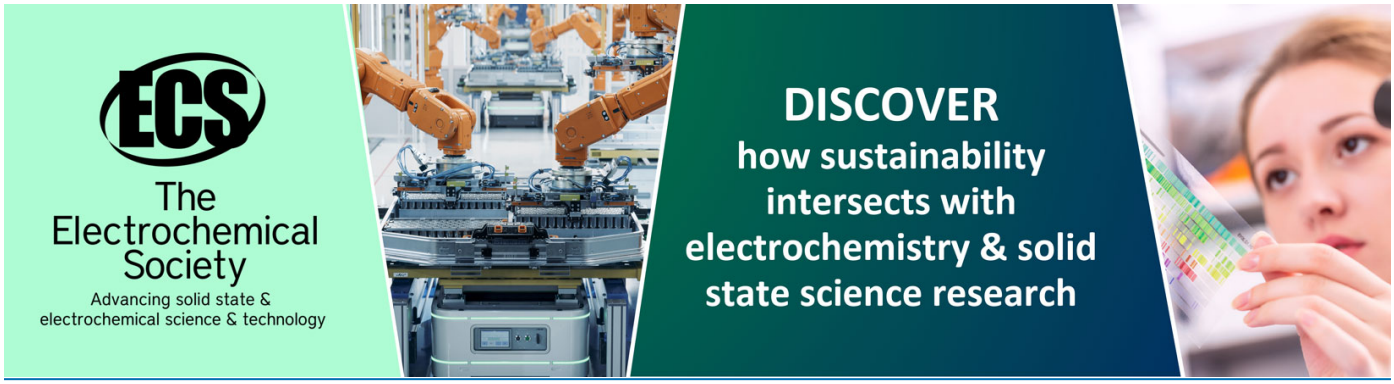
# WEB based online event displays for KASCADE-grande





Journal of Physics: Conference Series **119** (2008) 072028 doi:10.1088/1742-6596/119/7/072028

# WEB based online event displays for KASCADE-Grande

**H Schieler and A Weindl (KASCADE-Grande Collaboration)**

Forschungszentrum Karlsruhe, POB 3640, 76021 Karlsruhe, Germany

E-mail: schieler@ik.fzk.de, weindl@ik.fzk.de

**Abstract**. For three detector components of the KASCADE-Grande experiment, WEB based online event displays have been implemented. They provide, in a fast and simplified way, actual information about energy deposits and arrival times of measured events, and the overall detector status. Besides the aspect of being able to show air shower events to interested people wherever there is an internet access available, these event displays are an easy and highly useful tool for controlling and maintaining tasks from remote places. The event displays are designed as client-server applications, with the server running as independent part of the local data acquisition. Simplified event data are distributed via socket connections directly to the java applets acting as clients. These clients can run in any common browser on any computer anywhere on the planet.

## 1. Introduction

The KASCADE-Grande experiment is a multi-detector installation at the site of the Forschungszentrum Karlsruhe, Germany, to measure and study extensive air showers induced in the atmosphere by primary cosmic rays in the energy range from $10^{14}$ to $10^{18}$ eV [1].

Three of the detector components with their "flat" design are predetermined for 2-dimensional event displays. Figure 1 shows the site of the Forschungszentrum Karlsruhe with those detectors.

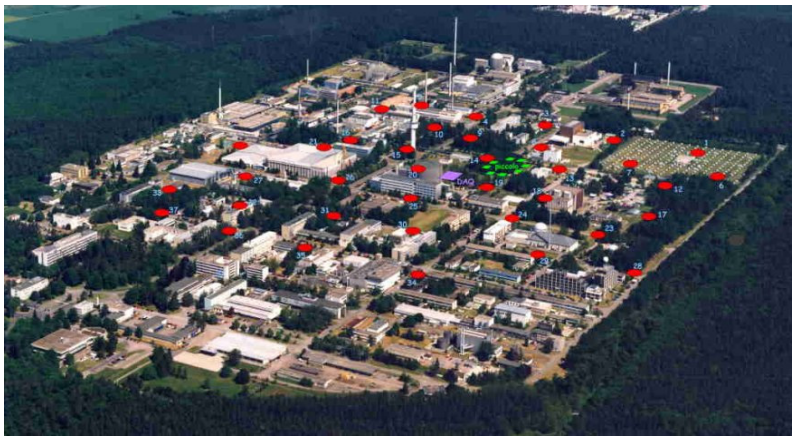


**Figure 1.** KASCADE-Grande at the Forschungszentrum Karlsruhe.

i. In the upper right of figure 1 you can see the KASCADE Array with its 252 detector stations arranged on a square of 200m x 200m with a spacing of 13m. Each station contains electron/gamma and/or muon detectors.
ii. The 37 stations of the Grande Array marked with red dots in figure 1 are distributed on a square of 700m x 700m and measure the charged component of the air showers.
iii. The green dots in figure 1 mark the 16 stations of the Piccolo Trigger Cluster arranged in 8 huts, also measuring the charged component.

