to retrieve database records. The main gain about executing the source code in other machine rather than in the server itself is the guarantee that the source code execution is not going to overload the server or break it. After the chosen slave machine executes the submitted source code, it returns a dictionary containing produced variables, that are going to be used to build graphical outputs for the end users. At this point, the user can build HTML objects, feeding them with the result produced with the source code submission. This graphical output can now be encapsulated as a plug-in. Several plug-ins are configured to compose dashboards, according to the end users and specific groups needs. If the data presented by a plug-in is not enough, the plug-in can be expanded in order to provide further information. This feature allows the installation of entire existing systems inside the Tile-in-ONE web platform.

Figure 3. Tile-in-ONE web platform dataflow. From source code implementation until entire embed systems.

4. Tile-in-ONE Infrastructure
Tile-in-ONE web platform is composed by two distinct environments, although fully compatible. The first one was set for testing new features and enriching existing ones. The second one was established for production purposes. Each environment has four slave machines. In this context, each slave machine instance can be assigned its own list of repositories, which allows slaves to work independently.
Moreover, if the setting changes over the time, it can be easily updated as it is centralized in a specific place for different web systems. Figure 4 illustrates the source code execution by slave machines: the source code is written on the client side. When submitting the code, a request is sent to the server. On the server side a slave machine is chosen according with the request content. After executing the received source code, the slave machine sends a response back to the main server, which will render it back to the client side.

![Figure 4. Infrastructure schema. Source codes are executed on slave machines to avoid server overload.](image)

By now, there are four slave machines at the moment: three configured to access or use specific TileCal data sources and tools, and one to execute generic source codes in case the main server does not identify either any data source request or more than one repository call. In total, there are five machines available for the testing environment (one main server and four slave machines) and five machines for the production environment. These machines are all virtual machines under CERN IT Cloud Infrastructure Service responsibility running Scientific Linux 6.

Figure 5 summarizes the existing slave machines and how they are configured. The cooldb slave machine is set to access the ATLAS Condition Database. It is automatically set to use latest athena release, so the users can prescind about it before executing a source code which accesses the detector conditions data. The tucs slave machine is pointing to the SVN repository that stores TUCS, the calibration group package used to generate plots. Tile-in-ONE tucs machine uses a version installed under a common user (tilebeam) at lxplus machine. TUCS package is based on Python language and uses packages as matplotlib and numpy. The oracle slave machine is configured to access TileCal oracle database, which stores information about reconstructed data. Finally, general slave machine has all the other machines settings and can also handle ROOT files available on TileCal EOS client. The need to have all previous settings is in case the main server identifies more than one data source request in the same submitted code.

![Figure 5. Slave machines and brief description of how they are set.](image)

5. Achieved Results
Currently, Tile-in-ONE web platform is supporting the calibration and the data quality groups by providing plug-ins within specific dashboards. In this section, three dashboards are illustrated.
5.1. Data Quality Dashboard

The Data Quality dashboard gathers plug-ins that support the main DQ group analysis. Two plug-ins are described.

(i) Channels masked on UPD4 but not on UPD1. This plug-in (figure 6) accesses COOL DB and retrieves two Bad Channels Lists: one list corresponds to the channels that were masked during data reconstruction phase (also known as UPD4) and the second one corresponds to the channels that were masked during data taking phase (also known as UPD1). Both lists should be synchronized. The main goal of this plug-in is to easily show the differences between the retrieved lists. In the case illustrated by figure 6(a), there is no channel masked in UPD4 that is not masked in UPD1.

(b) Expanded plug-in shows the Bad Channels Lists in a graphical way.

Figure 6. Channels masked on UPD4 but not on UPD1 plug-in.

