UTILITY FACILITY MONITORING SYSTEM ON LINUX-BASED IOC FOR THE KEKB

N. Yamamoto*, A. Akiyama, T. Katoh, J. Odagiri, KEK, Tsukuba, JAPAN
S. Matsuzaki, HITACHI,Ltd., Hitachi, Ibaraki, Japan
N. Ishibashi, HICOS, Inc., Hitachi-Naka, Ibaraki,JAPAN

Abstract

The latest version of Experimental and Industrial Control System(EPICS), R.3.14, is now executable on multiple platforms including the Linux operating system. A Linux-based PC as an Input/Output Controller (IOC) has an advantage over VME-board computers in the ability of running high level applications along with the IOC core software on the same computer. Such "all-in-one" EPICS system opens up a new way to build a cost-effective control system. As such a kind of system, we have developed a Linux-based IOC which interfaces the KEKB accelerator control system with a local control system of the utility facility of the accelerator. A client process running on the IOC gets the data from a data server computer in the local control system over a network. The data is put into a memory area shared by the process and the IOC core program. A simple device support accesses to the shared memory to get the data into EPICS records. The simple design allowed us to develop a reliable and available system in a few months. The configuration of the software and some experiences on the operation of the system are shown.

INTRODUCTION:UTILITY FACILITY MONITORING FROM AN ACCELERATOR CONTROL SYSTEM

After years of operation in TRISTAN and KEKB accelerators [2], we now are confident on the importance of monitoring environment changes around accelerators. Study[3] shows that air temperature, atmosphere pressure, building temperature, tunnel temperature and cooling water temperature can affects be beam in accelerators. The KEKB utility facility includes cooling water system, air conditioning system for accelerator buildings includes accelerators tunnel and electric power controls. Thus the controls of the utility facility is one of important source of information for accelerator operation.

In summer of 2002, the control system of KEKB utility facility was upgraded from old min-computer based system to the modern control system based on PLC and workstations. A serial serial communication line connecting the old facility control system and an accelerator control system is replaced ethernet connection in the new system. The new facility controls also includes a workstation as a gateway connecting the facility and accelerator control systems[Figure 1]. From the KEKB/PF-AR accelerator control system, which is based on EPICS toolkit, this gateway looks as another IOC, ie. CA server. We used latest EPICS release 3.14 running on Linux PC to setup this gateway. The iocsh program which become available in EPICS R3.14 is the key of the this gateway system. Using an iocsh, we just needed to develop a device support program to get data from the facility control system. Other functionality needed to setup gateway is already built in the iocsh program, including EPICS database and CA server/client.

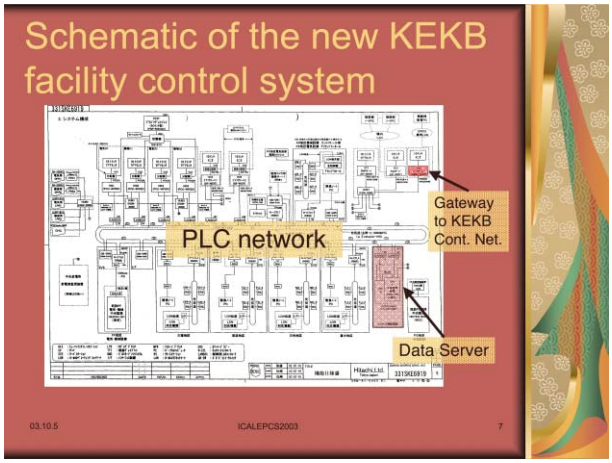

Figure 1: Schematic of the new KEKB facility control system

In the next section, we will review the iocsh in EPICS R3.14. The section describing the gateway and its software follows the next section. Finally, we will reports the performance of the system and discuss the future improvement.

IOCSH

The iocsh is a program provided by EPICS R3.14[4]. In the previous version of EPICS, ioc core programs are supported only on VxWorks[1]. On the VxWorks, tasks in ioc core share the memory space and the global symbol table. To make ioc core running on a Linux and other operating systems, iocsh is introduced to simulate this environment. In iocsh environment , tasks in ioc core are realized as thread of the ioc-sh process. OSI (Operating System Independent) library provides the functionality of global symbol table, on which many of ioc core tasks depends on.

Once you run an iocsh from a console, you can type in ioc core commands just like the Vx shell on VxWorks. From which you download EPICS databases and start ioc

* noboru.yamamoto@kek.jp

core tasks (iocInit). Iocsh provides all functionality of ioc core tasks, including EPICS database, CA server and sequencer. We can even use a device support routine, if it does not depend on the particular operating system using OSI library.

GATEWAY

The utility facility control system built in the summer of 2002 is based on the PLC from Hitachi. The system includes central processing unit based on Linux with proprietary realtime feature. The central processing units works also as a data server for the clients on the private ethernet using the proprietary protocol[Figure 2]. The gateway is provides based on the iocsh program discussed above by the Hitachi under the guidance of KEKB control group, KEK.