As shown in figure 2, each component of the KASCADE-Grande DAQ has its own local data acquisition. It is running and managing the detectors, taking data and sending them to the central data acquisition, where collecting and merging of the data and building common events is done. In parallel to that main data stream, the three mentioned components send their data to their corresponding web server applications, one instance running for each component.

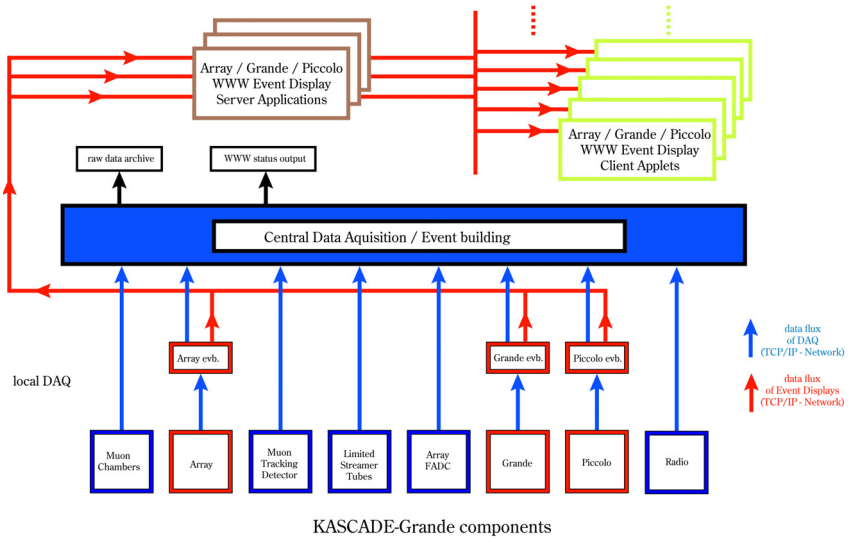



**Figure 2.** The KASCADE-Grande DAQ.

## 2. Server applications

The event displays are designed as client-server applications, with the servers running steadily as independent parts of the local data acquisition. The three separate server instances are created from one source code using *define* options at build time. They all look the same, except the background color of the windows is different. Figure 3 shows the screenshot of the Grande Server WWW Control window.

All three servers have the same functionality:

- handling connection to local DAQ
- receiving, processing and preparing data
- providing the socket for the clients
- managing client connections
- distributing data to the connected clients

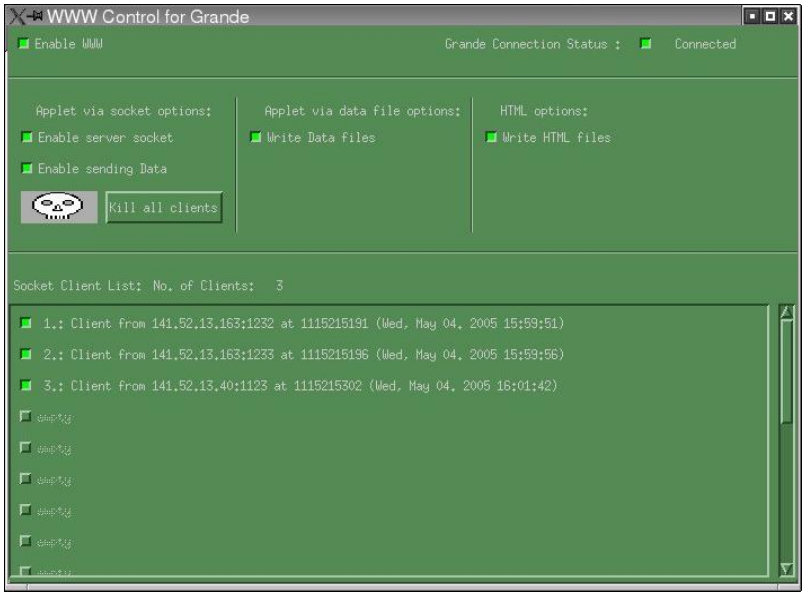



**Figure 3.** Grande Server WWW Control.

Server applications are written in *C* and *Motif* and are designed to compile and run on both *Linux* and *Digital Unix*.

## 3. Client applications

The multi-threaded client application is written in Java using *Swing*. There is basically the applet frame class with the controls and a class displaying the data (first thread). Behind that, as a second thread, there is the class managing the socket connection and receiving the data.

