COACK MULTI-SERVER SYSTEM WITH STARS

Takashi Kosuge, Yuuki Saito, Kazuyuki Nigorikawa, Hiroaki Katagiri, Akihiro Shirakawa, Hiromitsu Nakajima, Kenji Ito, Isamu Abe, Junichi Kishiro and Shin-ichi Kurokawa - KEK(High Energy Accelerator Research Organization), Japan

Abstract

A Central Control System (CCS)[1] for the Beamline Interlock System (BLIS) in the Photon Factory was developed using Component Oriented Accelerator Control Kernel (COACK)[2,3,4] in 2000, and it is working stably without system failure.

The CCS is connected to a private LAN with a firewall, can be monitored or controlled from anywhere within the LAN. Since it is often requested to monitor the CCS from outside of the firewall, it was decided to implement another COACK server for the outside clients.

One of the important issues for a COACK multi-server system is the synchronization among them, and we have found that Simple Transmission and Retrieval System (STARS) is effective in this synchronization.

STARS is a transferring message software for small control systems with TCP/IP socket, and it works on many kinds of operating systems. Since STARS was originally developed as an interface program of COACK for non-Windows system, it is relatively easy to implement the STARS architecture in the COACK system.

We will present the COACK multi-server system with STARS in this report.

1 COACK MULTI-SERVER SYSTEM

COACK is working efficiently and stably in several facilities, however, we think that there are several points to be resolved for the evolution of the COACK system.

A multiplexing of COACK servers is one of the issues including such as load sharing, security (read only), and available through a firewall.

Discussion of COACK multi server system is still being continued by the COACK development team consisting of members from various facilities.

At the same time, it was requested to access COACK server from outside the CCS of the BLIS, and we arrived at the conclusion that multi server system will work effectively on the system. Here we will demonstrate effectiveness of COACK multi server system for using the CCS as an example.

2 BEAMLINE INTERLOCK AND CENTRAL CONTROL SYSTEM

Over 20 beamlines are installed at the 2.5GeV and PF-AR electron storage rings at the Photon Factory for using synchrotron radiation efficiently. Each beamline is equipped with the BLIS, to avoid vacuum troubles and to protect the users from radiation hazards. Each BLIS is controlled by a Programmable Logic Controller (PLC) and it is connected to the PLC for the CCS, which is installed in every experimental halls (PF-2.5GeV, PF-AR NE, PF-AR N and PF-AR NW) with hard wiring. The CCS monitors the status of beamlines through the PLC of the CCS with a PC for a PLC interface and sends control signals to the BLISs.

2.1 CCS and COACK

The CCS is controlled by COACK stably since 1999 (in 2001, the BLIS of the PF-AR was connected to the CCS server). Layout of he entire CCS with COACK is shown in Fig.1.

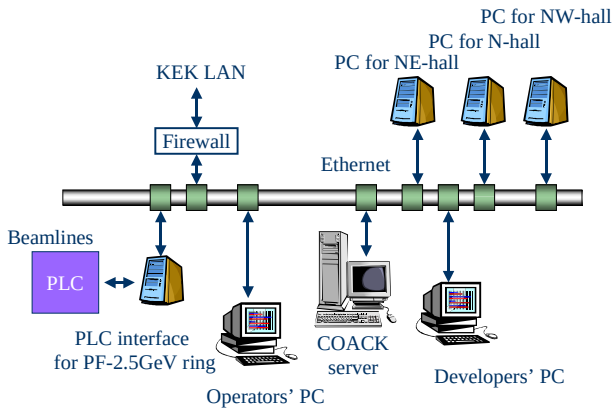

Figure 1: Hard ware composition.

There is a PC for COACK server and 4 PCs of PLC interface. Maintenance and development of programs are available from Developers' PC. Operators use an operators' PC for monitoring the system. All PCs of the CCS are connected to a private LAN, which has a firewall.

