Summer Student Report 2013: L1CT MIOCT presenter and BunchGrouper

Tomas Snellman
Supervisor:
Taylor Childers

2013-09-06

1 Introduction

For the summer I have been working as a part of the ATLAS Level 1 Central Trigger group.

I have had two separate projects. First I created a web-based monitoring tool, the Mioct Presenter. Mioct Presenter monitors the muon detector side of the Central Trigger Processor (CTP).

The second project was updating an existing application, the BunchGrouper application. BunchGrouper is used to combine the information of filled bunches in the two beams.

2 Mioct Presenter

Mioct Presenter is a Web tool, with a graphical user interface. It allows on-line monitoring of the muon detector system. It displays event rates, BCID timing information and event counter information of the MUCTPI (Muon to Central Trigger Processor Interface) and CTP system. It retrieves the displayed data from ATLAS' Information Service (IS) server, which provides ROOT histograms and other live data.

ATLAS had an application for monitoring the MUCTPI system. However to use it you had to be in the ATLAS control room or connect to it via SSH. A web based application can be accessed quickly by opening the appropriate web page with a web browser. Mioct Presenter uses Javascript which means that the application is executed on the user's computer when the web page is opened. Only

a minimum amount of text data is transferred from the server. When using the previous application the graphical output had to be sent over SSH.

Mioct Presenter is written in Javascript and can it is used with a web browser. To access IS Mioct Presenter uses the WebIS interface. WebIS runs on the server and allows accessing IS data via HTTP.

The interface is placed in a HMTL5 canvas element. Javascript allows drawing on the canvas. Mioct Presenter uses KineticJS framework, which is a framework for drawing on the canvas.

2.1 Graphical Interface

The initial view of L1CTPMioctPresenter displays information about the rates in the different regions of the muon detector. The muon detector cells are grouped into the forward endcap (endcap C), the backward endcap (endcap A) and the cylindrical barrel. The barrel is displayed as if it was cut open on one side and flattened out on the plane. The colors of the different sectors represent the current muon rate in that sector.

The muon detector is split up into eight MIOCT modules each for forward and backward side. The presenter indicates these with labels with orange and blue background. The MIOCT modules have numbers 4-11 on the forward side and 14-21 on the backward side.

In addition to the MIOCT module naming scheme the L1CTMioctPresenter has another naming scheme. This TGC/RPC expert scheme is shown with labels on gray background. The numbers go from 1 to 12 in the endcaps and from 1 to 16 in the barrel region.

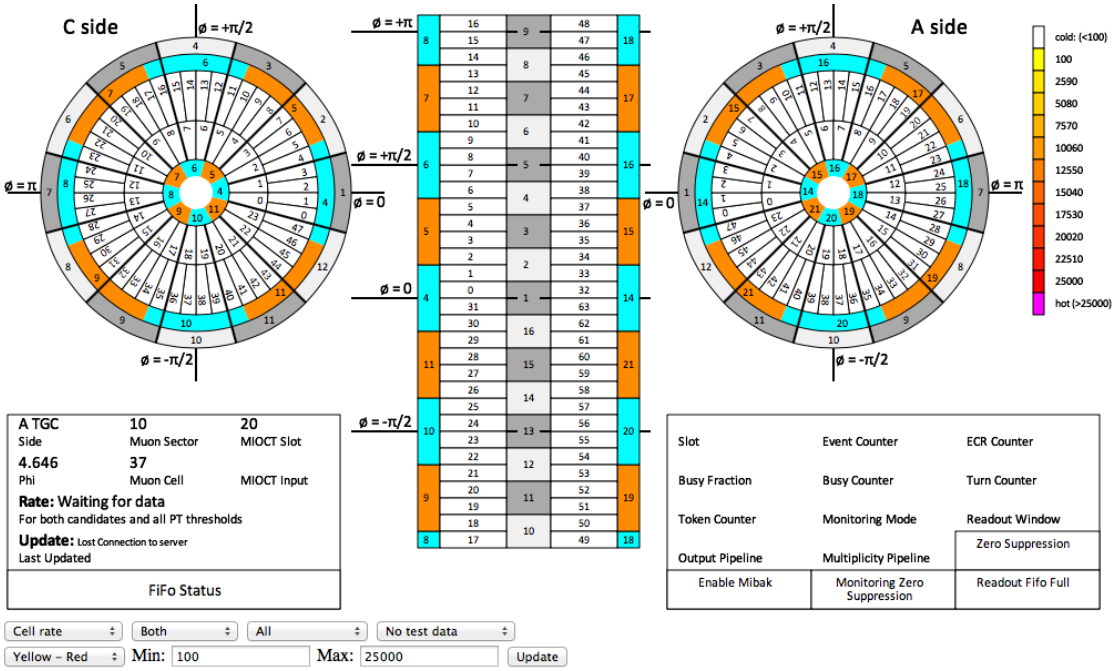
The muon detector cells on A and C side are divided into the forward TGCs, which are represented by the inner ring in the representer and the endcap TGCs which are represented by the outer ring. The forward TGCs are labeled 0-23 and the endcap TGCs 0-47.

The rectangular section in the centre represents the barrel RPCs. These are numbered from 0 to 63. Cells 0-31 are on the C side and cells 32-63 are on the A side. The TGC sector numbering is shown in the center with number 1-16 on grey background.

Text information is divided into two boxes. The lower left box shows information about the selected muon detector sector. The lower right box shows information about the selected MIOCT module. FiFo Status field is coloured based on the state of the MIOCT FiFos. IS shows Full, Empty, and in some cases Almost Full flags for the FiFos. A single Almost full flag gives a yellow colour and a single Full flag gives a red colour

The text information is chosen by hovering over a detector sector or clicking on a MIOCT module label. Hovering over the FiFo Status field shows a summary

of FiFo status.



3 BunchGrouper

BunchGrouper is an existing application written in C++ that I updated for the ATLAS upgrade during the long shutdown. BunchGrouper handles online per-bunch information in ATLAS. It can receive per-bunch information in different forms from ATLAS, the LHC, and other LHC experiments. Generic or custom algorithms can be applied and their result is being published to the ATLAS TDAQ Information Service or to the ATLAS conditions database

BunchGrouper reads and publishes histograms to ATLAS Online Histogramming (OH) service. The histograms are used to store the intensities of the two beams in LHC and combine this information to various BunchGroups. BunchGroups are binary histograms with 3564 bins, each representing one bunch crossing in LHC. For example the Physics BunchGroup indicates Bunch Crossings where both beams have a filled bunch. This indicates a collision. The information can be given to the trigger to only give accept signals when there should be collision.

The biggest upgrade I implemented was adding support for injection files. Injection files are files published by LHC. They give the series of beam injections used to fill the bunches. Previously one had to convert these files to a format understood by BunchGrouper. Now this conversion can be done in the BunchGrouper.

I added support for receiving messages in the BunchGrouper. The old version had already a system for passing on the commands, but they were not used for anything.

The old version of BunchGrouper had support for 8 BunchGroups. The new version has support for 16.

In addition I also corrected some minor bugs.