EPICS MySQL ARCHIVER INTEGRATION BETWEEN EPICS AND MySQL

A. Roy*, R.B. Bhole, S. Pal, D. Sarkar, VECC, Kolkata, India

Abstract

The performance evaluation and analysis of inter-system dependency of the various subsystems of the Superconducting Cyclotron (SCC) demand a well configured data logging, archiving and historic analysis facility for large number of control parameters along with on-line failure analysis facility of every system. Experimental Physics & Industrial Control System (EPICS) is used as development architecture of the control system of these systems with MySQL as database for large amount of relational data management. This combination requires integration between EPICS and MySQL server. For this purpose, MySQLArchiver as an EPICS Extension is developed for data logging and archiving of control parameters into MySQL database. This extension also provides a web based tool for online monitoring of control parameters and historic analysis of archived data. This paper describes the software architecture, implementation, as well as method of configuration for any other EPICS based control system as a utility. This facility is also elaborated with examples, web page views and experiences of deploying it in SCC.

INTRODUCTION

The performance of a particle accelerator is measured by its beam stability. The beam stability is achieved through fine tuning of the system parameters. The requirement of archiving the system parameters originates here. The Experimental Physics & Industrial Control System (EPICS), a standard open-source dual layer software tool for designing distributed control system, is widely used in large accelerators around the world for implementing supervisory control software. It has its own Channel Access (CA) archiver for storing system parameters into a proprietary database using channel access protocol. Since modern accelerators e.g. Superconducting cyclotron are comprised of various critical systems e.g. cryogenic system which are entirely vendor dependent. These systems have their own proprietary or third party supervisory system mostly without support for EPICS channel access. While fine tuning such accelerators with EPICS at supervisory control layer, EPICS channel archiver fails to store the third party system parameters. Again configuration and maintenance of the archiver demand expertise in EPICS tool set.

In contrast, interface to an open relational database e.g. MySQL, PostgreSQL etc. is more readily supported by most of the industry standard control software. The availability of expertise and support for these databases is plenty in industry. Hence an archiver based on such open relational database engine with channel access support will mitigate the requirement of storing control parameters from heterogeneous systems (EPICS and non-EPICS/third party) in a common platform. An EPICS archiver is developed using MySQL as database engine for archiving and retrieving the control parameters of Superconducting Cyclotron (SCC), Kolkata. MySQL is chosen due to its simplicity for installation and maintenance.

*r_ani@vecc.gov.in

SOFTWARE OVERVIEW

An archiver is a client tool, i.e. CA Client, from the perception of EPICS dual layer architecture. The main purpose of the archiver is to collect the control parameters using CA protocol from various Input Output Controllers (IOC) distributed over network and then it stores the values into a database. The user interface for retrieving the archived data for analysis purpose is also an integrated part of the archiver. The software architecture and other components of EPICS MySQL Archiver are described below.

Architecture

The software architecture of EPICS MySQL Archiver is shown in Fig.1. A multilayered architecture with a common memory resident database is adopted for this application.

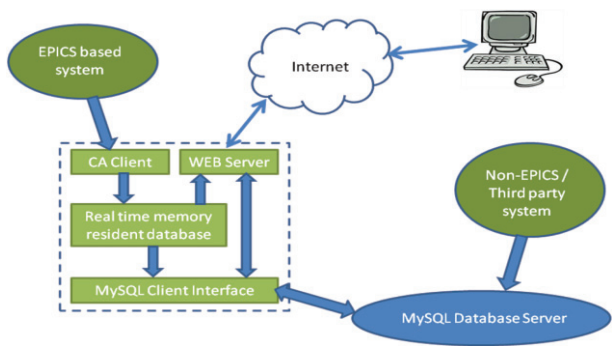

Figure1: Software architecture of EPICS MySQL Archiver.

The top layer of the software is consists of a CA Client interface and a Web server interface. The CA Client interface is dedicated for acquiring the values of the EPICS based parameters through CA link. The intermediate layer is a memory resident real time database (RT DB) keeping the latest values. The lowest layer is MySQL client layer dedicated for communicating with MySQL database server using SQL commands. This layer is responsible for archiving data from RT DB layer into database and also for retrieving archived data on demand. A web based user interface is provided through Web server interface. The user interface is equipped with tools for viewing system and subsystem wise online status of

ISBN 978-3-95450-124-3

all archived parameters, searching historic data of multiple system / subsystem parameters for analysis and downloading of the historic data with date & time stamp as spread sheet for offline analysis. The third party systems are interfaced directly with MySQL database server through respective standard client tools supported by them.

