ISBN: 978-3-95450-132-8 doi:10.18429/JACoW-IPAC2014-THPRO110

HIGH AVAILABILITY SOFTWARE ARCHITECTURE OF C-ADS CONTROL SYSTEM

Pengfei Wang[#], Jianshe Cao, Qiang Ye, IHEP, 100049, Beijing, China

Abstract

The control system of Accelerator Driven Sub-critical System (ADS) should be a high-availability (HA) system with fault tolerant architecture, due to the potential utilizations of the ADS, such as separating and transmuting irradiated nuclear fuel. This paper discusses the HA software architecture of ADS control system which mainly composed by four softwares, which are 1) low floor communication and control system---EPICS [1], 2) hierarchal programming framework of the accelerator —XAL [2], 3) monitoring and operating large scale control systems—Control System Studio (CSS) [1], 4) data storage and service infrastructure—HA database and server cluster. In addition, the recent development of ADS control system is briefly introduced in this paper.

INTRODUCTION

The ADS has been universally regarded as the most effective approach to dispose the long-lived nuclear wasted [3]. In 2011, the Chinese Academy of Sciences launched the “Strategic Priority Research Program” named “Future Advanced Nuclear Fission Energy”. This program has two sub-programs, and the ADS Project is one of them.

SOFTWARE ARCHITECTURE

Overall relationship of the softwares which included in the HA software architecture is shown in Fig. 1. Communication with the diagnostics and beamline devices is through the EPICS control system, or more specifically via the Java Channel Access (JCA) interface. CSS projects and XAL applications run on the Linux server cluster. They can read data from the HA database or write data to it.

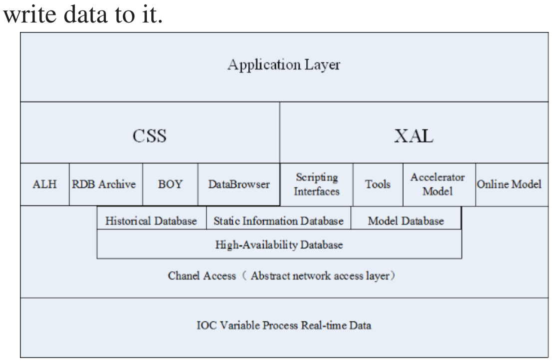

Figure 1: HA software architecture of ADS Control System.

EPICS

EPICS is a software environment originally written jointly by Los Alamos National Laboratory and Argonne National Laboratory. It is used to develop and implement distributed soft real-time control systems to operate devices such as particle accelerators, telescopes and other large experiments. Now, over 100 large scientific facilities throughout the world use the EPICS, such as NSRL, BEPCII and SSRF. ADS also uses EPICS as the underlying control system to communicate with accelerator hardware.

XAL

XAL is designed to be a flexible application framework for developing accelerator physics applications for beam commissioning [2]. XAL has advanced design concept and many features that users expect from modern applications. It has been adopted by many international accelerator laboratories and adopting it for ADS is a key subject in the long term. Use of this framework, the control system operators can write applications that can be applied to the accelerator. Virtual Accelerator is an important application in XAL. With the Virtual Accelerator, it is possible for an operator to judge whether setting parameters would be justified or not, examine the control system of the machine and practice the commissioning without a beam. The screenshot of ADS Virtual Accelerator is shown in the Fig. 2. ADS has some specific information, such as the bunchers in MEBT1 and RF cavities in CM1 use the new attribute ETL instead of the separate properties—the longitudinal electric field(E), the transit time factor (T) and the gap length (L). So proper modification is needed in order to better suit ADS online modeling purpose.

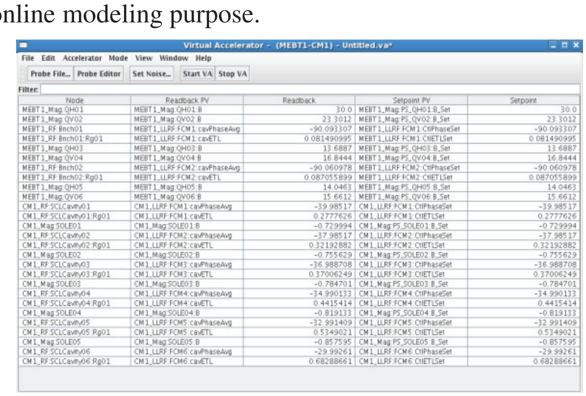

Figure 2: ADS virtual accelerator.

