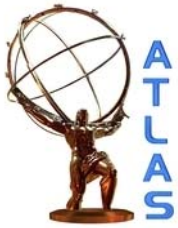
ATLAS Inner Detector Monitoring
(purpose, implementation and usage)

Arshak Tonoyan, Heidi Sandaker
University of Bergen

On behalf of ATLAS Inner Detector monitoring group

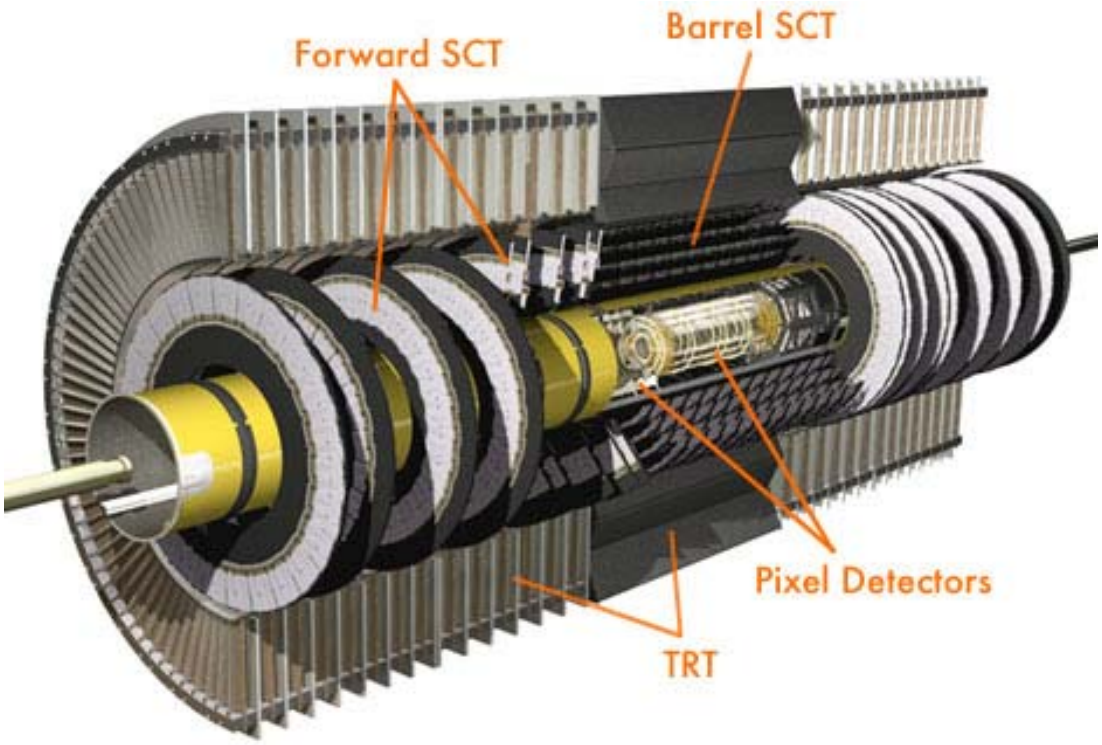
Nordic Workshop on LHC and Beyond
14 Jun 2008



Outline

- Quick introduction into ATLAS Inner Detector
- Individual sub-detectors monitoring
  - Pixel
  - SCT
  - TRT
- Global monitoring of Inner Detector
- Conclusion

ATLAS Inner Detector

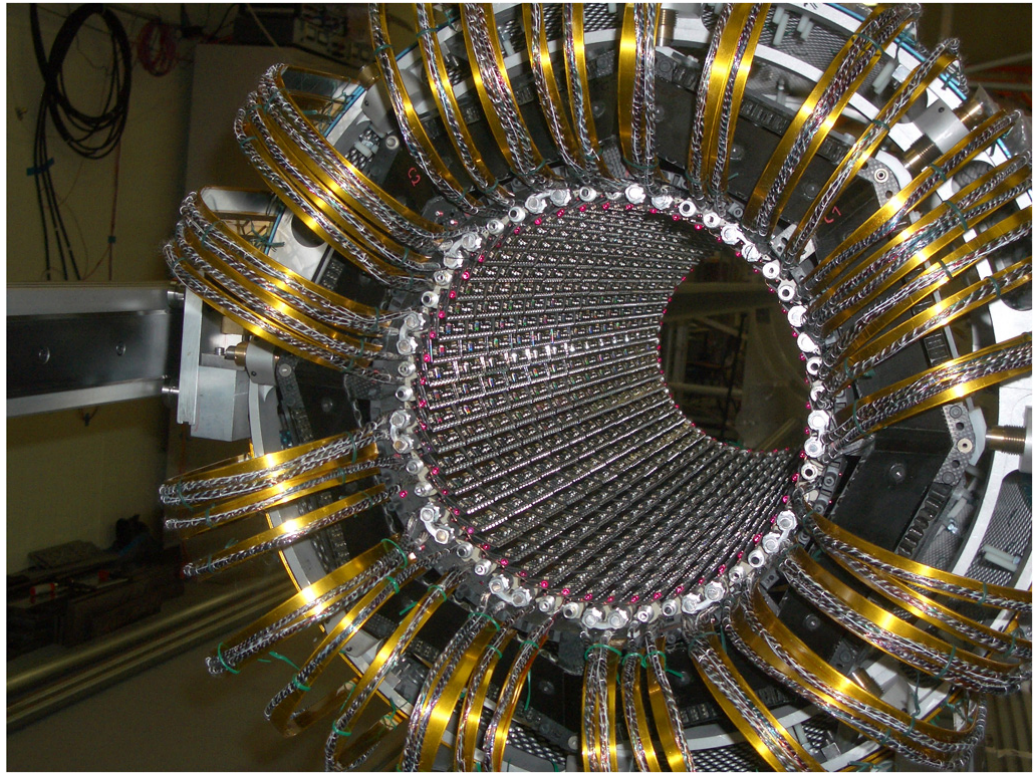


- Purpose is to provide
  - high precision efficient tracking
  - Vertex reconstruction
    - Primry
    - Secondary
  - Electron/pion separation
- Consists of 3 sub-detectors
  - Pixel detector
  - SemiConductor Tracker
  - Transition Radiation Tracker

