DEVELOPMENT OF LARGE, HIGH-RESOLUTION DISPLAY FOR SPRING-8 CENTRAL CONTROL ROOM

T. Hamano#, R. Fujihara, A. Yamashita,
Japan Synchrotron Radiation Research Institute JASRI/SPring-8, Hyogo-ken, JAPAN

Abstract

We developed a large, high-resolution display wall for the SPring-8 Central Control Room. The display wall consists of twelve LCD panels that are controlled using a personal computer (PC) cluster. A PC cluster consists of six PCs interconnected over Gigabit Ethernet. One PC cluster node controls two 40" (1366x768 pixels) LCD panels. The panels are arranged into a 6 x 2 segment to achieve a display resolution of 12 million pixels (8196x1536 pixels). A software program handles the 12 displays through one X Window server. Before deploying the real machine, we constructed a prototype display wall that consisted of 8 x 3 20" LCD panels to achieve a display resolution of 46 million pixels. This paper describes the prototyping, construction, configuration, testing, and usage of the high-resolution display wall.

INTRODUCTION

Previously, four plasma panels were used to display the accelerator and alarm status in the SPring-8 central control room. Those display had resolution of 1280x768 pixels in 50" size panel. They displayed alarm and accelerator status independently, so they cannot windows larger than one display. We decided to develop a larger, higher-resolution display wall to display more information. For example, sharing operator console displays would enable smoother operation.

We searched for commercially available solutions that would satisfy the following criteria.

1. High resolution: The display wall must display the entire operator console display that has a resolution of 8196x1536 pixels.
2. Size: The display must be larger than 600 cm and its thickness must be less than 50 cm.
3. Brightness.
4. 24-h operation.
5. Single-display operation.
6. Cost.

However, we were unable to find solutions that satisfied these criteria. For example, rear projection displays have poor resolution, large size, and high cost. A front projector has insufficient brightness and short bulb lifetime.

Therefore, we decided to manufacture our own large, high-resolution display using inexpensive LCD panels to satisfy the abovementioned criteria.

PROTOTYPE MACHINE

Design

The large, high-resolution display wall has been designed to provide several functionalities.

1. Display of the operator console and GUI window on a single display wall.
2. Realize a display wall that is larger and has higher resolution than a four-plasma display.
3. The four plasma displays are controlled independently, whereas the display wall is controlled using one keyboard and one mouse.

We built a prototype consisting of a 20.1" LCD (display resolution: 1600x1200 pixels) in 8 x 3 rows controlled by a PC cluster. One cluster node is connected to four display panels using a graphic device that can control four displays. A PC cluster consists of six nodes. (Fig. 1)

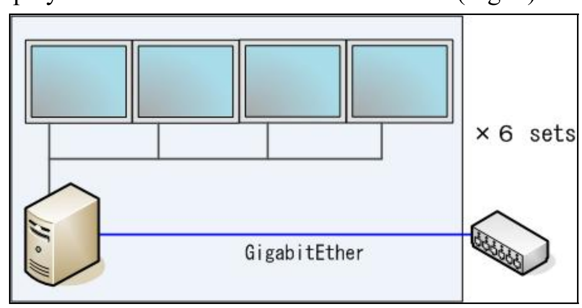

Figure 1: Connection of 4 LCDs to a display machine.

Development

We compared two software programs—Xdmx (Distributed Multihead X) [1] and Rocks Clusters[2].

1. Xdmx + Linux

We controlled the display wall using Linux (Fedora 10[3]) and Xdmx Version 1.3.00.

Xdmx handles the six network machines through one X Window server. We controlled the display wall using one keyboard and one mouse through Xdmx (Fig. 2).

#hamano@spring8.or.jp

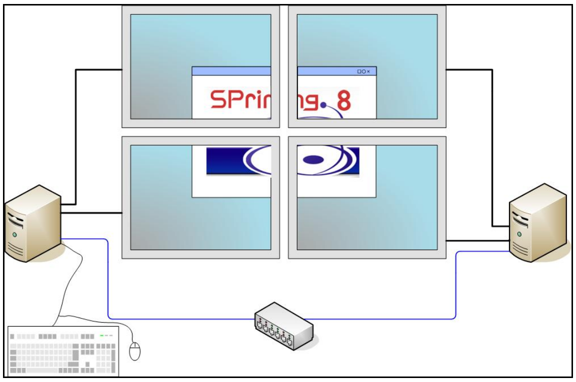

Figure 2: Control using one keyboard and one mouse.

We installed Fedora10 and Xdmx on machine1, and connected keyboard and mouse to a machine out of six machines (Fig. 3). One machine is operated in Fedora 10 Linux and uses NVIDIA NVS440 for graphic board.

One NVS 440 are able to be connected to four LCD panels through DVI (Digital Visual Interface) interfaces.

