Fermi National Accelerator Laboratory

FERMILAB-Conf-91/84

The Fermilab Experience: Integration of UNIX Systems in a HEP Computing Environment

Uday Pabrai
Fermi National Accelerator Laboratory
P.O. Box 500
Batavia, Illinois 60510

March 1991

* Presented at Computing and High Energy Physics Conference, KEK, Tsukuba, Japan, March 10-15, 1991.

Operated by Universities Research Association Inc. under contract with the United States Department of Energy

The Fermilab Experience: Integration of UNIX Systems In A HEP Computing Environment

Uday Pabrai
Fermi National Accelerator Laboratory
Batavia, IL

There is an increased emphasis within organizations to migrate to a distributed computing environment. Among the factors responsible for this migration are [1]:

1. A proliferation of high performance systems based on processors such as the the Intel 80x86, Motorola 680x0, RISC architecture CPU's such as MIPS Rx000, Sun SPARC, Motorola 88000 and Intel 860 series.

2. A significant reduction in hardware costs.

3. Configuration based on existing local area network technology.

4. The same (to a large extent) operating system on all platforms.

A characteristic of distributed computing is that communication takes the form of request-reply pairs. This is also referred to as the client-server model [2]. The client-server model is rapidly growing in popularity and in many scientific and engineering environments is replacing transaction-based and mainframe systems [3]. Over the last few years, Fermilab has been in the process of migrating to a client-server model of computing.

1. Objective

Integrate is based on the Latin word *integrare* which is defined as "to make into a whole by bringing all parts together; unify." The objective of this paper is to describe how UNIX systems and the associated technology have been integrated in a predominantly VAX/VMS and VM/CMS environment at Fermilab. The issues discussed include operating systems, network considerations, electronic mail, heterogeneous distributed computing, software development, graphical user interfaces, and terminal emulation.

2. UNIX is not UNIX is not UNIX...

UNIX systems available in the market-place today typically use different processor technologies and have slight variances in the operating system.

2.1. Processor Technology

Sun Microsystems SPARCstation series is based on the Scalable Processor Architecture (SPARC). Silicon Graphics and DIGITAL have based their UNIX workstations on MIPS R2000/R3000 processors. IBM RS/6000 systems use the Performance Optimisation With Enhanced Risc (POWER) as their RISC processor. The processor determines, among other things, how data is stored on the system. The SPARC processor is *Big Endian:* the lowest memory address holds the high-order byte of the integer. The MIPS processor can be configured as *Little Endian* or Big Endian. Little Endian machines are those where the lowest memory address contains the low-order byte of the integer. All Silicon Graphics systems are configured as Big Endian. It is important to understand what implication, if any, the processor architecture has on the design of applications.

2.2. Operating System Origin and Future Direction

The IBM AIX operating system is derived from AT&T UNIX SVR1 and includes AT&T SVR2 and SVR3 extensions that are required to conform to the AT&T Base System V Interface Definition (SVID) Issue 2, with certain exceptions. AIX for the RS/6000 conforms to the Portable Operating System for Computer Environments (POSIX) IEEE Standard 1003.1-1988. The AIX for RISC systems uses the following priority where conflicts or differences may exist: POSIX, SVID, 4.3 BSD, AIX/RT. The Sun SPARCstation operating system is called SunOS and is based on BSD 4.2, 4.3, includes features of AT&T SVR3 and some Sun enhancements. Sun is committed to delivering an implementation of SVR4 in the near future. It is expected that this release will be referred to as SunOS/SVR4.

Digital's ULTRIX operating system is based on 4.2 BSD with enhancements from 4.3 BSD and 4.3-5 BSD. The operating system is compatible at the source code level with AT&T's System V Interface Definition (SVID) Issue 2. The ULTRIX operating system is fully conformant with the IEEE Standard 1003.1 POSIX specification.

The Silicon Graphics IRIX operating system is based on System V Release 3 and includes some 4.3 BSD and SGI enhancements.

2.3. The Fermilab UNIX Seed Project

The objective of the Fermilab UNIX Seed Project was to distribute UNIX systems at various experiment sites to understand issues involved in configuring several regional computer/data centers based on UNIX and its associated technology. In an environment that is predominantly VAX/VMS based, it was critical to integrate the UNIX seed systems to assure their usability. Integration in this sense implied the following:

1. To provide local system managers, who had previously had no UNIX system administration experience, with the necessary tools to effectively administer their system - this included providing documentation, developing system management utilities, and "hands-on" training to configure their systems.

