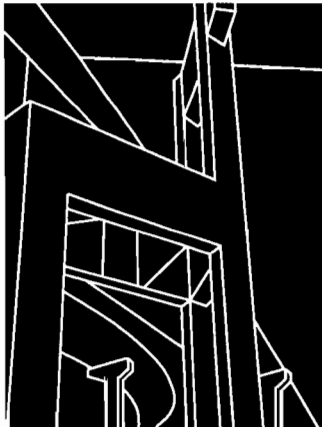RESEARCH REPORT 2912-S

EVALUATION, DESIGN, AND IMPLEMENTATION OF AN AUTOMATED PROCESS-CONTROL AND DATA-ACQUISITION SYSTEM FOR THE TEXAS MOBILE LOAD SIMULATOR (TXMLS)

Dennis S. Collier

CENTER FOR TRANSPORTATION RESEARCH
BUREAU OF ENGINEERING RESEARCH
THE UNIVERSITY OF TEXAS AT AUSTIN

DECEMBER 1996

1. Report No. TX-98/2912-S	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle EVALUATION, DESIGN, AND IMPLEMENTATION OF AN AUTOMATED PROCESS-CONTROL AND DATA-ACQUISITION SYSTEM FOR THE TEXAS MOBILE LOAD SIMULATOR (TXMLS)		5. Report Date December 1996
		6. Performing Organization Code
7. Author(s) Dennis S. Collier		8. Performing Organization Report No. 2912-S
9. Performing Organization Name and Address Center for Transportation Research The University of Texas at Austin 3208 Red River, Suite 200 Austin, TX 78705-2650		10. Work Unit No. (TRAIS)
		11. Contract or Grant No. Project 7-2912
12. Sponsoring Agency Name and Address Texas Department of Transportation Research and Technology Transfer Section/Construction Division P.O. Box 5080 Austin, TX 78763-5080		13. Type of Report and Period Covered Project Summary Report (9/95 — 8/96)
		14. Sponsoring Agency Code
15. Supplementary Notes Project conducted in cooperation with the Texas Department of Transportation.		

16. Abstract

The project consisted of using the following research methodology: evaluating computer hardware and software procured for the purpose of creating a process-control and data-acquisition system for the Texas Mobile Load Simulator; determining any additional hardware and software needed for the system, if necessary; hiring and supervising an independent software engineer whose job was to integrate the various system components; debugging the system on an on-going basis; and recommending changes to the original system design, even to the extent of suggesting new areas of related research.

The two major findings of the project are these: (1) cellular technology is too slow and too expensive to use as a transfer method for large quantities of data, such as that generated by the TxMLS; and (2) SCO UNIX is too complex and too expensive an operating system to use in any part of the computer system.

17. Key Words TxMLS, UNIX, Linux, process control, data acquisition		18. Distribution Statement No restrictions. This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.	
19. Security Classif. (of report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of pages 44	22. Price

Form DOT F 1700.7 (8-72)   Reproduction of completed page authorized

EVALUATION, DESIGN, AND IMPLEMENTATION OF AN AUTOMATED PROCESS-CONTROL AND DATA-ACQUISITION SYSTEM FOR THE TEXAS MOBILE LOAD SIMULATOR (TxMLS)

by

Dennis S. Collier

Research Report Number 2912-S

Research Project 7-2912

Automated Process-Control and Data-Acquisition for TxMLS

Conducted for the

TEXAS DEPARTMENT OF TRANSPORTATION

by the

CENTER FOR TRANSPORTATION RESEARCH
Bureau of Engineering Research
THE UNIVERSITY OF TEXAS AT AUSTIN

December 1996

IMPLEMENTATION RECOMMENDATIONS

The author of this report recommends that the following be implemented:

(1) The process-control and data-acquisition system developed jointly by the Texas Department of Transportation (TxDOT) and the Center for Transportation Research (CTR) should be used, as it represents a viable system for transferring pavement-wear data between the system's on-site and Austin, Texas-based Headquarters components. This system provides an automated system not previously available to TxDOT for streamlining movement of data between remote sites where conventional telephone communication is not likely to be available. The automated system will make the Texas Mobile Load Simulator (TxMLS) a more useful tool for pavement analysis.

(2) Further research is needed for examining alternative methods of transferring data between the two components cited above. Cellular modem communication provides a means of delivering the necessary data transfer. Unfortunately, it must be remembered that cellular communication is too expensive and too slow to be considered as a long-term solution for real-time monitoring of the system from the Austin site. Its use as a transfer mechanism is being implemented solely to meet system parameters in the original hardware/software specifications, a goal called for by TxDOT's Pavement Design Section. Therefore, it is suggested that microwave technology be considered as a more efficient communications method, both in terms of time and money spent. Consider the fact that the average cost to the State of Texas for an hour of cellular modem use is $16.20 and that the average data collected and transferred for TxMLS per day is 100 MB. The projected cost for transfer due to cellular modem speed is $1,080. Add to this the fact that sustained cellular modem speed makes it unlikely that one day's data can be transferred in a 24-hour period, and a problem is evident. Microwave beam-casting may prove to be a reasonable alternative for moving the large quantities of data quickly and economically.

