ESD-TR-92-051 MTR-11282

(2)

Ethernet Bandwidth Utilization of Datafull, Dataless, and Diskless Node Workstations and X-Terminals

By

J. E. Manley
S. P. Trieber

May 1992

Prepared for

Director, Intelligence Data Handling
System Program Office
Electronic Systems Division
Air Force Systems Command
United States Air Force

Project No. 558B

Prepared by

The MITRE Corporation
Bedford, Massachusetts

Contract No. F19628-89-C-0001

Approved for public release;
distribution unlimited.

REVIEW AND APPROVAL

This technical report has been reviewed and is approved for publication.

DEBRA J. NICELY, MAJ, USAF
Director, IDHS Architecture Division

FOR THE COMMANDER

MIKE PROWSE, LTC, USAF
Director, Intelligence Data Handling
System Program Office

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
May 1992

3. REPORT TYPE AND DATES COVERED
Final

4. TITLE AND SUBTITLE
Ethernet Bandwidth Utilization of Datafull, Dataless, and Diskless Node Workstations and X-Terminals

5. FUNDING NUMBERS
F19628-89-C-0001
558B

6. AUTHOR(S)
Manley, James E.
Trieber, Steven P.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
The MITRE Corporation
202 Burlington Road
Bedford MA, 01730

8. PERFORMING ORGANIZATION REPORT NUMBER
MTR-11282

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Director, Intelligence Data Handling System Program Office
Electronic Systems Division, AFSC (ESD/IC)
Hanscom AFB, MA 01731-5000

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
ESD-TR-92-051

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
As users of "open system technologies" migrate towards use of the Network File System (NFS), the X-Window System, and workstations configured as datafull, dataless, and diskless nodes, as well as X-Terminals, many questions arise as to network capacity planning. This report examines the Ethernet bandwidth utilization of three applications, all X-Windows based, in four different workstation/terminal configurations. Information is provided to facilitate replication of the data collection techniques for individual sites and network/application configurations. Several conclusions and recommendations are drawn which may be of interest to system designers considering network capacity planning.

14. SUBJECT TERMS
Capacity Planning
Network Performance
NFS
Performance Measurement
X Protocol

15. NUMBER OF PAGES
119

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR

EXECUTIVE SUMMARY

As users of "open system technologies" migrate towards use of the Network File System (NFS), the X-Window System, and workstations configured as datafull, dataless, and diskless nodes, as well as X-Terminals, many questions arise as to network capacity planning. This document examines each of the technologies mentioned above, as well as paging and swapping, all of which have a direct bearing on network bandwidth utilization. In addition, this report provides information which can be used by a system designer/planner to replicate the tests performed in our study, using his or her specific workstation configurations and applications.

Based on the results obtained in this study, several conclusions were drawn. These included:

- That application execution traffic rather than application startup traffic be used as the basis of network bandwidth capacity planning;
- That rough estimates of bandwidth consumption, and maximum number of supportable workstations can be estimated, for specific workstation configurations and applications;
- That the X-Protocol is extremely bandwidth intensive and that wholesale use of exported X displays should be carefully considered prior to adoption in any system architecture.

In addition, several recommendations were made which included:

- That the use of diskless node workstations be avoided;
- That use of NFS File Servers be encouraged;
- That use of dataless nodes offer a slight advantage over use of datafull nodes;
- That use of X-Terminals should be used cautiously;
- That network traffic behavior be used as additional application selection criteria;
- That the basic techniques described in this report be built upon and refined to produce more useful and accurate predictive results.

ACKNOWLEDGMENTS

This document has been prepared by The MITRE Coporation under Project Number 558B, Contract Number F19628-89-C-0001. The contract is sponsored by the Electronic Systems Division, Air Force Systems Command, United States Air Force, Hanscom Air Force Base, Massachusetts 01731-5000.

We would like to extend our thanks to Frank Robey in assisting us with the analysis of our base data, to the staff and management of G113 for use of their Sniffer, to Rajiv Kohli and Mark Workman for their review of this document, to the secretarial staff of G115 for their assistance in producing this document, and to Denise and Leslie for their understanding during the creation of this document.

TABLE OF CONTENTS

LIST OF FIGURES

SECTION 1

INTRODUCTION

The Intelligence Data Handling Systems(IDHS) community, Department of Defense Intelligence Information Systems (DODIIS) community, and the commercial-off-the-shelf (COTS) vendor community are all currently in the midst of migrating to "open system architectures." These architectures are based in large part on COTS workstation, Local Area Network (LAN), UNIX Operating System, and X-Window User Interface technologies.

Movement to "open system architectures" is not without its growing pains. One problem area facing system designers is that of workstation/network capacity planning;that is, determining how many user workstations can be effectively supported on a particular subLAN in a responsive, yet cost-effective manner.

1.1 PURPOSE AND OBJECTIVES OF THIS DOCUMENT

The purpose of this document is to share with the community the results of a series of experiments designed to capture the network loading effects of using datafull, dataless, and diskless workstations, as well as "X-Terminals." These results we believe will be useful in supporting capacity planning functions. We have detailed our methodology so that our "experiments" can be replicated in the field with actual IDHS and DODIIS applications.

The problem of capacity planning given the current suite of workstation technologies available to system planners and designers is shown in figure 1-1, "A Notional IDHS LAN Architecture." In this diagram, subLAN_1 depicts the architecture of a Network File System (NFS) File Server and a suite of client workstations it supports. NFS provides a network transparent file system capability which allows files shared by multiple users (client workstations) to be centrally located. NFS however is not without its costs, in that it generates network traffic while providing its services. The specific capacity planning problem facing designers is to determine the optimum number, "n," where "n" is the total number of workstations capable of being serviced on the subLAN without inundating the subLAN with NFS traffic. The problem is further complicated by shrinking Department of Defense (DOD) budgets, which do not permit system designers to simply segment their LAN into many subLANs, and replicate the NFS Servers on each subLAN. Similar capacity planning issues are faced on subLAN_2, which is X-Window Application Server based, and subLAN_3, which is a hybrid environment.

The goal of this study and paper was to begin the process of understanding the specific effects of these workstation technologies, in order to support the capacity planning needs of the community, as well as address current operational needs of the community.

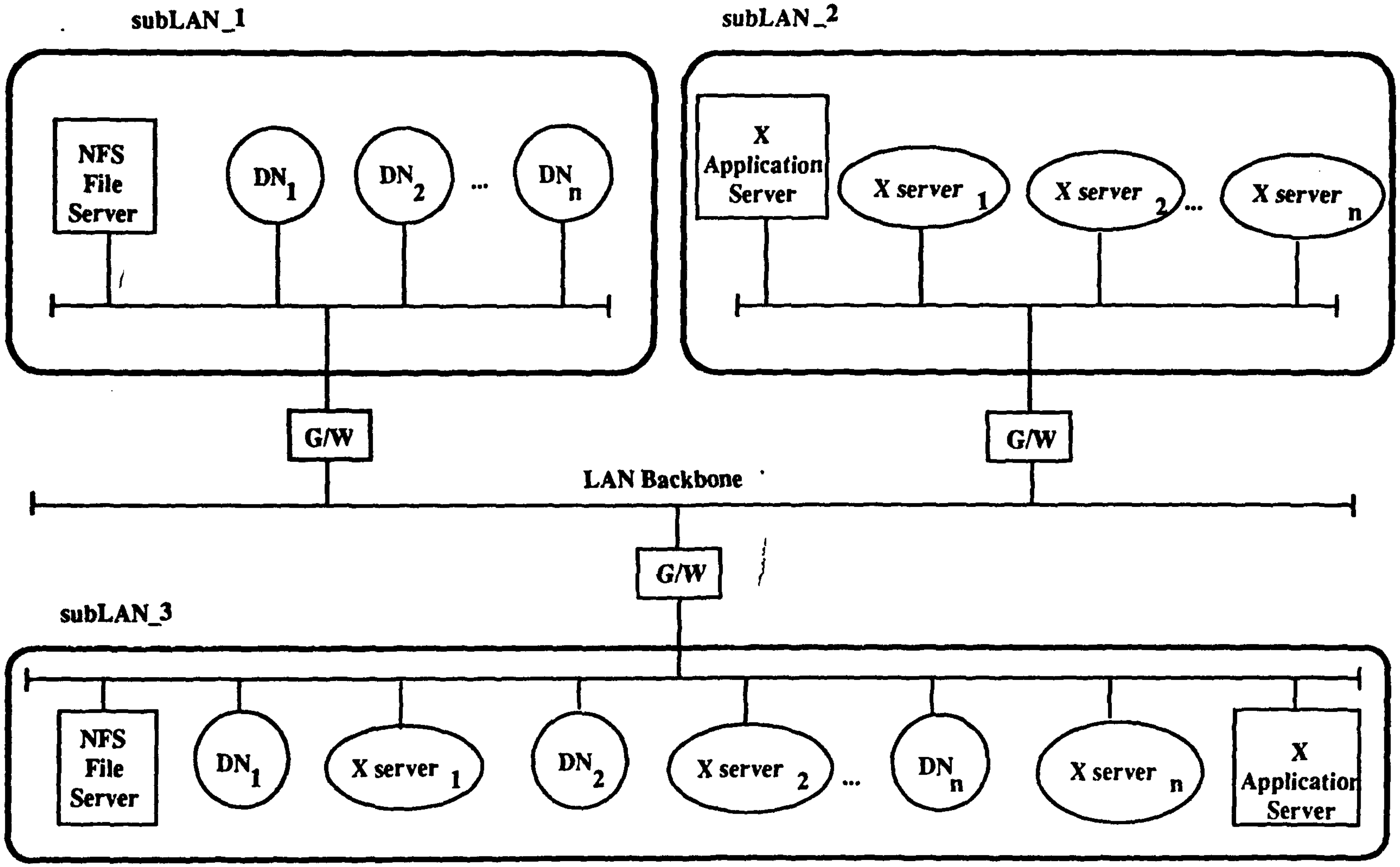

Figure 1-1. A Notional IDHS LAN Architecture

1.2 ORGANIZATION OF THIS DOCUMENT

Section 1, Introduction, provides a high-level description of the workstation/network capacity planning problems the IDHS and DODIIS communities are currently facing and states the goals this document hopes to achieve.

Section 2, Base Technology Description, discusses the base technologies involved with effort. These include NFS, X-Windows, datafull, dataless, and diskless workstations, and X-Terminals. Discussion of "paging and swapping" and how these two activities affect network performance are also included.

Section 3, Experimental Method, describes the specific testbed constructed for this effort and the methods used to simulate IDHS environments and capture the resulting network data.

Section 4, Data Analysis, presents the pertinent results of our experiments and describes our analysis techniques, and resulting observations.

Section 5, Conclusions, presents several conclusions resulting from our analysis, and illustrates how the data collected can be applied to the capacity planning problem.

Section 6, Recommendations, makes several specific recommendations to the IDHS community, based on the results of our efforts, and describes how this effort can serve as the basis for more refined modeling, and analysis techniques.

Appendix A describes the specific sequence of operations used to generate and collect data for each application tested, in each of the configurations.

Appendix B contains descriptions of the testbed configurations, and includes both key files and file system sizes.

Appendix C contains listings of the scripts and programs used to process and generate the data contained in appendix D.

Appendix D contains postprocessed data collected and generated by the Network General Sniffer LAN analyzer and vmstat, and nfsstat.

SECTION 2

BASE TECHNOLOGY DESCRIPTION

This study dealt with a number of related technologies, including the X-Window System, the Network File System(NFS), Datafull Node Workstations, Dataless Node Workstations, Diskless Node Workstations, and X-Terminals. In addition, an examination of operating system paging and swapping was required in order to understand the nature of the network traffic generated. This section provides a brief explanation of each of the technologies described above.

2.1 THE X-WINDOW SYSTEM

The X-Window System, developed at the Massachusetts Institute of Technology (MIT), is an industry standard for device-independent, graphical user interfaces for applications in a heterogeneous environment. It is a network-based, event-driven windowing system that conforms to the client-server model.

The client is the actual application which is displayed on the X-server such as the window manager or a map graphics package. Client applications make requests to the X-server such as getting window attributes or displaying text or graphics in a window. The X-server communicates with the clients by sending events to the client applications, such as an expose event if the application is exposed from being covered by another window. The X-server is responsible for all input and output devices including the display, mouse, and keyboard and it manipulates windows created by clients through the use of text, fonts, and graphics.

The interaction of requests and events can occur with the application executing locally and being displayed and manipulated locally as shown by System A in figure 2-1, "The X-Window System." The true benefit of the X-Window System is its ability to extend to the network model where the interaction between the client application and X-server occurs via the network. For example, the client application may execute on System A, but its display is exported to System B, allowing a user to access and manipulate the application as if it were executing locally.

The X-Windows protocol consists of the Xt Intrinsics, the X-Library, and the X-Toolkit which contains the widget set. As shown in figure 2-2, the client application is displayed on a remote X-Server by being sent on the network and traversing the X-Protocol stack to the X-Server on the remote host where the device drivers and independent device architecture takes over.

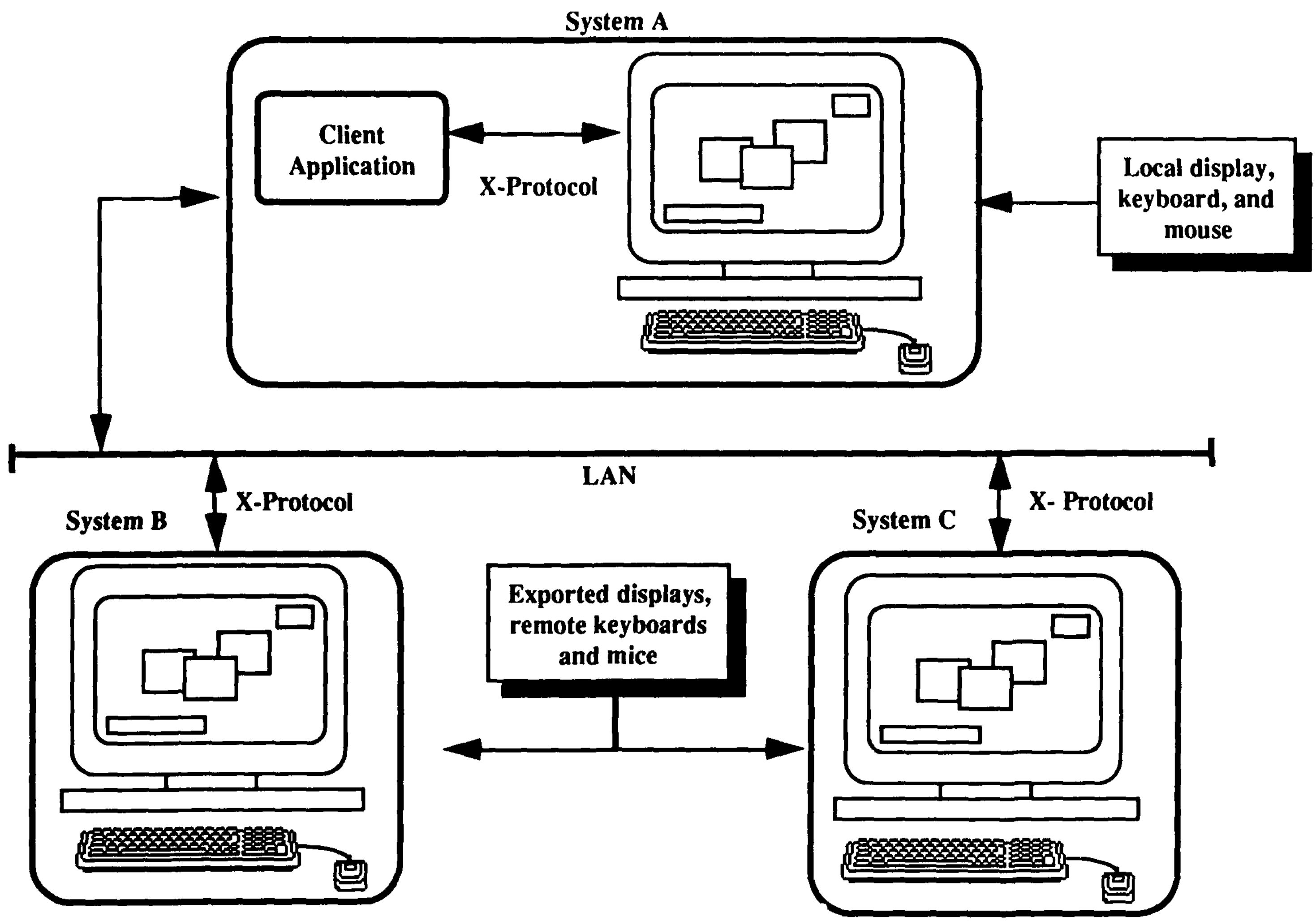

Figure 2-1. The X-Window System

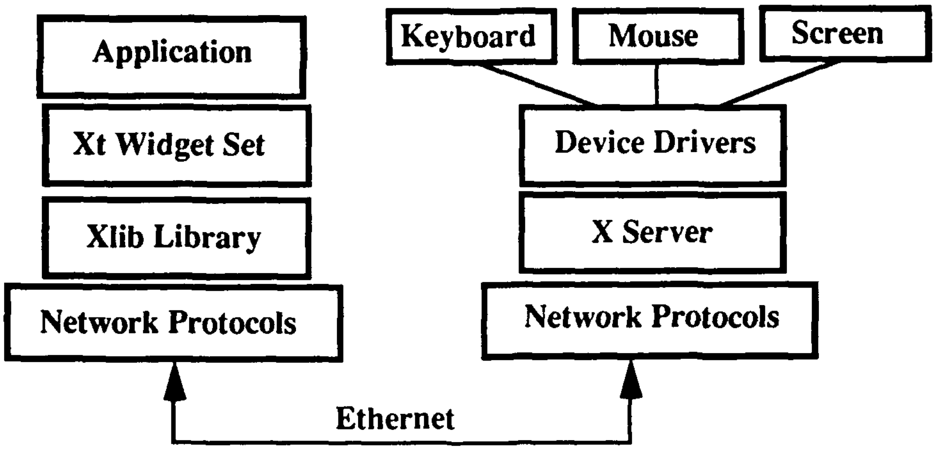

Figure 2-2. The X-Windows Protocol Stack

2.2 THE NETWORK FILE SYSTEM

The Network File System (NFS) is a network-based, stateless file system protocol that allows client workstations transparent, remote access to shared file systems. The main benefit of this type of configuration is that it allows applications to be centrally located on a file server and shared by many different client workstations. The configuration consists of an NFS file server which exports file systems or directories in the file system directory hierarchy to one or more clients using the /etc/exports file. The clients then mount the exported file systems to local mount points within the local directory structure. All manipulation to or from the remote file system occurs through a set of requests and replies handled by the NFS protocol. A client's file access request is translated to an NFS request which is sent over the network to the file server. The file server receives the request, processes the file system operation, and sends a response back to the client over the network. Each type of request is shown in table 2-1 with its hexadecimal equivalent which is stored in the network packet header.