Pixel Detector Monitoring

Pixel - Purpose

- **Aim of tools**
  - The PixelMonitoring is used both online and offline
  - Will monitor all relevant data which concerns the Pixel detector only
  - The online monitoring is optimised to look at:
    - Calibration of the detector
    - Identify modules that are noisy or turned off
    - Monitor the errors on the pixel bytestream
  - The offline tools are the same but with different histogram configurations
  - The pixel shifter should be notified about any problems of the order of minutes and be able to correct them

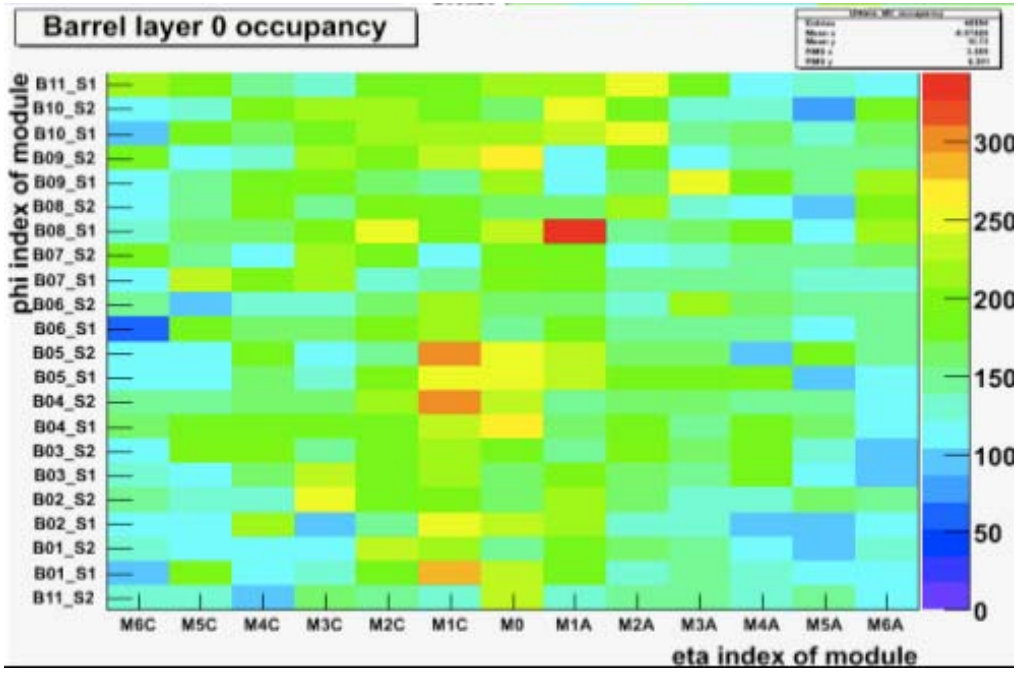


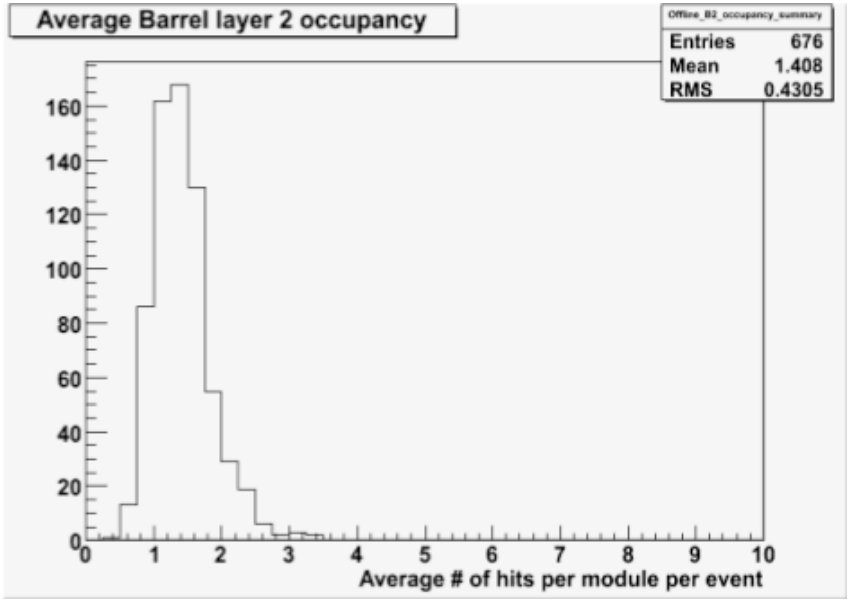
The Pixel Detector

Pixel – Example plots

Occupancy plots

- Easy and quick identification of dead and noise modules





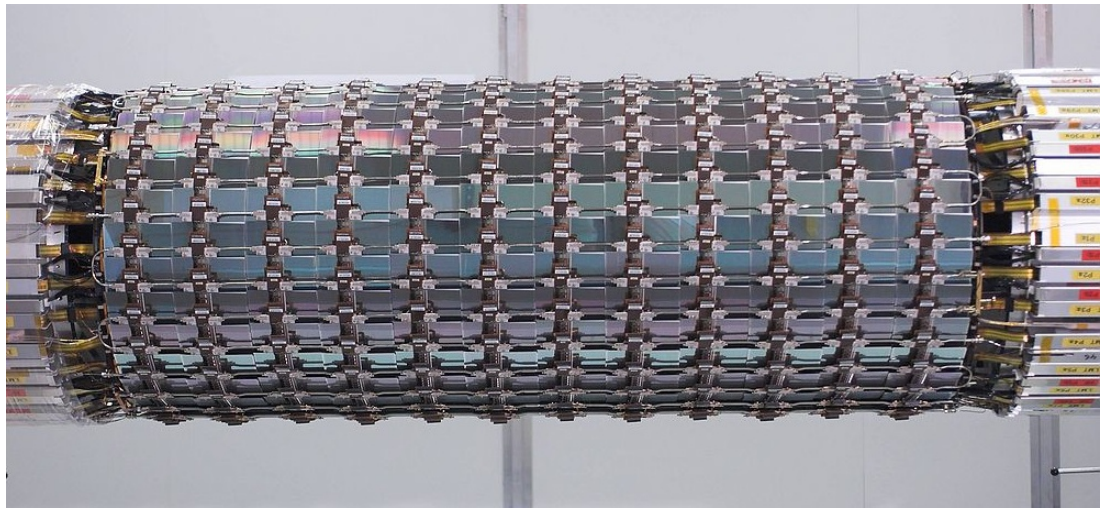
Occupancy summary plots

- One entry per all modules

SCT Detector Monitoring

SCT – Purpose

- **Aim of tools**
  - The SCT Monitoring should provide quick identification of any problems which may affect the data quality of this particular sub-detector
  - This includes all sub-detector specific information as well as SCT hits and track information
  - The monitoring can be divided into the following parts:
    - Offline
    - Online
    - ROS
    - DCS
    - Calibration