DISCLAIMERS

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the Federal Highway Administration or the Texas Department of Transportation (TxDOT). This report does not constitute a standard, specification, or regulation.

There was no invention or discovery conceived or first actually reduced to practice in the course of or under this contract, including any art, method, process, machine,

manufacture, design or composition of matter, or any new and useful improvement thereof, or any variety of plant, which is or may be patentable under the patent laws of the United States of America or any foreign country.

NOT INTENDED FOR CONSTRUCTION,
BIDDING, OR PERMIT PURPOSES

ACKNOWLEDGMENTS

TABLE OF CONTENTS

SUMMARY

The project consisted of using the following research methodology: evaluating computer hardware and software procured for the purpose of creating a process-control and data-acquisition system for the Texas Mobile Load Simulator; determining any additional hardware and software needed for the system, if necessary; hiring and supervising an independent software engineer whose job was to integrate the various system components; debugging the system on an on-going basis; and recommending changes to the original system design, even to the extent of suggesting new areas of related research.

The two major findings of the project are these: (1) cellular technology is too slow and too expensive to use as a transfer method for large quantities of data, such as that generated by the TxMLS; and (2) SCO UNIX is too complex and too expensive an operating system to use in any part of the computer system.

CHAPTER 1: INTRODUCTION

The Center for Transportation Research (CTR) has developed and fabricated the Texas Mobile Load Simulator—hereafter referred to as the TxMLS—as part of Research Project 7–1924. This engineering project was accomplished in cooperation with the Texas Department of Transportation (TxDOT) and Victoria Machine Works. The TxMLS is a one-of-a-kind device capable of simulating the load of up to approximately 50,000 truck axles per day on actual in-service or specially constructed roads anywhere in Texas.

As the device is used, it has been and will be necessary to acquire, decipher, and manipulate vast amounts of data related to pavement deterioration and environmental conditions. The Center for Transportation Research has been in a unique position to work in tandem with TxDOT personnel to design, set up, and evaluate a computer-based process-control and data-acquisition system for the TxMLS.

PROJECT SCOPE

Research Project 7–2912 was designed to meet two primary objectives. The first was to evaluate the merits of the automation system plan set forth by the independent software consulting firm Unicom Systems Development, Inc., and modify the scheme as necessary. This system was to be capable of receiving, transmitting, and processing the pavement data generated by the TxMLS. The integrated system consists of subsystems located in two separate sites: a process-control trailer that accompanies the TxMLS at a particular test site and a headquarters location in Austin, Texas.

The second objective was to implement the best data-acquisition and process-control system from that plan within the framework of the approximately $320,000 in computer and computer-related equipment and software previously procured by TxDOT's Pavement Design Section (DES) for this purpose. One necessary

requirement for accomplishing this task was to hire and supervise an independent software engineer whose goal was to integrate two related, but different, forms of the UNIX operating system, as well as to assist CTR and TxDOT in implementing the designed system. Testing of each facet of the system was imperative to ensure proper performance.

It must remembered throughout this report that the computer industry, together with its accompanying technology, is a very volatile and rapidly changing area. As a result, any automation plan cannot remain static, as too often it will become obsolete before it is implemented—or, certainly, at some time during its implementation phase. More will be mentioned about this problem and how it affected the research effort in a forthcoming chapter.

RESEARCH APPROACH

To achieve the objectives of the study, the research effort was divided into four tasks, with flexibility and open-mindedness toward completing these objectives as necessary subgoals within each task. The first goal was to evaluate hardware and software procured as a result of a plan developed prior to the beginning of Project 7–2912. TxDOT DES deemed it imperative that, wherever possible, the original equipment and software be used before any of it was abandoned. Working in tandem with DES personnel (particularly the original project director, and later with the software engineer), CTR constructed and tested the system called for by that plan. Equipment was tested for compatibility and marked as either suitable for the system or unsuitable and requiring replacement.

The second task was to determine whether any additional hardware and/or software was necessary for the system so as to complement the plan. Purchases were made on an as-needed basis.

A third task involved locating, hiring, and supervising the proper software engineering organization for integrating the subsystems together. This was a very important task, as the ability for users to analyze the data received from the TxMLS

site could not be accomplished without it. There are two versions of UNIX being employed by the various components of the system, and they must cooperate with each other so that data can be transferred from Intel-based computers to Sun SparcStations. Locating a qualified engineer proved to be both a time-consuming and a laborious undertaking. No one involved with the original plan ever imagined that the particular expertise required to tie the subsystems together would be so difficult to locate. Fortunately, the engineer who was hired proved to be extremely competent.

The last task involved the constant effort of debugging the system at each stage of its development, as well as improving throughput for the entire system wherever possible. Only then could problems be easily isolated and eliminated.

Two other tasks had been envisioned for this project. One involved installing an ISDN (Integrated Services Digital Network) line at the TxMLS site, with the intention of increasing data communication between the remote TxMLS location and a server located in Austin. The second involved creating an Internet link. Eventually both of these tasks were abandoned when support from TxDOT's automation division (Information Systems Division, or ISD) was not received. ISD expressed concern about costs and maintenance regarding the ISDN line; security issues caused the Internet issue to be foregone.

