CONDITIONS DATABASE AND CALIBRATION SOFTWARE FRAMEWORK FOR ATLAS MONITORED DRIFT TUBE CHAMBERS

M.Verducci, CERN, Geneva 1211, Switzerland
M.Cirilli, University of Michigan, Ann Arbor, MI 48109, USA
O.Kortner, Max-Planck-Institut für Physik, München 80805, Germany
D.Orestano, F.Petrucci, Università di Roma Tre and INFN Roma III, Roma 00146, Italy
J.Rothberg, University of Washington, Seattle, WA 98195 , USA
N.van Eldik, Z.van Kesteren, NIKHEF, Amsterdam 1098, The Netherlands
M.Woudstra, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

Abstract

The size and complexity of LHC experiments raise unprecedented challenges not only in terms of detector design, construction and operation, but also in terms of software models and data persistency. One of the more challenging tasks is the calibration of the 375000 Monitored Drift Tubes that will be used as precision tracking detectors in the Muon Spectrometer of the ATLAS experiment. An accurate knowledge of the space-time relation is needed to reach the design average resolution of 80 microns.

The MDT calibration software has been designed to extract the space-time relation from the data themselves, through the so-called auto-calibration procedure, to store and retrieve the relevant information from the conditions database, and to properly apply it to calibrate the hits to be used by the reconstruction algorithms, taking into account corrections for known effects like temperature and magnetic field.

INTRODUCTION

High pressure monitored drift tube (MDT) [1] chambers will be used as precision tracking detectors in the Muon Spectrometer of the ATLAS experiment at the Large Hadron Collider (LHC) at CERN [2]. An accurate knowledge of the space-time relation is needed to reach the design average resolution of 80 microns.

This document describes the software developed both to extract the calibration from the data (calibration framework) and to properly apply it to calibrate the hits to be used by the reconstruction algorithms (calibration service). It also provides an overview of the database structure that must be implemented to handle the calibration data.

The MDT calibration procedures, and in particular the possibility of autocalibration, have been extensively studied within the ATLAS muon community [3]. The experience gained from the analysis of test beam data, simulation and cosmic rays has been used to develop the software running on hits associated to selected muon track segments. The autocalibration requires multiple iterations over the same set of data, that are performed within a dedicated C++ framework interfaced to the ATLAS offline framework (ATHENA). The framework also allows the application of the autocalibration procedure separately to different regions (calibration regions) of the MDT spectrometer and provides some flexibility in the choice of the calibration regions. Flexibility is needed since size and number of the calibration regions are not yet defined.

This software structure is being exercised both on test beam and GEANT4 simulated data. Validation of the full calibration chain is based on the analysis of the tracking residuals and pull distributions. On simulation, direct comparison of the calibration results to the parameters entering in the digitization is also possible. Current plans are to apply the calibration chain to the cosmic ray events acquired during the detector commissioning in the cavern and to perform a calibration challenge on simulated data. The latter is aimed mainly at validating the handling of a realistic number of calibration regions and the organization of the calibration data in the database.

CALIBRATION MODEL

The basic response of an MDT detector is the threshold crossing time (tdc time) and the collected charge. Additional data necessary for MDT calibration are the trigger time, the bunch crossing identification and the measurement of the coordinate along the wire (provided by the trigger chambers) and the measurement of the temperature, of the magnetic field and of the gas composition, provided by the Detector Control System (DCS). To reconstruct tracks in MDT chambers we need to compute the drift time, to obtain the impact parameter applying the space-time relation and then to fit the track to the drift circles.

The calibration software is responsible of computing the required quantities and provide them to the reconstruction.

The first step is the computation of the drift time, t. It is obtained, as in (1), from the measured raw tdc counts, converted into a time, after subtraction of a tube characteristic constant, called t0. The corrected drift time, t_{drift}, is obtained correcting t for the various effects listed in Table 1.

$$\begin{aligned} t &= t_{TDC}(ns) - t0 \\ t_{drift} &= t + \Sigma_i \delta t_i(t) \end{aligned} \qquad (1)$$

Another set of corrections, listed in Table 2, accounts for the effect of the different environmental parameters: the r-t relation is defined at a nominal value of any given environmental parameter, and the measured drift time will in general be different from the value obtained using this nominal r-t relation. The corrections in Table 2 are applied knowing the parametrization of the dependence of the drift dime upon variations of these parameters. Detailed studies on some aspects of these corrections can be found in [4],[5],[6].

Time of flight	δt_f	Corrected knowing particle trajectory.
Position along the wire	δt_x	Correction of the propagation time along the wire. The second coordinate is provided by the trigger chambers.
Time slewing	δt_q	Correction of the the tdc measurement knowing signal amplitude.