#wangpf@ihep.ac.cn

ISBN: 978-3-95450-132-8 doi:10.18429/JACoW-IPAC2014-THPRO110

HA Database and Server Cluster

HA Server Cluster

HA server clusters provide continuous availability of services by eliminating single points of failure and by failing over services from one cluster node to another in case a node becomes inoperative [4]. ADS uses RedHat Cluster Suite (RHCS) shown in Fig. 3 as the HA server cluster. Advantages of RHCS include scalability, significantly lower in cost, flexible, easy to deploy, field-proven and certified.

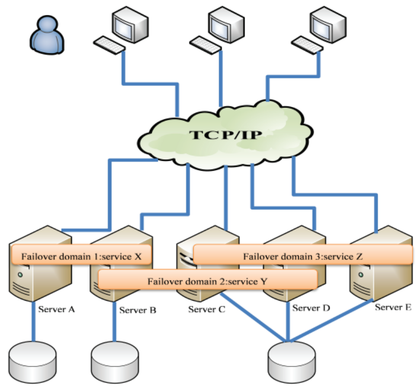

Figure 3. RHCS introduction.

Database

The accelerator database plays an important role in the accelerator control system. It is an integral part of daily operation. Compared with relational database, OO databases have some stability-related "early adopter" problems**.** Among the large relational database management system product, MySQL is extremely easy to use and is the world's most popular open source database. These factors led us choose MySQL as the ADS accelerator database.

HA Database System

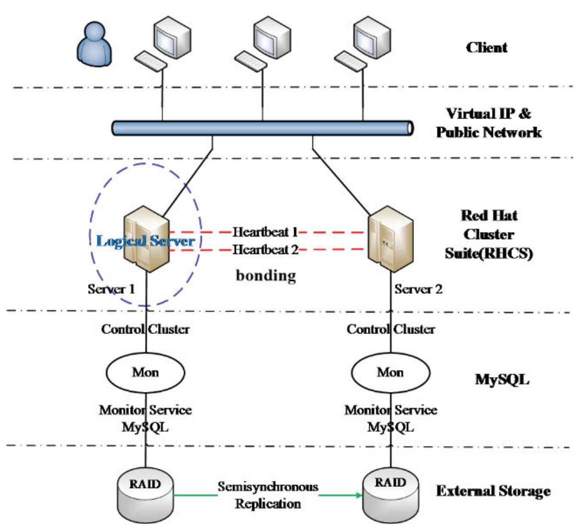

Figure 4: HA database system.

The schematic diagram of HA database system is shown in Fig. 4. ADS uses fencing, bonding, Mon and MySQL replication as the HA database solution. Fencing shown in Fig. 5 is a technology used to remove a cluster member from an active cluster, as determined by loss of communication with the cluster. Split-brain condition [5] is avoided to a certain extent by using the fencing technique. RHCS has more fencing technologies than other HA server clusters, such as Veritas Foundation Suite or Oracle Clusterware. The Linux bonding driver provides a method of combining multiple network interfaces in parallel to increase the reliability of the system. It is the best way to provide redundancy for the heartbeat. 'Mon' is a general-purpose scheduler and alert management tool used for monitoring service availability and triggering alerts upon failure detection [6]. The HA database system uses 'Mon' to monitor the MySQL service. If the MySQL service in master server fails, 'Mon' will instruct the RHCS to fail over the service from the master server to the slave server. The HA database system uses redundant arrays of inexpensive disks (RAID) as the external storage and uses MySQL replication for the purpose of backup. The most important benefit of using replication is that you can perform database backups using a slave server without disturbing the master. The master continues to process updates while the backup is being made.

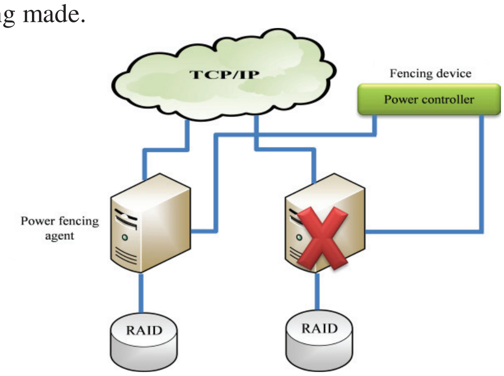

Figure 5: Fencing.

