The ATLAS Trigger Monitoring and Operation in proton-proton collisions

Martin zur Nedden, *for the ATLAS Collaboration*

Abstract—With the first proton-proton collisions at $\sqrt{s} = 900$ GeV in December 2009, the full chain of the ATLAS Trigger and Data Acquisition system could be tested under real conditions for the first time. Monitoring the performance and operation of these systems required a smooth and parallel running of many complex software tools depending on each other. They are the basis of rate measurements, data quality determination of selected objects and supervision of the system during the data taking. Based on the successful data taking experience with first collisions, the ATLAS trigger monitoring and operations performance are described.

I. THE ATLAS TRIGGER SYSTEM

In this note, the ATLAS Trigger monitoring system and first trigger operation experience with proton-proton collisions are described. In the first two sections, general aspects of trigger monitoring, including a short description of the trigger and the basic tools used to provide the necessary infrastructure are discussed. In the third section, the functionality of the main ATLAS trigger monitoring packages and their relation to the whole data acquisition environment are explained. Finally, a description of the first experience with proton-proton collision data is given followed by concluding remarks on the outlook for the upcoming data taking period in 2010/2011 at 7 TeV center-of-mass energy.

At the design LHC luminosity of $10^{34} \text{ cm}^{-2}\text{s}^{-1}$ and design bunch crossing frequency of 40 MHz, an average of 23 overlapping interactions will occur in each bunch crossing. The ATLAS trigger system is designed to filter out all but ≈ 200 Hz of the bunch crossings, while maintaining high efficiency for physics processes of interest.

Since the ATLAS detector is a general purpose detector, the trigger must be sensitive to a large number of different final state processes. The vast majority of the interesting physics processes produce trigger topologies with one or more high transverse energy (E_T) particles of various types such as leptons, photons and quarks (jets) or events with high missing transverse energy (E_T^{miss}) from undetected particles. For each of these types, one or more *trigger objects* is defined: μ , τ , e , γ , jet, b -jet and E_T^{miss} . The definition of a trigger object might include isolation criteria. Trigger decisions are mainly based on combinations of trigger objects with transverse energy or

momentum above various thresholds which are identified in the trigger processing.

The trigger decision is progressively refined in a three-level architecture which comprises a hardware trigger at the first level (L1) [1] followed by two software-based higher level steps collectively known as the High Level Trigger (HLT) [2]. The two levels of the HLT are known as level 2 (L2) and event filter (EF). Both run on large computer farms.

The L1 trigger is required to reduce the event rate from 40 MHz down to ≤ 75 kHz based solely on muon and calorimeter information. L1 is synchronized with the LHC bunch crossing frequency and provides a virtually dead-time free trigger decision with a latency of $2.5 \mu\text{s}$. While the decision is being made, the event information is kept in pipelines of a depth corresponding to the L1 latency. In addition to the trigger selection, the L1 trigger also provides Regions of Interest (RoI) for each trigger object which supply geometric information (η and ϕ , but with coarser granularity than the detector cells), E_T estimates and object type information to L2. Combinations of these RoIs are called *L1 items*.

A positive L1 trigger decision initiates the readout of detector data from the pipelines into a distributed buffer system (ReadOut Buffers or ROBs) where they are held pending the L2 decision. L2 takes the L1 trigger items and associated RoIs as input and further develops them, improving the resolution of the trigger objects and adding information from the tracking system. L2 requests a limited amount of information from the ROBs based on the geometric information in the RoIs and thus saves 98% of the data flow which would be needed for a full event build. A positive L2 trigger decision initiates the readout of the full event record from the ROBs and its transfer to a node of the EF farm. The design allows for a L2 output rate of about 3.5 kHz and an average latency of about 40 ms.

The EF applies optimized offline analysis algorithms to data within the RoIs identified by L2. Events accepted by the EF are routed to permanent storage. The design assumes an average latency of about 4 seconds and an EF output rate of ≈ 200 Hz.

The core HLT code which handles the successive running of algorithms and controls the information flow during the HLT decision process is called the *Trigger Steering* [3]. The Trigger Steering is initialized by a *Trigger Menu* which consists mainly of a collection of *trigger chains*. The trigger chains define the sequence of steps taken to derive the trigger decision starting from a particular input trigger item. An event can be discarded at any step of the trigger chain, thus avoiding the execution of the subsequent steps. A collection of trigger chains containing the same trigger object types is called a *trigger signature group*.

Manuscript received Mai, 2010. This work was supported in part by the Federal Ministry for Education and Research of Germany.