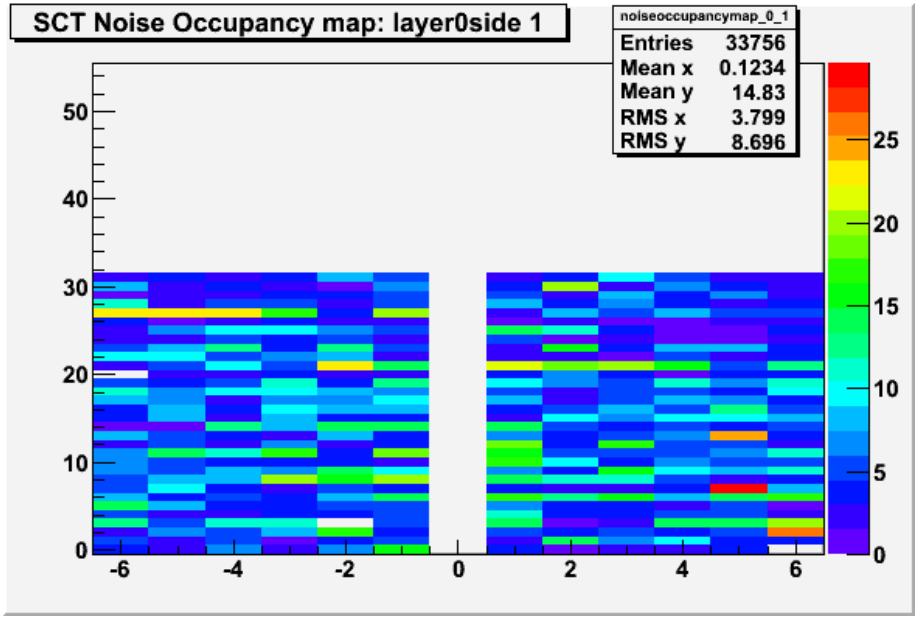


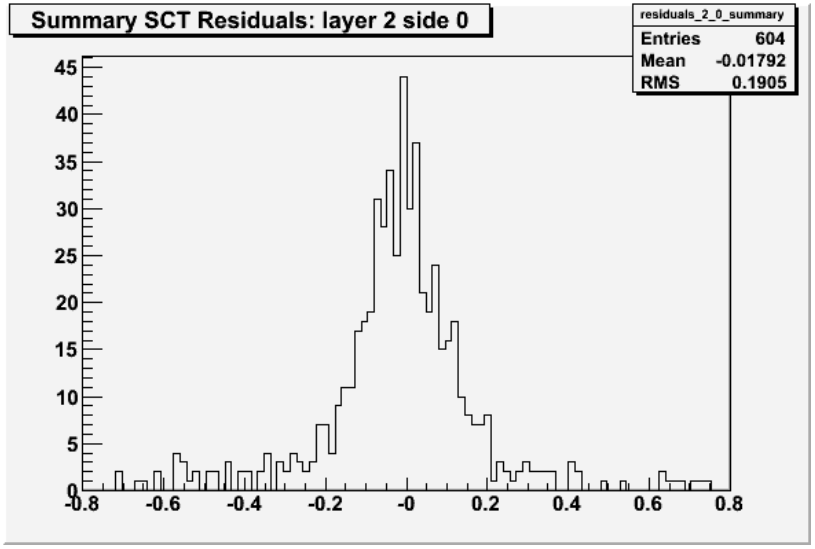
The SCT Barrel

SCT – Example plots

Noise occupancy plots

- Easy and quick identification of dead and noise modules/sides





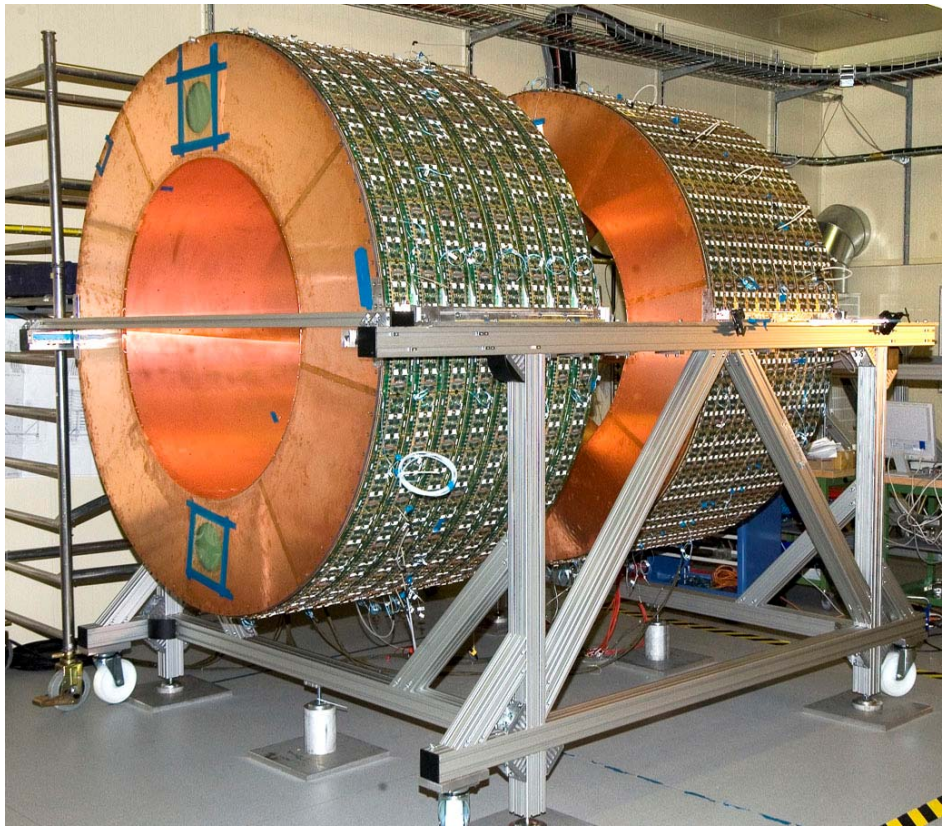
Residual plots

- Monitoring of tracking and alignment of modules

TRT Detector Monitoring

TRT – purpose

- **Aim of tools**
  - The monitoring should include timing, efficiency, noise, low level and high level hit response
  - It will look at three levels of read out chain (ROD, EF and offline)
  - The TRT monitoring consists of two tools;
    - the TRT Viewer
    - the Athena based monitorings