Table 2-1. NFS Request Identification Numbers

Type of Request	Hexadecimal ID
NULL	0
Get Attributes	1
Set Attributes	2
Root	3
Lookup	4
Read Link	5
Read	6

Table 2-1. NFS Request Identification Numbers (Concluded)

Type of Request	Hexadecimal ID
Write Cache	7
Write	8
Create	9
Remove	A
Rename	B
Link	C
Symbolic Link	D
Make Director	E
Remove Directory	F
Read Director	10
File System Status	11
N Procedure	12

The NFS protocol is platform independent; i.e., the protocol is not based on any one particular operating system, machine, network, or transport protocol. NFS transactions occur through Remote Procedure Calls (RPC) built on top of the eXternal Data Representation (XDR). The RPC allows for a procedure-oriented interface to remote program services. NFS is one of the programs which is accessed by providing a host address, program number, version number, and procedure number. The XDR standard provides a common machine independent means of representing a set of data types over a network. In our experiments the 8192 byte blocked data is passed on the network within the User Datagram Protocol (UDP) layered above the 802.3 TCP/IP packet header.

Figure 2-3, "The Network File System," shows an example setup of a network using NFS. It consists of three systems; an application file server (system A), a database file server (system C), and a workstation NFS client (system B). When a client workstation makes an NFS request, a network packet is sent from the client to the file server and a reply is received by the client from the file server. A typical NFS request and each of the steps involved is shown in figure 2-4, including the network protocol packet headers. Note that only the file handle and the RPC procedure call number had to be sent from the client.

In frame 1 the client sends a request to the file server asking for the file attributes of stated 32 byte file handle, as shown. In response to the request from the client, the file server processes the request, gathers the file system information and sends a reply back to the client. The reply contains information for the file attributes such as its type, mode, number of links, user ID, group ID, size, block size, number of blocks in the file, the file system ID, the last access time, modifications time, and node change time.

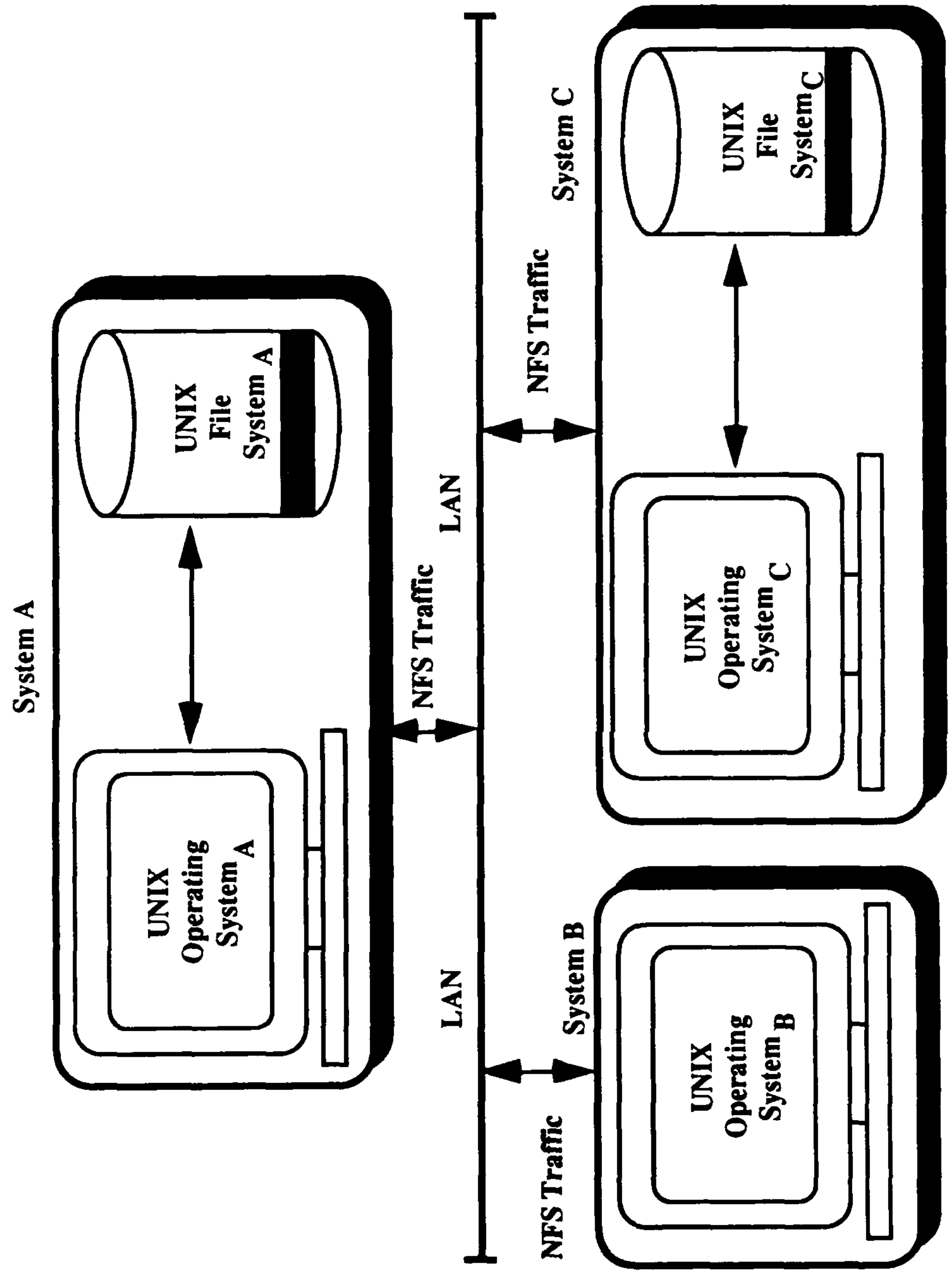

Figure 2-3. The Network File System

```
---------------- Frame 1 ----------------

CLIENT NFS:  ----- SUN NFS -----
CLIENT NFS:
CLIENT NFS:  Proc = 1 (Get file attributes)
CLIENT NFS:  File handle = 0000070600000001000A000000008E23
CLIENT NFS:            61D1B9EC000A00000000000021EE1A2FF
CLIENT NFS:
CLIENT NFS:  [Normal end of "SUN NFS".]

---------------- Frame 2 ----------------

SERVER NFS:  ----- SUN NFS -----
SERVER NFS:
SERVER NFS:  Proc = 1 (Get file attributes)
SERVER NFS:  Status = 0 (OK)
SERVER NFS:  File type = 1 (Regular file)
SERVER NFS:  Mode = 0100755
SERVER NFS:   Type = Regular file
SERVER NFS:   Owner's permissions = rwx
SERVER NFS:   Group's permissions = r-x
SERVER NFS:   Others' permissions = r-x
SERVER NFS:  Link count = 1, UID = 0, GID = 10
SERVER NFS:  File size = 516096, Block size = 8192, No. of blocks = 1024
SERVER NFS:  File system id = 1798, File id = 36387
SERVER NFS:  Access time       = 9-Aug-91 13:58:30.056788 GMT
SERVER NFS:  Modification time = 11-Oct-90 23:09:07.000000 GMT
SERVER NFS:  Inode change time = 20-Jun-91 19:18:20.024464 GMT
SERVER NFS:
SERVER NFS:  [Normal end of "SUN NFS".]
```

Figure 2-4. A get attributes (0x1) Request Between a Client (system B) and a NFS File Server

2.3 PAGING

The virtual memory system is a technique which allows the execution of processes that may not reside completely in memory. The main advantage of this schema is that user programs can be larger than physical memory therefore allowing the degree of multiprogramming to increase because more user programs may execute. It is based on transferring portions of a program image called pages in and out of memory. Local paging is managed by the "paging schedule" and occurs between the binary, which is resident on the disk, and memory, and the swap device, as needed, during program execution. The 4.3 Berkley Software Distribution (BSD) UNIX system pages based on what is known as "Fill-on-Demand Klustering."
When a portion of a program is not found in memory, an internal page fault occurs. A simple scenario of handling a page fault and subsequent page-in is described below and shown in figure 2-5, "Local Swapping and Paging."

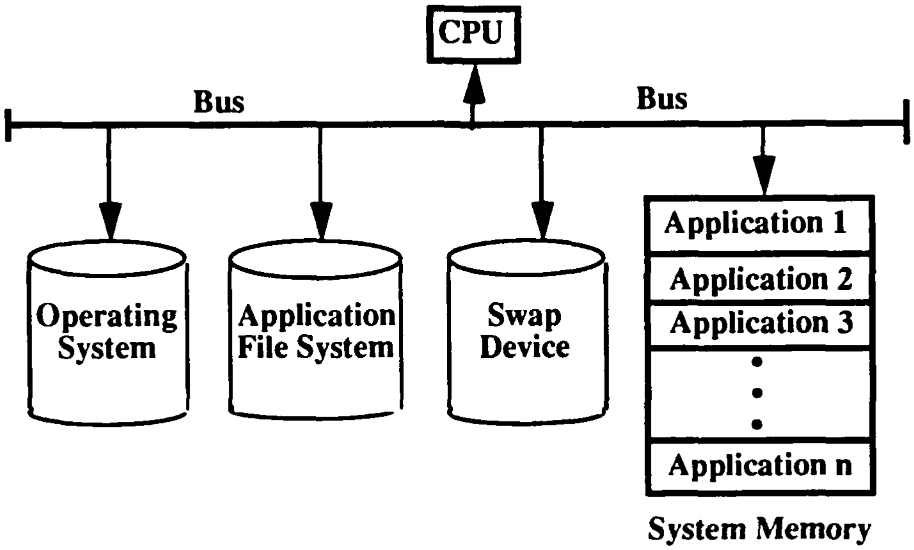

Figure 2-5. Local Swapping and Paging

- A check is made in the CPU of whether the reference was a valid or invalid memory access.
- If it was a valid reference, the scheduler fetches the page from the file system or the swap device.
- The page is written to a free frame in physical system memory.
- The page table is reset with the new reference.
- The instruction is restarted.

When user demand requires more memory than is available, page replacement occurs which involves a page-out and then a page-in. The steps in the page replacement process for a simple scenario is described below:

- The desired page to page in is found on the swap device or on the file system.
- If there exists a free frame use it.
- If not, a victim page is selected to be swapped out to the swap device or freed by the scheduler if the page was not accessed within a specified time interval.
- The page table reference is updated to no longer contain the victim page.
- If the page resided on the swap device it is reclaimed.

- The new page is written to the selected free frame in physical memory.
- The page table is reset with the new reference.
- The instruction is restarted.

The concept of paging can be extended to the network model as shown in figure 2-6, "Swapping and Paging via NFS", for each of the platforms except the X-terminal. Page-ins occur via the NFS network protocol for each of the user applications on each workstation. Page-outs occur locally to the swap device on the datafull and dataless workstations and remotely via the network on the diskless workstation. This type of network traffic increases the network load and increases the amount of network bandwidth utilization.

The 4.3 BSD paging procedure is based on a Least Recently Used (LRU) clock algorithm. Swap space is pre-allocated for the entire address space of a process at load time. When a page fault occurs, a cluster, which is equivalent to one logical page or multiple physical pages, is pre-paged into memory. The system page table is statically allocated. Pages are filled on demand by reclaiming the logical page from the free list or allocating a new page of memory for the fetched page. If free memory exists, *pagein()* allocates the number of page clusters the process requires, otherwise the process sleeps on the global variable *freemem* until more memory is available, then restarts the operation.

2.4 SWAPPING

The swap process completely removes a process from main memory, including the process page tables, the pages of the data and stack segments that are not already in swap space, the user structure, and the text segment. Swapping is similar in nature to paging in that paging reads in and writes out pages and swapping reads in and writes out the entire process.

The 4.3 BSD swapping procedure will swap a process in/out based on three conditions:

- If paging cannot free memory fast enough to satisfy demand,
- If the system page map becomes fragmented,
- If the process is inactive for more than 20 seconds.

A swap-out is based on the length of time the process has been asleep in memory and the size of the process. A swap-in is based on the length of time the process has been swapped out, the size of the process when it was swapped out, its nice value, and the amount of time the process was asleep since it last ran.

The swap-out procedure is as follows (reference figure 2-5);

- Map the user structure of the process into the kernel virtual memory.
- Allocate swap space for the user structure and page tables.

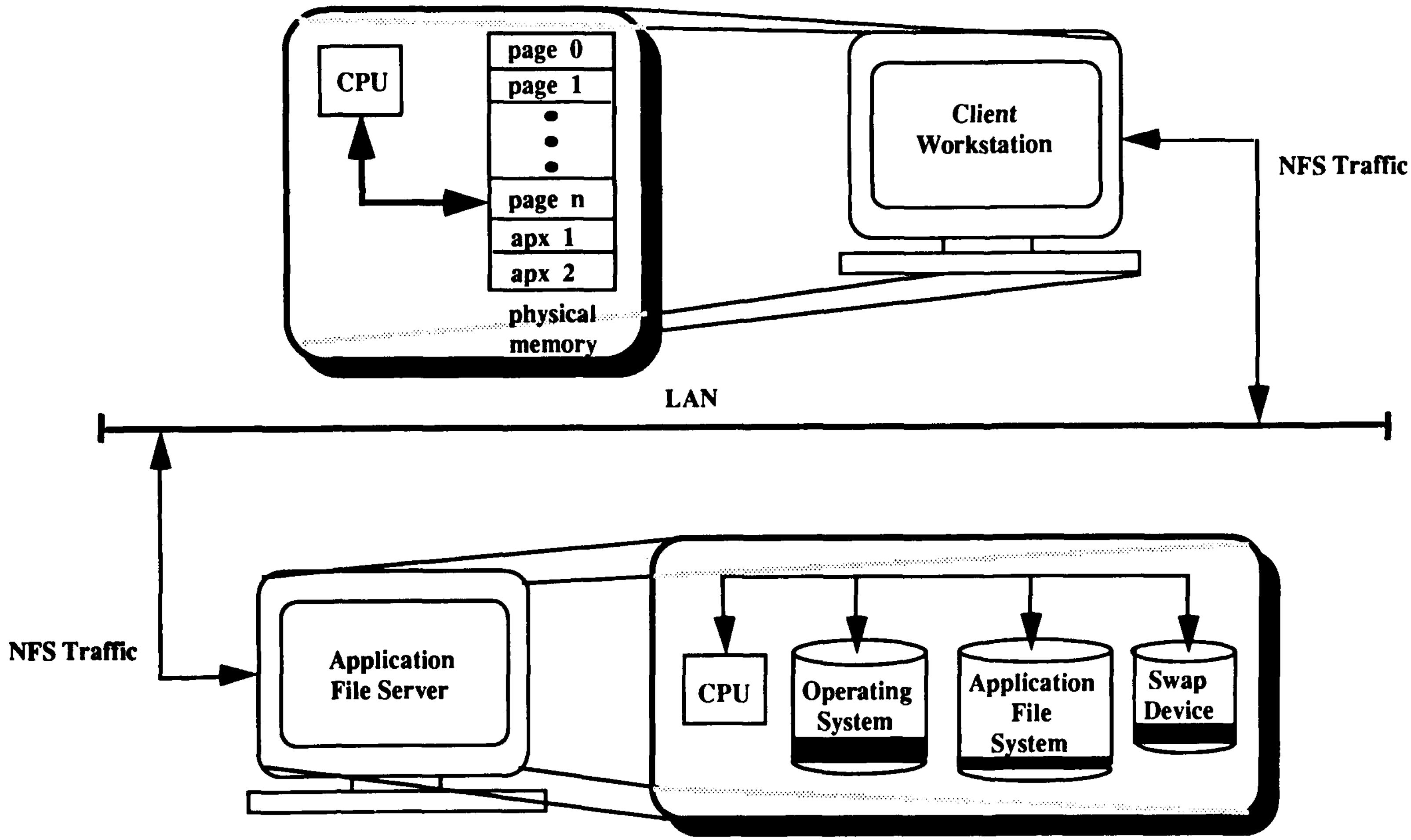

Figure 2-6. Swapping and Paging via NFS

- Release the text portion of the image.
- Forcibly page out all resident pages in the data and stack sections of the process.
- Write out the page tables to the swap space.
- Write out the user structure and kernel stack to the swap space.
- Release the user structure.
- Free the page tables.

