STATUS OF THE ACS-BASED CONTROL SYSTEM OF THE MID-SIZED TELESCOPE PROTOTYPE FOR THE CHERENKOV TELESCOPE ARRAY (CTA)

Peter Wegner, Bagmeet Behera, David Melkumyan, Stefan Schlenstedt, Torsten Schmidt, Ronny Sternberger, Stephan Wiesand, Deutsches Elektronen-Synchotron, DESY, Platanenallee 6, D-15738 Zeuthen, Germany

Oguzhan Anguner, Emrah Birsin, Igor Oya, Ullrich Schwanke, Institut für Physik, Humboldt Universität zu Berlin, Newtonstrasse 15, D-12489 Berlin, Germany

Matthias Fuessling, Universität Potsdam, Potsdam-Golm, Germany

Abstract

CTA, as the next generation ground-based very-high-energy gamma-ray observatory, is defining new areas beyond those related to physics; it is also creating new demands on the control and data acquisition system. With on the order of 100 telescopes spread over a large area with numerous central facilities such as a weather monitoring system, CTA will comprise a significantly larger number of devices than any other current imaging atmospheric Cherenkov telescope experiment. A prototype for the Medium Size Telescope (MST) of 12m diameter has been installed in Berlin and is currently being commissioned. The design of the control software of this telescope incorporates the main tools and concepts under evaluation within the CTA consortium in order to provide an array control prototype for the CTA project. The readout and control system for the MST prototype is implemented within the ALMA Common Software (ACS) distributed control middleware. The interfacing to the hardware is performed via the OPen Connectivity-Unified Architecture (OPC UA). The storage system of the prototype uses two different database systems: MySQL and MongoDB. MySQL keeps configuration data, while MongoDB stores monitoring data, log messages, alarm information and CCD images.

In this contribution the architecture of the MST control and data acquisition system, implementation details and first conclusions are presented.

THE CHERENKOV TELESCOPE ARRAY

The Cherenkov Telescope Array (CTA) project [1] is an initiative to build the next generation ground-based very high (VHE, E > 10 GeV) energy gamma-ray instrument. It will serve as an open observatory to a wide astrophysics community and will provide deep insights into the non-thermal high-energy universe.

The present generation of imaging atmospheric Cherenkov telescopes (H.E.S.S. [2], MAGIC [3] and VERITAS [4]) has in recent years opened the realm of ground-based gamma-ray astronomy in the energy range above a few tens of GeV. The Cherenkov Telescope Array will explore our Universe in depth in VHE gamma rays and investigate cosmic non-thermal processes, in close cooperation with observatories operating at other wavelength ranges of the electromagnetic spectrum, and those using other messengers such as cosmic rays and neutrinos.

Besides the anticipated high-energy astrophysics results, CTA will have a large discovery potential in key areas of astronomy, astrophysics and fundamental physics research. These include the study of the origin of cosmic rays and their impact on the constituents of the Universe, the investigation of the nature and variety of black hole particle accelerators, and the inquiry into the ultimate nature of matter and physics beyond the Standard Model, searching for dark matter and the effect of quantum gravity.

The design foresees a factor of 5-10 improvement in sensitivity in the current very high energy gamma-ray domain of about 100 GeV to some 10 TeV, and an extension of the accessible energy range from well below 100 GeV to above 100 TeV.

The CTA Observatory will consist of two sites, one in the southern hemisphere and one in the northern hemisphere, with gamma-ray telescopes of different sizes and designs.

The southern hemisphere array of CTA will consist of four types of telescopes with different mirror dish sizes in order to cover the full energy range. The northern hemisphere array would consist of two telescope types.

The low-energy instrumentation will consist of a few 24-metre-class telescopes (**Large Size Telescopes - LST**) with a moderate Field of View (FoV) of the order of 4-5°.

The medium energy range, from around 100 GeV to 1 TeV, will be covered by one type of telescope of the 10-12 metre class (**Mid Size Telescopes - MST**) with a FoV of 6-8° and another type of 9-10 metre class ("Schwarzschild-Couder") with similar FoV but better angular resolution.