CHAPTER 2: BACKGROUND

In 1992, personnel from TxDOT DES approached Unicom Systems Development, Inc., and requested that the firm design a fully integrated system capable of monitoring the TxMLS on-site. A private company was necessary, as no suitable expertise for such a system was resident within TxDOT. The system was to accept data from an acquisition system also located at the site. It was then to transmit that data from the TxMLS site pavement test sections back to a subsystem located in Austin, Texas, and be able in turn to transfer that data to Sun Microsystems workstations. The data would then be analyzed and models developed to predict pavement wear and failure.

ORIGINAL PLAN

Unicom produced two related documents (dated October 1992 and January 1993, respectively) that outlined how the on-site subsystem and the second, Austin-based subsystem would be constructed and what components and software would be necessary to form them. The reports seemed to be complete in their coverage, and therefore TxDOT approved funds in the neighborhood of $320,000, and DES was ordered to proceed.

The plan called for two computers, labeled "Manager" and "Supervisor," to be placed in a vehicle that would accompany the TxMLS at each test site. Each of these computers would be based upon Intel 80486 technology so that it could run MSDOS, Microsoft Windows®, and Santa Clara Operation (SCO) UNIX operating systems (OS). The first two OS's were necessary so that software incapable of running under UNIX could be used, such as that required for a weather station connected to the system. UNIX was chosen because it represented the only multiuser, multitasking OS at that time, and it was deemed desirable to have more than one user be able to use the system simultaneously and more than one software package run at any time. SCO was chosen specifically because it was in use by more than 90 percent of all Intel-based

5

microcomputers running some form of UNIX. Each computer was to be rack-mounted and employ optical disk drives for data storage.

A second subsystem called the "Headquarters" computer was to be similarly configured and located in Austin. The subsystems would communicate via cellular modems capable of moving data at speeds of up to 19,200 baud. The headquarters system would in turn be integrated into a network with Sun workstations running another form of UNIX called Solaris.

Unfortunately, for unknown reasons Unicom decided to disassociate itself from the project before any implementation of the design actually occurred. However, DES desperately wanted the plan to succeed as a necessary part of the overall TxMLS operation.

Therefore, CTR was approached in the summer of 1993 to continue where Unicom had stopped. Chapter 3 details CTR's involvement with the plan.

CHAPTER 3: PLAN EVALUATION AND IMPLEMENTATION

A BRIEF HISTORY

TxDOT DES first approached CTR in the summer of 1993 to implement the plan first set forth by Unicom Systems, with Project 7–2912 beginning that September. Unfortunately, it would be April of 1994 before approval would be granted by TxDOT's administration to proceed.

Although approval was not granted immediately as expected, CTR decided that any delay would prove costly from an automation perspective, as computer equipment has a very short lifespan before being overtaken by obsolescence. It was felt that eventually the project would be approved. Therefore, CTR teamed with then project director Mr. George Alexander and proceeded to inventory and evaluate the hardware and software that had been delivered by September of 1993.

FIRST SHORTCOMING

Unicom had suggested using SCO UNIX as the primary OS for both the manager/supervisor system and the Headquarters counterpart. Although it may be true that this form of UNIX is installed in over 90 percent of Intel-processor computers nationwide that use UNIX, its popularity comes at a price. First of all, it is an expensive OS solution. At the time of its purchase, a 512-user license was the sole way to obtain it, irrespective of the actual user need. As a result, TxDOT was forced to spend thousands of dollars for licenses that it did not need.

Another problem arose almost immediately. The OS was installed multiple times on the manager system, for example, only to fail for sundry reasons. Each installation was time consuming and frustrating.

The initial cost of purchasing SCO represented only a small fraction of its actual cost. It is overly complex to the extent that locating someone with SCO expertise proved to be very difficult. A nationwide search for a reliable software

7

engineer versed in SCO took 6 months and came at a cost of $50 per hour for his services. Approximately $20,00 was spent to pay for his help.

No one within TxDOT had any experience with SCO, thereby making internal support impossible. Training provided by private suppliers was expensive, not only because the classes were costly, but also because training could be obtained only by traveling to Dallas or Houston.

Technical support for the product was also expensive. It required a renewable, yearly payment to SCO to continue to receive any telephone support and no local support was available.

OTHER ISSUES

Fortunately, most of the hardware and software originally purchased proved reliable. Testing of the equipment proceeded with little difficulty. Very few items had to be returned or replaced.

Unfortunately, the project was delayed a second time when Mr. Alexander was stricken with a heart problem. The search for the software engineer was underway by that time, but little else could be accomplished while he convalesced. Eventually Mr. Alexander was replaced as project director by Dr. Dar-Hao Chen.

Only after the software engineer was hired would new items be ordered, as he attempted to improve the overall system design. This was expected, as he had the experience needed to evaluate any shortcomings in the plan as it was originally specified. For example, the motherboard for the Headquarters server system, with its Intel 80486 processor, was deemed too slow to process the incoming data from the remote site and simultaneously act as the network server for the Austin system. It was replaced with a faster Pentium processor board.