CSS

CSS is based on the Eclipse Rich Client Platform (RCP) and is a collection of useful tools: RDB Archive, "Best Ever Alarm System Toolkit" (BEAST), "Best OPI, Yet" —BOY, as well as some control system diagnostic tools. Compared With the old GUI tools: EDM and MEDM, the Operator Interface (OPI) of CSS applications is elegant, simple, easy to operate and common look-and-feel. Figure 6 below shows the screenshot of the MEBT1 control system. The middle part of the control system interface shows the devices in the MEBT1. If the users click the magnet or bunch icon, the corresponding setting interface will show in the lower part. The users can adjust parameters for the magnet or bunch. The value of SigmaX and SigmaY shown in the upper part will change according to the current configuration.

ISBN: 978-3-95450-132-8 doi:10.18429/JACoW-IPAC2014-THPRO110

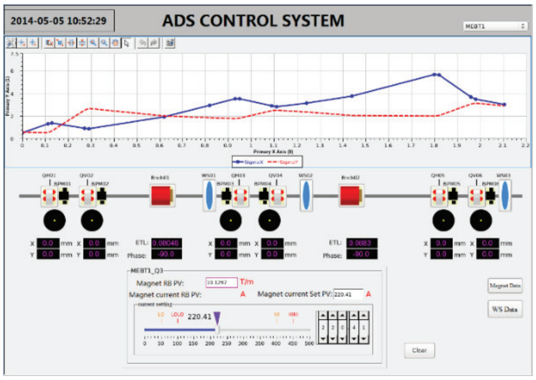

Figure 6: GUI of the ADS Control System.

Archive

ADS uses Relational Database Channel Archiver (RDB Archive) to archive operational data. As you can see from the left part of Fig. 7, the data was lost during the fail-over time for the slave node to take over the MySQL service. Proper modification is needed in order to better suit ADS data requirement. The flowchart of the new RDB Archive is shown in Fig. 8.

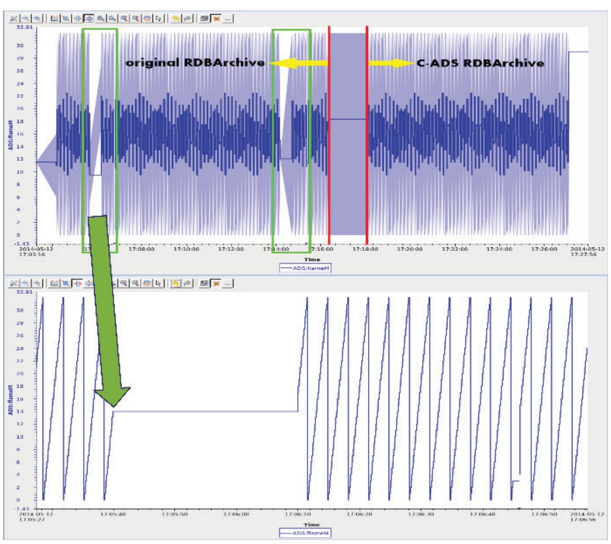

Figure 7: Output of data browser.

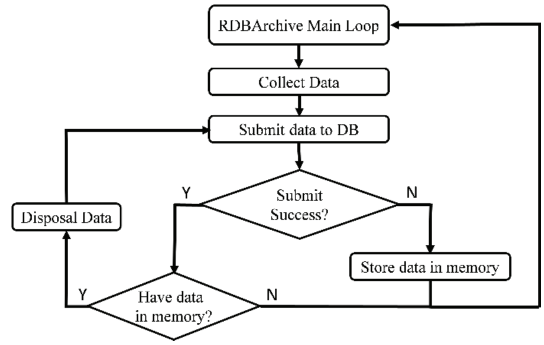

Figure 8: Flowchart of C-ADS RDB archive.

CONCLUSION

The ADS will benefit from the HA software architecture which mainly composed by EPICS, XAL, CSS and HA database and server cluster. We need to intensive study XAL, CSS, MySQL Replication and RHCS, keep a close watch on the new technology, strengthen the international exchanges and cooperation, set up the world-class accelerator control system.

REFERENCES

[1] http://aps.anl.gov/epics
[2] http://xaldev.sourceforge.net/
[3] Zhan Wenlong and Xu Hushan, "Advanced Fission Energy Program—ADS Transmutation System", Bulletin of the Chinese Academy of Sciences, 2012, 27(3).
[4] https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/4/html/Cluster Suite_Overview/ch.gfscs.cluster-overview-CSO.html
[5] http://linux-ha.org/wiki/Split_Brain
[6] http://directory.fsf.org/wiki/Mon