2. Test the viability of using third-party disk drives (such as CDC WREN V and WREN VII), 8MM tape drives (such as Exabyte 8200), and additional memory (1 MB and 4 MB SIMMs) to reduce the cost of configuring each UNIX system.

3. Configure UNIX and VMS systems to be nodes on both the Internet and DECnet networks to assure transparent end-user communication.

A total of 26 systems were configured at 9 experiment sites for the seed project. The systems included as a part of this project were the IBM Powerstation 320 (RS/6000), Sun SPARCstation 1+ and SLC, DECstation 5000, and Silicon Graphics 4D/25 and 4D/20.

Table 1 provides information on the configuration of the *seed* systems.

Specification	SPARC 1+	SGI 4D/25	IBM PS320	DEC 5000
CPU (MHz)	SPARC	MIPS R3000	POWER	MIPS R3000
Clock Speed (MHz)	25	20	20	25
MIPS	15.8	16	27.5	xxx
MFLOPS (Dbl Prec.)	1.7 Linpack	1.6 Linpack	7.4	3.0
Main Memory (Min/Max)	8/40 MB	8/32 MB	8/32 MB	8/32 MB
Cache	64 KB	96 KB	40 KB	128 KB
Std. Interfaces	SCSI-2 2 RS 423	SCSI 2 RS 232	SCSI 2 RS 232	SCSI 2 RS 232
Ethernet	15-pin	15-pin	15-pin+BNC	BNC
Operating System	SunOS	IRIX	AIX	ULTRIX
Communication	TCP/IP, NFS ONC RPC	TCP/IP, NFS	TCP/IP, NFS, NCS	TCP/IP, NFS NCS
FORTRAN Compiler	Sun	SGI	AIX XL	MIPS
Windowing System	OpenWindows Open Look	4Sight NeWS, GL, X	AIXwindows MOTIF	DECwindows MOTIF

Table 1: UNIX Seed System Specifications

3. Networking

3.1. Architectures & Protocols

UNIX systems use two protocols at the transport layer: the Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). At the network layer, UNIX systems use the Internet Protocol (IP). The use of these protocols enables UNIX systems to be hosts on the Internet network (also referred to as the DARPA/NSF Internet and the TCP/IP Internet).

VAX/VMS systems use the Network Services Protocol (NSP) as the communication protocol at the transport layer and the Routing Protocol (RP) at the network layer. NSP and RP are a part of the Digital Network Architecture (DNA). DECnet is the collective name for a family of communication products that allow DIGITAL operating systems to participate in the network. DNA specifies the functional layers in which DECnet software is arranged on each node, and the communication protocols through which the corresponding layers at different nodes communicate with each other. Figure 1 illustrates the OSI/Reference Model, DNA, and the Internet architectures.

3.2. Network Software Products

TELNET, *FTP*, *RLOGIN*, *RSHELL*, and *RCP* are some communication commands executed by users on UNIX hosts to interact with other UNIX systems on the Internet network. TGV's MultiNet product enables VAX/VMS nodes to be hosts on the Internet network. The MultiNet product consists of TCP/IP software, networking utilities, and programming libraries. The MultiNet product provides support for the following network protocols: IP, ICMP, TCP, UDP, ARP, RARP, FTP, TELNET, SMTP, DNS, NTP, FINGER, WHOIS, SYSTAT, BOOTP, TFTP, EGP, RIP, and SNMP.

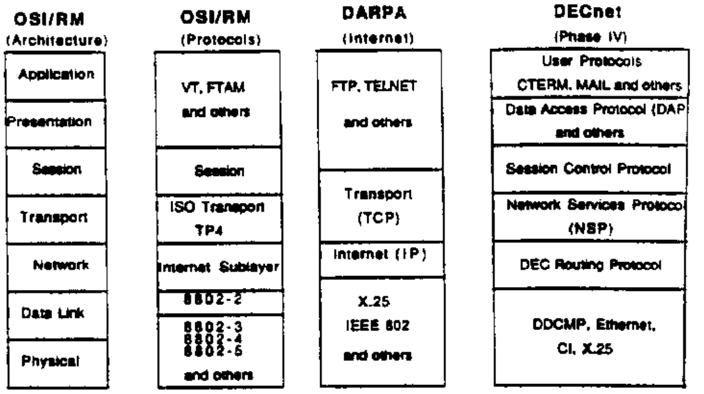

Figure 1: OSI/RM, DNA and Internet: Architecture & Protocols

SET HOST, *COPY*, and *DIR* are some commands executed by users on VAX/VMS nodes to interact with other VAX/VMS nodes on the DECnet network. Ki Research's KINET product enables UNIX systems to be nodes on the DECnet network. The KINET software provides support for the following network protocols: CTERM, DAP, MOP, and LAT.