The software engineer also worked to provide integrated services between the two versions of UNIX. This was by far the most difficult process in the entire plan. The two versions are sufficiently different to cause problems when interaction between them is necessary. However, the engineer was successful in creating the

communications protocols between them. One typical technical issue involved one of the Sun workstations, where X-Windows was adapted to facilitate administration standards. Motif 1.2.2 Developers Toolkit, a graphical user interface, was installed over Sun Solaris on the same Sun workstation. Motif window manager was configured as a Window session manager to provide compatibility between Sun and SCO window management. A reliable backup system was implemented as well.

Appendix A represents a system administrator's guide provided by the software engineer. Similarly, Appendix B is a simple users' guide. However, these guides would become somewhat unnecessary with the advent of Linux, as discussed in the section entitled "Last Concern."

SECOND SHORTCOMING

The software engineer also provided connectivity between the remote computer subsystem and the Headquarters system by establishing cellular modem links between them. However, it became apparent immediately that although the modems' links were capable of transferring data between the two subsystems, the transfer rate provided by cellular technology was inadequate for anything beyond small data sets. This was true even after connecting identical modems running at twice the baud rate of the original set—28,000 versus 14,400 baud.

LAST CONCERN

Although the software engineer was successful in integrating the two subsystems and providing necessary network services to ensure that data would be moved about among users properly, both CTR and TxMLS personnel agreed that the SCO issue would continue to pose a problem. An immediate problem occurred when the software engineer, his services strongly in demand, decided to leave the project for a job in California. For this reason, together with the problems cited above, it was decided that the SCO would be replaced with Linux, a freeware implementation of UNIX. An automation expert within DES was very familiar with this version and was able to replace the SCO components with their Linux counterparts. This is one

example of the willingness to be flexible mentioned in Chapter 1. Project team members came to the realization that although a great deal of money had been spent on SCO, the continuing costs and problems with it did not justify its use.

CHAPTER 4: LESSONS LEARNED

It is important to discuss what has been learned from this research project. In this way, similar problems may be avoided in the future. The most efficient way to accomplish this is to enumerate these lessons and then discuss each in some detail.

(1)   Avoid hiring a small software consulting agency, unless it has a proven history of success with state government and/or educational institutions. Many private companies are not acquainted with the constraints placed on state agencies regarding purchasing and the bureaucratic procedures necessary to receive approval for expenditures or changes to original contractual agreements. Others firms do not realize that although they will be compensated for services rendered, this remuneration is often not immediate. Some firms may become frustrated with delays that cannot be helped but are perceived within the private sector as avoidable.

A small firm cannot easily replace individuals within it should the need arise. Unicom Systems Development, Inc., the consulting firm responsible for specifying the hardware and software originally procured for the process-control and data-acquisition system, became disenchanted with the constraints placed on it by the system. As a result, it disassociated itself from the project *before* the system components were actually acquired but *after* the procurement procedure had begun. TxDOT was forced to scramble to find a replacement for the firm without adequate knowledge of how to proceed. Another small company from whom much of the original hardware was procured vowed never to deal with state agencies again because of delays in receiving compensation.

(2)   Beware of purchasing software, especially operating system software, if there is no experience with it *within TxDOT*. It is extremely important that internal support be available for any intricate computer system and its components. At the time of its purchase it was true that SCO UNIX represented the most

11

frequently used version of UNIX in Intel-processor microcomputers. That was its chief selling point. However, no one within TxDOT had any experience with it. The intricacies of this particular version of UNIX make it a costly alternative, not only in terms of dollars spent to acquire and maintain it but also in terms of training and gaining expertise with it. Several months and thousands of dollars were spent to find and employ a software engineer who was properly trained to integrate SCO into the overall system.

(3)   Avoid using cellular modems as a means of communicating between the field system components and the Headquarters counterpart. At the time of this writing (December 1996), cellular communication and its accompanying modems are too slow to move large data sets between the systems, and cellular represents a very expensive method of accomplishing such movement. Use of cellular technology was specified in the original Unicom Systems Development plan, but it should be used *only* if no other alternative is feasible. Real-time monitoring of the TxMLS cannot occur using cellular modems, as processor demands from Microsoft Windows® software slow down remote commands via modem to a virtual standstill. Any attempt to control the TxMLS from a remote location would be frustrating to Headquarters users, and, more importantly, dangerous to personnel located at the TxMLS site.

CHAPTER 5: RECOMMENDATIONS AND CONCLUSIONS

Project 7–2912 has proved to be a successful effort, both from the perspective of the Center for Transportation Research and from that of TxDOT. The spirit of cooperation between the agencies was very evident throughout the research project, in keeping with the nature of the Cooperative Research Program.