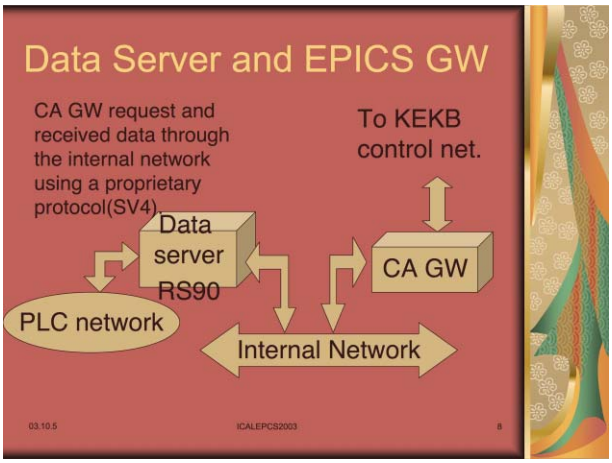

Figure 2: Data server and EPICS gateway

Gateway architecture

Generally speaking , a device support software which uses “slow” communication line needs to support asynchronous processing of EPICS records[5]. However, for this facility monitoring system, there is no needs to modify value or to send any request from the accelerator controls to the facility controls. And also we do not want disturb the operation of PLC data server by requesting data at high frequency. We took this advantage to simplify a device support software for SV4 PLC data server

To reduce transaction between the PLC data server and the gateway, a set of data is transferred at once. A thread in iocsh process stores this data in the shared memory area and send EPICS software event to the ioc core software using post_event () EPICS API. Record/Device support retrieve the data from the shared memory specified in the inp field of a record[Figure 3]. Currently AI and BI record types are supported.

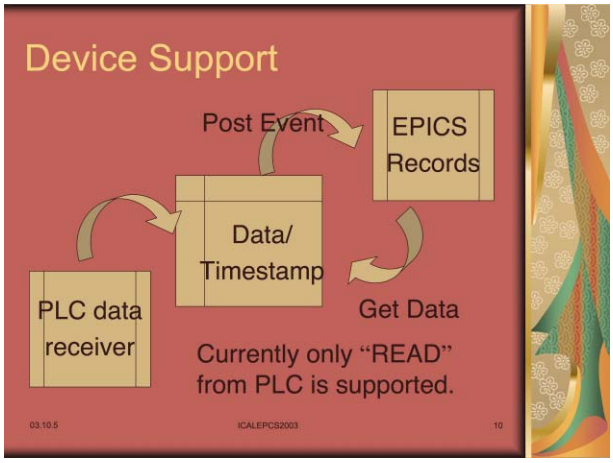

Figure 3: Device support

Gateway hardware/Software

As a gateway, we use PC-Linux box equipped with 1.8 GHz Intel Pentium 4 processor. It also is mounted 256 MBytes of memory. Red Hat Linux 7.3 with Linux 2.4.18-3 kernel from supports iocsh and the communication threads without any problem.

EPICSR3.14.2 is a bases of this gateway program. So far we don’t see any problem, in this configuration.

Performance

Currently the gateway reads 375 data points from the facility control system. To support these channels 32 threads are created in the iocsh process. Each thread occupies 3164 blocks of a main memory. Values are updated every 5 to 50 seconds depending on the update rate on the data server. An iocsh process just uses less than 1 % of CPU time. Since we started to use this gateway, the iocsh process have been running without any serious problem.

DATA LOGGING

Data from the facility control system are stored in the data archiving system, named kblogger. All 375 data from the facility control system are collected and stored into the archiving system. Tools such as kblogrd command or kekb logviewer are used to retrieve data from the archive. Figure 4 shows a sample history of data points in the facility control system.

CONCLUSION

Linux IOC is used to build gateway between EPICS base KEKB control system and the facility control system based on proprietary PLC controls. Device support software handles and hides the proprietary network protocol from EPICS side. Use of ioc-she on Linux reduced development time considerably in both implementation phase and software test phase.

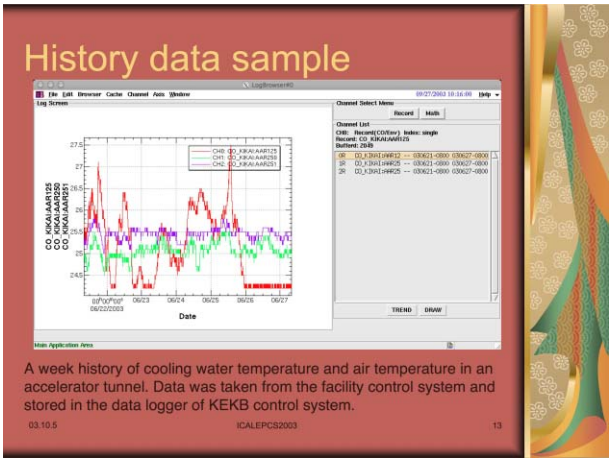

Figure 4: History data sample

System is running stable in the operation of KEKB. Success of this system encourage us to use Linux IOC in an accelerator control system.

REFERENCES

[1] VxWorks is a trademark of Wind River Co.

[2] S. Kurokawa and E. Kikutani, “Overview of the KEKB accelerators”, NIM Sec. A Vol. 499, Issue 1 ,February 2003, Pages 1-7

[3] M. Masuzawa et al., “Circumference Variations Observed at KEKB”, PROCEEDINGS OF THE 7TH INTERNATIONAL WORKSHOP ON ACCELERATOR ALIGNMENT, 2002, Hyogo, JAPAN, pp.288

[4] EPICS collaboration,“EPICS:Experimental Physics and Industrial Control System”, ;M. Kraimer et al., “EPICS: Input / Output Controller Application Developer’s Guide”,, Oct. 2003.

[5] J-I. Odagiri et al. ,“ EPICS Device/Driver Support Modules For Network-based Intelligent Controllers”, in these proceedings.