The swap-in procedure is as follows (reference figure 2-5):

- Any shared text for the process is locked against change.
- Resources are allocated for the page tables.
- Memory is allocated for the user structure and it is read back from the swap space.
- The page tables are read into memory.
- Swap space for the user structure and page tables is freed.
- The text area is attached with the text page tables being swapped in if necessary for a multiuser application.
- The process is returned to the run queue if it is runnable.

In our experiments, the dataless and datafull workstations took advantage of the local swapping scheme and the X-terminal was not affected by it. We could extend the idea of local swapping to the network model using the diskless workstation. Here the scheduler executes locally but the swap space resides remotely on a NFS file server as shown in figure 2.6. Every swap-in and swap-out must occur via NFS to a dedicated swap file on the server. The network traffic generated by this swapping activity increased the network load, reduced the performance of the network, and slowed the response time of an application and user requests.

SECTION 3

EXPERIMENTAL METHOD

This section discusses the testbed used in the network loading study, the manner in which the network traffic was geneıated using representitive IDHS applications, and the tools and methods used to collect the network traffic previously described.

3.1 TESTBED HARDWARE

Figure 3-1 shows the notional testbed configuration which was used in the experiments. The SPARCstation 1+ contained 16 MB of physical RAM and a 1.2 GB Seagate disk. The diskless, dataless, and datafull workstation was one in the same. It was reconfigured to act as the required type of workstation for each of the tests. The NFS file server and the X Applications server was one in the same and allowed both the X-Windows client applications to reside with the three test applications.

3.1.1 NFS File Server

The NFS file server was configured to use the nonsecure trivial file transfer protocol (tftp) server for booting purposes. The file server contained the kernel and the root partition for the diskless client which resided under the /export partition. It also contained a 50 MB special file which resided under the /export/swap partition for dedicated swap space usage by the diskless client. The NFS file server also contained the test user account which was mounted to each of the clients. A listing of the /etc/fstab and /etc/exports files are shown in appendices B.1 and B.2 respectivly.

3.1.2 Datafull Workstation

The datafull workstation had the partitions, shown in appendix B.3., and is considered to be a fully operational system which does not rely on any file servers for minimal use. The applications are mounted from the NFS file server to the /home1 partition. A listing of the /etc/fstab file and disk capacities are shown in appendices B.3 and B.6 respectively.

3.1.3 Dataless Workstation

The dataless workstation was configured to use one 8 MB partition on the 1.2 GB disk. This partition contained the root (/) partition and the kernel. A 50 MB swap partition was placed on the same disk. This type of configuration allows all swapping between the disk cache and memory to occur locally without network intervention. Paging of applications into memory still occurs via the network. The /usr partition is mounted from the the NFS file server and can be shared among many clients. This increases the load both on the network and on the NFS file server whenever system utilities are accessed. A listing of the /etc/fstab file and disk capacities are shown in the appendices B.4 and B.7 respectively.

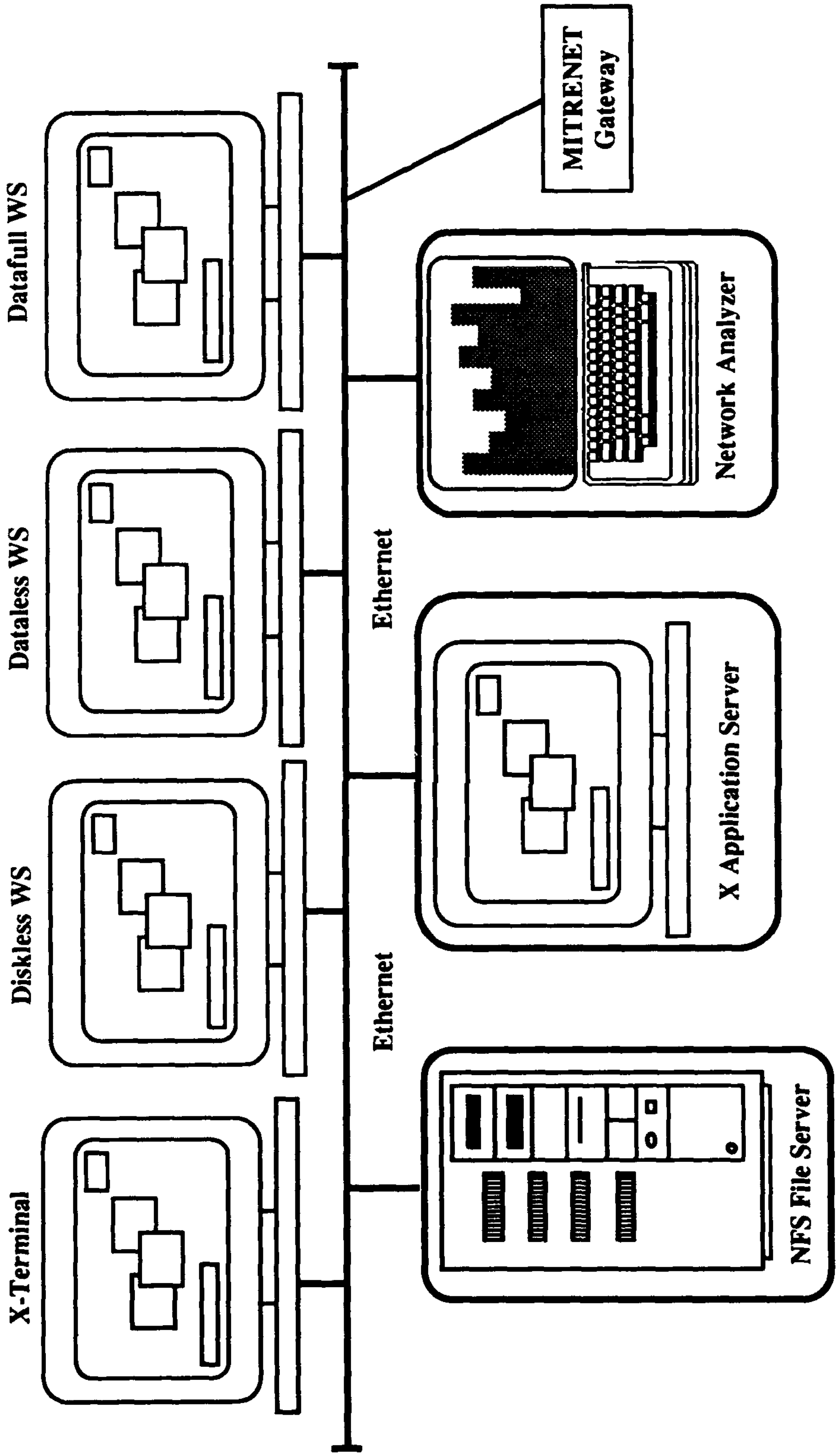

Figure 3-1. Notional Testbed

3.1.4 Diskless Workstation

The diskless workstation used in this analysis maintained all configuration aspects of the baseline configuration excluding the 1.2 GB disk which was detached. All paging and swapping traffic occurred via the network and the NFS protocol. A listing of the /etc/fstab files are shown in the appendix B.5.

3.1.5 X-Terminal

The X-Terminal that was used in the experiment is a Tektronix 4210, which requires other than standard fonts to be downloaded from a font server. The NFS file server contained the special fonts needed for BBN Slate to function properly. The backing store was enabled and RGB downloading occurred at boot time. The X Display Manager was exported to the X-Terminal's display allowing login to the NFS file server. After logging, in applications are started in the normal fashion similar to the other types of workstations.

3.2 TESTBED SOFTWARE

3.2.1 Motif Window Manager

The Motif Window Manager(mwm), from Open Software Foundation (OSF) is a user interface used to manage and manipulate multiple windows being displayed on one display. It is an X-Windows client application that executes like other X clients, but is used for opening, closing, moving, and resizing windows (clients). The Motif window manager was selected to be used in this study for its ease of use and functional capabilities.

3.2.2 Office Automation Package

Bolt, Beranak, and Newman's Slate Office Automation Package was selected for its highly functional capabilities of manipulating imagery, graphics, spreadsheets, audio, and text in a typical office environment. The ability to incorporate each of these functions into one package allows a user to have complete control over the work environment by having each of the main functions integrated into one package.

3.2.3 Imaging Package

Paragon Imaging represents a typical imagery package allowing the user to manipulate an image through the use of Zoom, Region of Interest Operations, (e.g., rotate, xray, scale, texture), Colormap Manipulation, Lut, and Overlay additions including text, circles, rectangles, lines, and polygons.

3.2.4 Mapping Package

Oilstock is a mapping package originally created by NSA and rewritten by MITRE to conform to the OSF Motif standards. This was accomplished by using the UIMX interactive design tool. This allowed the package to remain functionally equivalent to its predecessor but also contain a look and feel that conforms to industry standards. It displays World

Database II vector maps that can be manipulated through the use of the menu bar functions such as File, Criteria, Tracking, Geographics Features, Map Features, Intel Features, Map Detail, Feature Detail, and Projection.

3.3 TESTBED ANALYSIS SOFTWARE

3.3.1 Primary Data Analysis Tool

The Network General Sniffer was the primary data collection and analysis tool, and is an ethernet portable protocol analyzer, which offers protocol analysis, and real time network performance statistics. The Sniffer hardware is a Toshiba 80386 IBM PC clone with a special ethernet card to accept all packets being sent on the backbone. We selected the Ethernet Network PA-302 Sniffer for a number of reasons. It is very portable and lightweight, provides real time analysis capability, analyzes both NFS and X-Window protocol suites (no other analyzer offers this feature), resides as a separate entity on the network (i.e., the Sniffer does not place any load on the network because it is a data gathering tool), and the operation and control of the unit was straightforward and simple. We evaluated several other network protocol analyzers: the HP4900 series including the HP4972A, the LANalyzer, the SpiderAnalyzer P320, and the Vance ATS 1000. The PA-302 was the most flexible and upward compatible for our needs.

The Sniffer generates statistical information in real time which can be reported in many variations. The following is a summary of what the analyzer can accomplish.

Reports/Displays:

- Cable faults.
- Source/destination station addresses in both internet and aliased format.
- Station pair counts of frames and kilobytes sent between nodes.
- Station counts of frames and kilobytes sent from each node.
- Real time bar graph of the number of frames/kilobytes on the Ethernet.
- Percent of network utilization (this is a floating average of the percent of the Ethernet bandwidth being utilized) within a 1-1000 ms window. It is a percentage of total network bandwidth (1,250 Kbytes/sec) from 0 to 100 percent.
- Traffic generator (will send specified data to specific/all nodes).
- Actual data in each layer of the protocol.
- The number of good frames.
- The number of bad CRC (Cyclic Redundancy Check) frames.

- The number of fragmented frames.
- The number of bad alignment frames.
- The number of accepted frames.
- Pattern matching filters.
- Protocol address level filters (DLC, IP, etc.).
- Protocol suites (NFS, ATALK, etc.).
- Absolute time, delta time, relative time, network utilization, bytes, cumulative bytes.
- Hexadecimal view of the data.
- Capture termination when a search string or byte sequence (trigger) has been identified within a data packet.

Other Functions:

- Prints above displays and reports to a serial/parallel printer.
- Saves setups and captured data to a file which can be rerun for later analysis.

Statistical Data:

- Using the above tools the following information can be reported:
  - Percentage of Ethernet bandwidth utilized.
  - Relative time of application execution..
  - Percentage of NFS traffic including a breakdown of the type of traffic with respect to function (e.g., file search, attribute request, file open, acknowledgement).
  - Percentage of X-Window traffic including a breakdown of the type of traffic with respect to graphics, pure data, resource management, and X events.
  - Packets sent versus time graph.
  - Number of packets sent versus the type of packet graph.
  - Frequency calculations on data transmissions.
  - Average file access size.

- Average session duration.

- Mean waiting time from send to receive.

- Differentiate overhead from application data.

3.3.2 Secondary Data Analysis Tools

3.3.2.1 nfsstat

nfsstat is a UNIX-based utility which measures Network File System related statistics. It displays information about the number of NFS calls received (calls) and rejected (badcalls), and the counts and percentages for the various calls that were made. The client NFS display shows the number of calls sent and rejected, as well as the number of times a CLIENT handle was received (nclget), the number of times a call had to sleep while awaiting a handle (nclsleep), as well as a count of the various calls and their respective percentages. The server display shows the number of NFS calls received (calls) and rejected (badcalls).

nfsstat also displays Remote Procedure Call (RPC) statistical information for the server and client. The client display shows the total number of RPC calls received, total number of calls rejected, number of times no RPC packet was available when trying to receive, the number of packets that were too short, and the number of packets that had a malformed header. The server RPC display shows the total number of RPC calls sent, the total number of calls rejected by the server, the number of times a call had to be retransmitted, the number of times a reply did not match the call, the number of times a call timed out, the number of times a call had to wait on a busy CLIENT handle, and the number of times authentication information had to be refreshed.

The following is an example of a *nfsstat* call on a workstation that has remote mounted and exported file system.

```
%nfsstat

Server rpc:
calls    badcalls  nullrecv  badlen    xdrcall
55       0         0         0         0

Server nfs:
calls    badcalls
55       0
null     getattr   setattr   root      lookup    readlink  read
0 0%     44 80%    0 0%      0 0%      4 7%      0 0%      0 0%
wrcache  write     create    remove    rename    link      symlink
0 0%     1 1%      0 0%      0 0%      0 0%      0 0%      0 0%
mkdir    rmdir     readdir   fsstat
0 0%     0 0%      6 10%     0 0%
calls    badcalls  nullrecv  badlen    xdrcall
```

0	0	0	0	0

Client rpc:

calls	badcalls	retrans	badxid	timeout	wait	newcred	timers
335	0	0	0	0	0	0	0

Client nfs:

calls	badcalls	nclget	nclsleep			
335	0	335	0			
null	getattr	setattr	root	lookup	readlink	read
0 0%	128 38%	0 0%	0 0%	56 16%	0 0%	120 35%
wrcache	write	create	remove	rename	link	symlink
0 0%	5 1%	1 0%	0 0%	0 0%	0 0%	0 0%
mkdir	rmdir	readdir	fsstat			
0 0%	0 0%	25 7%	0 0%			

3.3.2.2 vmstat

vmstat is a UNIX-based utility which reports virtual memory statistics from the contents of the sum structure, giving the total number of several kinds of paging-related events which have occurred since boot. An example of the statistical data produced by a datafull workstation by *vmstat* is shown below.

```
datafull% vmstat -s
     130 swap ins
       0 swap outs
     100 pages swapped in
     108 pages swapped out
    4513 total address trans. faults taken
       7 page ins
       0 page outs
      11 pages paged in
       0 pages paged out
       0 sequential process pages freed
     227 total reclaims (0% fast)
     227 reclaims from free list
       0 intransit blocking page faults
       0 zero fill pages created
     247 zero fill page faults
       0 executable fill pages created
       0 executable fill page faults
       0 swap text pages found in free list
     227 inode text pages found in free list
       0 file fill pages created
       0 file fill page faults
       0 pages examined by the clock daemon
       0 revolutions of the clock hand
       0 pages freed by the clock daemon
```

1473 cpu context switches
20052 device interrupts
4751 traps
8133 system calls
6497 total name lookups (cache hits 70% per process) toolong 19

3.4 TEST PROCEDURE

The test procedure and data collection was divided into subparts for each of the four testbed configurations. This provided a more concise comparative analysis of data between platforms. The subprocedures included: logging in, using the X Display Manager, xdm; starting the Motif window manager, mwm; manipulating each package's functionality, and logging out.

The gateway to the MITRENET was disabled to isolate Ethernet segment and allow the Sniffer to capture only those frames related to the test nodes.

The human interaction response time has been considered in the evaluation of time variances in data being sent across the network. Each application's time delays were analyzed and a set of commands for each application was set up to reduce the amount of human interaction with the application. These procedures are described below, and in appendix A.

3.4.1 Office Automation Package

A clear user environment was set up for the BBN Slate office automation package to create default configuration files and documents in the user's file system. Slate was then started with the "Welcome-1.2.slt" document which contains a mixture of Slate capabilities. A detailed description of the the steps taken to generate and collect data for this application is contained in appendix A.1

3.4.2 Mapping Package

The Oilstock mapping package procedure is similar to the office automation package procedure and is described in appendix B.2.