Martin zur Nedden, Humboldt-Universität zu Berlin (e-mail: nedden@mail.cern.ch).

Antonio Sidoti, Humboldt-Universität zu Berlin (e-mail: antonio.sidoti@cern.ch).

Tomasz Blod, University of California Irvine (e-mail: tbold@mail.cern.ch).

To ease physics analysis, the selected events are stored into one or more streams depending on which chain(s) caused the trigger. The streams are defined in terms of the trigger objects. Examples of streams include “muons”, “electrons and gammas”, “ τ s, jets and missing E_T ” and, in addition to the physics-motivated streams, streams for calibration and debugging as well as an *express stream*. The latter fully overlaps with all other streams and contains a representative selection of events. It is processed immediately after a run is finished to provide the most up-to-date calibration constants for the first data reprocessing at the CERN IT center (the so-called *Tier0* center).

II. GENERAL ASPECTS OF TRIGGER MONITORING

A. Introduction

The ATLAS trigger and data acquisition systems are highly complex and require reliable, efficient and redundant monitoring systems to ensure proper online operation, good performance and the rapid recognition and localization of potential problems. Items to be monitored include the trigger rates at every decision step, the quality of the selected trigger objects and the technical performance of the trigger computer farms. Trigger rates are particularly valuable indicators of trigger performance since they are highly sensitive to detector malfunctions, LHC beam issues and instantaneous luminosity.

Over the last two years, a group of ATLAS institutes have developed, implemented and successfully tested the ATLAS trigger monitoring system. This has been realized within the framework of the global ATLAS Data Quality Monitoring (DQM) system. The Trigger Steering code controls the trigger decision computation and oversees the production of more than 2500 histograms which monitor detailed aspects of the trigger behavior. The histogrammed quantities include timing information, algorithm execution error codes, physics motivated parameters specified by the trigger signature groups, numbers of events, detector occupancies, multiplicities, and cut flows.

The trigger-specific DQM was developed based on the automatic analyses of histograms of physically motivated variables. The selection of reliable quality criteria required close cooperation with the ATLAS physics analysis groups. In the resulting system, the trigger data quality is being computed automatically during data taking and reprocessing. An evaluation of the quality of the trigger response on the selected data is provided in a coherent way to be used for physics analysis. To do so, the trigger data quality is provided in *flags* reflecting the behavior of the different trigger signatures derived from the online and offline monitoring informations. A Data Quality (DQ) flag in ATLAS, provided and stored for each sub-detector and trigger, can have five different values (indicated by different colours in the DQM display explained in section 3.2): good (green), flawed (yellow), bad (red), undefined (gray) and off (black). The online checks need to be confirmed by the offline event reconstruction in the Tier0, on the one hand to verify the online decision and on the other hand to clarify undefined flags. The final DQ decisions are stored into a database for further analysis.

The monitoring tools were tested during the ATLAS cosmic muon data taking periods of summer and fall of 2008. Based

on this experience, an internal evaluation of the tools took place and a new and improved concept was established and implemented during the LHC shutdown in 2008/2009. The new concept of the ATLAS Trigger monitoring system worked reliably during the first collision runs in 2009. The system ran smoothly and the monitoring tools proved to be sufficient to analyze the performance of the ATLAS trigger. They have been used successfully both online and within the offline event reconstruction and are an important part of the trigger and data taking process in ATLAS.

B. Design and General Tools

The ATLAS trigger and data acquisition monitoring is composed of a coherent set of independent software tools. The main requirement for the trigger monitoring tools is their independence and redundancy to allow a stable supervision of the system and fast and reliable recognition and resolution of problems. This requires efficient information sharing among the various applications and complex graphical user interfaces. In the development of these tools, the focus has been on the performance and scalability of the monitoring infrastructure.

The main building blocks required by a reliable and comprehensive trigger monitoring are, as discussed in Sec. II-A, the rate measurements, the assessments of the quality of the trigger objects and the survey tools to be used during the data taking process. Additionally, the monitoring information has to be stored in a convenient way for further more complex analyses and the selection of good quality runs for physics analysis. Furthermore, since ATLAS is a worldwide collaboration, the information has also to be easily accessible via the WWW. In this section, the main general services provided by the trigger and data acquisition (TDAQ) infrastructure, not only relevant for the trigger monitoring but for other parts of ATLAS as well, are presented.