Thus, networking products such as MultiNet and KINET, enable VAX/VMS systems and UNIX hosts to co-exist as nodes on the Internet and DECnet networks.

How does an Amdahl 5890-600E running IBM's VM/XA operating system be a node on the Internet network? A controller from Bus-Tech, Inc. (BTI) provides the Amdahl system with connectivity to Ethernet. The BTI controller is channel attached to the Amdahl system. IBM's TCP/IP for VM product provides support for the following protocols: TCP, IP, UDP, ICMP, ARP, FTP, TFTP, SMTP, and TELNET. Besides TELNET and FTP, the following applications are possible: TCPSEND (similar to the VM SEND command) and TCPNOTE (similar to the VM NOTE command) [4].

3.3. Network Hardware Devices

At Fermilab, bridges such as Digital's LAN Bridge 100 are devices used on the network to bridge traffic at the data link layer. The LAN Bridge 100 can filter IEEE 802.3/Ethernet frames. It also bridges Ethernet multicast frames - an example of which are Local Area VAXcluster (LAVc) packets. The bridge can also filter packets on the basis of the destination Ethernet address.

UNIX systems, inherently, use TCP/IP protocols to communicate at the network and transport layers. Typically, bridges are unable to filter layer 3 (network layer) or above packets. Thus, products such as Cisco Systems IGS router/bridge, need to be added to the network to route protocols such as TCP/IP, DECnet, ISO CLNS, Appletalk Phase 1 and 2, and LAT. Routers are network layer devices and are better suited to prevent local TCP/IP traffic from reaching the backbone network. All systems on one side of the Cisco router are on the same Internet subnetwork.

3.4. Electronic Mail

BITNET, SMTP and VAX Mail are the predominant e-mail protocols on the Fermilab network. Joiner Associates' JNET product enables VAX/VMS systems to be nodes on the BITNET network. TGV's MultiNet product enables VAX/VMS systems to be nodes on the Internet network. UNIX hosts such as Silicon Graphics, Sun SPARCstations and IBM RS/6000s run Ki Research's DEKnet product. This product enables a UNIX system to be a node on the DECnet network.

The Fermilab Mail Server Project

The objective of the Fermilab mail server project is to establish a gateway that supports a suite of e-mail protocols and provides a migration path to a X.400/X.500 based e-mail network [6]. The gateway would enable users on UNIX and VAX/VMS hosts to be able to send and receive BITNET, X.400, and VMS Mail without the installation of additional software on each system. In the near future, the gateway would provide a central repository of information pertaining to local usernames, local e-mail addresses, and e-mail gateways. Such a repository could eventually lead to the deployment of a White Pages service available to users on the wide-area network.

4. Heterogeneous Distributed Computing

4.1. Sockets and Streams

There are several interfaces that can be used by application programs to interact with communication protocols such as TCP/IP. Berkeley sockets and AT&T System V Streams (also referred to as Transport Layer Interface) are two common application program interfaces to TCP/IP. Figure 2 details socket system calls used for connection-oriented and connectionless client-server applications [6].

4.2. Remote Procedure Calls

ONC RPC

Sun Microsystems Open Network Computing (ONC) provides an environment to develop distributed applications and network services. ONC provides the tools to handle interprocess communication and data translation between systems on the network. Remote Procedure Call (RPC) is that component of ONC that enables client-server applications to use a network service in a manner similar to calling a local function.

Using RPC the client application makes a request (call) and waits for a response (return value). The server waits for the request, processes it, and sends back a return value.

Another component of ONC is eXternal Data Representation (XDR). XDR does the translation between the data representation used by the local system and a standard, machine independent representation. XDR defines a standard representation for the most commonly used primitive data types. XDR is typically used in conjunction with ONC RPC to handle data inconsistencies between systems sharing data via remote procedure calls [7].

