The DAQ system for the Fluorescence Detectors of the Pierre Auger Observatory

S. Argiro, University of Torino and INFN, A. Kopmann, O. Martineau,
H.-J. Mathes, Forschungszentrum Karlsruhe

Abstract

The Pierre Auger Observatory currently under construction in Argentina will investigate extensive air showers at energies above 10^{18} eV. It consists of a ground array of 1600 Cherenkov water detectors and 24 fluorescence telescopes to discover the nature and origin of cosmic rays at these ultra-high energies.

The ground array is overlooked by 4 fluorescence buildings equipped with 6 telescopes each. An independent local data acquisition (DAQ) is running in each building to readout 480 channels per telescope. In addition, a central DAQ merges data coming from the ground detectors and all fluorescence buildings.

The system architecture follows the object oriented paradigm and has been implemented mainly based on widespread open source tools for interprocess communication, data storage and user interfaces.

Each local DAQ is connected with further sub-systems for calibration, monitoring of atmospheric parameters and slow control.

After a prototype phase to validate the system concept since September 2003 the first telescopes started taking data in the final setup. The data taking will continue during the construction phase and the integration of all subsystems.

We present the design and the present status of the system currently running in two buildings with a total of 12 telescopes installed.

INTRODUCTION

The Pierre Auger Project [1] is an international effort to gather high statistics of cosmic rays at the highest energies. The installation will have an array of about 1600 particle detectors spread over 3000 km^2. Atmospheric fluorescence telescopes placed at the boundaries of the ground detector array will record showers that strike the array. The two air shower detector techniques working together form a powerful instrument for these studies.

After a successful operation of a prototype detector in the province Mendoza in Argentina for several months the construction of the final observatory started in 2003. At present more than 430 detector stations and two fully equipped fluorescence telescope stations are operational and the third one is expected to become operational soon.

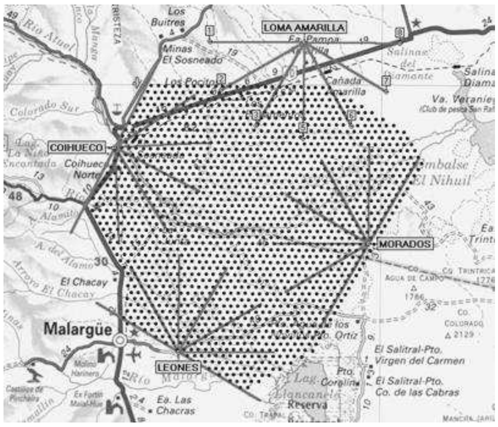

Figure 1: Layout of the southern observatory, the 4 FD telescopes are located at the borderline of the ground array (SD).

THE FD DETECTOR

The Auger fluorescence detector (FD) is expected to operate in conjunction with the surface detector (SD). Its primary purpose is to measure the longitudinal profile of showers recorded by the SD whenever it is dark and clear enough to make reliable measurements of atmospheric fluorescence of air showers. The integral of the longitudinal profile is used to determine the shower energy, and the speed of shower development is an indicator of the primary particles mass. The longitudinal shower profile measured by the fluorescence detector provides a model-independent measure of the electromagnetic shower energy. The hybrid detector will also have better angular resolution than the surface array alone. It is expected to be of value in studying tightly clustered arrival directions.

One observatory eye with a field of view of 180 $^\circ$ in azimuth and 30° in elevation is built from 6 telescopes each covering $30^\circ \times 30^\circ$. To achieve this wide field of view with a reasonable effort and good optical quality the layout of a Schmidt telescope is adapted. The elements of a telescope are the light collecting system (diaphragm and mirror) and the light detecting camera (a PMT array) which is composed of 440 hexagonal pixels arranged in a 20×22 matrix.