Table 1: Corrections to the measured tdc time.

Temperature	δt_T
B field	δt_B
Background	δt_{bg}
Gas composition	δt_g
Wire sag	δt_s

Table 2: Corrections to the the measured tdc time due to variations in the operating conditions. The correction is based on a parametrization of their effect.

t_{drift} is used as the nominal measured drift time. The quantity t_0 that appears in (1) is the relative delay between the different channels and is computed by fitting the rising edge of the drift time distribution. Fig. 1 shows a typical distribution of the drift times, along with the fit performed using the ad-hoc function in (2) [3].

$$\frac{dN}{dt} = P_1 + \frac{P_2 + P_3 \exp\left(\frac{P_5 - t}{P_4}\right)}{\left[1 + \exp\left(\frac{P_5 - t}{P_7}\right)\right] \cdot \left[1 + \exp\left(\frac{t - P_6}{P_8}\right)\right]} \qquad (2)$$

The second step of the calibration procedure is the calculation of the r-t relation. The ATLAS collaboration has developed a procedure (*autocalibration*) that uses the data themselves to determine the r-t relation, as described for example in [3]. The procedure is based on an iterative fit, which requires some thousands of muon tracks distributed over a wide angular range. The r-t relation is modified until the quality of the track fit is satisfactory.

Various different algorithms are being implemented and at a later stage, after some experience has been gained, the relative quality of the algorithms will be judged and the appropriate choice will be made.

The computed constants have to be stored in the database together with the parametrization used for the corrections. The outcome of the calibration procedure are the t_0 and the average measured charge for each tube and the parametrization of the r-t relation and the resolution for each calibration region.

In principle, the various effects changing the t_0 and the r-t relation could be parametrized, and a single r-t could be used for the whole detector. In practice, the required precision will not probably be achieved in this way as the corrections are not completely known and environmental conditions may not be correctly measured. Therefore we plan to divide the detector in a set of calibration regions such that the effect of the variations of the parameters listed in Table 2 could be neglected within one region. About 10000 regions are foreseen; this number depends on the balance between conflicting requirements: the statistical and systematical error, the time spent to collect the data, the time variation of the environmental conditions, the time required to process the data and the size of the data base.

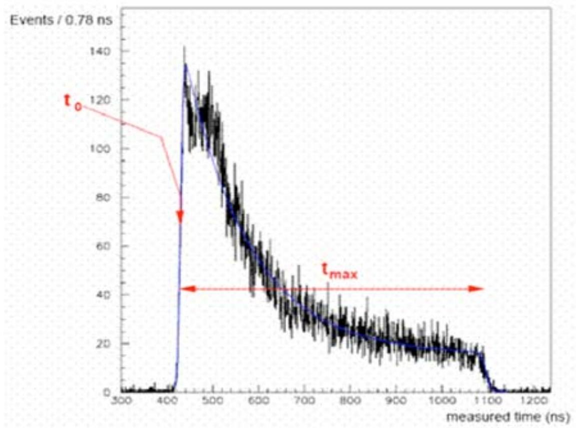

Figure 1: Typical distribution of the drift time (ns). The result of the fit with function (2) and the definitions of t_0 and of the maximum drift time are also shown.

CALIBRATION SOFTWARE

The calibration software has been designed having in mind some basic guidelines. The main issue is the capability to fully exploit the facilities provided by the ATLAS offline framework (ATHENA) in terms of data decoding, pattern recognition, tracking and database access. Computing algorithms must run within ATHENA, but they do not have any dependence on it. In addition, the calibration procedure should be independent from the particular reconstruction implementation, so that all the muon reconstruction packages available in the ATLAS software can be used. Last, we want to be able to easily switch between different calibration algorithms.

The calibration software is composed of 4 different parts, that are explained in detail in the following.

A. Calibration Service

The calibration service has to properly handle the application of all the corrections to the measured drift time, to convert it into a drift radius and an associated error. It provides this information to the reconstruction and to the calibration tasks.

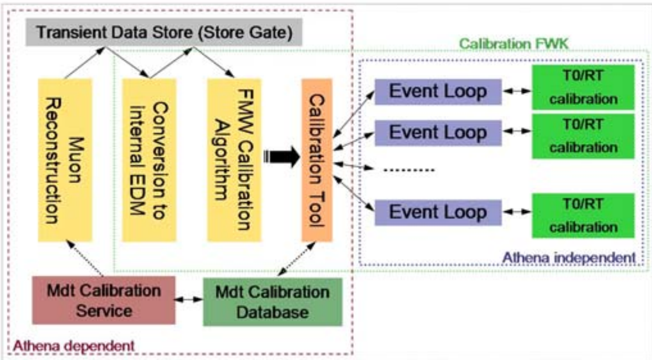

Figure 2: Diagram showing the different parts of the calibration framework and their interplay with the rest of the software.