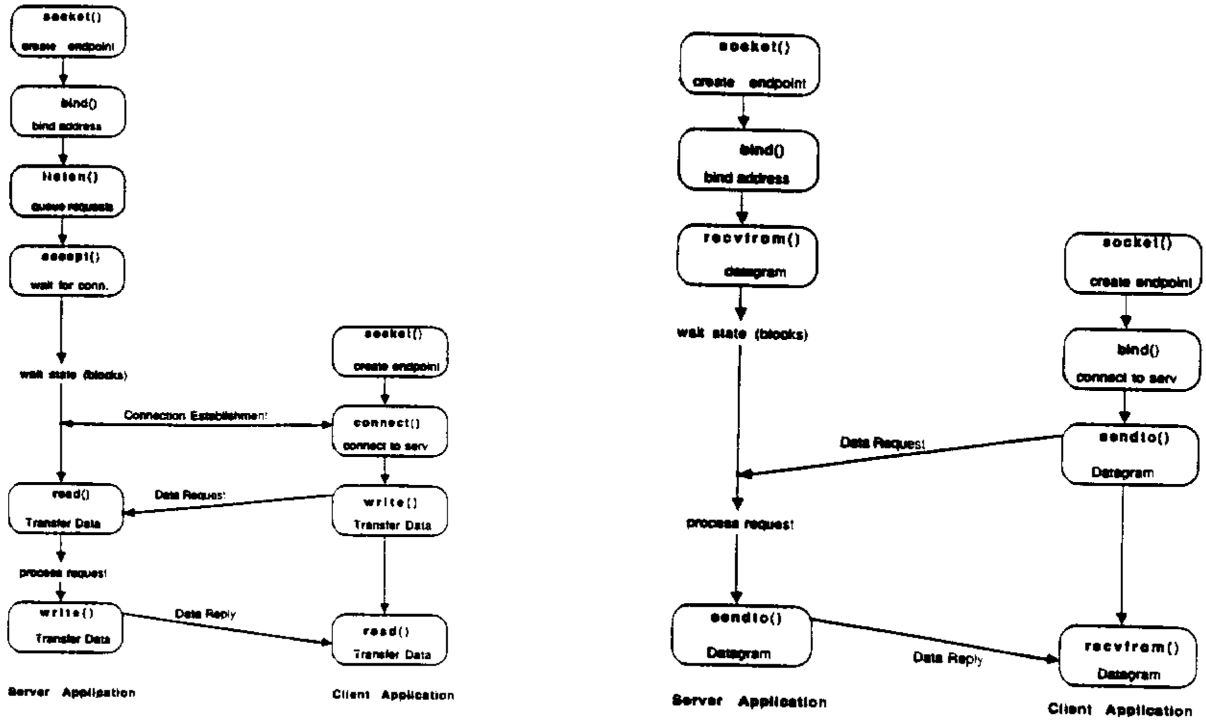

Figure 2: Connection-oriented and Connectionless Socket-based Applications [7]

NCS & Task Broker

Network Computing Standard (NCS) is an HP/Apollo distributed computing product that has been placed in the public domain. Today over 200 organizations have licenses to use NCS. NCS is included with Digital's ULTRIX and IBM's AIX operating systems.

NCS is based on using remote procedure calls to send compute intensive applications to powerful processors on the network. NCS can be used to break a computation task into *n* individual jobs for *n* systems on the network.

The major components of NCS are:

RPC RTE — The Remote Procedure Call Run Time Environment provides routines to enable local programs to execute procedures on remote hosts.

NIDL Compiler — The Network Interface Definition Language (NIDL) compiler generates most of the required calls to the run-time library. Typically, users do not use many RPC runtime routines directly but instead write interface definitions in NIDL.

LB — The Location Broker (LB) determines at run time which potential servers have the requested service, thereby eliminating the need for the end user to specify which system will service the RPC.

Task Broker is also an HP/Apollo distributed computing application that is designed to be used on the network by end users who need additional computation power and cannot modify their applications. The Task Broker basically transmits a bid on the network to find the least busy machine to process the job [8].

In the near future, it is expected that the Task Broker product will be integrated with NCS, thereby, enabling NCS applications to take advantage of Task Broker's technique for determining the least busy server [9].