The high energy instruments, operating above 10 TeV, will consist of a large number of small (4-6 metre diameter) telescopes (**Small Size Telescopes - SST**) with a FoV of around 10°.

ISBN 978-3-95450-139-7

THE MST PROTOTYPE

A prototype of the Mid Sized Telescope has been deployed in Berlin Adlershof (Germany) in May 2013.

Figure 1: The 12m Mid Sized Telescope prototype installed in Berlin Adlershof.

It has a Davies-Cotton [5] type reflector with a diameter of 12 m, and a focal length of 16 m (see Figure 2). The main goal of this prototype is to test a design of the mechanical structure and drive system, but other prototype instruments like CCD cameras pointing and monitoring and the Active Mirror Control (AMC) will also be tested. An important part of an imaging atmospheric Cherenkov telescope is the camera, consisting of about 2000 Photo Multiplier Tubes (PMTs) and an appropriate readout system. Since the Berlin region is not suitable for gamma ray detection due to the high background light, a mechanical dummy camera of 2.5 tons has been deployed, allowing functions such as the operation of the protective lids and temperature sensors to be tested. The prototype contains a drive system, five Charged-Coupled Device (CCD) cameras and a Weather Station (WS), plus various sensors designed to test the behaviour of the structure.

MST INSTRUMENTATION

Drive System

The drive system of the MST prototype is designed to resemble the expected operation modes of the CTA telescopes, allowing pointing of the prototype to any position and to track any astronomical object. The telescope will operate with two drives, one for azimuth and one for elevation.

The drive system of the prototype is composed, at a lower level, of the control of 6 motors (2 for azimuth and 4 for elevation) communicating via a Bosch-Rexroth programmable logic controller (PLC).

CCD Cameras

Five CCD cameras of the model Prosilica GC1350 are installed at the telescope; two in the centre of the dish, one in the dish axis and two in the dummy camera. The purpose for the CCD cameras is to measure the movement of the camera dummy, the mirror alignment, the pointing and calibration (online correction applied in order to obtain the real pointing direction of a telescope), the measurement and optimization of the optical Point Spread Function (PSF) and the monitoring of parts of the dish and/or the position of the pointing camera. These CCD cameras are interfaced via GigE Vision interface (allowing up to 1000 Mbit/s on Gbit Ethernet). The Allied Vision Technologies (AVT) PvAPI SDK allows controlling and capturing images from GigE Vision CCD cameras in a Linux environment. The CCDs will operate at a rate up to 10 Hz and, and excluding the emulated data sources, they will generate the largest fraction of data volume of the prototype.

Active Mirror Control

The design of the CTA telescopes makes use of a tessellated reflector composed of individual mirror facets (see [10] for details). Each individual mirror facet is attached to a triangular support, with two powered actuators and a fixed support (an AMC unit), which has the functionality to enable automatic mirror alignment. This allows online re-alignments of the LSTs, and if required MSTs as well, where the deformation caused by the weight of the telescope will cause misalignments depending on the telescope elevation. It also allows, for larger timescales, realignments in MSTs and SSTs to be performed. The dish of the MST prototype is already completely covered by a combination of real and dummy mirrors. Several AMC units of two different types are installed on the MST prototype (see [11] for details in the design concepts). The first type of unit communicates via XBee radio modules, creating a Wireless Personal Area Network (WPAN) that is accessed via a XBee receiver connected to an embedded computer via USB or RS-232 serial interface. The other type is interfaced via CAN-Bus, accessed via an Ethernet-CAN-Bus gateway.

ISBN 978-3-95450-139-7

Weather Station

A Weather Station has been installed nearby the MST prototype to continuously monitor the weather parameters, allowing correlation with the behaviour of the structure of the MST with changes in the environmental status like, for example, the wind speed or temperature.

This Weather Station is able to measure the wind speed and direction, as well as other quantities with the required accuracy and measurement rate.