B. Calibration Event Data Model (EDM)

The Calibration EDM is the collection of all the data objects needed by the software to communicate with its clients and to store information in the intermediate steps.

C. Calibration Framework

The Calibration Framework computes the calibration constants (t_0) and functions (r-t relation, resolution) and stores them into the ATHENA transient data store (StoreGate) and on ASCII files. It is a client of the reconstruction and it is made of different elements as can be seen from Fig. 2.

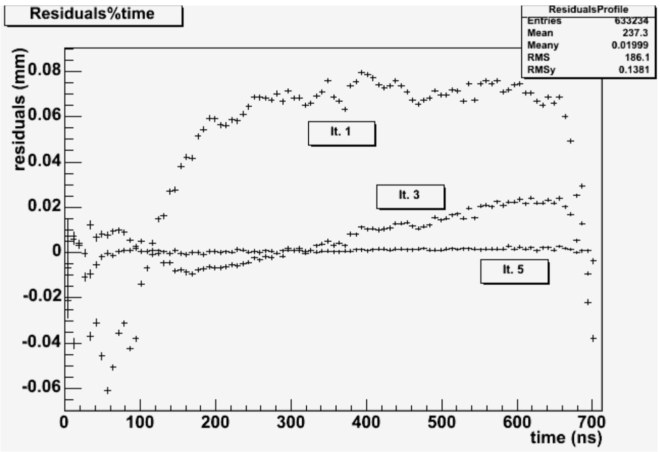

Figure 3: Tracking residuals (mm) as a function of the drift time (ns) after the 1st, the 3rd and the 10th iteration of the autocalibration..

The FMW Calibration Algorithm is the steering Athena algorithm that retrieves data from StoreGate and sends them to the tool that splits data between the different calibration regions and initializes one EventLoop for each region. In this stage the track segments are loaded into memory. The tool is also responsible of writing out the computed calibration constants. The EventLoop calls the algorithms that actually compute the constants.

Fig. 3 shows, on simulated data, the convergence of the autocalibration procedure implemented within this framework.

D. Calibration Database

The Conditions Database [10] stores all the data that describes the state of a system at a given time or, and it should contain all the information needed for the offline analysis. The MDT Calibration procedure, i.e. a calibration job running within ATHENA, produces a considerable amount information that is needed to validate the computation of the r-t relation but is not needed by the offline reconstruction. The database structure to handle MDT Calibration is therefore based on two different databases, as illustrated in Fig. 4: a Calibration DB, used to store all the information coming from calibration jobs, and a Conditions Database for the validated calibration constants needed by the offline reconstruction

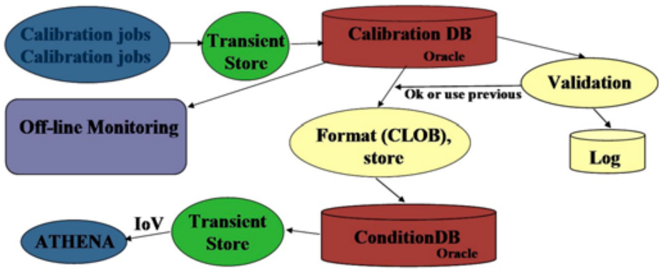

Figure 4: Diagram of the data base model.

The complete results of each calibration job will be copied in the Calibration DB, that will be developed in Oracle. At this stage, the calibration constants will be checked to ensure that they have been correctly computed. Also, the newly produced constants will be compared to those from the previous data taking period to decide whether the Conditions DB must be updated. In this case, only the information needed by the reconstruction is transferred to the Conditions DB. The Conditions DB will be implemented using the COOL interface [11], an LCG product that allows database applications to be written independently of the underlying database technology (this means that COOL databases can be stored in Oracle, SQLite or MySQL).

This model has the major advantage that we are free to change the private DB that is completely decoupled from the clients of the calibration and that this DB does not need to be shared should the calibration be performed in different computing centres.

The relevant calibration parameters will be transferred and stored into the Conditions DB using COOL tables via an Interval of Validity database [13]. The objects stored or referenced in COOL have an associated start and end time between which they are valid. COOL data is stored in folders, which are themselves arranged in a hierarchical structure of folder sets. Within each folder, several objects of the same type are stored, each with his own interval of validity range. These times are specified either as run/event, or as absolute timestamps, and the choice between formats is made according to meta-data associated with each folder. COOL is optimized to store and retrieve object(s) associated to a particular time.