The Archiver

The archiver is designed to log data along with date and time stamp, available in the database server, either in scan mode or in monitor mode. In scan mode, the process parameter data is read at a predefined interval using CAGet facility. The scan interval can be defined from 1 second to 999 seconds by the user. While in monitor mode, the archiver creates monitoring event to the respective IOC and receives data from IOC, whenever it changes. The archiver is implemented as a multithreaded application. There are dedicated threads for managing MySQL client interface for data insertion and retrieval, CA connection managements, CA callback, command prompt interface for debugging, a timer thread and web server thread. A dedicated thread is implemented to retrieve latest data, for display, from the database for non-EPICS systems logging data independently. There is user authenticated facility for modification, addition or deletion of logged parameters.

The open source libraries e.g. EPICS CA library, MySQL client library and 'gd' library are used for building the archiver. The MySQL client library is used for connection management with database, generating SQL queries and error handling. The gd library is required for generating historic plots as 'png' image. This format of image is chosen for the following reasons. Since the image of historic plot is to be uploaded along with web pages, minimising image footprint will improve the performance and 'png' results into minimum image size. Again historic plots are line plots with sharp contrast; hence png format is the best without any loss of information.

A configuration file, a text file containing the list of parameters to be archived along with system / subsystem names, archiving mode (scan / monitor), parameter type (EPICS / non-EPICS) and archiving interval (in case of scan mode), is required by the archiver for configuring the database. The archiver creates and maintains independent tables, with date column as the index column, at database for individual systems. The indexed tables enhance the searching efficiency. An example configuration file, used for Superconducting Cyclotron archiver, is shown in Fig.2. An example command, used from application 'Top' directory, to start the archiver is shown below.

bin/linux-XX/mysqlArch [host IP] [database name] [config file-1] [config file-2]...

where,

host IP = IP address of the machine running the archiver

database name = name of the database to be used for archiving the data, assumed to be residing in the same machine

config file - 1 = configuration files, multiple configuration files are supported

```
system Vacuum ca
sub Main_machine
pv VAC_MM_BC2 ca monitor 1
pv VAC_MM_BC3 ca monitor 1
pv VAC_MM_OVC ca monitor 1
pv VAC_LOLINER ca scan 1
pv VAC_UPLINER ca scan 1
sub Vault_beam_line
pv VAC_BTS_GL1 ca scan 1
system CRYO_DEL_SYS nc
sub PRESSURE
pv PI1 nc monitor 20
pv PI3 nc monitor 20
pv PI4 nc monitor 20
sub TEMPERATURE
pv TI1 nc monitor 20
pv TI20 nc monitor 20
pv TI3 nc monitor 20
pv TI4 nc monitor 20
```

Figure 2: Archiver configuration file.

Web Interface

A multilayer web based user interface is embedded in the archiver for monitoring, analysis and retrieval of archived data. This facilitates users to interact with the archiver from anywhere in the control LAN without installing any application. The various facilities included in the user interfaces are system view, sub-system view, online monitoring of parameters and connection status, historic analysis of upto five parameters of different systems, downloading of archived data and authenticated access to add/delete/modify parameters. The web interfaces of the above facilities are shown in Fig.3 to Fig.5.

MySQL Server

MySQL Server, Version 5.0.22, is deployed for archiving control system data in SCC. Control system history is stored in transaction safe *InnoDB* format. Database tables are placed in host local storage. An *Auto backup* mechanism is employed for protecting the archived data from hardware failure. In this mechanism, a snapshot of the whole database is stored in local storage and in tape drive using database dump technique on each day. System backups are taken on weekly and monthly basis also.

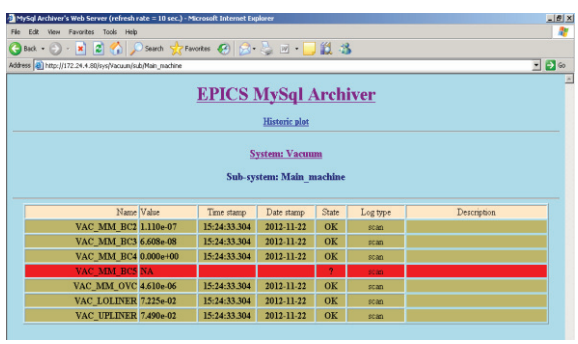

Figure 3: Sub-system page.