2.2 Newly requirements of CCS

The CCS with COACK is working satisfactory as mentioned before. Recently, monitoring CCS from outside the firewall has been requested with the condition shown below; load sharing, short construction time, and easy to implement. In order to meet this requirement, we decided to do mirroring with STARS.

3 DEVELOPMENT OF MULTI-SERVER

The COACK server of the CCS has virtual beamline images with tree structure (XML is used for internal database) and client programs are access data on these virtual images. We installed another COACK server for mirroring, which has same beamline images. In fact, the COACK multi-server system of the CCS is implemented by copying the status data of the virtual images in the main COACK server to the mirror server.

3.1 Mirroring with STARS.

STARS is a transferring message software for small control systems with TCP/IP socket which works on various kinds of operating systems (Fig.2). As it was originally developed as an interface program of COACK for non-Windows system, we can easily implement the STARS architecture in the COACK sytem.

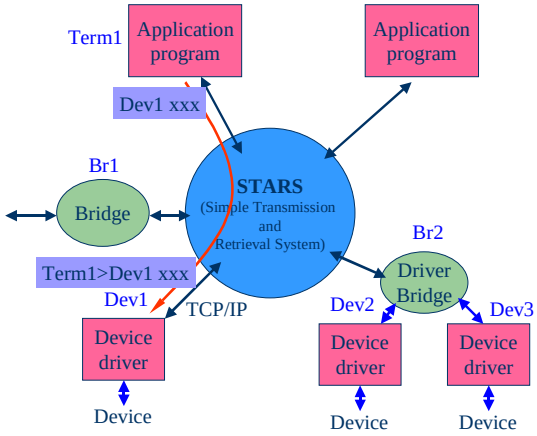

Figure2: STARS

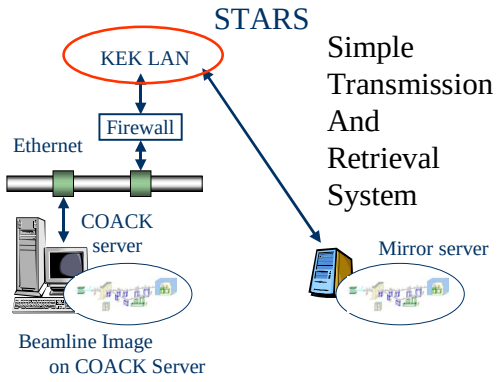

Figure 3: Mirroring with STARS

STARS and COACK server is connected with COACK bridge of STARS (Fig.3). Here we prepared 2 COACK bridges (for main COACK server and mirror server) and these 2 bridges are connected to same STARS server. A client program for mirroring (mirror client) is connected to the STARS server and it sends "flgon" command to the main COACK server firstly. If the status data of beamline image on main COACK server will be changed, information messages from main COACK is sent to mirror client with sending "flgon" command. When mirror client receives this information message, the client changes status data of mirror server with sending "setcache" command.

3.2 Advantages.

The COACK multi-server system with mirroring by STARS has some of advantages as follows.

Firstly, the mirroring is available through a firewall. Since the connection from a client to the STARS server with TCP/IP is well established, the bi-directionality of the network is not required with STARS.

Secondly, we can develop a load-sharing system with mirroring. When the load of the main COACK server is increased, it is possible to share the load by changing the connection of the client to the mirror server.

4 INSTALLATION

A mirroring system has now been installed into the CCS. Users are able to see beamline statuses from out side the firewall through the mirror server. Fig.4 shows a GUI client which is connected the mirror server.