3.4.3 Imaging Package

The Paragon imaging package procedure is similar to the office automation package procedure and is described in appendix B.3

3.5 COLLECTION AND ANALYSIS METHODOLOGY

The Sniffer had limited disk space and memory and as such it was necessary to subdivide the tests into smaller tests. After applying sample runs to the Ethernet and Sniffer, breakoff points were found where the Sniffer capturing could be stopped and the data saved because no data was being sent across the network between client and server. After each of the data segments was collected and saved to the Sniffer hard disk (e.g., TEST1A-TEST1F for Slate), they were trasnsferred to a UNIX platform so the results could be analyzed automatically using Bourne shell scripts as shown in appendix C.

The Sniffer saves its data in a compressed format to minimize disk space requirements. Unfortunately, the data can not be read by any other applications. In order to transfer the data to the UNIX platform, the data had to be printed out, frame by frame, to a file in an ASCII delimited format using the Sniffer print capability. The frames that were selected to be printed contained the following attributes set for display: Delta time, relative time, size in bytes, network utilization within a 100 ms braketing window, destination, source, protocol, and summary information.

The data was then backed up to 3.5-inch floppy disk using the DOS backup command and restored on the UNIX platform through a DOS emulator which had access to the UNIX file system.

On the UNIX platform, the Bourne shell script "go" was run, and all data was analyszed and totalled into summary sheets, figure 3-2.

The data final results were then moved from the UNIX platform to a Macintosh and imported into Microsoft Excel for further analysis.

Frame count: 454

Total Delta Time: 288.899
Mean Delta Time: 0.637746

Total Bytes: 419740
Mean Bytes: 924.537

Total NW Utilization Sniffer MB/Sec (%): 7875.9
Mean NW Utilization Sniffer MB/Sec (%): 17.3478

Mean NW Utilization Actual Bytes/Sec (%): 1452.9

NFS Procedure Totals

Null: 0
Get Attributes: 69
Set Attributes: 0
Root: 0
Lookup: 51
Read Link: 0
Read: 36
Write Cache: 0
Write: 45
Create: 9
Remove: 6
Rename: 0
Link: 0
Symbolic Link: 0
Make Dir: 0
Remove Dir: 0
Read Dir: 3
Stata File Sys: 0
N Proc: 0

UDP Continuation: 228

Figure 3-2. Result Totals After Running Sniffer Data Through Scripts

SECTION 4

DATA ANALYSIS

While the data collection approach of this set of experiments included the use of nfsstat and vmstat as described in section 3, data analysis was focused on the network traffic data collected by the Network General Sniffer. Furthermore, the analysis described in this section was focused on the "D" and "E" portions of each data collection run as described in section 3. Note that the "D" portion of the data collection run pertains to network performance associated with starting an application, while the "E" portion of the data collection run pertains to network performance associated with using an application. Appendix D contains full listings of the results of data collection runs 1A-F, 2A-F, and 3A-F, where 1, 2, and 3 refer to Slate, Oilstock, and Paragon respectively.

4.1 CONSOLIDATED APPLICATION DATA TABLES

Figures 4-1, 4-2, and 4-3 contain consolidated data, collected by the Network General Sniffer, for the Slate, Oilstock, and Paragon applications respectively, in each of the system configurations tested. An explanation of the data fields used in these figures is given below.

The consolidated application data tables contain a variety of useful data for each of the system configurations. As previously described, this data was obtained using the Network General Sniffer.

A frame is a unit of information transferred over the Ethernet. The first item listed in the application data tables is "Frame count." "Frame count" is equal to the total number of frames required to complete the sequence of user operations detailed in section 3 of this document.

"Total Delta Time" represents the summation of the time intervals between each frame transmission. "Mean Delta Time" represents the average time between frame transmissions.

"Total Bytes" is the total amount of information transferred over the Ethernet, expressed in byte units. "Mean Bytes" is the average number of bytes contained in one frame.

"Mean NW Utilization Actual Bytes/Sec" is equal to "Total Bytes"/"Total Delta Time" which represents the average network load generated by the particular application and system configuration, and is expressed in units of bytes/sec. A more meaningful expression for determining network bandwidth utilization is "Mean NW Utilization Actual Mbits/sec" which is equal to ["Mean NW Utilization Actual Mbits/sec" * 8]/1,000,000. This expression is more meaningful as it is expressed in the same units, Mbits/sec, that Ethernet's bandwidth is commonly expressed in. "Percent Utilization of Effective Bandwidth" is equal to "Mean NW Utilization Actual Mbits/Sec"/(5 Mb/s). 5 Mb/s is used as the effective working bandwidth of Ethernet, rather than

	Datafull	Dataless	Diskless	X-Terminal
Test 1D Slate Startup				
Frame Count	2989	3098	3199	2048
Total Delta Time	13.8958	17.1839	15.6195	47.4725
Mean Delta Time	0.00465054	0.00554856	0.00488415	0.0231913
Total Bytes	3186150	3254316	3357014	718402
Mean Bytes	1065.96	1050.46	1049.39	350.782
Mean NW Utilization Actual Bytes/sec.	229289	189382	214925	15133
Mean NW Utilization Actual Mb/s	1.8343	1.5151	1.7194	0.1211
Percent Utilization of Effective Bandwidth	36.69	30.3	34.39	2.422
Test 1E Slate Execution				
Frame Count	1254	1384	2787	13338
Total Delta Time	171.816	148.561	166.847	312.034
Mean Delta Time	0.137124	0.10742	0.0598875	0.0233961
Total Bytes	1132174	1242662	2919520	5251583
Mean Bytes	902.85	897.877	1047.55	393.731
Mean NW Utilization Actual Bytes/sec.	6589	8364	17498	16830
Mean NW Utilization Actual Mb/s	0.0527	0.0669	0.14	0.1346
Percent Utilization of Effective Bandwidth	1.05	1.34	2.8	2.69

Figure 4-1. Consolidated Slate Data

	Datafull	Dataless	Diskless	X-Terminal
Test 2D Oilstock Startup				
Frame Count	1452	1587	1587	1746
Total Delta Time	8.4765	7.8668	7.6545	19.431
Mean Delta Time	0.00584183	0.00496015	0.00482629	0.0111352
Total Bytes	1516692	1587910	1587910	342436
Mean Bytes	1044.55	1000.57	1000.57	196.126
Mean NW Utilization Actual Bytes/sec.	178929	201850	207448	17623
Mean NW Utilization Actual Mb/s	1.4314	1.6148	1.6595	0.1409
Percent Utilization of Effective Bandwidth	28.62	32.3	33.18	2.8
Test 2E Oilstock Execution				
Frame Count	2086	2098	2686	23168
Total Delta Time	95.3875	103.304	100.813	223.082
Mean Delta Time	0.0457494	0.0492626	0.0375469	0.00962929
Total Bytes	2475340	2477000	3193664	12374392
Mean Bytes	1186.64	1180.65	1189	534.116
Mean NW Utilization Actual Bytes/sec.	25950.4	23977.9	31679	55470.2
Mean NW Utilization Actual Mb/s	0.2076	0.1918	0.2534	0.4437
Percent Utilization of Effective Bandwidt:	4.16	3.84	5.06	8.88

Figure 4-2. Consolidated Oilstock Data

	Datafull	Dataless	Diskless	X-Terminal
Test 3D Paragon Startup				
Frame Count	1056	1131	1152	3637
Total Delta Time	7.1817	7.7405	7.4845	37.905
Mean Delta Time	0.0068073	0.00685	0.00650261	0.0104249
Total Bytes	1164984	1222318	1226976	1209134
Mean Bytes	1103.2	1080.74	1065.08	332.454
Mean NW Utilization Actual Bytes/sec.	162216	157912	163936	31899
Mean NW Utilization Actual Mb/s	1.2977	1.2633	1.3115	0.2552
Percent Utilization of Effective Bandwidth	25.95	25.27	26.23	5.1
Test 3E Paragon Execution				
Frame Count	855	957	1249	38986
Total Delta Time	288.884	329.049	374.962	464.601
Mean Delta Time	0.338271	0.344194	0.30045	0.011917
Total Bytes	745862	766830	1088278	6508030
Mean Bytes	872.353	801.285	871.319	166.932
Mean NW Utilization Actual Bytes/sec.	2582	2330	2902	14008
Mean NW Utilization Actual Mb/s	0.0206	0.0186	0.0232	0.1121
Percent Utilization of Effective Bandwidth	0.41	0.37	0.46	2.24

Figure 4-3. Consolidated Paragon Data

the theoretical 10 Mb/s bandwidth. It has been observed that when Ethernet bandwidth utilization exceeds approximately 5 Mb/s, frame collision rates increase to the point where more frames are retransmitted than get through to their destination on the first try.

4.2 FRAME COUNT ANALYSIS

An analysis of frame counts generated by each application in each of its respective system configurations was conducted to determine if the collected data tracked with expected behavior.

The expected behavior of the datafull node configuration was that "Frame count" would be directly proportional to paging activity resulting from application startup and execution. The expected behavior of the dataless node configuration was that its total "Frame count" would be equal to that of the datafull node configuration and any additional paging generated by accesses to the /user file system. Finally, the expected behavior of the diskless node configuration was that its total "Frame count" would be equal to or greater than that of the dataless node configuration, based on swap activity.

The expected behavior witnessed in all the three applications confirmed our assumptions that a datafull node configuration would generate the least network activity, a diskless node configuration would generate the most network activity, and a dataless node configuration generate activity between the two. Figure 4-4 illustrates the percentage relationships between frame counts for each of the application/configuration sets. The data was generated as follows:

[(Configuration2 - Configuration1) / Configuration1] * 100;

where Configuration2:Configuration1 is as indicated in figure 4-4.

		Dataless:Dataful	Diskless:Datafull	Diskless:Dataless
Slate	Test 1D	3.65	7.03	3.26
	Test 1E	10.37	122.25	101.37
Oilstock	Test 2D	9.3	9.3	0
	Test 2E	0	28.76	28.03
Paragon	Test 3D	7.1	9.09	1.86
	Test 3E	11.93	46.08	30.51

NOTE: Each number represents the percentage increase in frames of the first system configuration as compared to the second.

Figure 4-4. Frame Count Analysis Data

From figure 4-4, we again see that the expected behavior was observed for each configuration. In addition, we see that a dramatic increase in frame count occurs when we move from datafull and dataless configurations to a diskless node configuration.

4.3 TOTAL BYTE ANALYSIS

The expected behavior of the "Total Bytes" count was identical to that of Total "Frame count." As with the "Frame count" behavior, "Total Byte" counts for each of the configurations tracked with expected behavior, as can be seen in figures 4-1to 4-3.

4.4 TOTAL DELTA TIME ANALYSIS

In addition to representing the summation of time intervals between each frame transmission, it was empirically determined that a direct relationship exists between "Total Delta Time" and perceived user performance. Thus, we see from figures 4-1, 4-3, that perceived user performance was roughly comparable for datafull, dataless, and diskless workstation configurations. Also, we see from the same figures that the perceived user performance for the X-Terminal tested was three to five times slower than the various workstation configurations. We attribute this to the fact that the X-Terminal used in this study was a "first-generation" device and as such, suffered from performance limitations. Empirically, we have seen more comparable user perceived performance on newer "second-generation" X-Terminals and X Servers. Unfortunately, our schedule did not permit replication of our tests on alternate "X-Terminal devices."

Another behavior pattern observed was that in two of the three applications we tested, datafull workstations had a slightly lower "Total Delta Time" than dataless workstations. We attribute this to the fact that in the datafull configuration, applications do not have to page-in via NFS, rom the /usr file system, located on the NFS File Server.

4.5 BANDWIDTH UTILIZATION ANALYSIS

Analysis of the bandwidth utilization data revealed two major behaviors which will be of particular interest to the IDHS community and "distributed workstation," users, in general.

The first behavior pattern observed was that considerable bandwidth was consumed by datafull, dataless, and diskless workstation configurations when applications are first started. Bandwidth usage ranged from 25-34 percent. Fortunately, these high usage rates were short lived, ranging from 7-17 seconds, depending on the particular application and workstation configuration. Inspection of data from secondary sources, nfsstat and vmstat, confirmed our expectation that these high initial usage rates were due to the paging activity associated with loading an application into physical memory.

The bandwidth consumption rates described above contrasted sharply with those observed for applications executing on the X-Terminal. Application startups consumed bandwidth ranging from only 2.4-5.1 percent. Inspection of the raw Sniffer data and secondary nfsstat and vmstat data again confirmed our expectation that we were witnessing only the "end-result" of the application startup; that is, the user display, and that intense paging activity was in fact occurring on the "X Application Server." These concepts are illustrated in figure 4-5, "Application Startup Sequence."

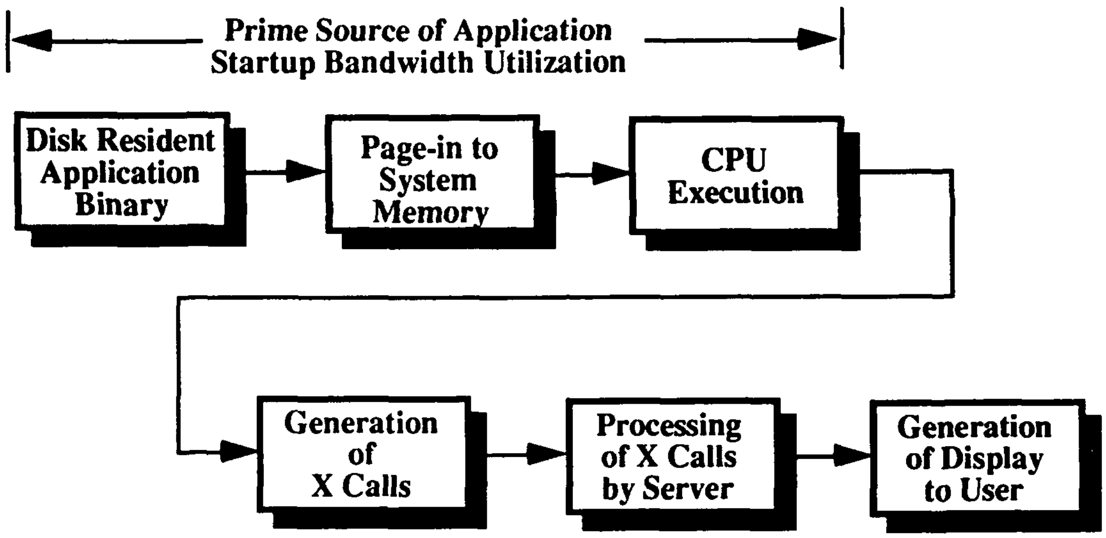

Figure 4-5. Application Startup Sequence

The second behavior pattern observed was that execution bandwidth was roughly comparable for particular applications executing on datafull and dataless workstations, but increased by factors of 25-100 percent for diskless node workstations. In addition, we observed that bandwidth consumption for exported X-Window applications (X-Terminal data), ranged from two to five times greater than that for datafull or dataless node operation.

We attribute the poor bandwidth behavior of the X-Terminal to the nature of the X-Protocol rather than to our specific X-Terminal. In general, we observed that X-Protocol behavior is characterized by significantly greater number of frames (10-45 times greater depending on the specific application), where the mean size ("Mean Bytes") of each frame is roughly 1/5-1/2 the size of datafull/dataless/diskless node configurations.

SECTION 5

CONCLUSIONS

We believe we were successful in meeting the overall goals of this research effort, which were to determine the characteristic behaviors of NFS and X-Protocol in a variety of workstation configurations; and to devise a basis for distributed computing capacity planning for IDHS programs. Detailed conclusions are cited below.

5.1 CONCLUSION 1: BASIS FOR CAPACITY PLANNING

Based on the significant differences observed in bandwidth utilization of datafull, dataless, diskless, and X-Terminals in application startup as opposed to application execution, we believe that bandwidth utilization figures associated with **application execution** be used as the primary basis for capacity planning.

Although application startup places a short, but severe strain on available network bandwidth, we believe that most operational sites are characterized by a somewhat "staggered" user startup schedule. This in turn produces a "randomizing" effect which distributes the startup load over a larger time interval. System planners should however consider this loading effect when designing a "con-ops" for their system/site.

Also, we believe that the approach taken in our execution tests are somewhat conservative in emulating the actual user/application behavior, in that we did not necessarily factor in "user think time" into our tests. This should have the net effect of producing a simulation which is closer to a worst-case scenario for system planners.

5.2 CONCLUSION 2: ESTIMATING OPTIMAL NETWORK CONFIGURATIONS

Discounting the simultaneous startup issue described above, and using bandwidth utilization data generated by an application(s) execution(s), we can make a rough estimate as to the approximate number of users a subLAN can support based on its mix of system configurations and applications. This estimate can be obtained by dividing the application/system bandwidth consumption percentage into 100 percent, yielding the approximate number of systems which can be supported for that application/system mix. A sample estimate graph is shown in figure 5-1, entitled "Estimate of Maximum Systems in a Homogeneous Application/System Environment."