Overall, the automation design originally created by Unicom Systems Development, Inc., proved to be a viable system. However, there were a few noteworthy exceptions. These were touched upon in the last chapter, particularly the use of cellular technology for transferring data from remote sites and employing SCO UNIX as an operating system for the on-site subsystem and for the Headquarters subsystem. Therefore, the following recommendations are made:

(1) Replace SCO UNIX with the freeware version of UNIX called Linux. This has already been accomplished with the mutual consent of TxMLS personnel and CTR. This version of UNIX is a no-cost solution to TxDOT, provides the necessary commands for the subsystems, and is known to TxMLS-assigned automation personnel. This makes it an attractive alternative to SCO UNIX. Another possibility is using either Microsoft Windows® NT or Sun Solaris exclusively throughout the system. The former is being implemented throughout TxDOT as part of the automation retooling plan and therefore is guaranteed support from the Information Systems Division. (ISD). The latter provides a one-version-of-UNIX solution throughout the process-control and data-acquisition system, and TxMLS personnel are familiar with it, as it is used on the Sun workstations.

(2) Consider replacing the cellular modems with a faster and—it is hoped—cheaper method of moving data from a remote TxMLS site to the Headquarters location. One possible means is microwave broadcasting. Further research may be necessary to determine whether this is in fact a feasible solution.

14

(3) Continue to investigate advances in computing in general. This area is perhaps the most volatile technology today, and it is almost guaranteed that better methods for throughput, data transfer, and system integration will be forthcoming.

(4) Maintain a spirit of flexibility. Within reasonable budget and time constraints, any facet of an automation plan must be allowed to change with the times.

(5) Attempt to keep the automation plan as simple as possible. Troubleshooting, training, implementation, and maintenance costs and problems will be minimized by reducing the complexity of the automation design.

These departures from the original plan should ensure future success with the system.

APPENDIX A:
SYSTEM ADMINISTRATOR'S GUIDE FOR SCO AND SUN SOLARIS

TEXAS DEPARTMENT OF TRANSPORTATION
DESIGN DIVISION
PAVEMENTS SECTION

MOBILE LOAD SIMULATOR INFORMATION SYSTEM

The MLS Information System has been established to provide support for network operations, to run engineering applications, and to accumulate and distribute data at remote sites during the data-acquisition phase.

Therefore, the MLS Information System network topology is made up primarily of two subnetworks. One subnet is in the RV, and the other is at the Design Pavement Section in Austin.

NETWORKING

General Network Overview

The MLS Information System is based on SUN Solaris Sparc LX workstations running SUN Solaris 2.3 and Intel-based machines running SCO UNIX Open Desktop/Open Server networked on Ethernet using the TCP/IP protocol.

The TCP/IP-based network is essentially made up of three C-Class subnets to provide inter-networking between the RV and Headquarters subnets adapting PPP (point-to-point) protocol over cellular communication phone lines.

To enable future demand for connections to an existing Novell Netware-based TxDOT network, an Olicom Token Ring NIC (Network Interface Card) and IPX/SPX protocol used on mlshqsrv. *drouter* and *nlogin* are two major tools to revise IPX/SPX nodes on networks and to connect them.

On SCO UNIX servers, *netconfig* can be used to revise and alter the network configuration.

The following are TCP/IP subnet addresses and general network information:

```
Headquarters:        201.100.100.32
RV:                  201.100.100.96
PPP Link:            201.100.100.64
```

```
Netmask:            255.255.255.224
Broadcast Address:  201.100.100.255
Domain Name:        txmls.txdot.GOV
```

Each individual node name and TCP/IP address associated with it can be found on /etc/hosts files as follows:

```
#  @(#)hosts      1.2 Lachman System V STREAMS TCP source
#  SCCS IDENTIFICATION
127.0.0.1    localhost
201.100.100.42   hqsrv          hqsrv.txmls.txdot.GOV
201.100.100.74   hqsrvpp        hqsrvpp.txmls.txdot.GOV
201.100.100.42   mlshqsrv       mlshqsrv.txmls.txdot.GOV
201.100.100.106  rvmgr          rvmgr.txmls.txdot.GOV
201.100.100.84   rvmgrp1        rvmgrp1.txmls.txdot.GOV
201.100.100.106  mlsrvmgr       mlsrvmgr.txmls.txdot.GOV
201.100.100.107  mlsrvspv       mlsrvspv.txmls.txdot.GOV
201.100.100.43   srvsta1        srvsta1.txmls.txdot.GOV
201.100.100.44   darhao         darhao.txmls.txdot.GOV
201.100.100.52   qms230_tcp     qms230_tcp.txmls.txdot.GOV
```

Taking future Internet access service requirements into account, DNS (Domain Name Server) has been configured.

The Headquarters SCO UNIX server (*mlshqsrv*) functions as the DNS master server, *darhao* as the DNS secondary server, and the Master SparcStation (*srvsta1*) as DNS cache-only server. The **nslookup** utility can be used to query node names and associated information on the network.

The DNS server configuration can be read at /etc/named boot file for each server and workstation as follows in mlshqsrv:

```
directory    /usr/lib/named/txmls.named.data
domain       txmls.txdot.GOV
```

```
primary          txmls.txdot.GOV txmls-hosts
cache            root.cache
primary          100.100.201.in-addr.arpa          txmls-
hosts.s.rev
primary          0.0.127.in-addr.arpa     named.local
```

The /usr/lib/named/txmls.named.data directory contains related configuration files for that specific node.

mlshqsrv at Austin and *mlsrvmgr* at RV have been configured as the Gateways to announce themselves to the rest of their respective subnets as routers to other subnets. Each workstation and server also provides router service to ease finding the route from source to destination on any node, no matter from where and what type request has been originated. Related files for this configuration are /etc/gateways and /etc/networks on mlshqsrv and mlsrvmgr, and /etc/gateways file on each node.