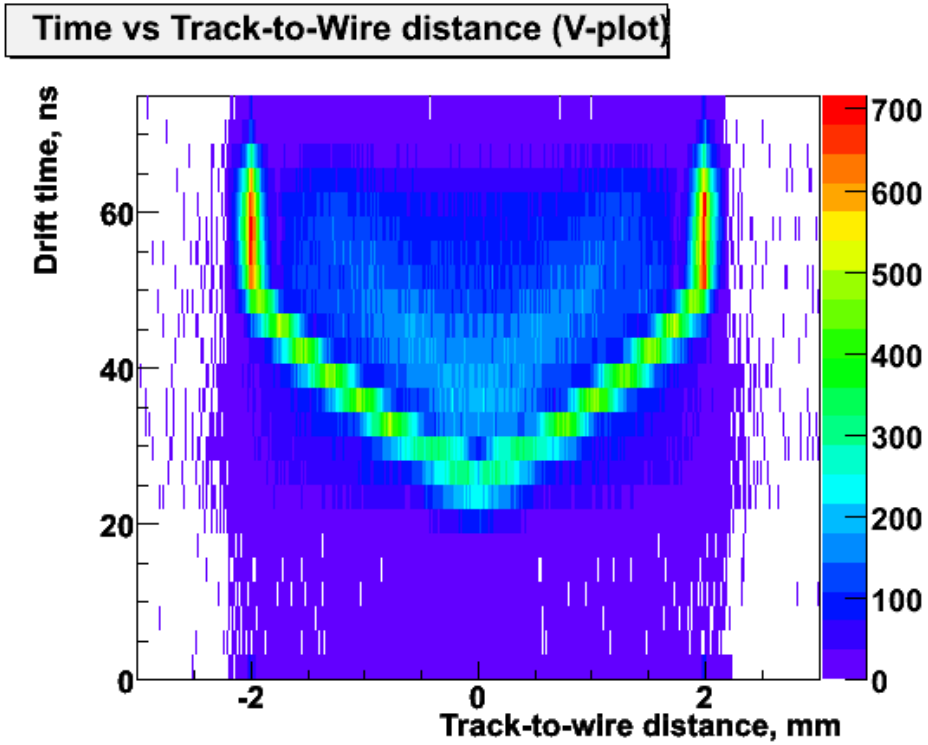


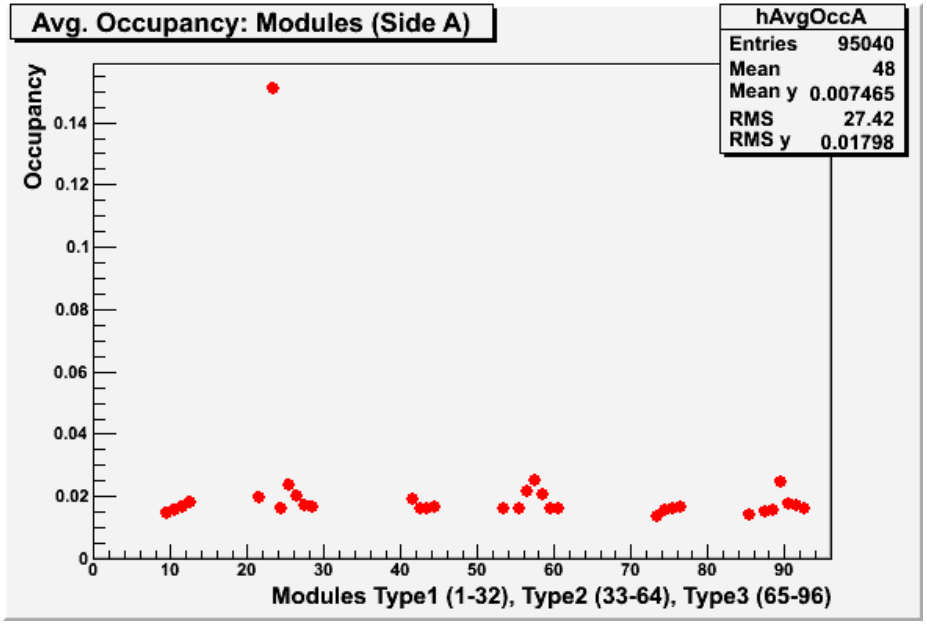
The TRT Barrel

TRT – Example plots

Drift time vs Track to wire distance plot

- Monitor R-T relation and compare with calibration





Occupancy plots

- Easy and quick identification of dead and noise modules

ID Global Monitoring

ID – purpose

- **Aim of tools**
  - Monitoring of synchronisation differences between 3 subdetectors: LVL1, BCID, matching of segments at the boundary surface of the 3 subdetectors.
  - Monitoring of noise occupancy correlation between the 3 subdetectors
  - Provide monitoring of hits, combined track information and residuals of combined tracks, looking for deviations in any ID part or phi sector
- **Particular aspects**
  - The ID Global Monitoring is a standalone tool which can work with any number of the subdetectors on. At least 2 subdetectors are needed for ID information to be produced, but one detector will be used for debugging purposes