The *Information Service* (IS) data repository is the backbone of the information sharing between the various online systems [4]. The data shared through IS ranges from simple quantities up to more complex objects as histograms. IS is used to share information between applications in a distributed environment, flowing around the repository which holds the information provided by the applications. A central part is the IS repository which holds the information provided by applications. This repository supports three main types of interactions. The *Information Provider* can create information in the IS repository and also update or delete an already existing piece of information. The *Information Reader* can get the value of the information from the repository by sending a request to it. The *Information Subscriber* can subscribe to the repository in order to be notified about the changes. Each time the Provider creates, updates or deletes information the Subscriber will be informed. Obviously, an application can use any combination of interactions described above. In addition to the interactions with the IS repository any application is able to send commands to any of the running IS Information Providers.

Based on IS, the *Online Histogram Service* (OHS) handles the sharing of histograms between the online applications.

Since the HLT is a distributed system over more than 2000 multi-core nodes, the histograms produced by the trigger need to be added up and handled in a correct way. This is done by an application called the *Gatherer*.

The monitoring information is permanently stored on a run-wise basis for additional cross-checks and analysis on the offline side. The storage of monitoring information is done by the *Monitoring Data Archiving* (MDA) application [6]. The histograms produced online are stored in root files and asynchronously transferred to the offline permanent storage.

C. ATLAS Trigger Data Quality Operation

In order to achieve well defined and understood trigger DQ flags, an offline cross-check of the basic variables used for their online determination is necessary. With a maximum delay of 36 hours after collection, the data are reconstructed and basic monitoring information is produced in parallel at Tier0. This offline monitoring (called Tier0 monitoring) is based on histograms containing physics distributions for the trigger objects and is organized in the same way as the online monitoring. The offline DQ looks at the same trigger objects as the online one enabling a direct comparison and allowing to confirm or resolve problems.

Generally, the trigger DQ decisions derived online are completed and verified on the offline side. At the end, only a final binary *good* / *bad* decision is provided for physics analyses. For instance, if on the online side the trigger DQ decision is *flawed*, this is clarified and turned into bad or good with additional information available after the Tier0 reconstruction. Fig. 1 shows the $\log E_T^{\text{miss}}$ distribution computed by the trigger online on the left and by the standard data reprocessing of the express stream in the Tier0 on the right. A good agreement of both distributions can be observed confirming the trigger DQ decision taken online. The mismatch of the distributions originate from the different event samples: where the online distribution from EF contains all events, the offline one is made by the events collected from the express stream only.

The trigger monitoring is handled similarly at the online and offline sides. Since ATLAS is a worldwide collaboration it is essential to have access to the trigger monitoring information also outside the ATLAS control room. Therefore, a web-based tool, the *Web Monitoring Interface* (WMI) has been designed to show all relevant trigger information after a run has been taken.

III. THE ATLAS TRIGGER MONITORING

In this section, the three main trigger monitoring applications are described: the trigger rates measurements, the derivation of the trigger data quality and the tools for the trigger supervision by the shift crew in the ATLAS control room. These tools are also the basis for the description of the performance of the trigger operation with the first proton-proton collisions in December 2009 discussed in the next section.

Fig. 1. Example of online and offline monitoring. The $\log E_T^{\text{miss}}$ distribution for a 900 GeV center-of-mass energy proton-proton collisions run; from the online Trigger Steering code (left) and from the standard first reconstruction pass at Tier0 (right). Both distributions can be checked with a specific application to derive both online and offline data quality flags. Since the online distribution corresponds to the full data set where the offline is made out of the express stream only, the distributions differ.

A. Monitoring of Trigger Rates

The L1 trigger rate calculations are based on event counting while the HLT rates are computed by summing up the rates pre-measured by the Trigger Steering code in each CPU of the HLT farm. All L1 and HLT event counts and rates are stored in histograms that are handled by IS. Additionally, the number of errors which occur during trigger execution are stored for each output stream, L1 item and HLT chain. Fig. 2 shows the number of events at the input of each trigger level, after each step during the trigger selection process and after pre-scale for a small sub-set of EF chains, for a 900 GeV center-of-mass energy proton-proton collision run. Most diagnostics related to the rates can be made with the information contained in this histogram.