5.3 CONCLUSION 3: USE OF "X-TERMINALS"

We believe that even if a faster X-Terminal had been used in our study, that the "Total Bytes" figures would have remained constant. While "Mean Bytes" could increase if an X-Terminal made better use of buffering, and possibly lower the total number of frames, we do not

	Slate	Oilstock	Paragon
Datafull	95	24	243
Dataless	74	26	270
Diskless	35	19	217
X-Terminal	37	11	44

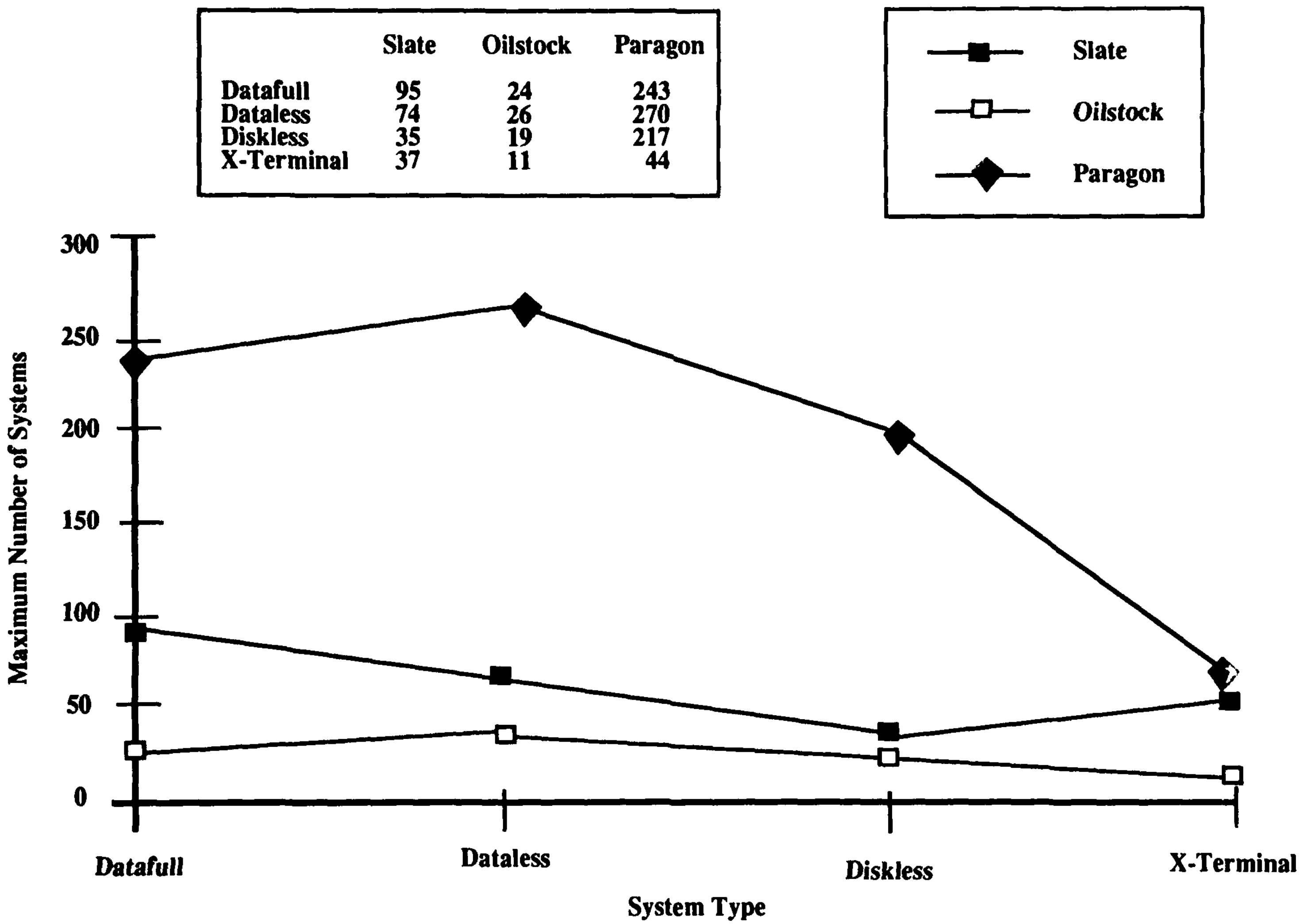

Figure 5-1. Estimate of Maximum Systems in a Homogeneous Application/System Environment

believe that the total number of frames would be lowered substantially. We do expect that the use of a higher performance X-Terminal would result in a decrease in "Total Delta Time," which would improve user perceived performance provided a network was lightly loaded. However, as stated above, "Total Bytes" would remain constant, and as such, drive up the already high bandwidth utilization rate of the "X-Terminal." This in turn would necessitate a reduction in the total number of "X-Terminals" capable of being serviced on a particular subLAN.

SECTION 6

RECOMMENDATIONS

In addition to the conclusions drawn in the previous section, we have included several recommendations which include ways in which the results of our study can be applied to the system engineering needs of the IDHS community, and ways in which these basic methods can be refined to provide more realistic, and hence, more accurate results for the community.

6.1 RECOMMENDATION 1: USE OF DISKLESS NODE WORKSTATIONS

From a network load standpoint, this configuration consistently exhibited the greatest amount of bandwidth utilization. While diskless node workstations were novel when first introduced, Winchester disk technology has advanced to the point where it is practical to provide each user workstation with a hard disk of at least 200 MB. The workstation can then be configured as either a datafull or dataless node as requirements dictate.

6.2 RECOMMENDATION 2: USE OF NFS FILE SERVERS

With the possible exception of very data intense applications (such as an interactive 3D map graphics system), use of NFS File Servers appears to be practical to the IDHS community. This is due to the relatively low loading exhibited with the test applications run on datafull and dataless node workstations. Use of central NFS servers minimizes system disk storage requirements and simplifies file system maintenance.

6.3 RECOMMENDATION 3: USE OF DATALESS NODE WORKSTATIONS

Dataless node configurations consume only slightly more bandwidth than do datafull configurations, and allow centralized management and storage of a sizeable file system. As such, we can make a general recommendation that systems take advantage of this configuration. However, as with all the recommendations presented here, implementation decisions should be based on the specific mix of applications present in a system.

6.4 RECOMMENDATION 4: USE OF "X-TERMINALS"

Based on our observations, we recommend a cautious approach to the use of X-Terminals in the IDHS environment. In addition to their potentially adverse effect on bandwidth utilization, system designers should not be lulled into thinking "X-Terminals" are a cheap solution. While the display head itself is relatively inexpensive, the system designer must allocate adequate application server resources in terms of raw CPU power, system memory, and system swap space.

6.5 RECOMMENDATION 5: USE OF NETWORK BEHAVIOR AS AN APPLICATION SELECTION CRITERIA

As was evidenced in our tests, specific applications exhibit a wide variety of paging and network related behaviors. The community should consider added network behavior (e.g.., low bandwidth utilization) as an additional criteria when selecting applications for their systems.

6.6 RECOMMENDATION 6: ENHANCEMENTS TO NETWORK LOADING MODELLING TECHNIQUES

When this task was initiated we were uncertain if meaningful data could be obtained. Upon analysis of our data, we were encouraged that this approach to network modelling has some promise. With that in mind, this technique could be enhanced through several follow-on activities. The first is to test the accuracy of our model against real IDHS systems, using similar performance monitoring tools. Provided our model mimics the real-world reasonably well, the next step should be to extend the model to deal with heterogeneous mixes of applications and workstation configurations. Following this, the final logical step would be to integrate Remote Terminal Emulator (RTE) technology into the model. RTEs allow user X-Window and console session activities to be captured into a file which then be "played back" to systems under test (SUT). RTEs provide a reliable mechanism to replicate a set of user generated activities, which includes "think-time." In addition, RTEs can be used to stress test systems by simulating many users at once from a single captured session, or set of sessions.

APPENDIX A

APPLICATION SESSION DESCRIPTION/DATA COLLECTION PROCEDURE

A.1 SLATE OFFICE AUTOMATION APPLICATION

Isolate all workstations which are not involved in this test by disconnecting them from the Ethernet fanout box. Log in as the test user on the client workstation and clear all unneeded files from the test account's file system. Start and exit Slate to set up the default configuration files. Execute a nfsstat -z and a vmstat -z as root after rebooting the client to clear all statistical information.

server:	login as root xset s off cd /misc script nfsstat -z nfsstat vmstat -z vmstat -s
sniffer:	set up sniffer to capture between server and client
client:	login as the test user via the X Display Manager
sniffer:	F10 to stop capture save buffer frames to disk as file: C:\STRIEBER\TEST1A restart capture (F10)
server:	nfsstat vmstat -s
client:	script nfsstat vmstat -s mwm &
sniffer:	F10 to stop capture save buffer frames to disk as file: C:\STRIEBER\TEST1B restart capture (F10)
server:	nfsstat vmstat -s

client: click in xterm
nfsstat
vmstat -s

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST1C
restart capture (F10)

server: nfsstat
vmstat -s

client: slate
after slate starts let it idle for 10 seconds

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST1D
restart capture (F10)

server: nfsstat
vmstat -s

client:

1. Scroll through the document displaying a color image, text, geometric graphics, digitized speech, a spreadsheet, a chart generated from the spreadsheet, a scanned black and white image, an enclosure, and multilingual text.
2. Scroll to the speech section.
3. Trigger a six second audio segment.
4. Scroll to the top of the document.
5. Use the menu bar to change the text font style of the entire document from Helvetica, 10 point, normal face to Helvetica, 18 point, bold-italic face.
6. croll down to the spreadsheet portion of the document.
7. Change the column labeled sin(x), row 1 from 0 to 5 degrees.
8. Change the column labeled cos(x), row 1 from 0 to 5 degrees.
9. Click in graphic and generate a new graph using the new spreadsheet figures.
10. Scroll to the bottom of the document.
11. Type in three lines of text.

12. Use the menu bar to add a raster image from disk.

13. Save the edited document.

14. Use the menu bar to exit the document editor.

15. Use the window manager to exit the file selection box.

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST1E
restart capture (F10)

server: nfsstat
vmstat -s

client: nfsstat
vmstat -s
exit
exit

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST1F

server: nfsstat
vmstat -s
exit
mv typescript test1.server
mv ~test/typescript /misc/test1.client

A.2 OILSTOCK MAP GRAPHICS APPLICATION

Isolate all workstations which are not involved in this test by disconnecting them from the Ethernet fanout box. Log in as the test user on the client workstation and clear all unneeded files from the test account's file system. Execute a nfsstat -z and a vmstat -z as root after rebooting the client to clear all statistical information.

server: login as root
xset s off
cd /misc
script
nfsstat -z
nfsstat
vmstat -z
vmstat -s

sniffer: set up sniffer to capture between server and client

client: login as the test user via the X Display Manager

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST2A
restart capture (F10)

server: nfsstat
vmstat -s

client: script
nfsstat
vmstat -s
mwm &

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST2B
restart capture (F10)

server: nfsstat
vmstat -s

client: click in xterm
nfsstat
vmstat -s

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST2C
restart capture (F10)

server: nfsstat
vmstat -s

client: map
after map starts let it idle for 10 seconds

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST2D
restart capture (F10)

server: nfsstat
vmstat -s

client:

1. Start the application within the binaries directory.
2. Select *Map Detail* from the menu bar.
3. Select *2* from the pull-down menu.
4. Select *Feature Detail* from the menu bar.
5. Select *9* from the pull-down menu.
6. Select *Geographic Functions* from the menu bar.

7. Select *Show Area* from the pull-down menu.
8. Type in *united states* at the prompt.
9. Select *Map Features* from the menu bar.
10. Select the toggle button labeled *Internal Boundaries.*
11. Select the toggle button labeled *Rivers.*
12. Select the toggle button labeled *Railroads.*
13. Select the toggle button labeled *Roads.*
14. Select the toggle button labeled *Cities.*
15. Select the toggle button labeled *Countries/Bodies of Water.*
16. Select *Intel Features* from the menu bar.
17. Select *LAT/LON Grid Lines* from the pull-down menu.

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST2E
restart capture (F10)

server: nfsstat
vmstat -s

client:
18. Select *Projection* from the menu bar.
19. Select *Orthographic Equitorial* from the pull-down menu.
20. Select *File* from the menu bar.
21. Select *Exit* from the pull-down menu.
22. Select the *Ok* button.

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST2F
restart capture (F10)

server: nfsstat
vmstat -s

client: nfsstat
vmstat -s
exit
exit

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST2G

server: nfsstat
vmstat -s
exit
mv typescript test1.server
mv ~test/typescript /misc/test1.client

A.3 PARAGON IMAGE PROCESSING APPLICATION

Isolate all workstations which are not involved in this test by disconnecting them from the Ethernet fanout box. Log in as the test user on the client workstation and clear all unneeded files from the test account's file system. Execute a nfsstat -z and a vmstat -z as root after rebooting the client to clear all statistical information.

server:
login as root
xset s off
cd /misc
script
nfsstat -z
nfsstat
vmstat -z
vmstat -s

sniffer:
set up sniffer to capture between server and client

client:
login as the test user via the X Display Manager

sniffer:
F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST3A
restart capture (F10)

server:
nfsstat
vmstat -s

client:
script
nfsstat
vmstat -s
mwm &

sniffer:
F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST3B
restart capture (F10)

server:
nfsstat
vmstat -s

client:
click in xterm
nfsstat
vmstat -s

sniffer:
F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST3C
restart capture (F10)

server: nfsstat
vmstat -s

client: demo
after slate starts let it idle for 10 seconds

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST3D
restart capture (F10)

server: nfsstat
vmstat -s

client:

1. Start the application within the user's home directory with the Kiev carrier as the default image.
2. Move the vdisplay_form under the image display to coordinates (10,650).
3. Move the pointer to the *Zoom* button and click.
4. Freeze the display on the zoom window.
5. Move the zoom window to the right of the image display to coordinates (700,30).
6. Zoom in on the Kamov-25 helicopter at zoom factor 2.
7. Select *Functions* from the vdisplay_form.
8. Resize the vdisplay_form_function_subform using the bottom right corner to 420x460.
9. Move the vdisplay_form_function_subform to (690,340).
10. Select *ROI* (Region Of Interest).
11. Select *ROI OPERATIONS*.
12. Select *vhenhance* from ROI Operations.
13. Move the right vertical ROI bar to 335.
14. Move the left vertical ROI bar to 255.
15. Move the bottom horizontal ROI bar to 220.
16. Move the bottom horizontal ROI bar to 155.

17. Click on extract within the ROI window.

18. Select *vhxray* from ROI Operations.

19. Click on extract within the ROI window.

20. Select *vrotate* from ROI Operations.

21. Click on extract within the ROI window.

22. Move the pointer to the *Overlays* button and click.

23. Move the vdisplay_form_xvoverlay_subform to (690,340).

24. Type in the input overlay filename and click on *Get Overlay* (the selected overlay contains lines, circles, rectangles, and text).