(ii) List of reconstructed runs. The List of Reconstructed Runs plug-in displays in a table format the runs reconstructed in the past 60 days. This plug-in guides the data quality group to perform offline analysis as it presents the data generated during the reconstruction as automatic status flags (from DQMF) and generated plots. Through expanded plug-ins, the user can even insert comments concerning the analyzed run modules. Figure 7(a) illustrates this plug-in and its expansions.
this example there is more than one expanded plug-in interface. It means that the entire system is embedded in Tile-in-ONE web platform. From plots icon on Figure 7(b), it is possible to access 

*TileComm Analysis* expansion (figure 7(c)), where the Data Quality user can access generated plots and write down a few comments.

(a) List of reconstructed runs during the past 60 days plug-in.

(b) Entire web system embedded on Tile-in-ONE platform. This is (c) *TileComm Analysis*. Presents information about all modules for a given run number. Accessed from WIS interface, through plots icon.

**Figure 7. Dashboard Web System** integrated in Tile-in-ONE platform.

5.2. Calibration Dashboard
The Calibration Dashboard displays plug-ins that support the calibration group analysis. So far, there are nine plug-ins in this dashboard, and all of them profits from TUCS package, which is installed within the Tile-in-ONE web platform. Figure 8 illustrates the COOL Status History plug-in that shows the change in the number of masked channels and the number of different calibration flags in the COOL status database from a date range.

6. Conclusions
The Tile Calorimeter of the ATLAS detector gathers a group of diverse Web systems to support different aspects of its operation. The softwares were developed in several phases of the experiment by various members who made use of different technologies and methods. Most of these systems access repositories, perform analysis algorithms and present the results through graphical outputs. Therefore, code replication and difficult maintenance are common difficulties that the collaboration faces.
The Tile-in-ONE Web Platform is a web based platform which provides a unified interface, which integrates systems and tools used by the Tile calorimeter, with a standard development technology and access to common services. A straight look at the Tile-in-One Server reveals four different components. The Dashboard component is responsible for gathering a set of applications, known as Plug-ins. Those plug-ins add specific features to Tile-in-ONE Web Client, allowing customization and integrating distinct analyses to the platform. The Playground corresponds to a restricted area, where collaborators can develop those plug-ins. The Jobs component provides control over the plugin processing at slave machines. The governance between those components, including user privileges and access control, is supported by the Setup component.

Tile-in-ONE is currently composed by the Calibration and Data Quality Dashboards, gathering 15 plug-ins that promotes collaboration among the TileCal members, supports code reuse and minimize maintenance efforts.

7. References
[1] The ATLAS Collaboration. The atlas experiment at the cern large hadron collider. *Journal of Instrumentation*, 3(08):S08003, 2008.
[2] ATLAS Tile Calorimeter Collaboration. Tile Calorimeter Technical Design Report, 1996. CERN/LHCC 96–42.
[3] Joao Antonio Tomasio Pina. The tilecal/atlas community detector control system. *51st Nuclear Science Symposium and Medical Imaging Conference*, 2:1201–1204 p.p., 2004. on Behalf of TileCal Community.
[4] M. Verducci. Atlas conditions database experience with the lcg cool conditions database project. *Journal of Physics: Conference Series (JPCS)*, 119 - part 4, 2008.
[5] TileCal Universal Calibration Software (TUCS). Accessed on May, 2015. https://twiki.cern.ch/twiki/bin/viewauth/Atlas/TileCalibrationTucs.
[6] C. Maidantchik; F. Ferreira; F. Grael; A. Sivolella; L. Balabram. Web system for data quality assessment of tile calorimeter during the atlas operation. *Journal of Physics: Conference Series (JPCS)*, 331 - part 4, 2011.
[7] ATLAS HLT/DAQ/DCS Group. ATLAS High-Level Trigger, Data Acquisition and Controls Technical Design Report, 2003. ATLAS TDR-016.
[8] P. Calafiura and W. Lavrijsen and others. The athena control framework in production, new developments and lessons learned. *Journal of Physics: Conference Series*, pages 456–458, 2004.
[9] A. Corso-Radu; H. Hadavand; M. Hauschild; R. Kehoe; S. Kolos. Data quality monitoring framework for the atlas experiment at the lhc. *IEEE Xplore Digital Library*, 55, 2008.