Monitoring the trigger rates is essential to adjust the trigger pre-scale factors in order to make optimal use of the available bandwidth or to react on changes of the LHC beam conditions and potential problems of the sub-detectors. The trigger rates are calculated every 10 seconds for all trigger chains at each

Fig. 2. Event counts histogram for the trigger rate measurement in ATLAS. On the x-axis, all output streams, L1 items and HLT chains are displayed, while on the y-axis the decision steps are shown. For readability, only a small part of the x-axis is shown here. The size of the rectangles correspond to the number of events found of a certain trigger chain at the given step. Since the HLT chains are composed of different number of steps, when applicable, crossed boxes indicate the non-existence of the step in the corresponding trigger chain.

level within the Trigger Steering, summed up by the Gatherer and stored in a specific IS server keeping them in memory for several hours. The rate information is stored in a histogram similar to the one shown in Fig. 2. In order to provide the trigger rates in a convenient way to the ATLAS shift crew, a package called *Trigger Rate Presenter* (TRP) has been developed. It is a collection of distributed applications, where one group performs CPU intensive calculations while other groups are used to display the data. The guiding principles of the design are the modularity of the various applications and their scalability. It consists of dedicated software applications performing rate summation, computation and publication to IS. Fig. 3 illustrates the TRP functionality. The so-called *adapters* read the relevant information from IS and, within one central server application, calculate the rates at all levels and all trigger chain steps. Since this is the most CPU intensive application, it is vital to decouple it from the graphical presentation. From IS, the rates are distributed to the trigger data base for storage and to both a dedicated GUI to be displayed in the control room and the WWW. The TRP turned out to be one of the most central trigger diagnostic tools during the commissioning with the first proton-proton collision data in 2009.

A Graphical User Interface (GUI) to display the rates in the ATLAS control room has been written using the Qt [7] development framework. This GUI displays the rates time trend plots and tables (as presented in the example shown in Fig. 4). A plug-in of WMI, which is a part of the TRP package, exports the trigger rates to web pages publishing the rates of the last 24 hours to the public network. Using the MDA mechanism introduced in Sec. II-B, an application archives the trigger rates in ROOT files permanently for later studies.

Fig. 3. Schema of the Trigger Rate Presenter of ATLAS: the trigger rates calculated in the Trigger Steering are summed up by the adapter software and sent to the IS. From there they are distributed to the TRP GUI for rate displays in the control room, for storage on a DB and to WWW to be displayed outside of the control room. The thin rectangles at the top left represent the output of other applications

Fig. 4. Display (GUI) of the online Trigger Rate Presenter that shows the rates in the ATLAS control room during data taking. The basic display is the summary rate yield over time for L1, L2 and EF. Selected individual trigger levels can be organized in table views contained in independent windows presenting the rates of the individual trigger chains. The tables are sortable by rate. Rate yields as a function of time can be provided for any user-defined selection of trigger items from the table window.

B. Monitoring of Trigger Objects Data Quality

The data quality monitoring of the ATLAS trigger is part of the general ATLAS DQM and is realized based on a common software infrastructure. The Trigger DQM analyses the properties of the trigger objects corresponding to physics object candidates and used by the trigger for the event selection (i.e. candidates for muons, electrons, photons, taus, jets, b -jets and missing energy). Triggers based on these objects are bundled in the so-called *trigger signature groups*. Additionally, trigger signature groups are defined for the trigger chains for cosmic muons, minimum bias and B -physics. For each trigger signature group, a data quality flag is formed and saved into

a database.

A dedicated software, the *Data Quality Monitoring Framework* (DQMF), makes an automatic evaluation of selected histograms¹, based on pre-defined tests. Test examples include checking the distributions width and mean or the Kolmogorov-Smirnov discriminant with respect to a reference histogram.

The trigger signature data quality flags are stored in the ATLAS conditions DB to allow assessment of the data quality during the physics analysis. A highly configurable *Data Quality Monitoring Display* (DQMD) shows the DQ results obtained with DQMF and focuses the attention of the shifter to the histograms failing the automatic checks. The information flow is organized hierarchically starting at the top level showing the various ATLAS sub-detector status, down to the lowest level checks performed on single histograms. On the offline side, the histograms filled during data reprocessing at Tier0 are evaluated automatically using an offline DQMF system, delivering the analogous DQ flags for the trigger as the online DQMF. With the help of the offline DQ flags, unclear DQ decisions on the online side are overwritten, yielding a conclusive *good* / *bad* decision.

The interface for the evaluation of the DQ flags is shown in Fig. 5. Every single histogram check is performed with its own threshold and reference computing a result in terms of “good”, “uncertain” or “bad”. These DQ results are successively grouped into *DQ regions* yielding the final DQ result in the corresponding DQ flag. The data quality is derived separately for L2 and EF, and then merged.