25. Click on *Close* from the vdisplay_form_xvoverlay_subform.

26. Click on *Functions* from the vdisplay_form.

27. Click on *Input/Output* from the vdisplay_form_function_subform.

28. Enter the output file name including path (home1/users/test/ship2.ovr).

29. Click on *OUTPUT IMAGE*.

30. Click on *Close* from the vdisplay_form_function_subform.

31. Click on *Quit* from vdisplay_form.

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST3E
restart capture (F10)

server: nfsstat
vmstat -s

client: nfsstat
vmstat -s
exit
exit

sniffer: F10 to stop capture
save buffer frames to disk as file: C:\STRIEBER\TEST3F

server: nfsstat

```
nfsstat
vmstat -s
exit
mv typescript test1.server
mv ~test/typescript /misc/test1.client
```

APPENDIX B

TESTBED SYSTEM CONFIGURATIONS

B.1 FILE SERVER /ETC/FSTAB FILE

```
/dev/sd2a /            4.2 rw   1 1
/dev/sd2g /usr         4.2 rw   1 2
/dev/sd2d /export      4.2 rw   1 3
/dev/sd2e /export/swap 4.2 rw   1 6
/dev/sd2f /home1       4.2 rw   1 4
/dev/sd2h /var         4.2 rw   1 5
```

B.2 FILE SERVER /ETC/EXPORTS FILE

```
/usr -root=diskless:dataless
/home -root=diskless:dataless:datafull
/home1 -access=datafull,root=datafull
/var/spool/mail -access=diskless:dataless:datafull
/export/root/diskless   -access=diskless,root=diskless
/export/swap/diskless   -access=diskless,root=diskless
```

B.3 DATAFULL NODE /ETC/FSTAB FILE

```
/dev/sd2a  /     4.2 rw 1 1
/dev/sd2d  /home  4.2 rw 1 3
/dev/sd2g  /usr  4.2 rw 1 2
/dev/sd2h  /var  4.2 rw 1 4
nfserver:/home1 /home1 nfs rw 0 0
```

B.4 DATALESS NODE /ETC/FSTAB FILE

```
/dev/sd3a on / type 4.2 (rw)
nfserver:/export/exec/sun4.sunos.4.1.1 on /usr type nfs (rw)
nfserver:/export/exec/kvm/sun4c.sunos.4.1.1 on /usr/kvm type nfs (rw)
nfserver:/export/share/sunos.4.1.1 on /usr/share type nfs (rw)
nfserver:/home1 on /home1 type nfs (rw)
nfserver:/home/nfserver on /home/nfserver type nfs (rw)
```

B.5 DISKLESS NODE /ETC/FSTAB FILE

```
nfserver:/export/root/diskless  / nfs rw 0 0
nfserver:/export/exec/sun4.sunos.4.1.1 /usr nfs ro 0 0
nfserver:/export/exec/kvm/sun4c.sunos.4.1.1    /usr/kvm nfs ro 0 0
nfserver:/home/nfserver /home/nfserver nfs rw 0 0
nfserver:/home1 /home1 nfs rw 0 0
nfserver:/var/spool/mail       /var/spool/mail nfs rw 0 0
nfserver:/export/share/sunos.4.1.1      /usr/share nfs rw 0 0
```

B.6 DISK CAPACITY OF DATAFULL NODE

```
Filesystem        kbytes   used  avail capacity  Mounted on
/dev/sd2a          8999   3706   4393   46%   /
/dev/sd2g         46279  38805   2846   93%   /usr
/dev/sd2d         34847     10  31352    0%   /home
/dev/sd2h         13815     93  12340    1%   /var
nfserver:/home1     588438 150091 379503   28%   /home1
```

B.7 DISK CAPACITY OF DATALESS NODE

```
Filesystem        kbytes   used  avail capacity  Mounted on
/dev/sd3a          7799   3987   3032   57%   /
nfserver:/export/exec/sun4.sunos.4.1.1
                 144239  43817  85998   34%   /usr
nfserver:/export/exec/kvm/sun4c.sunos.4.1.1
                 144239  43817  85998   34%   /usr/kvm
nfserver:/export/share/sunos.4.1.1
                 144239  43817  85998   34%   /usr/share
nfserver:/home1    588438 150091 379503   28%   /home1
nfserver:/home/nfserver
                   8999   4206   3893   52%   /home/nfserver
```

APPENDIX C

POSTPROCESSING SCRIPTS

C.1 CONV

```
#!/bin/sh
#
# This script will convert the DOS-based data file to UNIX and the new data will be totaled.
if [ -f $1 ]
then
echo "Processing Data . . . Please Wait"
echo ""
echo "Converting from DOS to UNIX format . . ."
dos2unix $1 $1.new
echo "Selecting only the frame header lines . . ."
grep nfserver $1.new > $1.good
rm $1.new
echo "Totaling statistics . . ."
/home1/users/strieber/doc/AFIA/Notes/total $1.good > $1.totals
rm $1.good
echo "Totalling NFS statistics . . ."
/home1/users/strieber/doc/AFIA/Notes/greps $1 >> $1.totals
echo "Done"
else
 echo "File not found: " $1
fi
```

C.2 TOTAL

```
#!/bin/sh
# total3: Total Delta Time
# total5: Total Bytes
# total6: Total Sniffer Network Bandwidth Utilization (100 ms window)
#
awk '
 BEGIN { FS="," }
{
  total3 += $3
  total5 += $5
  total6 += $6
  ++count
}
 END {
   print "Frame count: ", count
```

```
    print ""
    print "Total Delta Time: ", total3
    print "Mean Delta Time:  ", total3 / (count - 1)
    print ""
    print "Total Bytes: ", total5
    print "Mean Bytes:  ", total5 / count
    print ""
    print "Total NW Utilization Sniffer MB/Sec (%): ", total6
    print "Mean NW Utilization Sniffer MB/Sec (%):  ", total6 / count
    print ""
    print "Mean NW Utilization Actual Bytes/Sec (%):  ", total5 / total3
} ' $1
```

C.3 GREPS

```
#!/bin/sh
proc0=`grep "Proc = 0 (" $1 | wc -l`
proc1=`grep "Proc = 1 (" $1 | wc -l`
proc2=`grep "Proc = 2 (" $1 | wc -l`
proc3=`grep "Proc = 3 (" $1 | wc -l`
proc4=`grep "Proc = 4 (" $1 | wc -l`
proc5=`grep "Proc = 5 (" $1 | wc -l`
proc6=`grep "Proc = 6 (" $1 | wc -l`
proc7=`grep "Proc = 7 (" $1 | wc -l`
proc8=`grep "Proc = 8 (" $1 | wc -l`
proc9=`grep "Proc = 9 (" $1 | wc -l`
proc10=`grep "Proc = 10 (" $1 | wc -l`
proc11=`grep "Proc = 11 (" $1 | wc -l`
proc12=`grep "Proc = 12 (" $1 | wc -l`
proc13=`grep "Proc = 13 (" $1 | wc -l`
proc14=`grep "Proc = 14 (" $1 | wc -l`
proc15=`grep "Proc = 15 (" $1 | wc -l`
proc16=`grep "Proc = 16 (" $1 | wc -l`
proc17=`grep "Proc = 17 (" $1 | wc -l`
proc18=`grep "Proc = 18 (" $1 | wc -l`
udp=`grep "continuation ID=" $1 | wc -l`

echo ""
echo "NFS Procedure Totals"
echo ""
echo "  Null:           " $proc0
echo "  Get Attributes: " $proc1
echo "  Set Attributes: " $proc2
echo "  Root:           " $proc3
echo "  Lookup:         " $proc4
echo "  Read Link:      " $proc5
echo "  Read:           " $proc6
```

```
echo "  Write Cache:   " $proc7
echo "  Write:         " $proc8
echo "  Create:        " $proc9
echo "  Remove:        " $proc10
echo "  Rename:        " $proc11
echo "  Link:          " $proc12
echo "  Symbolic Link: " $proc13
echo "  Make Dir:      " $proc14
echo "  Remove Dir:    " $proc15
echo "  Read Dir:      " $proc16
echo "  Stata File Sys: " $proc17
echo "  N Proc:        " $proc18
echo ""
echo "UDP Continuation: " $udp
```

C.4 GO

```
#!/bin/sh
#
go.datafull
go.dataless
go.diskless
go.xterminl
```

C.5 GO.DATAFULL

```
# datafull

cd /home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2a
/home1/users/strieber/doc/AFIA/Notes/conv t2aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2adf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2anf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2b
/home1/users/strieber/doc/AFIA/Notes/conv t2bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2bdf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2bnf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2c
/home1/users/strieber/doc/AFIA/Notes/conv t2cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2cdf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2cnf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2d
/home1/users/strieber/doc/AFIA/Notes/conv t2dboth.csv
```

```
/home1/users/strieber/doc/AFIA/Notes/conv t2ddf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2dnf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2e
/home1/users/strieber/doc/AFIA/Notes/conv t2eboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2edf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2enf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2f
/home1/users/strieber/doc/AFIA/Notes/conv t2fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2fdf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2fnf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2g
/home1/users/strieber/doc/AFIA/Notes/conv t2gboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2gdf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2gnf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/paragon/test3a
/home1/users/strieber/doc/AFIA/Notes/conv t3aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3adf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3anf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/paragon/test3b
/home1/users/strieber/doc/AFIA/Notes/conv t3bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3bdf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3bnf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/paragon/test3c
/home1/users/strieber/doc/AFIA/Notes/conv t3cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3cdf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3cnf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/paragon/test3d
/home1/users/strieber/doc/AFIA/Notes/conv t3dboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3ddf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3dnf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/paragon/test3e
/home1/users/strieber/doc/AFIA/Notes/conv t3eboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3edf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3enf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/paragon/test3f
/home1/users/strieber/doc/AFIA/Notes/conv t3fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3fdf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3fnf.csv
```

```
cd /home1/users/strieber/doc/AFIA/results/datafull/slate/test1a
/home1/users/strieber/doc/AFIA/Notes/conv t1aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1adf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1anf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/slate/test1b
/home1/users/strieber/doc/AFIA/Notes/conv t1bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1bdf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1bnf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/slate/test1c
/home1/users/strieber/doc/AFIA/Notes/conv t1cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1cdf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1cnf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/slate/test1d
/home1/users/strieber/doc/AFIA/Notes/conv t1dboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1ddf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1dnf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/slate/test1e
/home1/users/strieber/doc/AFIA/Notes/conv t1eboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1edf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1enf.csv

cd /home1/users/strieber/doc/AFIA/results/datafull/slate/test1f
/home1/users/strieber/doc/AFIA/Notes/conv t1fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1fdf.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1fnf.csv
```

C.6 GO.DATALESS

```
# dataless

cd /home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2a
/home1/users/strieber/doc/AFIA/Notes/conv t2aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2adal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2anf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2b
/home1/users/strieber/doc/AFIA/Notes/conv t2bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2bdal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2bnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2c
/home1/users/strieber/doc/AFIA/Notes/conv t2cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2cdal.csv
```

```
/home1/users/strieber/doc/AFIA/Notes/conv t2cnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2d
/home1/users/strieber/doc/AFIA/Notes/conv t2dboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2ddal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2dnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2e
/home1/users/strieber/doc/AFIA/Notes/conv t2eboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2edal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2enf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2f
/home1/users/strieber/doc/AFIA/Notes/conv t2fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2fdal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2fnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2g
/home1/users/strieber/doc/AFIA/Notes/conv t2gboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2gdal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2gnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/paragon/test3a
/home1/users/strieber/doc/AFIA/Notes/conv t3aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3adal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3anf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/paragon/test3b
/home1/users/strieber/doc/AFIA/Notes/conv t3bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3bdal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3bnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/paragon/test3c
/home1/users/strieber/doc/AFIA/Notes/conv t3cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3cdal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3cnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/paragon/test3d
/home1/users/strieber/doc/AFIA/Notes/conv t3dboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3ddal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3dnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/paragon/test3e
/home1/users/strieber/doc/AFIA/Notes/conv t3eboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3edal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3enf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/paragon/test3f
```

```
/home1/users/strieber/doc/AFIA/Notes/conv t3fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3fdal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3fnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/slate/test1a
/home1/users/strieber/doc/AFIA/Notes/conv t1aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1adal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1anf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/slate/test1b
/home1/users/strieber/doc/AFIA/Notes/conv t1bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1bdal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1bnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/slate/test1c
/home1/users/strieber/doc/AFIA/Notes/conv t1cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1cdal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1cnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/slate/test1d
/home1/users/strieber/doc/AFIA/Notes/conv t1dboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1ddal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1dnf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/slate/test1e
/home1/users/strieber/doc/AFIA/Notes/conv t1eboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1edal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1enf.csv

cd /home1/users/strieber/doc/AFIA/results/dataless/slate/test1f
/home1/users/strieber/doc/AFIA/Notes/conv t1fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1fdal.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1fnf.csv
```

C.7 GO.DISKLESS

```
# diskless

cd /home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2a
/home1/users/strieber/doc/AFIA/Notes/conv t2aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2adil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2anf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2b
/home1/users/strieber/doc/AFIA/Notes/conv t2bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2bdil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2bnf.csv
```

```
cd /home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2c
/home1/users/strieber/doc/AFIA/Notes/conv t2cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2cdil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2cnf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2d
/home1/users/strieber/doc/AFIA/Notes/conv t2dboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2ddil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2dnf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2e
/home1/users/strieber/doc/AFIA/Notes/conv t2eboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2edil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2enf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2f
/home1/users/strieber/doc/AFIA/Notes/conv t2fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2fdil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2fnf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2g
/home1/users/strieber/doc/AFIA/Notes/conv t2gboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2gdil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2gnf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/paragon/test3a
/home1/users/strieber/doc/AFIA/Notes/conv t3aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3adil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3anf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/paragon/test3b
/home1/users/strieber/doc/AFIA/Notes/conv t3bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3bdil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3bnf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/paragon/test3c
/home1/users/strieber/doc/AFIA/Notes/conv t3cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3cdil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3cnf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/paragon/test3d
/home1/users/strieber/doc/AFIA/Notes/conv t3dboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3ddil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3dnf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/paragon/test3e
/home1/users/strieber/doc/AFIA/Notes/conv t3eboth.csv
```

```
/home1/users/strieber/doc/AFIA/Notes/conv t3edil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3enf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/paragon/test3f
/home1/users/strieber/doc/AFIA/Notes/conv t3fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3fdil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3fnf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/slate/test1a
/home1/users/strieber/doc/AFIA/Notes/conv t1aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1adil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1anf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/slate/test1b
/home1/users/strieber/doc/AFIA/Notes/conv t1bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1bdil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1bnf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/slate/test1c
/home1/users/strieber/doc/AFIA/Notes/conv t1cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1cdil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1cnf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/slate/test1d
/home1/users/strieber/doc/AFIA/Notes/conv t1dboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1ddil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1dnf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/slate/test1e
/home1/users/strieber/doc/AFIA/Notes/conv t1eboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1edil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1enf.csv

cd /home1/users/strieber/doc/AFIA/results/diskless/slate/test1f
/home1/users/strieber/doc/AFIA/Notes/conv t1fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1fdil.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1fnf.csv
```

C.8 GO.XTERMINL

```
# xterminl

cd /home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2a
/home1/users/strieber/doc/AFIA/Notes/conv t2aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2axt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2anf.csv
```

```
cd /home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2b
/home1/users/strieber/doc/AFIA/Notes/conv t2bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2bxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2bnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2c
/home1/users/strieber/doc/AFIA/Notes/conv t2cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2cxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2cnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2d
/home1/users/strieber/doc/AFIA/Notes/conv t2dboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2dxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2dnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2e
/home1/users/strieber/doc/AFIA/Notes/conv t2eboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2ext.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2enf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2f
/home1/users/strieber/doc/AFIA/Notes/conv t2fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2fxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2fnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2g
/home1/users/strieber/doc/AFIA/Notes/conv t2gboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2gxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t2gnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3a
/home1/users/strieber/doc/AFIA/Notes/conv t3aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3axt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3anf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3b
/home1/users/strieber/doc/AFIA/Notes/conv t3bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3bxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3bnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3c
/home1/users/strieber/doc/AFIA/Notes/conv t3cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3cxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3cnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3d
/home1/users/strieber/doc/AFIA/Notes/conv t3dboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3dxt.csv
```

```
/home1/users/strieber/doc/AFIA/Notes/conv t3dnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3e1
/home1/users/strieber/doc/AFIA/Notes/conv t3e1both.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3e1xt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3e1nf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3e2
/home1/users/strieber/doc/AFIA/Notes/conv t3e2both.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3e2xt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3e2nf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3f
/home1/users/strieber/doc/AFIA/Notes/conv t3fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3fxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t3fnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/slate/test1a
/home1/users/strieber/doc/AFIA/Notes/conv t1aboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1axt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1anf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/slate/test1b
/home1/users/strieber/doc/AFIA/Notes/conv t1bboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1bxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1bnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/slate/test1c
/home1/users/strieber/doc/AFIA/Notes/conv t1cboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1cxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1cnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/slate/test1d
/home1/users/strieber/doc/AFIA/Notes/conv t1dboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1dxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1dnf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/slate/test1e
/home1/users/strieber/doc/AFIA/Notes/conv t1eboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1ext.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1enf.csv

cd /home1/users/strieber/doc/AFIA/results/xterminl/slate/test1f
/home1/users/strieber/doc/AFIA/Notes/conv t1fboth.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1fxt.csv
/home1/users/strieber/doc/AFIA/Notes/conv t1fnf.csv
```

C.9 DIRECTORY STRUCTURE USED TO STORE SNIFFER DATA AND RESULTS

```
datafull/     dataless/     diskless/     xterminl/

/home1/users/strieber/doc/AFIA/results/datafull:
oilstock/     paragon/      slate/

/home1/users/strieber/doc/AFIA/results/datafull/oilstock:
scripts/      test2a/       test2c/       test2e/       test2g/
sniffer/      test2b/       test2d/                     test2f/

/home1/users/strieber/doc/AFIA/results/datafull/oilstock/scripts:
test2.client            typescript.client
test2.server            typescript.server

/home1/users/strieber/doc/AFIA/results/datafull/oilstock/sniffer:
test2a.enc    test2c.enc    test2e.enc    test2g.enc
test2b.enc    test2d.enc    test2f.enc

/home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2a:
t2aboth.csv             t2adf.csv               t2anf.csv
t2aboth.csv.totals      t2adf.csv.totals        t2anf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2b:
t2bboth.csv             t2bdf.csv               t2bnf.csv
t2bboth.csv.totals      t2bdf.csv.totals        t2bnf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2c:
t2cboth.csv             t2cdf.csv               t2cnf.csv
t2cboth.csv.totals      t2cdf.csv.totals        t2cnf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2d:
t2dboth.csv             t2ddf.csv               t2dnf.csv
t2dboth.csv.totals      t2ddf.csv.totals        t2dnf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2e:
t2eboth.csv             t2edf.csv               t2enf.csv
t2eboth.csv.totals      t2edf.csv.totals        t2enf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2f:
t2fboth.csv             t2fdf.csv               t2fnf.csv
t2fboth.csv.totals      t2fdf.csv.totals        t2fnf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/oilstock/test2g:
t2gboth.csv             t2gdf.csv               t2gnf.csv
t2gboth.csv.totals      t2gdf.csv.totals        t2gnf.csv.totals
```