In the air fluorescence technique, the atmosphere can be assimilated to the radiator of a calorimeter. Therefore it is important to monitor its varying optical properties while precisely calibrating the PMT camera. In particular, the DAQ has been designed to manage, besides regular data taking, the following ancillary tasks:

- the absolute calibration which is rather time consuming and thus carried out only at larger time intervals
- the relative calibration carried out at the beginning of each night shift to monitor the short-term variation of the cameras electronics and the parameters of the optical system
- a constant monitoring of the atmospheric parameters using various systems in the field like LIDAR, weather stations, roving lasers and aerosol phase function and horizontal attenuation monitors
- the highest energy events will require an immediate measurement of the actual atmospheric conditions in the vicinity of the shower track to have a good knowledge of all parameters which influence the reconstruction in order to minimize the systematic uncertainties (shoot-the-shower philosophy).

Some of these mentioned systems produce actively light which would interfere with the FD DAQ operation thus the DAQ and/or the front-end must be locked against artificial light sources when these light sources are active. In the case of artificially generated events like laser shots, a tag must be produced in order to check the trigger and reconstruction performance of hardware and software.

Due to the difference in design of the SD and FD detectors, different energy thresholds for the detection of air showers follow. The FD threshold is lower than the SD threshold which means that FD is capable of triggering SD sub-threshold events. Events seen in both detectors will allow cross-checks and cross-calibration. To use the possibilities of the so called hybrid-trigger, the time given by the GPS system is used to tag all events in order to allow a synchronization of distant detector system with an accuracy better than 100 nsec. As the data buffering capability of the SD stations and the SD radio network bandwidth is limited, certain time constraints have to be respected in order to enable the cross-trigger possibility for sub-threshold events.

Finally, the distance between different FD detector sites and the central DAQ (CDAS) requires the possibility of remote operation and even to the greatest possible extent an automatic operation following a certain programmed ahead schedule.

THE LAYOUT OF THE DAQ

Fig. 2 gives an overview about the structure of a single telescope station (FD Eye). Each of its 6 PMT cameras is controlled by the disk-less so called MirrorPCs which are connected to the EyePC via the Eye-LAN. The EyePC hosts the software for event building, the run control and the disks for system, program and data storage. It provides the Eyes computer systems with the basic networks services (routing, name service, DHCP/WLAN,...). All tasks usually denoted as slow control tasks are carried out by an independent Slow Control PC. This system has additionally to fulfill all functions relevant for the safety of the telescope hardware. The described set of computers is completed by the GPS clock and a few more computers or devices coupled via the LAN in charge of controlling the various calibration and atmospheric monitoring systems.

As operating system for the computers running the DAQ, the calibration and most of the atmospheric monitoring tasks Linux has been chosen. It allows to carry out most of the tasks on standard PCs. The MirrorPCs are operated as disk-less clients whereas for the EyePCs systems which provide a higher redundancy (RAID) are chosen. The main design goals of the system, low price, good testability, high reliability and flexibility of the trigger has been met by the consequent use of FPGAs for the front-end electronics [10]. The use of FPGAs together with large event buffers in the hardware helps to implement most of the real-time aspects already in hardware - the software is then less time critical.

The software is mainly coded in C++ and follows the object oriented approach, i.e. object oriented design patterns have been used where applicable and feasible.

Each telescope building is coupled to the central computer site of the observatory via a micro wave link. During the nightly data taking only fast trigger information is exchanged with the Central Data Acquisition System (CDAS), but the data are stored locally and are mirrored to the central computer site during daytime. As one step during this data mirroring process, the files from different FD Eyes and SD are merged together yielding common data files for the whole observatory.

Telescope level DAQ

The DAQ at the telescope level has the tasks to

- handle triggers generated by the trigger logic (SLT) or external signals
- apply higher level trigger criteria
- fully or partly read out the data of the event
- send the event data to the central event builder running on the EyePC
- gather monitoring information for the purpose of supervision and automatic operation
- host various daemons necessary to set parameters of the acquisition hardware and allow diagnostic software the access to the hardware (RPC)

The interface between hardware and the MirrorPC is based on the Fire-Wire (IEEE-1394) protocol. The advantage of this approach is that standard device drivers available in newer Linux kernels can be used. The design allows high throughput for readout of larger data blocks. The bulk of our data consists of the FADC digitization results of the