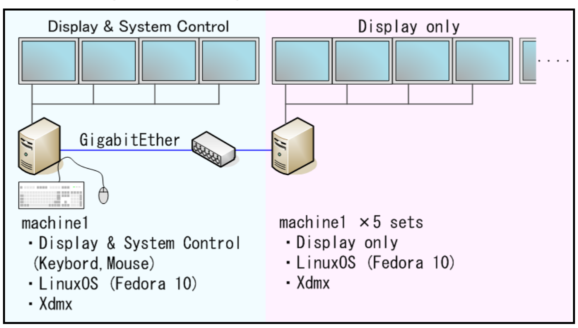

Figure 3: Fedora 10 and Xdmx installed on machine1.

We setup an X Window server system on every machine and then setup the xhost program. The xhost program lists hostnames allowed to connect to the X Window server, and enables the X Window server to be controlled. The xhost program enables the display machines to be controlled using the control machine.

We successfully displayed one X Window over 24 displays using Xdmx. However, the display speed was unsatisfactory. This may be attributable to the network performance of the control machine. The control machine controls all other nodes, and therefore, it has to handle heavy network traffic. We added a fast network interface card (NIC) to the control machine to improve the network performance. Two-port bonding (link aggregation[4]) was used for load balancing.

We measured the throughput among the display machines from the control machine.

- Physical port: 188 Mbps (average).
- Bonding port: 298 Mbps (average).

However, did not feel improvement of the speed on display wall machine.

2. Rocks Cluster + Viz Roll

Another attempt was made on Rocks Cluster 5.0 and optional packages Viz Roll 5.0. Rocks Cluster software is PC cluster tool. This tool was developed based on the CentOS 5 [5] by University of California, San Diego (UCSD)[6]. optional packages Viz Roll 5.0 is multi head display tool to configure the screen. Rocks employ a dedicated machine as a controller (Fig. 4).

Figure 4: Control machine and display machines.

We installed Rocks Cluster server (front-end) on the control machine and Rocks Cluster client (tile node) on the six display machines (Fig. 5).

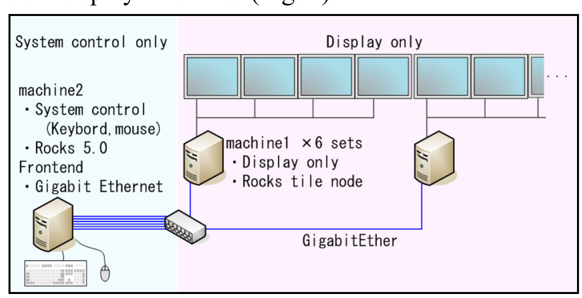

Figure 5: Rocks Cluster and network bonding.

Rocks Cluster and Viz Roll handle 24 displays as one display. The front-end machine has its own monitor that displays a reduced version of the overall display.

Rocks Cluster can be set up more easily than Xdmx because it integrates the OS and add-on packages.

To balance the network traffic, we installed seven NICs in the control machine. One port was used for external communication, and the other six ports were used for point-to-point connections to the display machines using vlan with an intelligent switch. We improved the network load balancing of the control machine through network port bonding. This enables us to realize smooth movement of windows and smooth playback of a video file.

We measured the time required to display a large image (12561x3350 pixels, 11 MB JPEG image).

Xdmx required approximately 20 s, whereas Rocks Cluster required only 2 s. Therefore, we found the display speed of Rocks Cluster to be satisfactory.

Figure 6: Display wall (SPring-8 Central Control Room). In the center of the display, sub-screens display real-time NTSC video.

INSTALLATION IN THE CONTROL ROOM

Based on the success of the prototype machine, we constructed a display wall in the SPring-8 Central Control Room. (Fig. 6)

We adopted Rocks Cluster as the control software. In addition to its superior performance, Rocks Cluster afforded the following advantages:

- Easy to install.
- Easy to manage and control.
- Easy to reinstall the OS in case of problems.

We arranged a 40" LCD (1366x768 pixels) in 2 x 6 rows. The total number of pixels is 8196x1536 = 12589056 pixels. One tile node machine controls two LCD panels through an NVIDIA GeForce GTX 285[7] graphic controller.

The contents of each operator console can be displayed on the display wall using Virtual Network Controller (VNC) to share information between operators. A Real VNC [8] implementation can increase or decrease the display size for smooth operation.

The LCD panel has an additional function that displays an NTSC video screen on the child screen. This is used to display a real-time screen monitor and spectrum analyzer.

CONCLUSION

We controlled and integrated multiple displays as a single large display using Rocks Cluster and Viz Roll. We have now begun using the large display wall, and expect that the advantages offered by the use of the large, high-resolution display will be beneficial for further developments.

REFERENCES

[1] http://dmx.sourceforge.net/.
[2] http://www.rocksclusters.org/wordpress/.
[3] http://fedoraproject.org/.
[4] http://grouper.ieee.org/groups/802/3/ad/index.html.
[5] http://www.centos.org/.
[6] http://www.ucsd.edu/.
[7] http://www.nvidia.com/object/product_geforce_gtx_285_us.html.
[8] http://www.realvnc.com/.