22nd International Conference on Computing in High Energy and Nuclear Physics

CHEP2016

October 10-14, San Francisco, USA

Visualization of Historical Data for the ATLAS Detector Controls - DDV

Julian Piotr Maciejewski, Stefan Schlenker (CERN, Geneva, Switzerland)

ABSTRACT

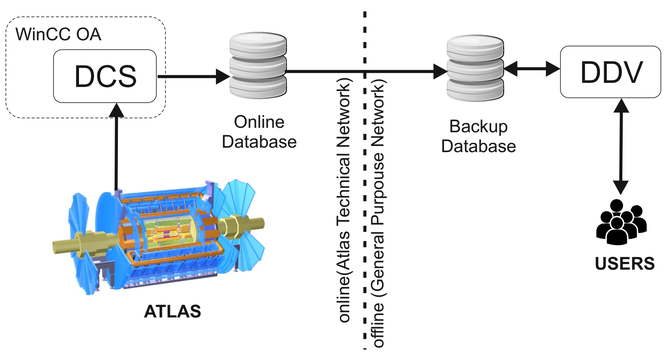
The ATLAS experiment is one of four detectors located on the Large Hardon Collider (LHC) based at CERN. Its detector control system (DCS) stores the slow control data acquired within the back-end of distributed WinCC OA applications. The data can be retrieved for future analysis, debugging and detector development in an Oracle relational database.

The ATLAS DCS Data Viewer (DDV) is a client-server application providing access to the historical data outside the experiment network. The server builds optimized SQL queries, retrieves the data from the database and serves it to the clients via HTTP connections. The server also implements protection methods to prevent malicious use of the database.

The client is an AJAX-type web application based on the Vaadin (framework build around Google Web Toolkit (GWT)) which gives users the possibility to access the data with ease. The DCS metadata can be selected using a column-tree navigation or a search engine supporting regular expressions. The data is visualised by a selection of output modules such as a java script value-over time plot or a lazy loading table widget. Additional plugins give the users the possibility to retrieve data in ROOT format or as ASCII file. Control system alarms can be visualized in a dedicated table. Python mock-up scripts can be generated by the client, allowing the user to query the pythonic DDV server directly, such that the users can embed the scripts into more complex analysis programs. Users can store searches and output configurations as XML on the server to share with others by URL or embed in HTML.

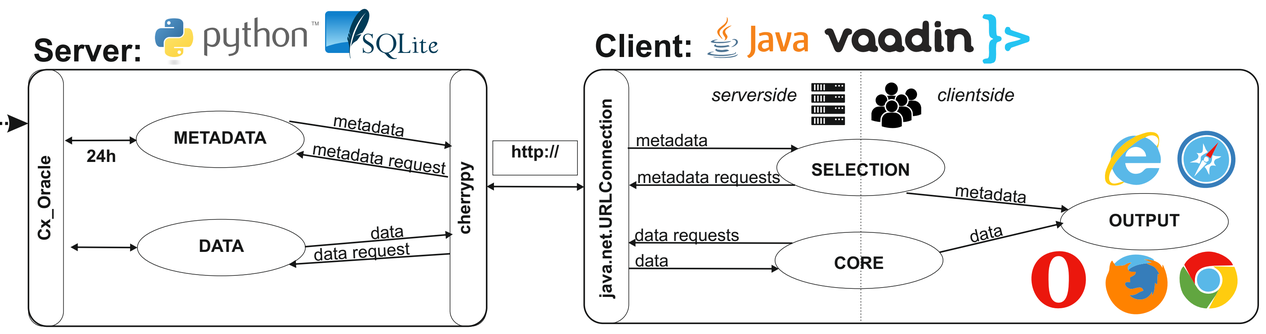
In a recent major release of DDV, the code was migrated to use the Vaadin Java Web framework for the user interface implementation, greatly improving aspects such as browsers and platform independence. The update has helped with reducing development and maintenance timescales. In addition, the tool now supports and visualizes metadata evolution which allows users to access the data consistently over decades. It is able to trace changes of hardware mappings or changes resulting from back-end software migration/restructuring. Furthermore, users can use DDV to review database insert rates e.g. to spot elements causing excessive database storage consumption. The client now also provides each user a usage history which is stored on the server allowing quick access to previously used configurations. Finally, the application has been generalised to be compatible with any other WinCC OA based RDB archive which allowed it to be set up for other control systems of the CERN accelerator infrastructure without any additional development.

