CONTROL SYSTEMS ON LOW COST COMPUTERS

Ernest L. Williams Jr., Greg S. Lawson, ORNL, Oak Ridge, TN 37830, USA

Abstract

Computing infrastructures for accelerator control systems can now be built using low cost, commodity-type computers. The commonly used EPICS control system toolkit has been adapted to facilitate its deployment on Pentium-based PCs running the Linux operating system, as well as on other inexpensive platforms.

Low-cost, reliable enterprise-class computers can be obtained from many commercial vendors for such control system network-based services as Domain Name Service (DNS), Network Information Services (NIS), and Dynamic Host Configuration Protocol (DHCP). Control system file servers can also be configured using inexpensive rack-mounted PCs and multiple monitor operator consoles can be built with Linux-based PCs or standard X-windows software with high performance video graphic adapters. At the front end of the control system, input-output controllers can be configured also as PCs running embedded Windows.

Several accelerator laboratories, including the Spallation Neutron Source (SNS) at the Oak Ridge National Laboratory are taking advantage of these possibilities. This paper will review the use of inexpensive computers in the accelerator community, with particular emphasis on the issues and applications at SNS.

1 INTRODUCTION

Commodity based personal computers (PCs) are starting to play a larger role in the accelerator community. PCs running the Linux operating system (OS) have already begun replacing the traditional SUN Solaris and HP-UX control system file servers. Many of the accelerator labs have introduced the PC to achieve the flexibility of choosing non-proprietary hardware and software. This flexibility allows a combination of both an "open source" and a commercial software mix at some labs. For example, the control system enterprise file servers, database servers, and network support servers can be run using Linux while smart embedded front-end processors can be based on Windows Embedded XP. The EPICS developer community has strived to make the control system software toolkit LINUX and WIN32 ready. The recent introduction of an operating system independent (OSI) version of EPICS (3.14.x) [1] facilitates its use on commodity-type PCs running the OS of your choice. The large base of PC vendors helps to optimize cost, reliability, and availability. We discuss below some of the control system integration issues, vendor support, reliability, and remote management. We also summarize how some accelerator labs deploy and use commodity-type PCs at their facility.

2 ARCHITECTURE

The low cost of PC-based servers allow the controls system infrastructure to use a de-centralized "black-box" appliance architecture (DBA). The DBA architecture allows each PC to perform one job and one job only. This reduces the load on each server and increases the reliability. Think of this is as "one box" - "one service". For example, one box for DHCP, one box for DNS, one box for NIS, one box for an EPICS archiver, one box for an EPICS alarm handler, one box for an EPICS front-end process boot server, etc… Figure 1 shown below illustrates the concept of DBA.

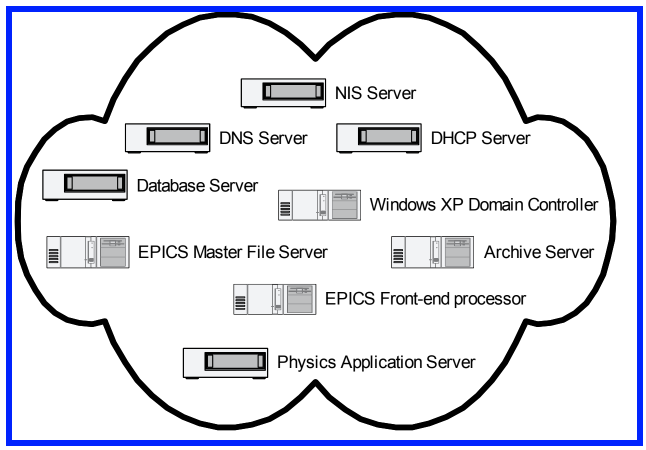

Figure 1: De-centralized "black-box" appliances

The DBA architecture also lends itself to taking advantage of external storage solutions such as network attached storage (NAS) appliances. The cost of external storage continues to go down. Having the option of using multiple PCs gives the control system designer freedom to choose the CPU architecture specific for the application. Taking a look at the various choices with Intel processors:

- Intel Itanium 2 ---- High-end number crunching servers
- Intel Xeon ---- High-end file server applications
- Intel Pentium 4 --- Human machine interface (HMI) workstations.
- Intel Pentium 4 Mobile --- Mobile HMI used for control system maintenance activity.