4.3. Fermilab & TCP/IP Based Applications

Any distributed application may involve designing a client process that runs on a local system and uses the services provided by a server process that may be running on a remote system. The client process not only needs to know the Internet address of the remote system but also a way to identify the server process on that system [10]. Port numbers are addresses by which processes can be identified on TCP(UDP)/IP based systems. A port number is an unsigned 16 bit integer that takes on values between 0 and 65536.

With the proliferation of UNIX systems on the Fermilab network it was important to determine a methodology to assign port numbers for TCP/IP based client-server applications. [10] describes how port numbers can be assigned statically (hard-coded within the client-server application) or dynamically (determined by the operating system). Also, the portmapper network service provides a standard way for client applications to determine the port number of any remote server application. Server applications on start up, request an available port number from the system and then register their services and the corresponding port numbers with the portmapper. ONC RPC applications are uniquely defined by a program number. Fermilab's Data Communications Group assigns unique port numbers between 20,000 and 30,000 and RPC program numbers between 25000000 (Hex) and 26000000 (Hex) for client-server applications that are either based on the central facility systems or that execute on a large number of distributed systems.

4.4. The Fermilab FORTRAN Compiler Evaluation Project

FORTRAN is the language of choice at HEP computing environments primarily because of its advantages for scientific applications and statistical and mathematical analysis. The Fermilab Programming Language Support requires FORTRAN language compilers to be ANSI FORTRAN 77 compliant. The additional required features are a subset of VAX/VMS FORTRAN language extensions and are categorized into two groups. The first group includes all features that must be supported without any failures. The second group includes the not so critical features with a maximum of two failed features [11]. The Fermilab FORTRAN compiler evaluation project examined FORTRAN compilers on the IBM RS/6000, Sun SPARCstation, Digital DECstation, Sony NEWS, and Silicon Graphics Personal IRIS systems. All compilers tested met Fermilab FORTRAN compiler mandatory requirements.

This project also compared the new ANSI Fortran 90 standard with Fermilab FORTRAN language requirements.

5. Windowing Systems

X, also referred to as the X Window System, is a network transparent window system. The X Window System consists of three major components: the X server, Xlib, and the toolkit intrinsics. X is based on an asynchronous network protocol, called X protocol. In order to mask protocol encoding and transport interactions the X Window System includes an interface library. Xlib is the C language interface library. The X protocol and X library do not specify menus, scroll bars, dialog boxes or how an application should respond to user input. Toolkits, such as the X toolkit, provide menus, scroll bars, dialog boxes etc. Graphics libraries, such as the GKS library, transform abstract object descriptors into graphics requests. Toolkits, graphics libraries and user interface management systems can be implemented on top of Xlib. Two major user interface toolkits and window managers are MOTIF and Open Look.

5.1. Open Look & Open Windows

Open Look is an interface standard promoted by Sun Microsystems and AT&T. Open Windows is Sun's composite window system supporting X, SunView and Network Extensible Window System (NeWS) applications. All Fermilab Sun SPARCstations run the Open Window software. Open Windows version 2.0 includes XView (Sun's toolkit for Open Look), AT&T intrinsics (referred to by Sun as Open Look Intrinsics Toolkit (OLIT)), and Sun NeWS Development Environment. Integrated Computer Solutions offers an OSF/MOTIF product for Sun SPARCstations.

5.2. MOTIF

MOTIF is the interface standard promoted by the Open Software Foundation (OSF). OSF/MOTIF includes a MOTIF toolkit, User Interface Language (UIL), and a OSF/MOTIF window manager.

The IBM RS/6000 system uses the AIXwindows environment as its GUI. AIXwindows is based on OSF/MOTIF and consists of AIXwindows toolkit, window manager, and desktop. AIXwindows also contains Graphics Library (GL) which is a high function graphical interface library and enhanced X-Windows Display Postscript. The Display Postscript Interpreter provides a single image model for both display and printer data streams (independent of the window system).

Digital's DECstation series uses the DECwindows environment as its GUI. DECwindows is based on X, OSF/MOTIF, and Adobe's PostScript page description language. DECwindows includes industry standard graphics libraries such as PHIGS and GKS. The DECwindows architecture provides a common user interface between ULTRIX and VMS [12].