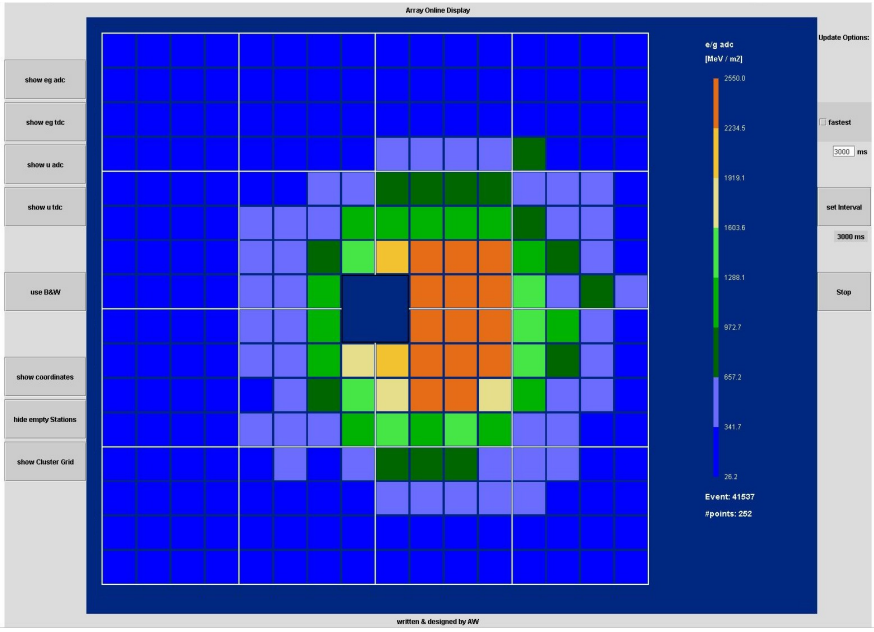



**Figure 4.** The KASCADE Array applet.

A client is started by calling a web site with the applet integrated. The client-server connection is established automatically on load and displaying events starts immediately. Figure 4 shows the KASCADE Array applet with display area and the left and right control panels.

The applet offers some control possibilities: For example the interval between displayed events can be set in milliseconds or the applet can be switched to mode "fastest". The first mode means, events are polled by the applet from the socket reader class and passed to the display. The second mode means, data packages are passed to the display class via the applet as soon as they arrive. Another control feature is to select which data representation of the event should be shown. Data packages are always sent complete, meaning one package contains a full single event measured by the detector. So users can switch between energy deposit and arrival time representation of the same event. For the KASCADE Array with its two types of detectors, one can switch between muon and electron/gamma and their representation, too. In addition, some display details can be switched "On" or "Off".

Using *JavaScript* to be informed about resize events of the browser window and to query for the actual window size, the applet fits the maximum available area. The display area itself can also be enlarged by double-clicking on the applet, which hides or shows the left and right control panels.

## 4. Compatibility and performance

The applet application has been tested successfully using several versions of *Sun's Java Runtime Environment* (1.3.x-1.6.x) in different browsers and versions (*Netscape* since 6.2; *Opera* since 6; *IE* since 6.x; *Firefox* since 1.0; *Konqueror* since 3.2) running on different *Linux* and/or *Windows* systems.

The online display client-server system is NOT designed to guarantee for every event to be displayed. A major design goal was indeed, to guarantee that the online displays don't disturb the main data acquisition stream in any way. Due to this, servers receive and process data only on idle CPU times. The web servers don't process and distribute (and therefore the applets can't show) each single event. Another limitation is the available client-server connection bandwidth. An array data package is about 8 kB in size. If only a modem connection is available one wouldn't succeed in seeing the full 5 Hz event rate (mean event rate of KASCADE-Grande). In addition to the server and connection limitations, the hardware where the client is running may also be a restricting factor. Depending on the java runtime environment, its integration in the browser, the display features switched on or the refresh rate selected, some machines may reach their CPU power limits. The final limitation is the observer himself, since 5 Hz is a high rate of events to follow by eye. It is also impossible to catch a certain event to view both of it's representations, energy deposits and arrival times.

But even connected via LAN and with the fastest options chosen, in average only every second or third event is displayed.

## 5. Summary

We designed and implemented WEB based online event displays for three detector components of KASCADE-Grande. These client-server applications provide, in a user friendly way, a fast and easy access to information about the detector status. In addition, they allow interested people to observe air showers online wherever a computer with a browser and a connection to the WWW is available.

## References

[1] Antoni T et al. 2003 The cosmic-ray experiment KASCADE *Nucl. Instrum. Methods* A **513** 490–510