ID - Implementation

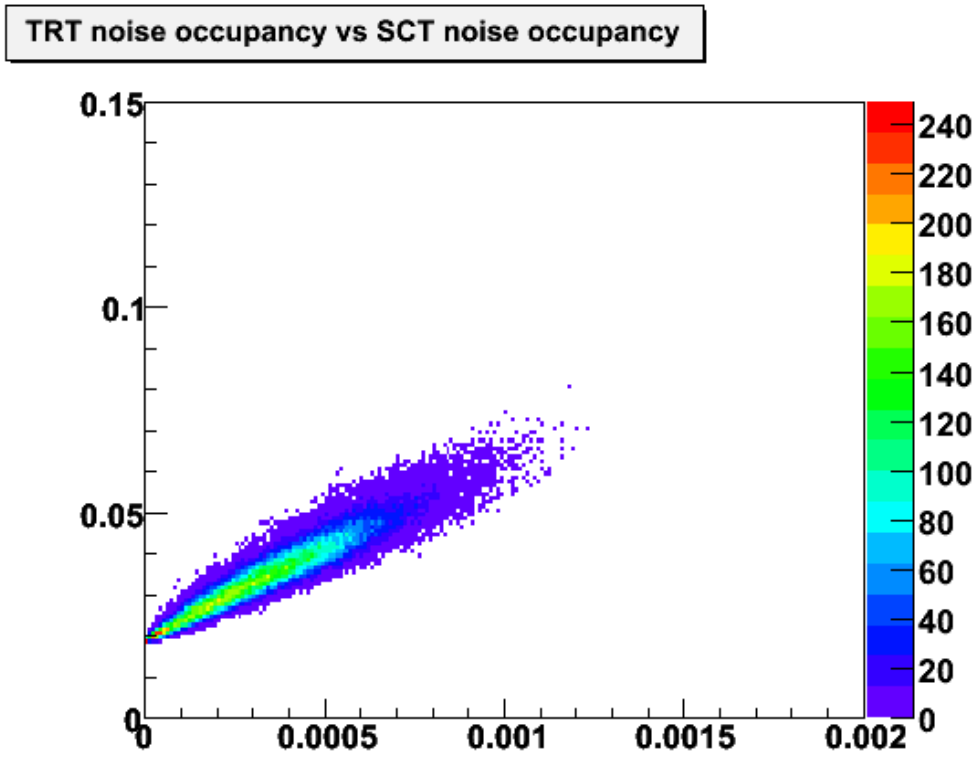
- **Current implementation**
  - **(cosmics)**
    - **InDetGlobalMonitorig package was developed for SCT+TRT tested on surface during summer 2006**
    - **Fully tested with cosmic muons**
    - **Since mid-2007 when Pixel detector was in place new functionalities were added to monitor it as well**
    - **Now Pixel is monitored on the same level as SCT and TRT**
  - **Full Dress Rehearsal – 2 (FDR2)**
    - **FDR2 is simulated events passing through the chain of Atlas Data Acquisition, triggers etc. as they are events from real detector.**
    - **Most of the monitoring tools were tuned for cosmics (histogram ranges, memory usage etc)**
    - **One have to retune for collision data and even drop some tools like TopBottom Monitoring**
    - **Due to memory consumption and algorithms migration issues lite version of InDetGlobalMonitoring was used during FDR2**