DCS DATA FLOW



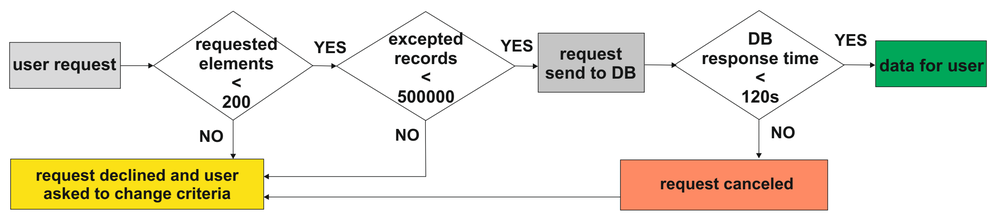
ATLAS DCS systems archive the detector parameters (data point elements) to the online database, while at the same time they can present a historical view of these parameters serving the online monitoring needs. The database structure is defined by the proprietary schema of the WinCC OA RDB archiver application. For security and performance reasons, the online database is accessible only to systems within the ATLAS Control Network (ATCN). Nevertheless, a backup database - a complete replica of the online database - is in place to serve users within the CERN General Public Network (GPN). DDV is connected to the backup database and aims to cover the needs for the offline access of operational data worldwide. This backup database is using the Oracle Active Data Guard mechanism. It prevents data loss and downtime simply and economically by maintaining a synchronised physical replica (backup) of a production database (online) without impacting database performance. DDV accesses data by forming proper SQL queries and using cx_Oracle to connect to backup database.

ARCHITECTURE



This illustrates the different building blocks that DDV it is composed of. A SQlite database is used at the server side and caches all the DCS metadata updated daily from the Oracle database. Furthermore, SQLite is configured in memory mode that means that instead of hard disk files, information can be stored in memory file objects, optimizing the response time of the requests. At the other end, the users' web browser represents the DDV client which fetches the data from the DDV server via HTTP requests. The client has been developed using Vaadin - a Java web application framework for creating rich and interactive web applications running on a Tomcat web server and supporting all common web browsers. The aim of the selection component is easy navigation among the metadata (i.e. column tree view widget or search panel) as well as constructing metadata requests. The core component is responsible for building proper data requests to the server acquiring data and sending it to the chosen output. Those requests combine selected metadata and time periods. Due to its flexible interface and its generic and modular approach, DDV could be easily used in other experiments which use the WinCCOA RDB Archiver.

DB PROTECTION MECHANISM



Intending to serve a high number of users, DDV aims to be more than an interface to the database data. DDV validates and certifies each request with a minimum-response-time cost and finally propagates it to the DB. This protection mechanism is organised in three levels(diamonds on the figure above). The second protection mechanism which performs a light test-request with a time period significantly shorter than the one requested. The number of returned rows is translated to a data rate (Hz) and by extrapolating to the full query time, a decision is made whether this request is acceptable or not. The decision parameters were chosen empirically based on users queries.

FEATURES

SERVER

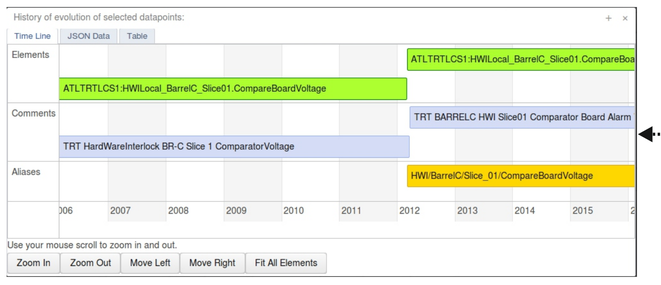
- Allows direct http queries to the server within CERN network giving the opportunity for users to write their own scripts independent of the web browser client part
- Server implements database protection mechanism
- RbdAPI integrated (API that allows to access the data stored by WinCC OA Rdb Archiver in the Oralce database from an external application using stored procedures instead of creating sql queries)

CLIENT

1. Navigate among data point elements (DPEs) by name, comment and alias
2. Easy navigation mode trough column-tree widgets
3. Search among data point elements (simple and regular expressions)
4. Navigate among historical deleted data point elements
5. Query for alarm history and DCS logs
6. Quickly reload your last queries
7. Multiple output formats:
   - Plot
   - Table (lazy loading)
   - ROOT (data as b-trees)
   - TXT
8. Input rates of selected data point elements or the whole project to spot possible DPEs with high insert rate
9. Explore metadata history
10. Save the output and its configuration such as plot colours and share as URL link or embed in HTML page
11. Selection based on data validity

MAIN USER INTERFACE

METADATA EVOLUTION



Every data point element apart from its unique element name can have an alias and a description. These metadata values may change over time. DDV is able to track and visualize those changes in a time line or table.