MST CONTROL SOFTWARE

Introduction

Whereas the MST prototype is designed as a dedicated instrument for testing various telescope hardware properties and features in advance, it is in addition used as test bed for significant parts of the general CTA Array Control and Data Acquisition (DAQ) software environment (also called ACTL) [6], in particular the framework ALMA Control Software (**ACS**) [7] and the low level software interface protocol OPen Connectivity-Unified Architecture (**OPC UA**) [8]. The initial design concepts and first test implementations for the prototype were already presented in the previous ICALEPCS conference [9].

The ACS framework was chosen as a result of comparing the complexity and functionality of the array control and data acquisition software requirements between the ALMA project and CTA. ACS comprises a container-component software approach using CORBA distributed object communication services. Services provided by ACS, like event handling, notification, alarming, and GUI interfacing, are being tested as integral part of the MST prototype control software, which from the beginning was designed to take advantage of the ACS framework.

In order to standardize the interface to as many hardware devices as possible, OPC UA is under evaluation inside the CTA consortium. At the MST prototype the drive system is already providing a native OPC UA server implemented inside the drive controlling PLC. For other devices like the CCD cameras and the weather station OPC UA servers have been developed using the Prosys OPC UA Java Software Development Kit (SDK). With the support from of the ACS development team from ESO the ACS DevIO, a low level device communication abstraction layer, was extended by a Java DevIO class implementing basic OPC UA client functionality. This opens the opportunity to integrate hardware with integrated OPC UA server software being connected to ACS by using an ACS standard method.

The control software for the prototype has been installed on an on-site server system based on a Dell PowerEdge R720xd with 2 CPUs (12 CPU cores), 128GB main memory and more than 6TB local disk space. The operating system used is Scientific Linux, version 6, which is compatible to RedHat Enterprise 6. An automatic installation procedure for the OS and the software framework installation based on the RedHat Package manager (RPM) has been applied.

The On-Site system at the prototype site in Berlin Adlershof can be accessed via a WAN connection from a 15km distant remote development environment (2 Dell R510 servers) at DESY in Zeuthen. The DESY data-centre is connected to the MST prototype site by a broadband connection with a bandwidth of about 40 MByte/s. Apart from using the WAN connection for remote login and system monitoring, ACS features a remote deployment opportunity of software which was created on the local development system in Zeuthen.

Device Control

As the **drive system** already contains a native OPC UA server inside the firmware it can be directly accessed via the OPC UA DevIO mechanism of ACS. A set of basic functions has been implemented for initial drive tests. Figure 2 gives an impression of interfacing the drive system via an ACS GUI, it shows drive system functions on the right hand site and some plots of drive movements induced by ACS control commands.

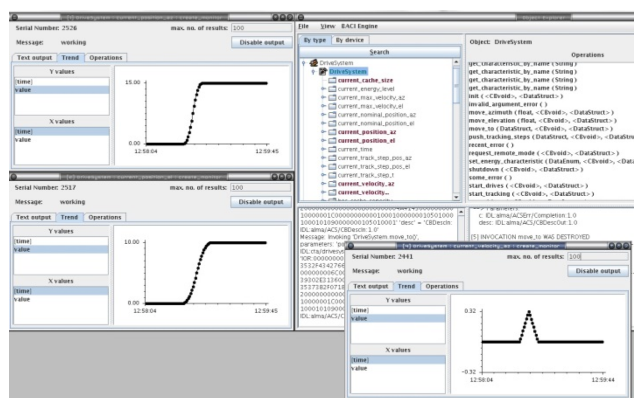

Figure 2: Generic ACS GUI interface to the drive system.

Figure 3 shows a schematic view of the **CCD camera** software design. OPC UA servers will be used as the low level access to the CCD cameras. On a higher level, the ACS part of the controlling software is implemented by using an ACS Java container with two components:

- CCD controller – interface to properties and methods on the OPC UA Server by using the Java DevIO mechanism. The ACS configuration database is used to define the default values and parameter limits of each CCD unit.
- CCD image server - obtains the CCD images from the OPC UA server by using the subscription mechanism of the Java DevIO. By separating the image server from the controller, it is possible to test different data transport mechanisms with the CCD frames in the distributed system.