/home1/users/strieber/doc/AFIA/results/datafull/paragon:

scripts/	test3a/	test3c/	test3e/
sniffer/	test3b/	test3d/	test3f/

/home1/users/strieber/doc/AFIA/results/datafull/paragon/scripts:

test3.client	typescript.client
test3.server	typescript.server

/home1/users/strieber/doc/AFIA/results/datafull/paragon/sniffer:

test3a.enc	test3c.enc	test3e.enc
test3b.enc	test3d.enc	test3f.enc

/home1/users/strieber/doc/AFIA/results/datafull/paragon/test3a:

t3aboth.csv	t3adf.csv	t3anf.csv
t3aboth.csv.totals	t3adf.csv.totals	t3anf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/paragon/test3b:

t3bboth.csv	t3bdf.csv	t3bnf.csv
t3bboth.csv.totals	t3bdf.csv.totals	t3bnf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/paragon/test3c:

t3cboth.csv	t3cdf.csv	t3cnf.csv
t3cboth.csv.totals	t3cdf.csv.totals	t3cnf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/paragon/test3d:

t3dboth.csv	t3ddf.csv	t3dnf.csv
t3dboth.csv.totals	t3ddf.csv.totals	t3dnf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/paragon/test3e:

t3eboth.csv	t3edf.csv	t3enf.csv
t3eboth.csv.totals	t3edf.csv.totals	t3enf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/paragon/test3f:

t3fboth.csv	t3fdf.csv	t3fnf.csv
t3fboth.csv.totals	t3fdf.csv.totals	t3fnf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/slate:

scripts/	test1a/	test1c/	test1e/
sniffer/	test1b/	test1d/	test1f/

/home1/users/strieber/doc/AFIA/results/datafull/slate/scripts:

test1.client	typescript.client
test1.server	typescript.server

/home1/users/strieber/doc/AFIA/results/datafull/slate/sniffer:

test1a.enc	test1c.enc	test1e.enc
test1b.enc	test1d.enc	test1f.enc

/home1/users/strieber/doc/AFIA/results/datafull/slate/test1a:
t1aboth.csv t1adf.csv t1anf.csv
t1aboth.csv.totals t1adf.csv.totals t1anf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/slate/test1b:
t1bboth.csv t1bdf.csv t1bnf.csv
t1bboth.csv.totals t1bdf.csv.totals t1bnf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/slate/test1c:
t1cboth.csv t1cdf.csv t1cnf.csv
t1cboth.csv.totals t1cdf.csv.totals t1cnf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/slate/test1d:
t1dboth.csv t1ddf.csv t1dnf.csv
t1dboth.csv.totals t1ddf.csv.totals t1dnf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/slate/test1e:
t1eboth.csv t1edf.csv t1enf.csv
t1eboth.csv.totals t1edf.csv.totals t1enf.csv.totals

/home1/users/strieber/doc/AFIA/results/datafull/slate/test1f:
t1fboth.csv t1fdf.csv t1fnf.csv
t1fboth.csv.totals t1fdf.csv.totals t1fnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless:
oilstock/ paragon/ slate/

/home1/users/strieber/doc/AFIA/results/dataless/oilstock:
scripts/ test2a/ test2c/ test2e/ test2g/
sniffer/ test2b/ test2d/ test2f/

/home1/users/strieber/doc/AFIA/results/dataless/oilstock/scripts:
test2.client typescript.client
test2.server typescript.server

/home1/users/strieber/doc/AFIA/results/dataless/oilstock/sniffer:
test2a.enc test2c.enc test2e.enc test2g.enc
test2b.enc test2d.enc test2f.enc

/home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2a:
t2aboth.csv t2adal.csv t2anf.csv
t2aboth.csv.totals t2adal.csv.totals t2anf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2b:
t2bboth.csv t2bdal.csv t2bnf.csv
t2bboth.csv.totals t2bdal.csv.totals t2bnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2c:

t2cboth.csv	t2cdal.csv	t2cnf.csv
t2cboth.csv.totals	t2cdal.csv.totals	t2cnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2d:

t2dboth.csv	t2ddal.csv	t2dnf.csv
t2dboth.csv.totals	t2ddal.csv.totals	t2dnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2e:

t2eboth.csv	t2edal.csv	t2enf.csv
t2eboth.csv.totals	t2edal.csv.totals	t2enf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2f:

t2fboth.csv	t2fdal.csv	t2fnf.csv
t2fboth.csv.totals	t2fdal.csv.totals	t2fnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/oilstock/test2g:

t2gboth.csv	t2gdal.csv	t2gnf.csv
t2gboth.csv.totals	t2gdal.csv.totals	t2gnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/paragon:

scripts/	test3a/	test3c/	test3e/
sniffer/	test3b/	test3d/	test3f/

/home1/users/strieber/doc/AFIA/results/dataless/paragon/scripts:

test3.client	typescript.client
test3.server	typescript.server

/home1/users/strieber/doc/AFIA/results/dataless/paragon/sniffer:

test3a.enc	test3c.enc	test3e.enc
test3b.enc	test3d.enc	test3f.enc

/home1/users/strieber/doc/AFIA/results/dataless/paragon/test3a:

t3aboth.csv	t3adal.csv	t3anf.csv
t3aboth.csv.totals	t3adal.csv.totals	t3anf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/paragon/test3b:

t3bboth.csv	t3bdal.csv	t3bnf.csv
t3bboth.csv.totals	t3bdal.csv.totals	t3bnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/paragon/test3c:

t3cboth.csv	t3cdal.csv	t3cnf.csv
t3cboth.csv.totals	t3cdal.csv.totals	t3cnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/paragon/test3d:

t3dboth.csv	t3ddal.csv	t3dnf.csv
t3dboth.csv.totals	t3ddal.csv.totals	t3dnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/paragon/test3e:

t3eboth.csv	t3edal.csv	t3enf.csv
t3eboth.csv.totals	t3edal.csv.totals	t3enf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/paragon/test3f:

t3fboth.csv	t3fdal.csv	t3fnf.csv
t3fboth.csv.totals	t3fdal.csv.totals	t3fnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/slate:

scripts/	test1a/	test1c/	test1e/
sniffer/	test1b/	test1d/	test1f/

/home1/users/strieber/doc/AFIA/results/dataless/slate/scripts:

test1.client	typescript.client
test1.server	typescript.server

/home1/users/strieber/doc/AFIA/results/dataless/slate/sniffer:

test1a.enc	test1c.enc	test1e.enc
test1b.enc	test1d.enc	test1f.enc

/home1/users/strieber/doc/AFIA/results/dataless/slate/test1a:

t1aboth.csv	t1adal.csv	t1anf.csv
t1aboth.csv.totals	t1adal.csv.totals	t1anf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/slate/test1b:

t1bboth.csv	t1bdal.csv	t1bnf.csv
t1bboth.csv.totals	t1bdal.csv.totals	t1bnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/slate/test1c:

t1cboth.csv	t1cdal.csv	t1cnf.csv
t1cboth.csv.totals	t1cdal.csv.totals	t1cnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/slate/test1d:

t1dboth.csv	t1ddal.csv	t1dnf.csv
t1dboth.csv.totals	t1ddal.csv.totals	t1dnf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/slate/test1e:

t1eboth.csv	t1edal.csv	t1enf.csv
t1eboth.csv.totals	t1edal.csv.totals	t1enf.csv.totals

/home1/users/strieber/doc/AFIA/results/dataless/slate/test1f:

t1fboth.csv	t1fdal.csv	t1fnf.csv
t1fboth.csv.totals	t1fdal.csv.totals	t1fnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless:

oilstock/	paragon/	slate/

/home1/users/strieber/doc/AFIA/results/diskless/oilstock:

scripts/	test2a/	test2c/	test2e/	test2g/
sniffer/	test2b/	test2d/		test2f/

/home1/users/strieber/doc/AFIA/results/diskless/oilstock/scripts:

test2.client	typescript.client
test2.server	typescript.server

/home1/users/strieber/doc/AFIA/results/diskless/oilstock/sniffer:

test2a.enc	test2c.enc	test2e.enc	test2g.enc
test2b.enc	test2d.enc	test2f.enc	

/home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2a:

t2aboth.csv	t2adil.csv	t2anf.csv
t2aboth.csv.totals	t2adil.csv.totals	t2anf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2b:

t2bboth.csv	t2bdil.csv	t2bnf.csv
t2bboth.csv.totals	t2bdil.csv.totals	t2bnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2c:

t2cboth.csv	t2cdil.csv	t2cnf.csv
t2cboth.csv.totals	t2cdil.csv.totals	t2cnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2d:

t2dboth.csv	t2ddil.csv	t2dnf.csv
t2dboth.csv.totals	t2ddil.csv.totals	t2dnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2e:

t2eboth.csv	t2edil.csv	t2enf.csv
t2eboth.csv.totals	t2edil.csv.totals	t2enf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2f:

t2fboth.csv	t2fdil.csv	t2fnf.csv
t2fboth.csv.totals	t2fdil.csv.totals	t2fnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/oilstock/test2g:

t2gboth.csv	t2gdil.csv	t2gnf.csv
t2gboth.csv.totals	t2gdil.csv.totals	t2gnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/paragon:

scripts/	test3a/	test3c/	test3e/	
sniffer/	test3b/	test3d/		test3f/

/home1/users/strieber/doc/AFIA/results/diskless/paragon/scripts:

test3.client	typescript.client
test3.server	typescript.server

/home1/users/strieber/doc/AFIA/results/diskless/paragon/sniffer:

test3a.enc	test3c.enc	test3e.enc
test3b.enc	test3d.enc	test3f.enc

/home1/users/strieber/doc/AFIA/results/diskless/paragon/test3a:

t3aboth.csv	t3adil.csv	t3anf.csv
t3aboth.csv.totals	t3adil.csv.totals	t3anf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/paragon/test3b:

t3bboth.csv	t3bdil.csv	t3bnf.csv
t3bboth.csv.totals	t3bdil.csv.totals	t3bnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/paragon/test3c:

t3cboth.csv	t3cdil.csv	t3cnf.csv
t3cboth.csv.totals	t3cdil.csv.totals	t3cnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/paragon/test3d:

t3dboth.csv	t3ddil.csv	t3dnf.csv
t3dboth.csv.totals	t3ddil.csv.totals	t3dnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/paragon/test3e:

t3eboth.csv	t3edil.csv	t3enf.csv
t3eboth.csv.totals	t3edil.csv.totals	t3enf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/paragon/test3f:

t3fboth.csv	t3fdil.csv	t3fnf.csv
t3fboth.csv.totals	t3fdil.csv.totals	t3fnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/slate:

scripts/	test1a/	test1c/	test1e/
sniffer/	test1b/	test1d/	test1f/

/home1/users/strieber/doc/AFIA/results/diskless/slate/scripts:

test1.client	typescript.client
test1.server	typescript.server

/home1/users/strieber/doc/AFIA/results/diskless/slate/sniffer:

test1a.enc	test1c.enc	test1e.enc
test1b.enc	test1d.enc	test1f.enc

/home1/users/strieber/doc/AFIA/results/diskless/slate/test1a:

t1aboth.csv	t1adil.csv	t1anf.csv
t1aboth.csv.totals	t1adil.csv.totals	t1anf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/slate/test1b:

t1bboth.csv	t1bdil.csv	t1bnf.csv
t1bboth.csv.totals	t1bdil.csv.totals	t1bnf.csv.totals

```
/home1/users/strieber/doc/AFIA/results/diskless/slate/test1c:
t1cboth.csv          t1cdil.csv           t1cnf.csv
t1cboth.csv.totals   t1cdil.csv.totals    t1cnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/slate/test1d:
t1dboth.csv          t1ddil.csv           t1dnf.csv
t1dboth.csv.totals   t1ddil.csv.totals    t1dnf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/slate/test1e:
t1eboth.csv          t1edil.csv           t1enf.csv
t1eboth.csv.totals   t1edil.csv.totals    t1enf.csv.totals

/home1/users/strieber/doc/AFIA/results/diskless/slate/test1f:
t1fboth.csv          t1fdil.csv           t1fnf.csv
t1fboth.csv.totals   t1fdil.csv.totals    t1fnf.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl:
oilstock/       paragon/        slate/

/home1/users/strieber/doc/AFIA/results/xterminl/oilstock:
scripts/        test2a/         test2c/         test2e@         test2g/
sniffer/        test2b/         test2d/         test2f@

/home1/users/strieber/doc/AFIA/results/xterminl/oilstock/scripts:
test2.client         typescript.client
test2.server         typescript.server

/home1/users/strieber/doc/AFIA/results/xterminl/oilstock/sniffer:
test2a.enc      test2c.enc      test2e.enc      test2g.enc
test2b.enc      test2d.enc      test2f.enc

/home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2a:
t2aboth.csv          t2anf.csv            t2axt.csv
t2aboth.csv.totals   t2anf.csv.totals     t2axt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2b:
t2bboth.csv          t2bnf.csv            t2bxt.csv
t2bboth.csv.totals   t2bnf.csv.totals     t2bxt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2c:
t2cboth.csv          t2cnf.csv            t2cxt.csv
t2cboth.csv.totals   t2cnf.csv.totals     t2cxt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2d:
t2dboth.csv          t2dnf.csv            t2dxt.csv
t2dboth.csv.totals   t2dnf.csv.totals     t2dxt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/oilstock/test2g:
```

```
t2gboth.csv         t2gnf.csv         t2gxt.csv
t2gboth.csv.totals  t2gnf.csv.totals  t2gxt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/paragon:
scripts/    test3a/    test3c/    test3e1/    test3f/
sniffer/    test3b/    test3d/    test3e2/

/home1/users/strieber/doc/AFIA/results/xterminl/paragon/scripts:
test3.client    typescript.client
test3.server    typescript.server

/home1/users/strieber/doc/AFIA/results/xterminl/paragon/sniffer:
test3a.enc    test3c.enc    test3e1.enc    test3f.enc
test3b.enc    test3d.enc    test3e2.enc

/home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3a:
t3aboth.csv         t3anf.csv         t3axt.csv
t3aboth.csv.totals  t3anf.csv.totals  t3axt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3b:
t3bboth.csv         t3bnf.csv         t3bxt.csv
t3bboth.csv.totals  t3bnf.csv.totals  t3bxt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3c:
t3cboth.csv         t3cnf.csv         t3cxt.csv
t3cboth.csv.totals  t3cnf.csv.totals  t3cxt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3d:
t3dboth.csv         t3dnf.csv         t3dxt.csv
t3dboth.csv.totals  t3dnf.csv.totals  t3dxt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3e1:
t3e1both.csv         t3e1nf.csv         t3e1xt.csv
t3e1both.csv.totals  t3e1nf.csv.totals  t3e1xt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3e2:
t3e2both.csv         t3e2nf.csv         t3e2xt.csv
t3e2both.csv.totals  t3e2nf.csv.totals  t3e2xt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/paragon/test3f:
t3fboth.csv         t3fnf.csv         t3fxt.csv
t3fboth.csv.totals  t3fnf.csv.totals  t3fxt.csv.totals

/home1/users/strieber/doc/AFIA/results/xterminl/slate:
scripts/    test1a@    test1c@    test1e@
sniffer/    test1b@    test1d@    test1f@
```

/home1/users/strieber/doc/AFIA/results/xterminl/slate/scripts:

test1.client	typescript.client
test1.server	typescript.server

APPENDIX D

POSTPROCESSING SCRIPTS

D.1 Slate Office Automation Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
After Login (test1a)				
Frame Count	111	233	344	193
Total Delta Time	4.0334	3.6631	6.5972	6.2713
Mean Delta Time	0.0366673	0.0157892	0.0192338	0.032663
Total Bytes	22542	81570	122464	26440
Mean Bytes	203.081	350.086	356	136.995
Mean NW Utilization Sniffer MB/sec. (%)	4.16441	6.71309	6.41648	0.832487
Mean NW Utilization ActualBytes/sec.	5588.83	22268	18563	4216.03
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	87	49	79	0
Set Attributes	0	0	4	0
Root	0	0	0	0
Lookup	8	98	139	0
Read Link	0	20	25	0
Read	0	10	10	0
Write Cache	0	0	0	0
Write	0	0	2	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0

D.1 Slate Office Automation Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Read Dir	12	26	36	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	3	27	42	0
After starting mwm (test1b)				
Frame Count	901	1006	1074	763
Total Delta Time	15.4464	21.9054	19.8017	32.7148
Mean Delta Time	0.0171627	0.0217964	0.0184545	0.0429328
Total Bytes	1030030	1068052	1091312	131668
Mean Bytes	1143.21	1061.68	1016.12	172.566
Mean NW Utilization Sniffer MB/sec. (%)	30.0362	30.8916	27.1885	1.16529
Mean NW Utilization actual Bytes/sec.	66684.1	48757.5	55112	4024.72
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	29	65	96	0
Set Attributes	0	0	8	0
Root	0	0	0	0
Lookup	29	58	69	0
Read Link	0	12	15	0
Read	193	179	180	0
Write Cache	0	0	0	0
Write	0	2	2	0
Create	2	2	2	0

D.1 Slate Office Automation Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	19	24	33	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	584	597	605	0
After clicking in xterm (test1c)				
Frame Count	14	54	78	141
Total Delta Time	0.1014	4.5069	10.3853	11.7068
Mean Delta Time	0.0078	0.0850358	0.134874	0.08362
Total Bytes	1988	7580	10844	24414
Mean Bytes	142	140.37	139.026	173.149
Mean NW Utilization Sniffer MB/sec. (%)	1.76643	1.48481	1.95987	0.713475
Mean NW Utilization Actual Bytes/sec.	19605.5	1681.87	1044.17	2085.45
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	14	50	66	0
Set Attributes	0	0	8	0
Root	0	0	0	0