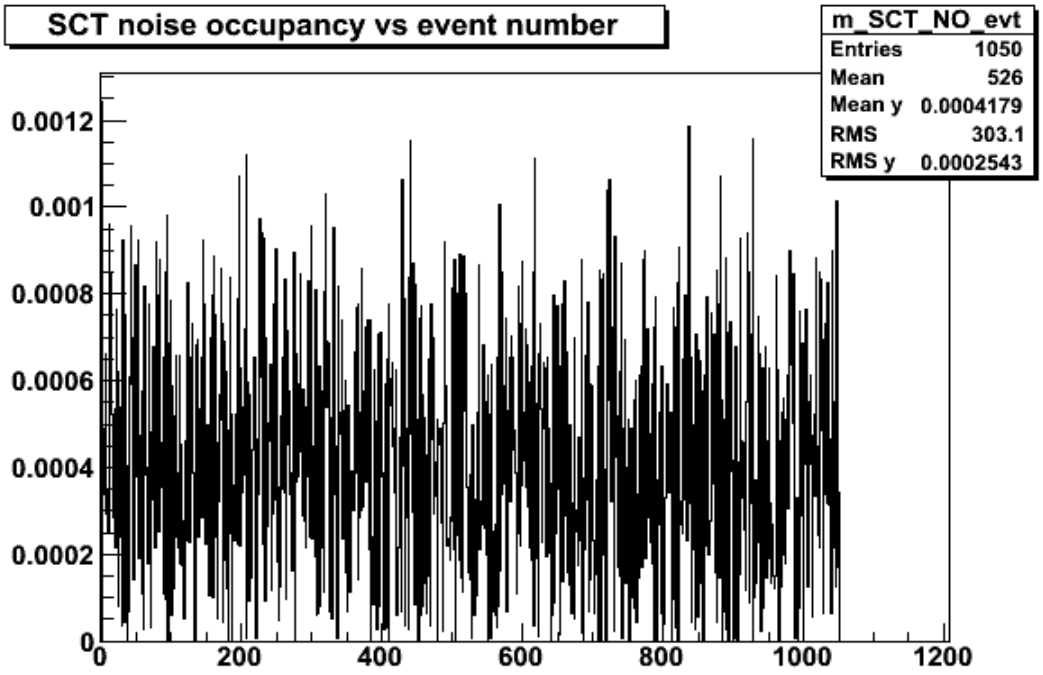
Existing tools

- **InDetGlobalSynchMonTool (monitoring of LVL1, BCIDs)**
- **InDetGlobalHitsMonTool (monitoring of number of hits, residuals on combined tracks)**
- **InDetGlobalSegmentsMonTool (matching of TRT, SCT & Pixel segments)**
- **InDetGlobalTopBottomMonTool (matching parameters of top and bottom tracks)**
- **InDetGlobalNoiseOccupancyMonTool (Pixel, SCT & TRT noise occupancies)**

NoiseOccupancy MonTool

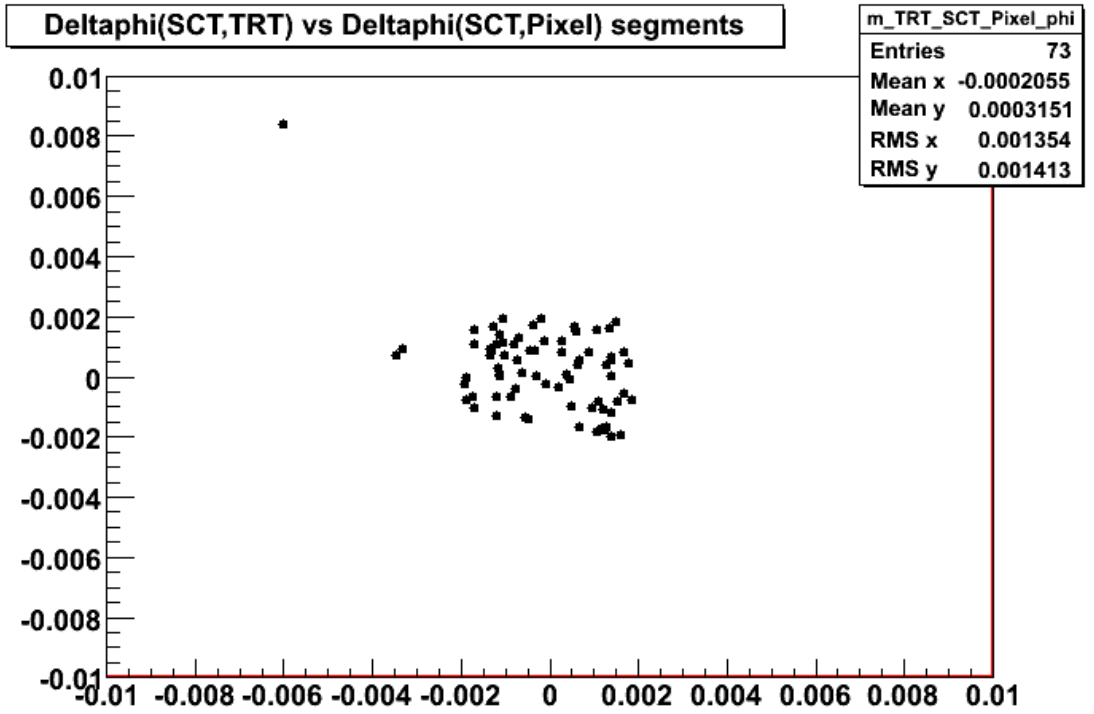
- Plots from FDR2 data
- Show clear correlation between SCT and TRT noise
- For debug purposes one can look at noise level of each detector as well