ISBN 978-3-95450-124-3

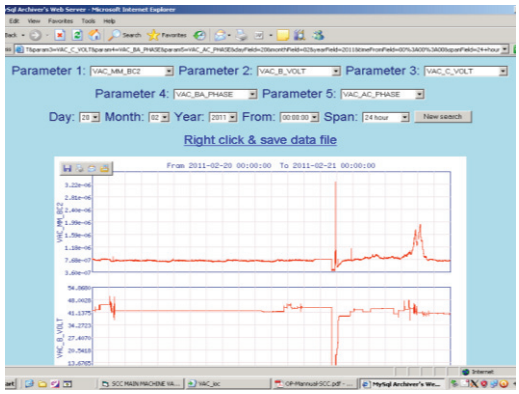

Figure 4: Historic analysis page.

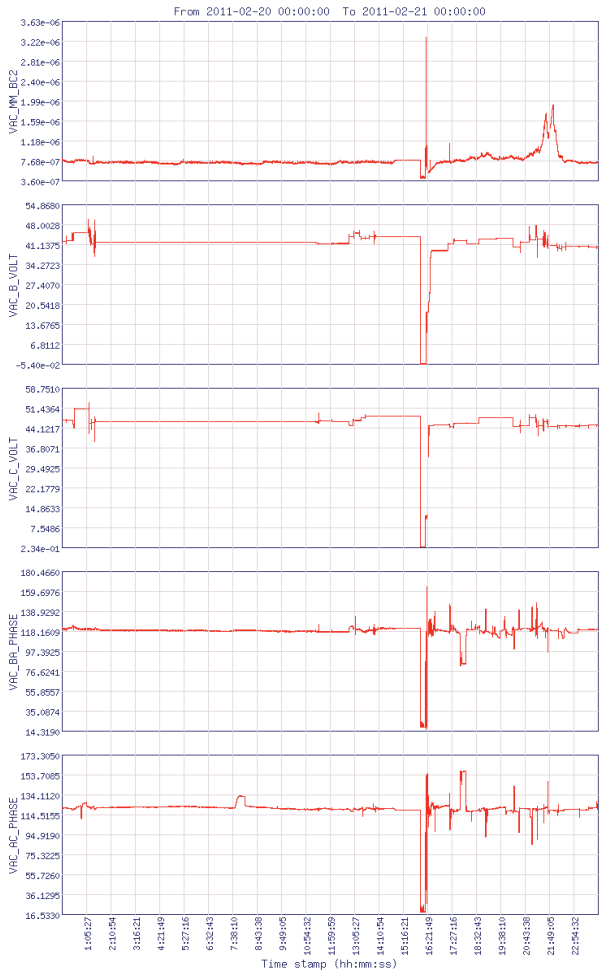

Figure 5: An example historic analysis view.

SYSTEM OVERVIEW

The example EPICS MySQL Archiver is deployed on a Dell PowerEdge 2950 server having dual Intel Xeon E5320 (2 x 4MB cache), Intel 5000X chipset, 4 x Hot swappable SAS drive of approximately 300GB with RAID 5 configuration. The system is running on RedHAT Linux with standard Linux Ext3 file system.

CONCLUSION

An EPICS MySQL Archiver is installed for archiving SCC control parameters. At present around 100 control parameters of various systems e.g. Vacuum, RF, Main magnet, Cryogen Delivery System (CDS), LHe plant, are being logged into the database with a frequency of 50 parameters per second. Among the various systems, the CDS is a non-EPICS system and logs parameter data directly to the database using a third party MySQL Client interface. As all parameters are being logged with a common time stamp available at database server, hence it helps to provide a snapshot with respect to time of all parameters irrespective of EPICS/non-EPICS system. This system is running satisfactorily for a considerable period of time resulting into a data volume around 200GB. It also helps to resolve various inter-system dependent issues in the cyclotron.

Although the example system is deployed on Linux operating system, the same application can be ported on Windows operating system.

ISBN 978-3-95450-124-3