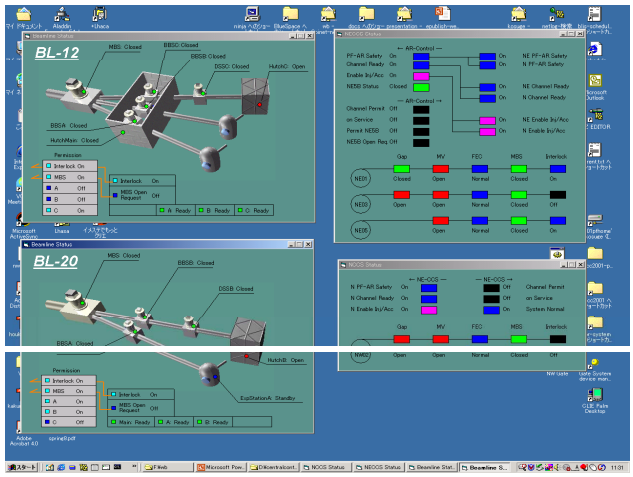

Figure 4: GUI client connected mirror server

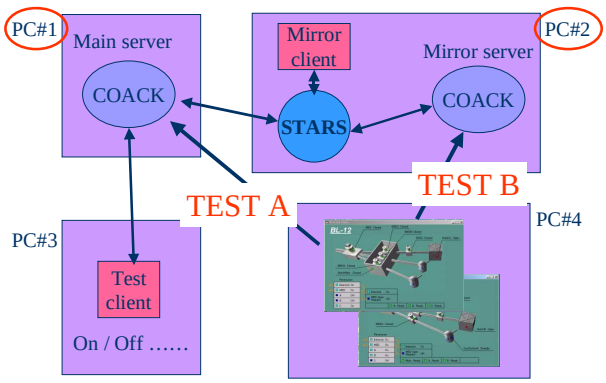

Figure 5: Test of load sharing

5 LOAD SHARING TEST

We tested the effectiveness of mirroring for load sharing with a test bench. A total of 4 PCs was used for the test shown Fig.5.

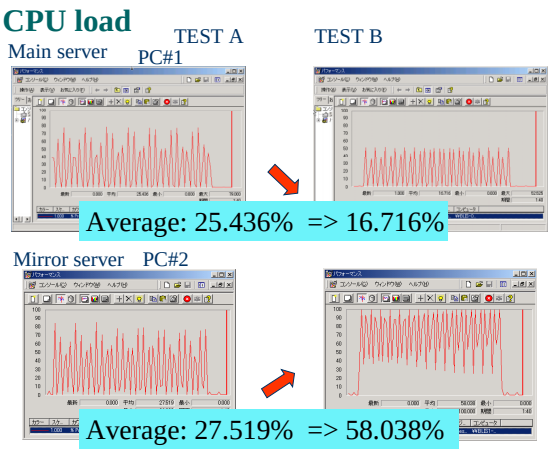

Figure 6: Test result

The main COACK server is running on PC#1 and the mirror server is running on PC#2. A test client who is running on PC#3 and connected to the main COACK server, frequently sends different status data to the server. PF#4 is for GUI clients, in the case of test "A", the GUI clients are connected to the main COACK server and in the case of test "B", the GUI clients are connected to the mirror server. Fig.6 shows the CPU loads of the main COACK server and the mirror server. The CPU load of the main server was decreased in case "B" and the CPU load of the mirror server was increased. This result shows that load sharing has been done satisfactory by mirroring using STARS.

6 SUMMARY

We have developed COACK multi server system with STARS and verified that mirroring with STARS is effective. At the same time, we have demonstrated the effectiveness of the multi server system in COACK. COACK will be upgraded to support multi server system in future.

7 REFERENCES

[1] T. Kosuge, et al., "COACK APPLICATION FOR THE BEAMLINE INTERLOCK SYSTEM AT THE PHOTON FACTORY", PCaPAC2000, Hamburg, 2000

[2] I. Abe, et al., "COACK-II PROJECT ON ACCELERATOR CONTROL KERNEL DEVELOPMENT", ICALEPCS'99, Trieste, 1999

[3] T. Kosuge, et al., "Application Server and Pushing Technology on COACK-II", ICALEPCS'99 Trieste, 1999

[4] I. Abe, et al., "Recent status on COACK project", PCaPAC2000, Hamburg, 2000