The *netstat -rn* command lets the user verify the routing table on each node.

POINT-TO-POINT PROTOCOL (PPP) OVERVIEW

The Headquarters server (mlshqsrv) and RV manager (mlsrvmgr) provide PPP (point-to-point) connections to each other with modems over a cellular link. PPP will provide a maximum 19,200 baud speed in the best circumstances. However, modems will connect to each other at the best speed available to establish and continue communication. Practically, that speed has been found to be 9,600 baud in most trials, with a corresponding average throughput of 4,800 characters per second (cps).

The following is a list of files associated with PPP configuration on each server:

```
/etc/ppphosts
/etc/pppauth
```

Before PPP, the researchers also had to configure UUCP with related files as follows:

```
/usr/lib/uucp/Devices
/usr/lib/uucp/Dialers
/usr/lib/uucp/Systems
/usr/lib/uucp/Permissions
/etc/default/atdialZyXEL_2    contains    definitions
    for ZyXEL modem.
/etc/default/atdialUSRSportster    contains
    definitions for US Robotics Sportster modem.
/etc/inittab
/etc/gettydefs
```

HEADQUARTERS SITE OVERVIEW

The Headquarters subnet is made up of two SUN SparcStations and one SCO UNIX server. The SCO UNIX Server (*mlshqsrv*) provides a gateway to the RV and can run MSDOS-based applications under UNIX X-Windows/Motif window manager. This server is also loaded with the complete SCO UNIX Development System (C compilers and more).

SUN Solaris workstations (*srvsta1*, *darhao*) have been configured to run OSF/Motif window manager and installed with engineering applications.

MLSHQSRV (Headquarters Server)

This machine has the following technical specifications and capacities:

- Dual processor motherboard with one Intel Pentium 90 MHz processor.
- Six EISA and four PCI slots.
- 32 MB RAM memory.
- One SCSI 420 MB HD drive.

- PCI-based Adaptec Wide SCSI and Fast SCSI control card.
- PCI-based Imagine 128 graphics card with 4MB SRAM.
- Wantek Tape Unit.
- EISA-based Digiboard concentrator card with 16 RS-232C ports.
- Olicom 16-bit Token Ring NIC.
- Western Digital Ethernet card.

Use *hwconfig -h* to get the following hardware configuration on UNIX:

device	address	vec	dma	comment
fpu	–	13	–	type=80387
serial	0x3f8-0x3ff	4	–	unit=0 type=Standard nports=1
serial	0x2f8-0x2ff	3	–	unit=1 type=Standard nports=1
floppy	0x3f2-0x3f7	6	2	unit=0 type=135ds18
floppy	–	–	–	unit=1 type=96ds15
console	–	–	–	unit=vga type=0 12 screens=68k
tape	0x338-0x33c	5	1	type=wangtek
eisarom	–	–	–	eisa (1.3.0)
smc/wdn	0x240-0x25f	10	–	type=8013WC addr=00:00:c0:ce:13:80
adapter	0xf400-0xf4ff	9	–	type=alad ha=0 id=7 fts=sto
epca	0x1005-0x1005	–	–	mem=0x000D8000 ports=16 EISA C/X V5.8.0
disk	–	–	–	type=S ha=0 id=0 lun=0 ht=alad fts=sdb
Sdsk	–	–	–	cyls=402 hds=64 secs=32
otr	0xa30-0xa3f	15	3	Olicom 16/4 ISA Adapter 0 (at 16 Mbps)

Use the *custom* utility to see loaded software:

Products Currently Installed

SCO Open Server Enterprise System

SCO Open Desktop Development System

Accelerated-X(tm) Advanced X-Window Display Server

```
Adaptec 7800 Family Driver for SCO 3.2v4.2. [Version d1.7]
MenuPort Interface (MPI) Package
SCO Open Systems Software Release 3.0 Release Supplement
Olicom otr Token-Ring STREAMS LLI driver
PA Driver Enhancements
```

Use *fdisk* and *divvy* to see HD Disk Partition tables and file systems for a specific HD as follows:

```
# fdisk -f /dev/rdsk/0s0
# divvy /dev/rdsk/0s0
# fdisk -f /dev/rdsk/0s0
1. Display Partition Table
2. Use Entire Disk for UNIX
3. Use Rest of Disk for UNIX
4. Create UNIX Partition
5. Activate Partition
6. Delete Partition
Enter the userr choice or q to quit: 1
Current Hard Disk Drive: /dev/rdsk/0s0
```

Partition	Status	Type	Start	End	Size
1	Active	UNIX	9664	25663	16000
3	Inactive	DOS(ext)	3200	9599	6400
4	Inactive	DOS	1	3199	3199

```
Total disk size: 25728 tracks (65 reserved for masterboot
and diagnostics)
Press <Return> to continue
```

To change each individual port configuration on Digiboard Concentrator unit, use the *mpi* (Menu Port Interface) utility. Naming conventions adapted to address

each individual port name are /dev/ttyA?? for modem ports and /dev/ttya?? for terminal ports.