5.3. VT100/VT200 and 3270 Emulation at Fermilab

Even though UNIX systems are increasing in numbers at Fermilab, a significant amount of computing is still done on VAX/VMS machines. It is thus critical to be able to run interactive applications on VMS systems from UNIX hosts. The problem, however, is not of connectivity but instead how does one map VT100/VT200 keys on a Sun, SGI, IBM RS/6000 and Data General keyboard?

On a Sun SPARCstation to emulate VT100/VT200 keys one can execute the command, **xterm -tn vt100**. This command, however, does not map all VT100/VT200 keys. **crttool-2.0** is a public domain software that provides complete VT100/VT200 emulation for Sun SPARCstations [13]. On the Silicon Graphics, IBM RS/6000 and the Data General AViiON system the .Xdefaults file in the user's HOME directory has to include key translation information in order to provide correct VT100/VT200 emulation [14], [15], [16].

Also, **tn3270** is a public domain software that allows users to emulate an IBM 3270 keyboard when accessing an IBM or Amdahl systems from a SUN SPARCstation. [17] describes how to install the product on a Sun SPARCstation.

6. Conclusion

This paper describes how UNIX systems and their associated technology have been integrated in Fermilab's computing environment. The associated technology includes:

- Configuring UNIX systems with third-party peripheral devices
- Developing distributed applications based on Berkeley sockets and remote procedure calls
- Evaluating software development tools especially FORTRAN compilers on UNIX systems
- Investigating VT100/VT200 and 3270 emulation issues

Various projects were initiated at Fermilab to better understand integration issues associated with UNIX systems. Some of these projects included: UNIX Seed, FORTRAN compiler evaluation, Mail Server, TCP/IP & Distributed Applications and Terminal Emulation.

7. References

[1] Pabrai, U.: *Network Security Design for UNIX Systems in a Distributed Environment*, IIT Master of Science Thesis, pp. 1, August 1990.

[2] Tanenbaum, A.: *Computer Network*, Second Edition, New Jersey: Prentice Hall, pp. 455-461, 1988.

[3] Lidinsky, W.P.: *Data Communication Needs*, IEEE Network Magazine, Vol. 4, No. 2, pp. 28-33, March 1990.

[4] Pabrai, U., Dowat, C. & Weber, J.: *Understanding and Using Computer Networks*, Fermilab Computing Division GG0009, pp. 6.18-6.19, June 1990.

[5] Pabrai, U. et. al.: *Mail Server Project*, Fermilab Computing Division, February 1991.

[6] Stevens, W.R.: *UNIX Network Programming*, New Jersey: Prentice Hall, pp. 261-262, 1990.

[7] Corbin, J. and Silveri, C.: *Open Network Programming*, UNIX World Magazine, pp. 115-128, December 1989.

[8] Sharp, B.: *The Right Connection*, HP Professional Magazine, Vol. 4, No. 3, pp. 36-42, March 1990.

[9] King, P.: *Team Computing*, HP Professional Magazine, Vol. 4, No. 3, pp. 28-34, March 1990.

[10] Abarghoui, F., Pabrai, U. & Weber, J.: *Port Numbers and TCP(UDP)/IP Socket-based Networking Applications*, Fermilab Computing Division TUN008, February 1991.

[11] Abarghoui, F., Pabrai, U. et. al.: *FORTRAN Compiler Evaluation of Fermilab Mandatory Requirements*, Fermilab Computing Division TN0057, January 1991.

[12] McGregor, S.A.: *An Overview of the DECwindows Architecture*, Digital Technical Journal, Vol. 2 No. 3, pp. 9-15, Summer 1990.

[13] Shah, H.: *VT100/VT200 Emulation on a Sun SPARCstation*, Fermilab Computing Division TSUN011, December 1990.

[14] Shah, H.: *VT100/VT200 Emulation on a SGI Workstation*, Fermilab Computing Division TSGI004, December 1990.

[15] Shah, H.: *VT100/VT200 Emulation on an IBM Workstation*, Fermilab Computing Division TIBM001, February 1991.

[16] Shah, H.: *VT100/VT200 Emulation on a Data General Workstation*, Fermilab Computing Division TSUN011, December 1990.

[17] Hathaway, J. and Shah, H.: *IBM 3270 Emulation on a Sun SPARCstation*, Fermilab Computing Division TSUN016, February 1991.