To help with the storage of many objects of identical structure (e.g. chambers or tubes of the same chamber), objects in COOL folders can be optionally identified by a channel number (or channel ID) within the folder. Each channel has its own intervals of validity, but all channels can be dealt with together in bulk updates or retrieves.

In our case, the channel ID can be a chamber or a sector of chambers of the Muon Spectrometer.

COOL implements each folder as a relational database table, with each stored object corresponding to a row in the table. COOL creates columns for the start and end times of each object, and optionally the channel_ID and tag if used. Several other columns are also created (e.g. insertion time and object ID), to be used internally by the COOL system, but these are generally of no concern to the user. The remaining payload columns are defined by the user when the table is created.

The MDT calibration data has been stored as an inline BLOB in the database, i.e. defining the payload to be a large character object (CLOB) which has internal structure invisible to the COOL database. COOL is then responsible only for storing and retrieving the CLOB, and its interpretation is up to the client (other possibilities to store data are: inline payload or reference payload).

The CLOB appears as a sequence of characters, with a header and all the values separeted by a comma; the header typically contains all the information needed by the reconstruction job.

The retrieving and storing of the data inside the Athena framework is possible using the IoVService, an interface with the COOL tables via IoV range, and a packing and unpacking algorithm of the data (currently read from ascii file).

The IoVService permits to access to the right CLOB data in time with the event, and to retrieve the payload, the unpacking algorithm changes the format of the CLOB structure, obtaining the calibration constants and finally stores them in the Transient Data Store TDS to be used by the reconstruction algorithms via the Calibration Service.

The full chain described is almost complete and we are finalizing a preliminary version to be ready for the cosmic runs in the cavern.

REFERENCES

[1] M.Livan, “Monitored drift tubes in ATLAS.”, Nucl. Inst. And Meth. in Phys. Res A384 (1996) pp. 214.

[2] ATLAS Muon Collaboration, “The ATLAS Muon Spectrometer Technical Design Report”, CERN-LHCC/97-22 [Online], May 31 1997, Available: http://atlas.web.cern.ch/Atlas/GROUPS/MUON/TDR/Web/TDR.html

[3] A.Biscossa et al., “Calypso: a full scale MDT prototype for the ATLAS muon spectrometer.”, Nucl. Inst. And Meth. in Phys. Res A419 (1998) pp. 331.

G.Avolio et al., “Test of the first BIL tracking chamber for the ATLAS muon spectrometer”, Nucl. Inst. And Meth. in Phys. Res A523 (2004) pp. 309.

[4] P.Bagnaia et al., “Charge dependent corrections to the time response of BIL chambers at the Roma TRE cosmic test site”, Nucl. Inst. And Meth. in Phys. Res A533 (2004) pp. 344.

[5] J.Dubbert et al., “Resolution and Efficiency of Monitored Drift-Tube Chambers with Final Read-out Electronics at High Background Rates”, ATLAS note ATL-MUON-2004-002 [Online], March 4 2004, Available: http://weblib.cern.ch/Home/ATLAS_Collection/ATLAS_Notes/Muon.

[6] A.Baroncelli et.al., “Study of the drift properties of high pressure drift tubes for the ATLAS muon spectrometer”, IEEE NS Symposium Conference Record, 1 (2004) 427

[7] A.Baroncelli et al., “Calib user manual”, ATLAS note ATL-MUON-2002-004, 2002, available at: http://weblib.cern.ch/Home/ATLAS_Collection/ATLAS_Notes/Muon

[8] E.Etzion, S.Rosati., “System Test of the ATLAS Muon Spectrometer in the H8 Beam at the CERN SPS”, ATLAS note ATL-MUON-2004-026, 2002, Available at http://weblib.cern.ch/Home/ATLAS_Collection/ATLAS_Notes/Muon

[9] A.Baroncelli et al., “Assembly and test of the BIL tracking chambers for the ATLAS muon spectrometer”, Nuclear Instruments and Methods in Physics Research A 557 (2006) 421-435.

[10] R.Hawkings, “Conditions Database accounts and conventions”, available at http://atlas.web.cern.ch/Atlas/GROUPS/DATABASE/project/online/doc/conddb_conventions.pdf

[11] Proceedings of CHEP’04 Conference, Interlaken, Switzerland, 2004, published on InDiCo, Available on the CERN InDiCo server http://indico.cern.ch/

[13] A.Valassi , COOL Web Page, Available at http://lcgapp.cern.ch/project/CondDB/