To change graphics options, use the */usr/bin/X11/Xsetup* and */usr/bin/X11/Xaccel* utilities that come with the XAccelerator product.

SRVSTA1 (Server SparcStation)

This machine is a SparcStation LX and runs the Solaris 2.3 UNIX Operating System. "srvsta1" has 72 MB RAM. Its has the greatest disk space capacity with three hard disk units. All Sun Solaris 2.3 Operating System components, including "man" (manual) pages, have been installed, along with a C language compiler, which has one machine license.

Disk partitions for each HD unit and associated file systems on each partition can be configured and viewed on Solaris 2.3 machines with the *format* and *newfs* commands. Hard disk partition naming adapted by Solaris is /dev/rdsk/c0t0d0s0 for raw device name and /dev/dsk/c0t0d0s0 for block device name.

This machine also exports three file systems to the rest of the nodes in the local subnetwork.

The /usr/etc/exportfs command provides a means to export file systems to the network. All exported file systems can be seen in the /etc/xtab file at any particular moment using the *df -f NFS -v* command, which also lists file systems with their allocated disk units and available capacities.

Complete definitions of exported files systems and their respective paths can be found in the /etc/exports file; these file systems are related to the following:

- File system and associated directory, which contains the system manual pages.
- File system and associated directory, which contains the OSF/Motif manual pages.
- /compilers directory, which contains the C language compiler itself and related files and manual pages.

The /compilers/.profile file in this machine contains all necessary environment variables and their respective settings regarding the C compiler, the OSF/Motif window manager, ABAQUS, SAS, and Mathematica applications, which are available on *darhao* as installed.

DARHAO (Engineering SparcStation)

This is another SUN Solaris 2.3 machine with such engineering applications as SAS, ABAQUS, and Mathematica installed. This machine also imports the aforementioned file systems from SRVSTA1.

RV (Victoria) Site Overview

While the Headquarters site in Austin is targeted to use and analyze the data via engineering applications, the RV site is responsible for collecting data from the MLS and DAV data-acquisition subsystems and the weatherstation. Making data available to the Austin site is another responsibility of this site.

Each SCO UNIX system has an MSDOS partition available on its first SCSI disk unit, which contains MSDOS-based EISA configuration and diagnostic utilities and device drivers for add-on cards and peripherals.

The RV site is made up of two SCO UNIX server and data-acquisition/control PC workstations, one for the MLS and another for the DAV. Each node on this site has been connected via an Ethernet-based local area network.

MLSRVMGR (Manager)

The SCO UNIX server named *mlsrvmgr* has the following hardware components:

- Intel 486 50 MHz processor and EISA motherboard.
- 48 MB RAM.
- Two SCSI hard drives, each 420 MB.
- One SCSI optic disk unit.

- One EISA SCSI control card.

- A 1 MB graphics card.

- One Wantek tape controller and drive unit.

- One Digiboard eight serial Intelligent Multi I/O board.

- One Digiboard four serial nonintelligent serial port.

- One Ethernet card.

mlsrvmgr uses PPP to connect to *mlshqsrv* (Austin) via one serial port on the Intelligent Multi I/O board over a cellular phone. This server also directs connection to the weatherstation via one port on the nonintelligent digiboard I/O card. SCO UNIX is capable of dedicating intelligent I/O ports to DOS sessions running under SCO OpenDesktop. In this way, the researchers are able to run the weatherstation program as a DOS session and collect and monitor data.

The "/etc/weather" executable script file contains necessary parameters and command lines for this session.

Another thing that is worth mentioning is that the optic disk drive on this node has been configured as a DOS drive that is also accessible to UNIX. Therefore, an alternative means of porting data with a maximum 320 MB capacity has been provided.

In OpenDesktop, MSDOS partitions are available with the following naming conventions:

C:\ always refers to UNIX / (root) file system.

D:\ always refers to UNIX user's home directory.

E:\ always refers to first bootable MSDOS partition, if available.

Any other DOS partition can be associated with any arbitrary drive name with the following command:

```
dos +af:=/dev/dsk/devicename devicename
```

in the format of /dev/dsk/?s?, where ? stands for the SCSI id and partition number.

26

MLSRVSPV (Supervisor)

This SCO UNIX node has the following hardware components:

- Intel 486 50 MHz processor and EISA motherboard.
- 32 MB RAM.
- Two SCSI hard drives, one with 420 MB, the other with 2 GB.
- One EISA SCSI control card.
- A 1 MB graphics card.
- One Ethernet card.

This machine has the greatest disk space capacity, and for this reason therefore some vacant file systems are available to be exported to the rest of the subnetwork.

The file system "/usr/dosuser/export2dos" has been exported to the MLS and DAV workstations. This file system has been configured as an MSDOS file system and contains two subdirectories, "mls" and "dave," to be imported by Microsoft Windows®-based PC workstations, one for the MLS, the other for DAV.