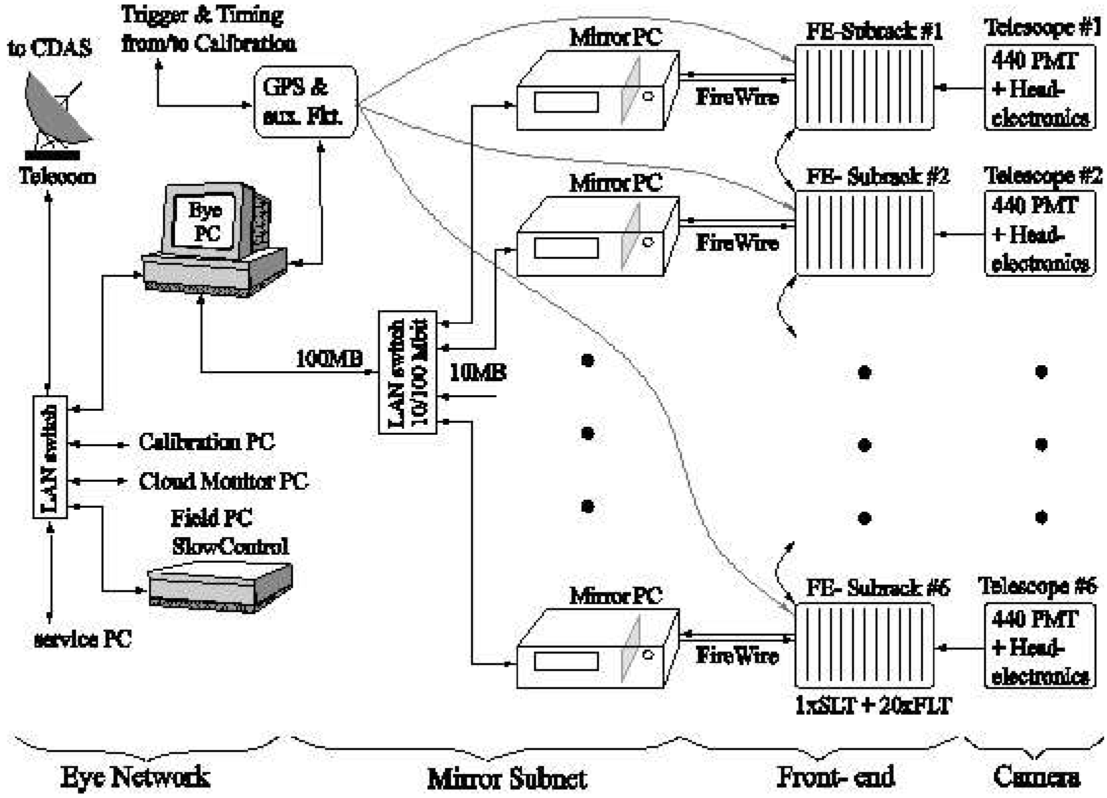

Figure 2: The hardware layout of the DAQ system at telescope station level.

PMTs with a resolution of 100 nsec. The PMT trigger rate (T1) is controlled to a value of 100 Hz. For typical conditions the rate of PMT patterns with a track like signature (T2) is if the order of 0.3 Hz. The software trigger algorithm selects about 5 % of these triggers as possible shower candidate events, whereas the expected rate of showers per camera is of the order of $3 - 4h^{-1}$.

As communication means between the processes different low and high-level protocols are used. At the mirror PC level, where the hardware readout is performed, the data structures are mapped into shared memory. On the other side, the transport to the Eye level and the storage is taking advantage of the ROOT I/O system [3]. The ability of data objects to be streamable intrinsically allows all types of I/O.

The communication layer used for all demands of the run control is based on the CORBA protocol [4]. Though this protocol is not particularly well suited for online purposes, it offers several advantages in terms of flexibility and location independence. In addition, it is easy to switch between the test and deployment environment almost transparently. In particular the FD DAQ is using the open source MICO CORBA implementation [5].

The central CORBA facility which allows us to achieve this flexibility is the CORBA Naming Service whose daemon is started at boot time. This tool is a part of the chosen CORBA implementation.

Eye level DAQ

The EyePC in each telescope building provides the basic infrastructure for the Eyes computer system and the DAQ. All central services of the DAQ are located on the EyePC - in particular the already mentioned CORBA Naming Service and a daemon for the GPS clock. All communication with external systems is performed by processes running on this system. At the current stage of the DAQ development the EyePC hosts also the GUIs for the run control and for the event display.