Fig. 5. Display of the online data quality monitoring framework to survey the trigger objects quality. The trigger DQ flags described in the text are listed on the left panel. This example shows the check of a L2 calorimeter algorithm error histogram for an e/γ trigger object as a function of η . The bin contents above a certain threshold are compared to a reference histogram to derive the DQ decision. The displayed test is successful.

C. Monitoring of System Functionality and Performance

The last aspect of trigger monitoring discussed in this paper concerns the diagnostic tools used by the shift crew to survey the trigger behavior in the ATLAS control room during data taking. Since more than 2000 online histograms are available for the trigger, an efficient tool to display and manipulate them has been implemented: the *Online Histogram*

Presenter (OHP). This application uses Qt [7] and ROOT and is configured via an XML[5] file. A small number of representative histograms (less than ten) per trigger signature group are presented in a tree structure or in configurable pre-defined windows. In OHP, reference histograms can be displayed and superimposed on the histograms in question.

OHP works with a mixed pull/push mode interacting with the OHS servers: notifications are pushed from the OHS server to OHP every time an histogram is updated. Only when the histogram is actually displayed, the histogram object is retrieved from the OHS server and updated in OHP. A sophisticated caching mechanism minimizes the required network bandwidth.

An example of the setup provided for the ATLAS control room is given in Fig. 6. Here, the plug-in for the general HLT overview is presented, showing the panel of the HLT streaming. As for DQMF, the plug-ins are organized in terms of signature groups as can be seen on the left hand side of the selection panel. The basic concept here is to provide one single panel per signature group allowing a fast check.

Fig. 6. Display of the Online Histogram Presenter for the survey of the trigger in the ATLAS control room by the shift crew. Individual signature groups can be selected in the panel on the left hand side. The corresponding pre-defined histograms are displayed in the central region. The example shows the behavior of streaming including the correlation of the streams (as in Fig. 9).

IV. TRIGGER OPERATIONS WITH FIRST PROTON BEAMS

A. Experience with Single LHC Beams

The priorities of ATLAS for the first beam interactions were reliability and stable operation, requiring a simple trigger configuration based on L1 trigger decisions only. The HLT was running in a transparent mode where the events were routed to different output data streams according to their L1 decision. Algorithms, except those crucial for the routing task, were not exercised or executed online. In October 2009, ATLAS received the first LHC beam events of the year in the form of the so-called beam splash events: one single beam consisting of one single bunch sent around the accelerator and stopped by collimators situated 140 m upstream of the detector along the LHC ring. A precious handful of beam splash events was collected by ATLAS and provided the first very important beam-based reference to adjust the detectors and L1 trigger timing.

¹These histograms are published to IS by the Trigger Steering.

B. Commissioning the Trigger System with Collisions

After measurements performed with the beam splash events, a single LHC beam and then two counter-rotating ones were sent around the whole LHC ring. At that time, the L1 decision was based on the signals from electrostatic *Beam Pickups* (BPTX)² and two planes of dedicated *Minimum Bias Trigger Scintillators* (MBTS [8])³ situated around the beam line at $2.1 < |\eta| < 3.8$. Collectively, these systems provided high selection efficiency for collisions with a low enough trigger rate that there was no need for an additional rejection at the HLT. All the L1 decisions were relying on a coincidence with the beam pickup signals: only very rarely and for short periods of time (e.g. when the beam pickup signals were below the trigger threshold), the *bunch group mechanism*⁴ was used in order to recover the beam-synchronous events not triggered by the beam pickups. In this case, the bunch identifier known to contain protons was used as a trigger.

Several of the runs in 2009 were taken by collecting 32 separate time samples of all the ATLAS Liquid Argon Calorimeter (LAr, electromagnetic calorimeter) channels for each event for commissioning purposes. The event size was thus very large (≈ 14 MBytes) and had an impact on the maximum output rate that could be written into disk. In this condition, in order to maintain the output bandwidth requirements within the ATLAS TDAQ capabilities and avoid data acquisition dead time, the L1 beam pickup signals (with a rate of ≈ 11 kHz) were highly pre-scaled. The collision rate, defined by the rate of events where a time coincidence of at least one hit in both MBTS planes are produced, was unaffected. Only when reducing the number of time samples of the LAr channels, the pre-scale of the beam pickup signals could be reduced and, hence, the overall data taking rate increased.

The commissioning of the HLT was undertaken in several steps. At the beginning HLT was only routing the events to the different output streams without executing any event processing/reconstruction algorithm. *Monitoring mode* was the second step, where the HLT selection algorithms are running but no events are discarded. The potential decision is anyway stored in the events and can be monitored and cross-checked offline. Finally, the HLT was run online in its designed acceptance/rejection mode. In the first two steps, the functionality

and HLT selection could be tested, debugged and validated offline by analyzing the events and the potentially rejected ones in particular. Following these three steps, the ATLAS data taking was not jeopardized by any potential problem the HLT selection might have introduced.