These workstations run ONNET 2.0, which is a TCP/IP-based PC/NFS networking software that enables file system sharing and distribution of network services. With this application each PC node can see the imported file systems as one of its local drives, such as D: or E:.

The MLS workstation runs Windows-based data acquisition/monitoring/control applications residing on its local D:\mls directory. This is the mlsrvspv:/usr/dosuser/export2dos/mls file system that makes it possible to collect and access the data simultaneously from the UNIX side as well.

The DAV workstation imports the "/usr/dosuser/export2dos/dave" file system/directory as one of its local drives in a similar way and for the same reason as expressed in the previous paragraph.

mlsrvspv also accommodates a parallel/serial printer through one of its serial ports. Printer configuration and printer names can be seen using the ***sysadmsh*** utility.

This utility provides basic and essential system configuration and administration functions for system administrators.

28

APPENDIX B:
A SIMPLE USER'S GUIDE FOR UNIX

SOME FUNDAMENTAL COMMANDS

TO BOOT THE SYSTEM

- Since all the SCO servers are configured with at least a 50-MB bootable MSDOS partition, the users may want to go strictly to MS-DOS/Windows and skip UNIX. To do this, all the users have to do is type *dos* when the users see the UNIX command as follows:

  - *boot*: dos <enter> — if the users simply hit <enter>, the users trigger UNIX booting.

  - *boot*: <enter>

- On Solaris 2.3 workstations the users should type <boot> to start UNIX booting.

TO SHUT DOWN THE SYSTEM

- On SCO UNIX, type shutdown -y -g0 <Enter>

- On Solaris 2.3 type, shutdown -y -g0 -i0 <Enter>

TO LOG IN TO A SPECIFIC NODE OR TO COPY FILES BETWEEN NODES

- *rlogin*

  This command enables any user on any TCP/IP node to log in to a specific server with the following command line options:

  On Solaris 2.3:

  $ rlogin [-l *username*] *hostname* <enter>

  On SCO UNIX:

  $ rlogin *hostname* [-l *username*] <enter>

  Here *hostname* could either be a real hostname, such as *darhao,* or an IP address, such as 201.100.100.43.

32

- *rcp*

  This command enables any user on any TCP/IP node to copy files between nodes with the following command-line syntax:

  ```
  rcp [ -p ] filename1 filename2 ; or
  rcp [ -rp ] filename1 filename2
  ```

  Here the *-p* option allows a user to retain file ownership and access right properties while copying files between nodes, whereas the *-rp* option provides recursive copy for subdirectories. *Filename* has the following naming convention: ***username@hostname:filename***
  For instance,

  ```
  rcp darhao@darhao:/tmp/sample.txt   root@srvsta1:/usr/can
  ```

- *ftp*

  This command provides another means of transferring files between nodes using the TCP/IP file transfer protocol.

  ```
  ftp <enter> and
  ftp> ? <enter> lists all available options.
  ```

- *rsh* or *rcmd*

  These commands are for executing commands and/or scripts on other nodes without explicitly performing a log in into them. These commands stand for **Remote Shell /Command execution**. Generally, these commands have the following syntax, and **rsh** is available on Solaris, whereas **rcmd** is available on SCO UNIX.

  ```
  rsh [ -l login ] [ - n ] hostname "command itself" <enter>
  ```

- *Copying files from/to diskette and tape drives*

  On SCO UNIX servers the users can use either the *doscp DRV:\filename1 filename2* or *doscp filename1 DRV:\filename2* commands to transfer files from/to floppies. Here, *DRV:* represents a symbolic drive name such as *A:* or *B:*.

  The *DOSCP* command can also provide line-feed character elimination via a command line option to strip the '^M' character to support compatible data transfer between the MSDOS and UNIX formats.

  Another tool that provides character stripping for DOS versus UNIX file format compatibility is the *dos2UNIX* command that resides either on the SCO UNIX or on the Solaris 2.3 platform. More information regarding this tool can be obtained through the *man* utility, using a command such as *man dos2UNIX <enter>.*

  Commands on SCO UNIX such as *doscat* and *dosdir* provide compatible file operations for DOS file formats on UNIX sessions. Here *doscat* corresponds to the *cat* command in UNIX and the *type* command in DOS.

  The *TAR* command in both SCO UNIX and Solaris 2.3 system facilitates compatible archiving on UNIX systems. It can be used in the following circumstances:

  - *to create tape archives*

    ```
    tar -cvf  /dev/tape [filename] <enter>
    ```

  - *to extract files from tape media*

    ```
    tar -xvf /dev/tape [ filename ] <enter>
    ```

  - *to copy files recursively from directories*

    ```
    cd fromdir; tar cf - . | (cd  todir; tar xfBp - ) <enter>
    ```

  - *to copy files from nodes on Ethernet*

    ```
    tar cvfb - 20 filenames | rsh hostname dd of=/dev/tape
    obs=20 <enter> or
    ```

```
cd fromdir; tar cf - . | rsh hostname ( cd todir; tar xvf -)
<enter>
```