ISBN 978-3-95450-139-7

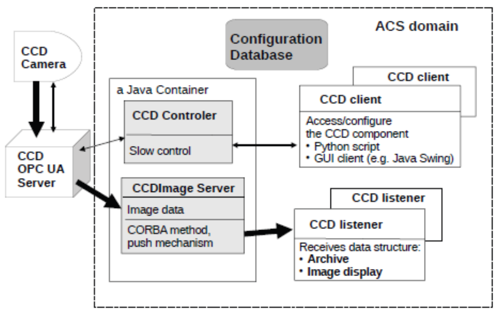

Figure 3: Implementation of the MST CCD camera control software.

In an initial test, the ACS-based software has been used to control and read out a CCD camera that was observing the sky in a clear night while the drive system was tracking a sky position at sidereal speed. The processing of the obtained images allowed the extraction of the pointing direction of the CCD camera. Figure 4 shows a picture taken with a CCD camera pointing at the sky while the MST prototype was tracking. The exposure time was 5 s; the image has been processed with the Astrometry.net software [12]. The circles denote stars found in a comparison with star catalogue. The green lines connect stars that have been used to obtain the pointing direction of the CCD camera.

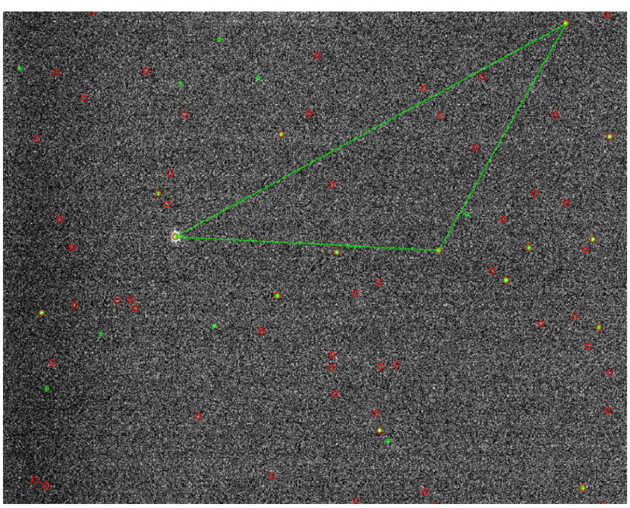

Figure 4: CCD camera pointing at the sky during tracking (courtesy Louise Oakes).

The control software for the different units for **Active Mirror Control** (AMC) has been implemented by connecting the ACS DevIO layer directly. For the units using the XBee communication, a preliminary version of a C++ ACS component based on the libxbee library has been implemented. The AMC hardware has been successfully tested using C++ under ACS. Figure 5 shows the connection of the AMC communication layers to the ACS framework. One type of unit communicates via XBee radio modules, creating a Wireless Personal Area Network (WPAN) that is accessed via a XBee receiver connected to a PC via USB or RS-232 serial interface. The other type of unit is interfaced via CAN-Bus, accessed via an Ethernet-CAN-Bus gateway. In order to simplify the control of the AMC system and the mirror alignment procedures, the higher level interfacing to both unit types are unified. This can be achieved with a higher lever ACS component which sends the common instructions for the alignment procedures, acting as client for the lower level ACS components that will be different for each of the two AMC concepts. The mirror alignment procedures use as input images obtained with the CCD cameras described above, and therefore the higher-level component will be synchronized with the operation of the CCDs.

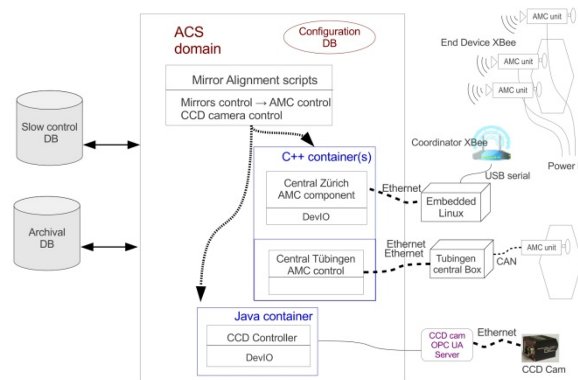

Figure 5: Schema of AMC-ACS interfacing.