During the time when the trigger decision was based on the L1 decision and HLT was only streaming the events, the HLT was extensively exercised offline using the already taken splash, single beam and collision data events. Basic cross-checks of errors, crashes, algorithm performance and trigger decision were extensively done using also the trigger monitoring histograms produced in the Trigger Steering. The offline trigger monitoring infrastructure proved to be very useful during this time. Several of the early collision runs were reprocessed multiple times with improved trigger configurations in order to maximize the smoothness of the online execution of the HLT algorithms. Around 30.000 events were reprocessed in order to make sure that the HLT was running reliably and could be safely turned on for the online data taking.

Once convinced of its robustness, we started running the HLT system online in monitoring mode, i.e. running the events through both L2 and EF and accepting all of them irrespective of the HLT decision. In this mode, in addition to the physics chains, several calibration chains were executed. As a result, ATLAS exercised and collected events for several calibration streams. In a very small number of events errors occurred during the HLT online processing and were therefore routed to the so-called debug streams: all these events were then automatically reprocessed offline in order to understand the reason of the error. All of them turned out to contain hits in large portions of the muon system caused by cosmic showers. The muon tracks reconstruction of these events was therefore exceeding timeout boundaries and the events were routed to the debug stream.

Finally, in December 2009 and after having run in monitoring mode for several days, one L2 and one EF inner detector based trigger algorithm were included online in order to have an alternative method to the MBTS system to select inelastic interactions [8]. All events triggered by either of the beam pickup signals at L1 were processed through this L2 minimum bias trigger algorithm, which uses the Inner Detector silicon sub-detectors [9] (the pixel and the silicon central tracker (SCT)) to detect central detector activity by counting space-points⁵ in both detectors. This algorithm would then reject events having a too small number of space points. In EF, charged tracks in the silicon detector are reconstructed, employing tracking tools identical to the offline track reconstruction software. One silicon track reconstructed with $p_T > 200$ MeV is then required to accept the event.

The introduction of HLT rejection allowed better event selection with respect to the indiscriminate pre-scale of the L1 beam pickup triggers. This pre-scale was progressively reduced during the first run where HLT was introduced online: the L2 and EF rates were constantly monitored and pre-scale

²The Beam Pickups (BPTX) stations are beam position monitors provided by the LHC machine, but operated by experiments for timing purposes. They are comprised of four electrostatic button pick-up detectors, arranged symmetrically in the transverse plane around the LHC beam pipe. The signals are used by the trigger system and by a monitoring system for the LHC beams and timing signals.

³The Minimum Bias Trigger Scintillators are a trigger dedicated system located on the cryostat in front of the forward calorimeter. The system consists of two wheels of scintillator panels, each mounted face-on to the beam axis 3.5 meters on either side of the interaction point. The 32 individual panels provide independent triggers, giving a clear indication of when and where a track passes parallel to the beam axis.

⁴The different bunch identifiers around the LHC machine are grouped into seven different categories according to the LHC content in the bunch. These categories include empty, filled or unpaired bunches. Empty bunches can be used to trigger on cosmic muon events while filled bunches are the ones containing potential proton collisions. The L1 decision primitives are generated based on the LHC bunch crossing in ATLAS at a given time, and can be combined by the L1 decision hardware like any other detector-based primitive.

⁵The Inner Detector space-points are 3-dimensional hit representations formed from hit clusters.

factors adapted to maximize the exploitation of the available bandwidth.

Regardless of whether the HLT was running online, offline reprocessing of the data through the HLT code was still performed for a selected number of runs. This procedure was used to help commission and understand the different triggers that include tracking, muon or calorimeter cluster reconstruction. A special trigger configuration was created in order to better exercise all the high level trigger algorithms, and offline tracking or calorimeter cluster reconstruction efficiencies were compared to those obtained by the trigger. This exercise was particularly necessary to help commission the calorimeter-based triggers: the data from the LAr calorimeters used during the automatic event reconstruction at the Tier0 is based on the raw calorimeter readout (calorimeter samples), while during the online HLT algorithm reconstruction the already calculated energies from the samples performed by the LAr readout electronics are used. Therefore, a comparison of online and Tier0 reconstructed quantities is not appropriate in this case: it is only after the HLT offline reprocessing that the properties of clusters reconstructed online and offline can be compared.