Since Moore's Law continues to hold even after 38 years; current PC bus architectures will have to change to keep up with the continued enhancements in the processor world. The next generation bus technology is needed due to the increased I/O demands for high-bandwidth and low latency control system applications. Applications such as real-time video over Gigabit Ethernet, fiber channel data storage applications, and high-performance graphics applications require better bus performance. PCI-64 boasts a transfer rate of 264 Mbytes/sec. What will be

0-7803-7739-9

next after PCI bus? PCI-X offers a short term interim solution with a bus throughput of about one Gigabyte per second. PCI Express a serial I/O technology will provide bandwidth up to 16 Gigabytes per second and supports chip-to-chip level interfaces. Infiniband is another promising prospective bus technology. Infiniband is a serial switch-based architecture and specification for moving data between processors and I/O devices. This new bus technology offers a throughput of up to 10 Gigabytes per second. There are already products based on Infiniband on the market for external peripherals. The standard PC serial (i.e. RS232) interface is being taken over by Universal Serial Bus (USB) and FireWire (IEEE 1394). The next generation commodity-type PC will probably use PCI Express for chip level interfaces and graphics adapters while storage and other high-speed peripherals will use the Infiniband technology.

3 REMOTE MANAGEMENT AND MAINTENANCE

Most facilities deploy a combination of centralized and distributed computers across the control system enterprise and must deal with remote management and maintenance issues. Remote Health monitoring and PC hardware management can be subdivided into three categories:

1. Low-level system monitoring/management
2. Operating system monitoring/management
3. Network system monitoring/management

Remote Health monitoring and PC hardware management can be done effectively using industry standard approaches. Remote management is now a commodity since; most PC vendors now provide the Intelligent Platform Management Interface (IPMI) standard especially for their high-end servers. IPMI is a standardized, abstracted, message-based interface to intelligent platform management hardware [2]. Health monitoring hardware based on IPMI is used to track system parameters such as fan speed, temperature, voltages, processor statistics, etc… Facilities with heterogeneous server environments can take advantage of the interoperability of IPMI. Remote power-off/reset, firmware updates, and hardware inventory is also supported via IPMI. The promoters of IPMI include well-known manufactures such as Intel, Dell, Hewlett Packard, and NEC. All SNS control system servers are IPMI compliant. The SNS uses the DELL PowerEdge server line for many of the applications. DELL provides a third-party software tool called Dell OpenManage (see figure 2) that integrates with their IPMI compliant servers

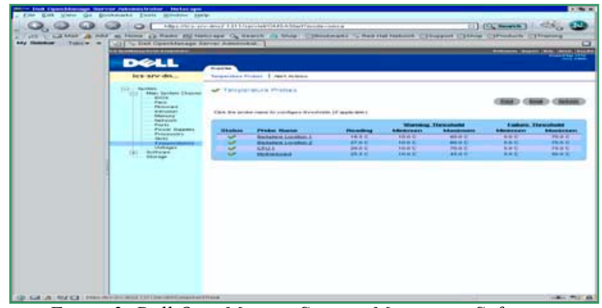
Figure 2: Dell OpenManage Systems Monitoring Software

In a heterogeneous environment the challenge of keeping tabs on the operating systems services and daemons can be difficult. There are many commercial and open-source solutions that will do the job. The idea is to select a software solution that is client/server based which queries monitored services across the network and displays system status on a "user friendly" interface. A software solution that uses the simple network management protocol (SNMP) will also help maintain interoperability across systems. Critical network services that may support the control system network such as DNS, NIS, and DHCP usually run on a server class PC and must be checked for availability. Automatic notification of the appropriate control system or information Technology (IT) personnel can be challenging if the enterprise is behind a firewall. The need to perform periodic software updates, bug fixes, and security patches remotely across many platforms warrants reliable and robust software distribution tools. The SNS control system will standardize on Microsoft System Management Software (SMS) for software distribution management on beam diagnostics Windows XP systems.

The SNS has also standardized on the Red Hat Linux OS and will use a combination of open source and commercial solutions for software management and system monitoring. In a client/server environment the network health is also critical and must be monitored as well. The open source community and commercial vendors provide many cost effective solutions for monitoring and managing networks. The SNS uses a combination of HP OpenVIEW and CISCO Works 2000 to monitor and manage the accelerator network. We are currently at work to integrate the health monitoring of the entire control system infrastructure into EPICS.