Event building The most important tasks from the data taking point of view of the EyePC is the event building. Besides the pure event building task based on the GPS derived timing information recorded with each sub-event, a few more requirements influenced the development of the event builder:

- built events must undergo a software trigger algorithm which flags physical events (T3)
- in order to allow to send a cross-trigger to the DAQ of the SD detector, all steps must be done within a certain time limit (actually 5 sec)
- events inserted by various calibration and atmospheric monitoring light sources must be detected, flagged and sorted out in order to prevent overflowing the hybrid trigger

The event builder is built around a dispatcher implementing a reactor pattern and which starts handlers for incoming connections and the different objects sent by the Mirror level DAQ systems on demand. Events are built after a certain timeout in order to account for transport delays and the CDAS timing requirements. Then they are passed to the T3 (software trigger) algorithm and receive their specific tag. Depending on the type of this tag events are finally sank into one or more data sinks and physics events are sent as trigger information to the SD. The design of these data sinks makes it easy to plug in further sinks if the experiment would require it.

Run control The run control task is divided into two tasks the Eye Run Control GUI (implemented with the Qt toolkit [6]) and the so called Eye Status Manager (ESM) which runs as a daemon in the background and has the control over all program instances in the DAQ system. The main task of the ESM is to control the system configuration, change it on request and distribute the messages of the run control to all software and hardware components in the system. It is supported by a daemon on each node with the specific task to invoke and manipulate processes. In principle this can be seen in a very simplified picture as a kind of CORBA Implementation Repository.

This partitioning which means basically a separation of the run control logic from the presentation layer has the advantage to leave the DAQ intact in case of network failures between the remote FD eye and the operators console and for allows for automatic scripting which is anticipated for the future.

Slow control interface The Slow Control System is implemented on a Field PC running with the 4CONTROL PLC [7] (Programmable Logic Controller) software based on Windows NT. Sensors and actuators are connected to the Field PC via a wide spread field bus standard (ProfiBus/EN 50170) which allows to use commercially available hardware and shortens the development time.

The communication with the tasks running on the Slow Control PC can in principle be achieved via the OPC protocol (OPC = OLE for Process Control) which is a communication standard for automation systems based on DCOM. This requires however additional proprietary software to be installed on the DAQ system and it turned out that this causes also some problems to be integrated smoothly with other software packages on Linux. In particular, incompatibilities in the binary format of third party libraries together with the fast change of compiler versions on the Linux side have been identified as causing these problems.

However by using the new XML based extension of the OPC standard [8], the communication is then based only on easier manageable exchange of XML packets via native TCP/IP sockets.

Logging Important for the monitoring of the performance of a distributed DAQ application is a central logging system. It provides to all applications of the DAQ a mean to send uniformly formatted logging messages to the same stream. We intend to accomplish this task with the use of the CMLOG [9] logging package.

STATUS AND OUTLOOK

We have developed a distributed system for the DAQ of the fluorescence detectors of the Auger Observatory. It has been successfully used for several months on the field. In parallel, we are still in the process of developing and fine tuning the system. The main activities of the further development are:

- validation and improvement of the present trigger algorithms
- identify and cure potential performance bottlenecks
- improvements of the detector and data quality monitoring
- integration with the central logging facility

The use of CORBA has shown its potential for integrating distributed application. However, network or other hardware failures which have some influence on this communication layer made the code writing, especially the error handling a challenging task.

REFERENCES

[1] http://www.auger.org

[2] AUGER Collaboration et al., Nucl.Inst. and Meth. **A 523** (2004) 50

[3] http://root.cern.ch

[4] http://www.omg.org

[5] http://www.mico.com

[6] http://www.trolltech.com/products/qt/index.html

[7] http://www.softing.com

[8] F. Ivanitz, Computers and Automation **4** (2004) 78; http://www.opcfoundation.org

[9] http://www.slac.stanford.edu/grp/cd/soft/cmlog/

[10] H. Gemmeke for the AUGER Collaboration, Proc. of 27th ICRC, Hamburg, (2001) 699