C. First Measurements

Fig. 7. L1 (black) and HLT (red) output trigger rates for a typical run with stable beam flag. Also shown is a collision trigger rate at L1 (green line), requiring hits on the minimum bias scintillator counters and filled bunches for both beams. The pink line labeled L2 Inner Detector activity represents a filtering algorithm at the L2 trigger, which accepts events based on space point counts in the Inner Detector (see text in section 4.2). This L2 algorithm receives 5 % of all filled bunches as input from L1. Assuming both the L1 collision trigger and the L2 algorithm are highly efficient for collision events, the ratio the two lines should roughly reflect this fraction (modulo the difference in the acceptance of both triggers). The moment the L2 algorithm is enabled is clearly visible with the start of a non zero event rate on the L2 line. At the same time, the L1 pre-scales are reduced and thus an increase of the L1 output rate can be observed which, compensated by the L2 rejection, yields a similar HLT output rate. The dips in HLT and L1 output rates just before this moment are due to the short pause needed to change trigger setup. The HLT output rate (which represents the rate of events recorded to disk) does not change visibly, as it is dominated by a constant rate of monitor triggers.

In November 2009 the first collisions at a center of mass energy of $\sqrt{s} = 900$ GeV were observed in ATLAS, followed in December 2009 by the highest ever man-made collider interactions at a collision energy of $\sqrt{s} = 2.36$ TeV. The extended periods of stable single beam in either direction, yielding many beam-halo events⁶, allowed fine timing of the sub-detectors and the validation of the beam triggers.

⁶Upstream of the detector, protons from the LHC beam collided with atomic nuclei of gas atoms producing a halo of muons, roughly parallel to the beam in the beam pipe. These are the so-called beam-halo events.

Fig. 8. Total execution time per trigger algorithm (sequences) in ms to process events at L2 (left) and EF (right). For readability, only a small part of the trigger algorithm on the x-axis are shown. Note, that the vertical scale is logarithmic. The different execution times of the trigger algorithms can be seen clearly. The mean execution times at L2 and EF are ≈ 27 ms and ≈ 130 ms respectively.

Around 917.000 events were collected overall during November and December 2009 corresponding to an integrated luminosity of $\approx 20 \mu\text{b}^{-1}$. Out of these, a total number of 538.000 events could be recorded with an overall data taking efficiency of more than 90% in the so-called stable LHC beam period. The peak LHC luminosity achieved was $\mathcal{L} = 7 \cdot 10^{26} \text{ cm}^{-2}\text{s}^{-1}$. The main triggers used to record these events were the L1 collision triggers provided by the two planes of Minimum Bias Trigger Scintillators. The HLT (both L2 and EF) first ran in pass-through mode and only later with limited algorithms active. Fig. 7 shows the trigger rate versus time during a typical stable beam run of this period. The different lines represent the L1 output, HLT output, MBTS L1 output and L2 inner detector activity rates respectively. All plots shown in this section are based on the Trigger Steering online histograms.

The execution of the HLT is demonstrated with some representative histograms produced online by the Trigger Steering during collision runs at $\sqrt{s} = 900$ GeV recorded in December 2009. Fig. 8 shows the total time for the trigger algorithm execution (*trigger sequences*) per event processed at L2 on the left and EF. Online average execution times for L2 and EF are found to be ≈ 27 ms and ≈ 130 ms respectively, compatible with the required time constraints of 40 ms for L2

Fig. 9. At the top, event counts in HLT streams and their correlation. It can be observed in particular that the express stream is fully overlapping with the other physics streams. At the bottom, an example of the event statistics for EF chains directly contributing to the express stream is shown.

and 4 seconds for EF. The time spent in EF is significantly smaller than the designed value due to the fact that only a very small fraction of the trigger selections are exercised with the 900 GeV center-of-mass energy proton collisions: on average, events at this energy are significantly less busy than at high energies and design luminosity. Therefore, the full EF potential has not yet been exploited.

The upper plot of Fig. 9 shows all active streams and their correlations. This plot is one of the crucial histograms for online diagnostics and is a powerful tool to locate potential Trigger, DAQ or sub-detector problems very quickly. For example, the presence of unexpected events in the debug streams or the absence of expected events within the calibration streams may indicate problems with the trigger or sub-detectors. Any deviation of the exclusiveness of debug streams or a non-correlation of the express streams with the physics streams are also indications of potential problems with the trigger system.