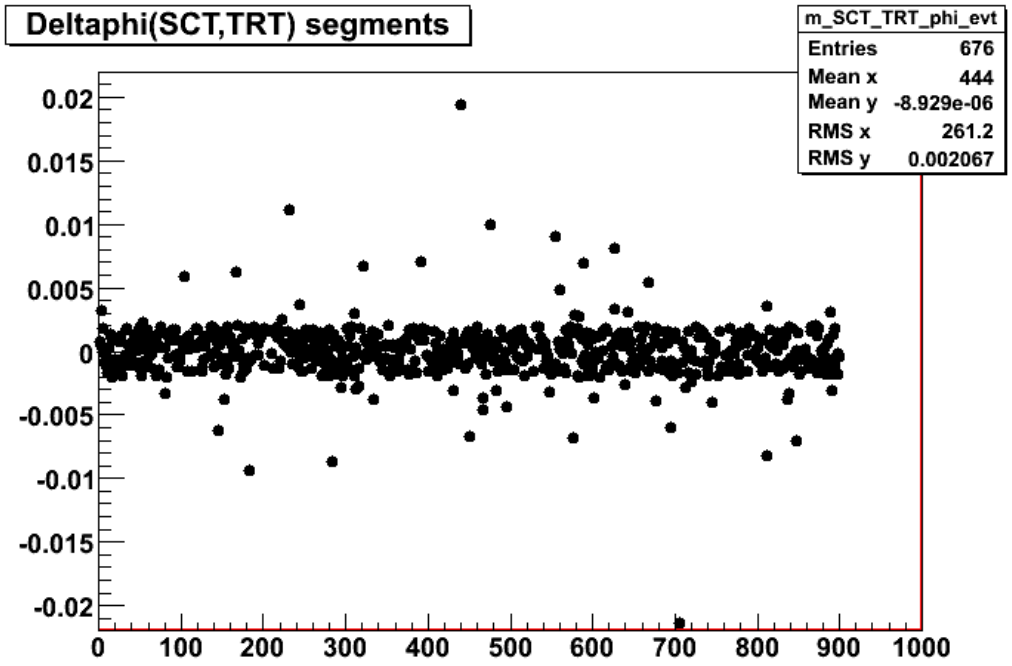




Segments MonTool

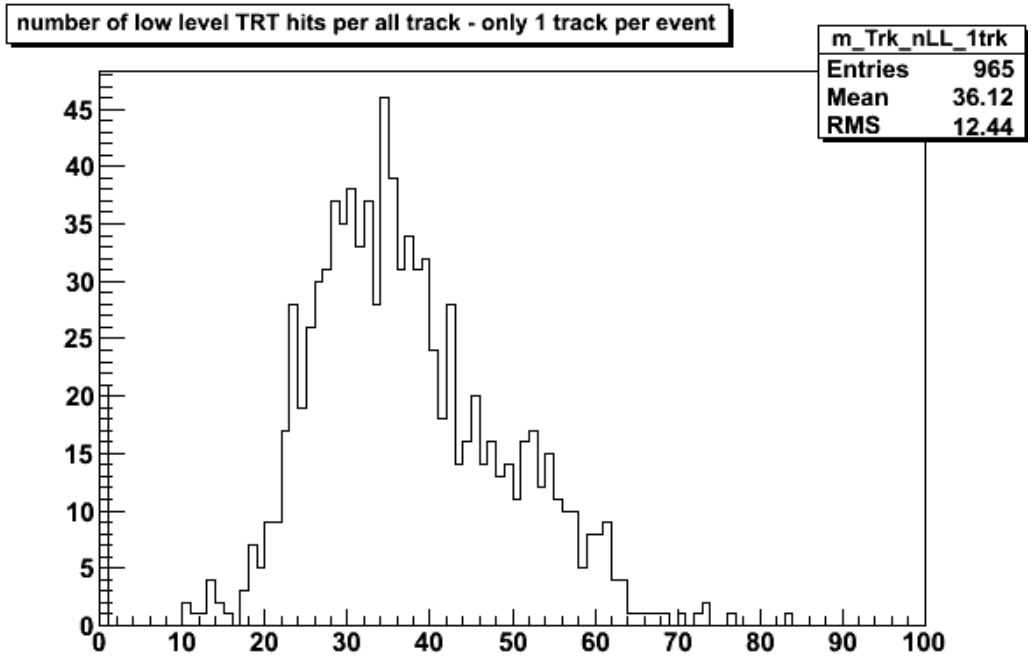
- Plots from simulated cosmic events
- Check delta_phi of track segments reconstructed independently in Pixel, SCT & TRT
- Mean should be ~0 unless detectors are misaligned or not synchronised