For the prototype **Weather Station** an OPC UA server has also been implemented with the OPC UA Java SDK, which uses the RXTX library to communicate with the datalogger of the weather station via the serial line. At a higher level, the data are accessed via a Java ACS component. With the actual implementation, the data are stored on disk by using an ACS Python client.

Storage System

Configuration Data: An ACS system needs information that includes configuration and deployment data for Components, Containers, and the ACS Manager. The default implementation of the Configuration Database (CDB) consists of a set of XML files parsed against some corresponding XSD files. ACS also provides an alternative implementation of the CDB that uses a relational database such as HSQLDB or Oracle DB to store this information. The database-oriented implementation was modified to run at the MST prototype with MySQL.

Monitoring Data: Each Component has a number of Component Properties. Properties are used to monitor (and eventually control) the corresponding entity such as

ISBN 978-3-95450-139-7

a physical device or a business component that implements certain business logic. A generic monitoring system has been developed. It consists of a front-end that automatically detects any Component and continuously acquires its Property data plus a back-end that stores the data. Back-ends using MongoDB and the ACS logging system are already implemented, back-ends using MySQL and HDF5 files are alternatively planned as well. The back-end for MySQL utilizes an object-relational mapper that includes additional dialects for SQLite, Postgresql, Oracle and others.

Log Messages and Alarm Information: All log messages and alarms are transmitted to central services that provide interfaces for consumers of these data. Two consumers—one for logs and another for alarms—were realized for the MST control system. They store all data in MongoDB. The risk of filling up the storage system with excessive log information is excluded by using a capped collection or alternatively a "time of life" collection.

CCD Images: CCD images represent the biggest part of data at the MST prototype. They are stored with GridFS, which is an abstraction layer on top of MongoDB to store and retrieve "files" plus meta data. Having the images accessible alongside with their meta data is very handy while querying a specific image against its meta information. The CCD software contains a specialized CCD listener that receives image data from the CCD image server and writes it to GridFS.

ACKNOWLEDGMENT

We thank the whole MST prototype group at Humboldt University Berlin and DESY Zeuthen, engineers and scientists, for their excellent work. We also like to thank the ACS experts from ESO Garching, in particular G. Chiozzi and H. Sommer, for their support.

We gratefully acknowledge support from the agencies and organisations listed at the following URL: http://www.cta-observatory.org/?q=node/22

REFERENCES

[1] CTA Consortium, “Design Concepts for the Cherenkov Telescope Array,” Experimental Astronomy, Vol. 32, Number 3 (2011), pp. 193-316.

[2] HESS website: http://www.mpi-hd.mpg.de/hfm/HESS/

[3] MAGIC website: http://magic.mppmu.mpg.de/

[4] VERITAS website: http://veritas.sao.arizona.edu/

[5] J.M. Davies and E.S. Cotton, “Design of the Quartermaster Solar Furnace,” Solar Energy Vol. 1, Issues 2-3 (1957), pp. 16-22.

[6] P Wegner et al., “Simultaneous operation and control of about 100 telescopes for the Cherenkov Telescope Array,” J. Phys.: Conf. Ser. Vol. 396 (2012), 012052; doi:10.1088/1742-6596/396/1/012052.

[7] G. Chiozzi et al., “The ALMA Common Software: A Developer Friendly CORBA Based Framework”, SPIE Proc. Vol. 5496 (2004), pp. 205-218.

[8] OPC UA foundation: http://www.opcfoundation.org/

[9] I. Oya et al., “A Readout and Control System for a CTA Prototype Telescope,” ICALEPCS 2011; www.JACoW.org

[10] A. Föster et al., “Mirror Development for CTA,” Proc. 32nd International Cosmic Ray Conference, Beijing, China, 2011.

[11] B. Beher et al. “Development of the ACS+OPC UA Based Control System for a CTA Medium Size Telescope Prototype,” in Software and Cyberinfrastructure for Astronomy II, SPIE Proc. Vol. 8451 (Sept. 24, 2012); doi:10.1117/12.923773.

[12] Astrometry web page: http://astrometry.net/

ISBN 978-3-95450-139-7