The lower plot of Fig. 9 shows the event count of EF chains within the express stream. This is an example of how event counts for a representative selection of chains from all streams can be studied. The real-time availability of these histograms is

very useful to provide fast feed-back before the reconstruction at Tier0, that starts with a latency of 36 hours.

Regions with high trigger activity can be easily located with only one plot shown to the shift crew. In case of problems like “hot” (high statistics) or empty regions, a detailed analysis can then be made making use of the various RoI plots from L2. Fig. 10 shows the (η/ϕ) distribution of the L1 RoIs as seen by L2. On the left, the L1 jet RoIs position are shown (triggered by the L1 calorimeter system) while the L1 muon RoIs position (triggered by the L1 muon system) are shown on the right. The region showing high activity in the center of the muon RoIs is not caused by a hot muon cell but is devoted to a feature in the trigger code. Fig 11 shows the (η/ϕ) distribution of all L1 RoIs (both calorimeter and muon systems together) accepted by EF. The (η/ϕ) grid is chosen to have the same granularity as the detectors to allow a fast correlation between detector problems and “hot” or “empty” trigger towers. Furthermore, Fig 11 reflects the dominance of cosmic trigger chains during the 900 GeV running, leading to a higher activity at $(\phi \in (0.9, 2.1)/\eta \in (0, 1.1))$ where the big access shaft to the experiment is located.

Overall, the main diagnostic tools for the online and offline trigger monitoring were available and worked reliably with collision data. A high order of consistency between the redundant software tools as requested for a reliable monitoring system has been achieved.

V. CONCLUSIONS AND FURTHER STEPS

The requirements, implementation and functionality of the ATLAS trigger monitoring were shown. A method to derive and present the trigger rates at all trigger levels and intermediate steps during data taking, as well as an offline diagnostic of whole runs was developed and proved to work well and reliably. Additionally, the framework to assess event data quality for what concerns the trigger selection was implemented and satisfactorily tested.

With cosmics muon events, single LHC beams and proton-proton collision data, it could be shown that the ATLAS trigger and data acquisition system is working reliably. All the monitoring systems implemented so far have satisfactory functionality and deliver the necessary support to run the ATLAS trigger system. A tool for fast trigger checks in the control room has been implemented and tested. The development phase has concluded, but with the first data in 2010 many new challenges have to be faced for the trigger monitoring. In order to produce reliable trigger DQ flags, the behavior of the system has to be understood better with real data. As a consequence the parameters of the framework, like the thresholds of the DQ tests and the reference histograms, have to be optimized making use of the first data. This requires an intensive cooperation within the ATLAS trigger groups. The offline diagnostic tools, based on the monitoring tools to be processed during the Tier0 standard reconstruction or even more during the first reconstruction using the express stream will also be extended with the experience of the first real collision data.

The trigger operation focus is now shifting from development and commissioning to a constant monitoring of the

system, the optimization and continuous adjustment of its parameters, as well as assessment of its performance.

REFERENCES

- [1] ATLAS Collaboration, *Level-1 Trigger Technical Design Report*, August 1998, ATLAS TDR-012
- [2] ATLAS Collaboration, *The ATLAS HLT, DAQ and DCS Technical Design Report*, October 2003, ATLAS TDR-016
- [3] N. Berger et al., *IEEE Trans.Nucl.Sci* **55** (2008) 165
- [4] W. Vandelli et al., *IEEE Transactions on Nuclear Science (TNS)* **54** (2007) 609
- [5] The Extensible Markup Language, <http://www.w3.org/XML/>
- [6] P. F. Zema, *The Monitoring Data Archiving Service for ATLAS 2006*, IEEE Nuclear Science Symposium Conference Record, N02-5
- [7] The Qt-Cross Platform Application, <http://doc.trolltech.com/>
- [8] W. Bell et al, *MBTS trigger efficiency for the minimum bias analysis using Inner Detector tracks from pp interactions at $\sqrt{s} = 900$ GeV*, February 2010, ATL-COM-DAQ-2010-003
- [9] ATLAS Collaboration, *The ATLAS Inner Detector Technical Design Report*, April 1997, ATLAS TDR-04 and ATLAS TDR-05

Fig. 10. η vs ϕ distribution of the L1 RoIs (with the granularity of the sub-detectors) monitored by L2 for the hadronic calorimeter (left) and muon system (right). This allows a fast correlation of doubtful trigger rates with detector effects like “hot” detector cells.

Fig. 11. η vs ϕ distribution of the L1 RoIs as received by the EF. Here, a map of all L1-RoIs is shown, regardless of the sub-detector originating them.