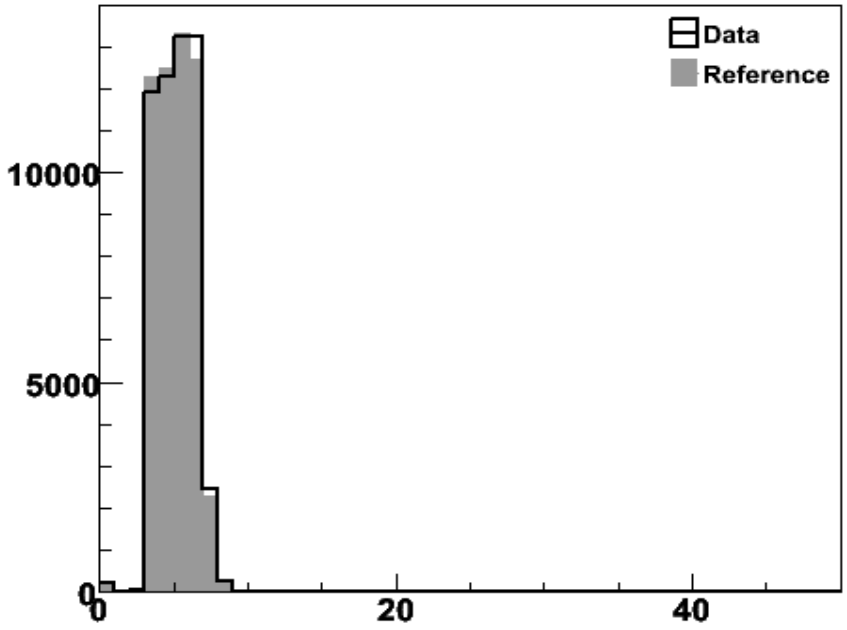




Hits MonTool

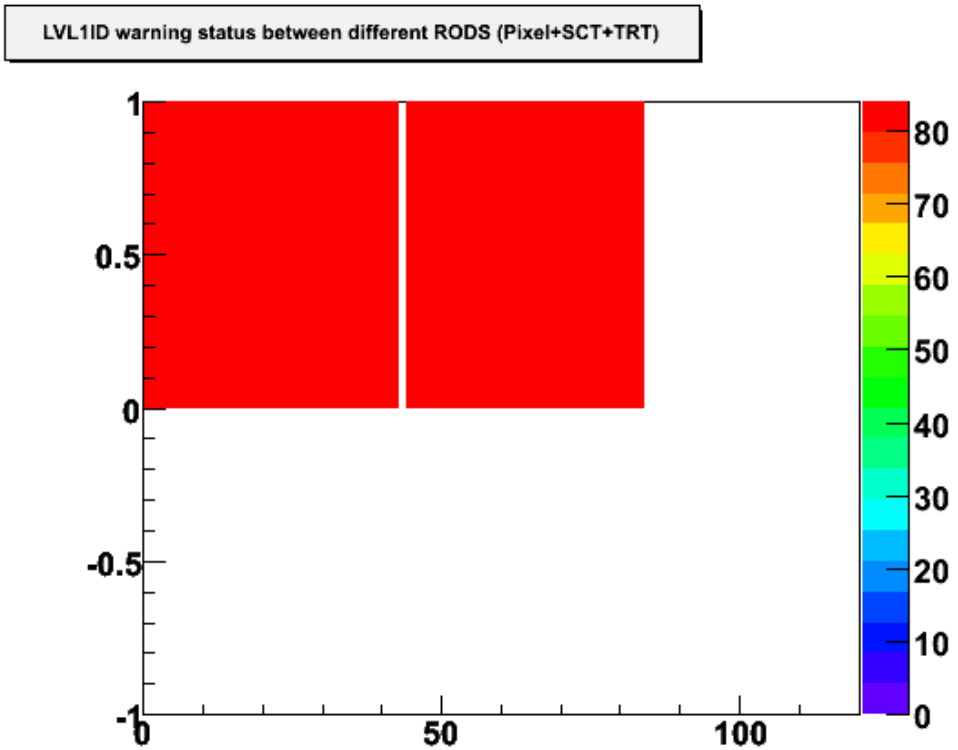
- Plots from cosmic muons and FDR2
- Tool is mostly needed to control if tracking works as it is supposed
- This quantity is monitored by particular sub-detectors as well
- Another purpose is to cross check these histograms and histograms from Pixel, SCT and TRT monitoring packages





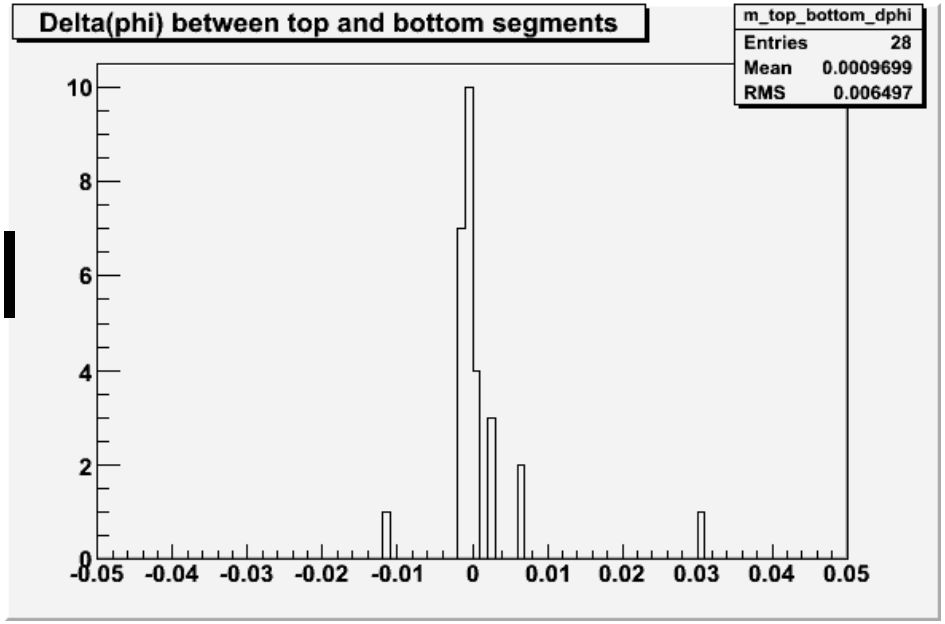
Synchronisation MonTool

- Checks LVL1ID and BCID between subdetectors and even between different parts of a particular subdetector
- If synchronised 0 is filled but is any mismatches -1 will be flashed
  - The gap in plot is introduced to separate RODs from different sub-detectors



TopBottomMonTool

- Compares track parameters between tracks reconstructed in top and bottom sectors of the detector
- Not useful for collision data in current shape



Conclusion

- Monitoring (both online and offline) is the first step of understanding the detector
- Monitored quantities allow quickly observe possible problems and solve them
- Monitoring is in a good shape both at individual sub-detector level and at Global level
- But still many issues exist
  - Code is not always written in an optimal way
    - Recently all monitoring code has been reviewed and changes are being implemented now
  - Memory and CPU consumption is rather high
  - Not all the histograms are optimized