4 FRONT-END PROCESSORS

The days of the minicomputer and CAMAC I/O as a front-end processor (FEP) have long been over. We find that VMEbus/VXIbus is still the most popular FEP implementation. However, another player has been carving out its place as a FEP in the accelerator community. Well, now commodity-type PCs are consistently becoming the "first draft" pick. Rack mount PCs offer an amazing amount of processing power for the cost. The large quantity of I/O modules

available for PCs continues to increase allowing accelerator control systems engineers flexibility in design and implementation.
The SNS plans to use about four hundred rack-mount PCs as FEPs for the beam diagnostics control system. There are already beam diagnostic FEPs installed and running on the SNS front-End section of the LINAC. The FEPs for the beam diagnostics are thought of as network attached diagnostics (NADS) [3]. The Beam Diagnostics group has chosen the NAD approach to provide a stand-alone FEP for each beam diagnostic sensor. An example of the SNS Beam Position Monitor (BPM) NADS versus the traditional VME-based FEP is shown in figure 3.

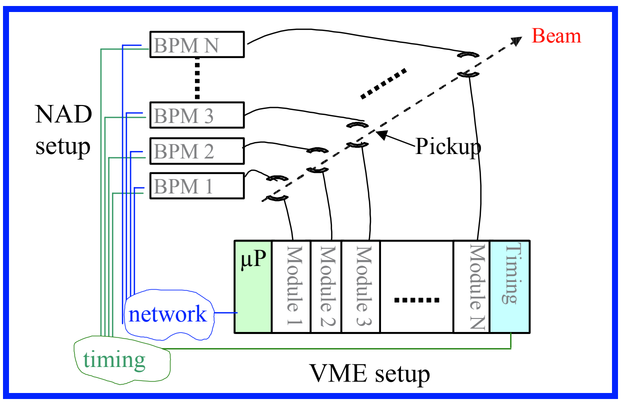

Figure 3: The NAD versus VME configuration

Currently the NADS are running on top of Windows XP Embedded OS. The NAD applications are developed and run in LabVIEW. Careful configuration of the OS and application programming with LabVIEW ensures the reliability and robustness of the system. LabVIEW was selected as the programming environment for the NADS because of its rapid software prototyping ability. The PC-based platform chosen for the NADS allows the selection of an OS and software toolkit to be application driven. For applications that require a hard real-time OS a commercial solution such as VxWorks or an open source solution like RTEMS can be used. Real-time performance can also be achieved by putting more of the processing burden in Field Programmable Gate Arrays (FPGA) s. Many of the PCI I/O cards used for the beam diagnostics sensors are built around FPGAs. The use of operating system independent software such as EPICS R3.14 gives the control system engineer the freedom to architect the combination of commodity-type PC and software for a sound FEP blueprint.

5 CONCLUSION

An outlook towards the future shows commodity-type PCs playing an ever-increasing role in the accelerator community. The continuing advances in PC processor, memory, peripherals, and bus technologies will keep performance high and costs low. Commercial software companies are embracing the open-source model and delivering high-end applications on PC platforms. Do low cost PCs imply a decrease in high availability? Many hardware and software vendors are evolving their technical support model to provide high-quality 24x7 support. Even the open source OS software vendors are evolving the software configuration and release cycle to provide better performance and stability. One example of this is the recent introduction of the "Linux Advanced Server and Workstation" from Red Hat Inc. A survey of other accelerator labs on the use of PCs for control systems showed that integration of PCs into new and existing facilities is still on the rise. The Swiss Light Source (PSI) has led the way in building their entire control system infrastructure on commodity-type PCs except for the use of VME-based FEPs. The SNS will also continue to deploy its control system on top of commodity-type PCs with a mix of both VME/VXI and PC front-end processors.

6 REFERENCES

[1] L. Dalesio, "EPICS: Recent Applications and Future Directions," PAC 2001, Chicago, Illinois., USA, May 5-7, 2003

[2] http://developer.intel.com/design/servers/ipmi

[3] W. Blokland, T. Shea and M. Stettler, "Network Attached Devices at SNS," to be published at DIPAC 2003, Mainz, Germany, May 5-7, 2003