D.1 Slate Office Automation Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Lookup	0	0	0	0
Read Link	0	4	4	0
Read	0	0	0	0
Write Cache	0	0	0	0
Write	0	0	0	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	0	0	0	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	0	0	0	0
After application start (test1d)				
Frame Count	2989	3098	3199	2048
Total Delta Time	13.8958	17.1839	15.6195	47.4725
Mean Delta Time	0.00465054	0.00554856	0.00488415	0.0231913
Total Bytes	3186150	3254316	3357014	718402
Mean Bytes	1065.96	1050.46	1049.39	350.782
Mean NW Utilization Sniffer MB/sec. (%)	29.688	32.8566	28.5494	2.86024
Mean NW Utilization Actual Bytes/sec.	229289	189382	214925	15133

D.1 Slate Office Automation Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	129	142	155	0
Set Attributes	0	0	0	0
Root	0	0	0	0
Lookup	211	219	231	0
Read Link	28	60	61	0
Read	594	567	607	0
Write Cache	0	0	0	0
Write	0	0	18	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	22	17	22	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	1790	1825	1879	0
After application completion (test1e)				
Frame Count	1254	1384	2787	13338
Total Delta Time	171.816	148.561	166.847	312.034
Mean Delta Time	0.137124	0.10742	0.0598875	0.0233961

D.1 Slate Office Automation Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Total Bytes	1132174	1242662	2919520	5251583
Mean Bytes	902.85	897.877	1047.55	393.731
Mean NW Utilization Sniffer MB/sec. (%)	18.7571	18.2817	25.1538	1.71632
Mean NW Utilization Actual Bytes/sec.	6589.46	8364.65	17498.2	16830.1
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	176	199	204	0
Set Attributes	8	8	7	0
Root	0	0	0	0
Lookup	117	116	113	0
Read Link	20	42	41	0
Read	150	176	370	0
Write Cache	0	0	0	0
Write	103	104	311	0
Create	8	8	7	0
Remove	0	0	0	0
Rename	4	4	4	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	0	0	0	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	614	670	1585	0

D.1 Slate Office Automation Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
After Logout (test1f)				
Frame Count	114	537	905	743
Total Delta Time	13.4008	16.5844	14.3003	31.521
Mean Delta Time	0.118591	0.030941	0.0158189	0.0424811
Total Bytes	22864	435850	769054	171849
Mean Bytes	200.561	811.639	849.783	231.291
Mean NW Utilization Sniffer MB/sec. (%)	1.57105	22.4622	19.7375	2.49149
Mean NW Utilization Actual Bytes/sec.	1706.17	26280.7	53778.9	5451.89
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	104	145	176	0
Set Attributes	0	0	23	0
Root	0	0	0	0
Lookup	2	5	21	0
Read Link	0	41	51	0
Read	2	88	200	0
Write Cache	0	0	0	0
Write	2	4	8	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0

D.1 Slate Office Automation Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Read Dir	0	0	0	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	4	232	386	0

D.1.1 Slate Office Automation vmstat Data

Test	Datafull	Dataless	Diskless	X-Terminal
CLIENT RESULTS				
After Login				
swap ins	130	55	45	70
swap outs	0	0	0	0
pages swapped in	100	44	36	56
pages swapped out	108	84	88	84
page ins	7	12	8	10
page outs	0	0	0	0
pages paged in	11	21	14	11
pages paged out	0	0	0	0
After starting mwm				
swap ins	180	100	90	130
swap outs	0	0	0	0
pages swapped in	144	80	72	104
pages swapped out	156	128	124	116
page ins	122	127	123	113
page outs	0	0	0	0
pages paged in	240	250	243	228
pages paged out	0	0	0	0
After application completion				
swap ins	395	315	285	475
swap outs	0	0	0	0
pages swapped in	312	256	228	384
pages swapped out	332	304	284	420
page ins	665	693	764	533
page outs	20	20	155	23
pages paged in	1298	1427	1389	1122
pages paged out	317	375	303	302

D.1.1 Slate Office Automation vmstat Data

Test	Datafull	Dataless	Diskless	X-Terminal
SERVER RESULTS				
After Login				
swap ins	10	45	45	50
swap outs	0	0	0	0
pages swapped in	8	36	36	40
pages swapped out	24	48	32	40
page ins	0	0	9	10
page outs	0	0	0	0
pages paged in	0	0	9	11
pages paged out	0	0	0	0
After starting mwm				
swap ins	50	110	100	110
swap outs	0	0	0	0
pages swapped in	40	88	80	88
pages swapped out	52	100	76	112
page ins	48	36	55	113
page outs	0	0	0	0
pages paged in	289	129	236	228
pages paged out	0	0	0	0
After clicking in xterm				
swap ins	90	130	135	160
swap outs	0	0	0	0
pages swapped in	76	104	108	128
pages swapped out	88	116	104	144
page ins	48	36	55	113
page outs	0	0	0	0
pages paged in	289	129	236	228
pages paged out	0	0	0	0

D.1.1 Slate Office Automation vmstat Data

Test	Datafull	Dataless	Diskless	X-Terminal
After application start				
swap ins	130	235	225	215
swap outs	0	0	0	0
pages swapped in	108	188	180	176
pages swapped out	120	200	176	224
page ins	219	115	305	450
page outs	0	2	55	0
pages paged in	1137	300	1171	926
pages paged out	0	47	160	0
After application completion				
swap ins	255	325	350	430
swap outs	0	0	0	0
pages swapped in	204	280	280	348
pages swapped out	220	292	276	368
page ins	257	154	418	532
page outs	0	46	179	23
pages paged in	1232	400	1448	1121
pages paged out	0	135	400	302
After Logout				
swap ins	300	385	380	550
swap outs	0	0	0	0
pages swapped in	240	308	308	440
pages swapped out	252	320	312	444
page ins	257	154	468	543
page outs	0	53	185	24
pages paged in	1232	400	1531	1144
pages paged out	0	205	411	304

D.2 Oilstock Map Graphics Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
After Login (test2a)				
Frame count	149	243	680	199
Total Delta Time	3.8736	17.6107	6.3275	10.6403
Mean Delta Time	0.026173	0.0727715	0.00931885	0.0537389
Total Bytes	45330	67806	525612	26602
Mean Bytes	304.228	279.037	772.959	133.678
Mean NW Utilization Sniffer MB/sec. (%)	21.8183	4.73753	30.214	12.9204
Mean NW Utilization Actual Bytes/sec.	11702.3	3850.27	83067.9	2500.12
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	44	72	76	0
Set Attributes	0	0	3	0
Root	0	0	0	0
Lookup	66	98	132	0
Read Link	0	20	22	0
Read	7	6	78	0
Write Cache	0	0	0	0
Write	0	2	2	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0

D.2 Oilstock Map Graphics Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Remove Dir	0	0	0	0
Read Dir	16	26	45	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	13	17	274	0
After starting mwm (test2b)				
Frame count	887	1017	1139	739
Total Delta Time	17.132	14.7787	22.5208	26.8555
Mean Delta Time	0.0193363	0.014546	0.0197898	0.0363896
Total Bytes	1012858	1061794	1138994	130136
Mean Bytes	1141.89	1044.05	999.995	176.097
Mean NW Utilization Sniffer MB/sec. (%)	64.2062	27.6898	26.2956	15.1214
Mean NW Utilization Actual Bytes/sec.	59120.8	71846.2	50575.2	4845.79
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	28	83	113	0
Set Attributes	0	0	8	0
Root	0	0	0	0
Lookup	30	57	80	0
Read Link	0	12	16	0
Read	115	184	201	0
Write Cache	0	0	0	0

D.2 Oilstock Map Graphics Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Write	2	0	2	0
Create	2	2	2	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	18	26	33	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	574	592	630	0
After clicking in xterm (test2c)				
Frame count	21	59	80	144
Total Delta Time	0.6614	17.8253	10.1241	12.551
Mean Delta Time	0.03307	0.307333	0.128153	0.0877692
Total Bytes	10650	15958	11128	24644
Mean Bytes	507.143	270.475	139.1	171.139
Mean NW Utilization Sniffer MB/sec. (%)	31.6114	3.50475	1.53925	15.0563
Mean NW Utilization Actual Bytes/sec.	16102.2	895.244	1099.16	1963.51

D.2 Oilstock Map Graphics Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	14	48	68	0
Set Attributes	0	0	8	0
Root	0	0	0	0
Lookup	0	0	0	0
Read Link	0	4	4	0
Read	2	2	0	0
Write Cache	0	0	0	0
Write	0	0	0	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	0	0	0	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	5	5	0	0
After application start (test2d)				
Frame count	1452	1587	1587	1746
Total Delta Time	8.4765	7.8668	7.6545	19.431
Mean Delta Time	0.00584183	0.00496015	0.00482629	0.0111352

D.2 Oilstock Map Graphics Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Total Bytes	1516692	1587910	1587910	342436
Mean Bytes	1044.55	1000.57	1000.57	196.126
Mean NW Utilization Sniffer MB/sec. (%)	59.3639	27.7809	27.3998	16.4386
Mean NW Utilization Actual Bytes/sec.	178929	201850	207448	17623.2
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	86	79	82	0
Set Attributes	0	0	0	0
Root	0	0	0	0
Lookup	144	135	137	0
Read Link	0	77	77	0
Read	196	307	302	0
Write Cache	0	0	0	0
Write	0	0	0	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	2	2	2	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	851	886	886	0

D.2 Oilstock Map Graphics Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
During application execution (test2e)				
Frame count	2086	2098	2686	23168
Total Delta Time	95.3875	103.304	100.813	223.082
Mean Delta Time	0.0457494	0.0492626	0.0375469	0.00962929
Total Bytes	2475340	2477000	3193664	12374392
Mean Bytes	1186.64	1180.65	1189	534.116
Mean NW Utilization Sniffer MBB/sec. (%	64.9763	14.4375	18.9854	2.67764
Mean NW Utilization Actual Bytes/sec.	25950.4	23977.9	31679	55470.2
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	84	90	91	0
Set Attributes	0	0	0	0
Root	0	0	0	0
Lookup	8	7	8	0
Read Link	0	2	0	0
Read	290	548	567	0
Write Cache	0	0	0	0
Write	0	0	130	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0

D.2 Oilstock Map Graphics Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Read Dir	0	0	0	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	1417	1417	1822	0
After application completion (test2f)				
Frame count	689	750	2065	22734
Total Delta Time	151.856	155.73	162.687	499.132
Mean Delta Time	0.220721	0.207917	0.0788214	0.0219563
Total Bytes	776422	819604	2402354	12746856
Mean Bytes	1126.88	1092.81	1163.37	560.696
Mean NW Utilization Sniffer MB/sec. (%)	13.5046	14.8557	25.3046	3.58968
Mean NW Utilization Actual Bytes/sec.	5112.88	5262.98	14766.7	25538.1
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	56	75	78	0
Set Attributes	0	0	0	0
Root	0	0	0	0
Lookup	7	8	8	0
Read Link	0	6	8	0
Read	173	181	354	0
Write Cache	0	0	0	0
Write	0	0	192	0
Create	0	0	0	0

D.2 Oilstock Map Graphics Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	0	0	0	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	441	461	1327	0
After Logout (test2g)				
Frame count	282	484	968	744
Total Delta Time	13.5361	16.5476	21.1773	29.0071
Mean Delta Time	0.0481712	0.03426	0.0219	0.0390405
Total Bytes	232108	360692	840168	171637
Mean Bytes	823.078	745.231	867.942	230.695
Mean NW Utilization Sniffer MB/sec. (%)	15.4932	10.8894	18.9895	2.49316
Mean NW Utilization Actual Bytes/sec.	17147.3	21797.2	39673	5917.07
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	94	140	167	0
Set Attributes	0	0	21	0

D.2 Oilstock Map Graphics Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Root	0	0	0	0
Lookup	2	6	21	0
Read Link	0	39	49	0
Read	54	94	224	0
Write Cache	0	0	0	0
Write	2	2	6	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	0	0	0	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	122	184	424	0

D.2.1 Oilstock Map Graphics vmstat Data

Test	Datafull	Dataless	Diskless	X-Terminal
CLIENT RESULTS				
After Login				
swap ins	55	80	45	70
swap outs	0	0	0	0
pages swapped in	40	68	36	52
pages swapped out	68	76	92	68
page ins	7	7	687	0
page outs	0	0	0	0
pages paged in	22	11	1312	0
pages paged out	0	0	0	0
After starting mwm				
swap ins	100	145	90	125
swap outs	0	0	0	0
pages swapped in	76	116	72	96
pages swapped out	120	128	128	112
page ins	123	122	802	0
page outs	0	0	0	0
pages paged in	253	240	1541	0
pages paged out	0	0	0	0
After application completion				
swap ins	290	435	285	495
swap outs	0	0	0	0
pages swapped in	228	348	228	392
pages swapped out	276	364	288	436
page ins	755	767	1532	210
page outs	20	23	171	58
pages paged in	1542	1568	2891	379
pages paged out	358	413	337	394

D.2.1 Oilstock Map Graphics vmstat Data

Test	Datafull	Dataless	Diskless	X-Terminal
SERVER RESULTS				
After Login				
swap ins	0	45	40	50
swap outs	0	0	0	0
pages swapped in	0	36	32	36
pages swapped out	20	56	44	48
page ins	0	14	14	0
page outs	0	0	0	0
pages paged in	0	45	16	0
pages paged out	0	0	0	0
After starting mwm				
swap ins	40	85	80	105
swap outs	0	0	0	0
pages swapped in	36	68	64	80
pages swapped out	48	88	76	96
page ins	0	64	65	0
page outs	5	0	0	0
pages paged in	0	338	305	0
pages paged out	41	0	0	0
After clicking in xterm				
swap ins	90	130	105	140
swap outs	0	0	0	0
pages swapped in	72	104	84	108
pages swapped out	84	128	96	128
page ins	0	64	65	0
page outs	5	0	0	0
pages paged in	0	338	305	0
pages paged out	41	0	0	0
After application start				

D.2.1 Oilstock Map Graphics vmstat Data

Test	Datafull	Dataless	Diskless	X-Terminal
swap ins	125	210	170	195
swap outs	0	0	0	0
pages swapped in	100	168	140	152
pages swapped out	112	192	152	204
page ins	99	160	165	0
page outs	38	0	0	2
pages paged in	444	778	754	0
pages paged out	436	0	0	9
During application execution				
swap ins	210	325	300	330
swap outs	0	0	0	0
pages swapped in	168	260	240	260
pages swapped out	180	280	252	296
page ins	208	333	325	106
page outs	59	22	105	43
pages paged in	1484	1984	1880	175
pages paged out	582	316	292	320
After application completion				
swap ins	315	440	450	450
swap outs	0	0	0	0
pages swapped in	252	352	360	364
pages swapped out	268	372	372	400
page ins	229	355	419	202
page outs	61	44	210	58
pages paged in	1598	2102	2190	366
pages paged out	595	511	500	394
After Logout				
swap ins	360	470	490	565
swap outs	0	0	0	0

D.2.1 Oilstock Map Graphics vmstat Data

Test	Datafull	Dataless	Diskless	X-Terminal
pages swapped in	288	380	392	448
pages swapped out	300	400	404	452
page ins	235	387	471	267
page outs	65	45	216	58
pages paged in	1604	2187	2298	482
pages paged out	672	512	509	394

D.3 Paragon Image Processing Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
After Login (test3a)				
Frame Count	128	227	350	197
Total Delta Time	1.6069	5.433	16.0027	5.164
Mean Delta Time	0.0126528	0.0240398	0.045853	0.0263469
Total Bytes	34680	65382	108132	26540
Mean Bytes	270.938	288.026	308.949	134.721
Mean NW Utilization Sniffer MB/sec. (%)	3.50047	4.98445	6.55163	0.83401
Mean NW Utilization Actual Bytes/sec.	21581.9	12034.2	6757.11	5139.43
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	30	57	97	0
Set Attributes	0	0	4	0
Root	0	0	0	0
Lookup	66	97	141	0
Read Link	0	20	26	0
Read	6	6	6	0
Write Cache	0	0	0	0
Write	0	0	4	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0

D.3 Paragon Image Processing Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Read Dir	18	26	35	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	8	17	32	0
After starting mwm (test3b)				
Frame Count	903	1000	1146	746
Total Delta Time	19.3532	18.4574	22.93	24.5959
Mean Delta Time	0.0214559	0.0184759	0.0200262	0.0330146
Total Bytes	1030314	1067048	1101648	130498
Mean Bytes	1140.99	1067.05	961.298	174.93
Mean NW Utilization Sniffer MB/sec. (%)	34.4501	28.4175	25.71	1.31621
Mean NW Utilization Actual Bytes/sec.	53237.4	57811.4	48044	5305.68
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	29	62	151	0
Set Attributes	0	0	8	0
Root	0	0	0	0
Lookup	30	57	79	0
Read Link	0	12	15	0
Read	161	186	190	0
Write Cache	0	0	0	0
Write	0	0	2	0
Create	2	2	2	0

	Datafull	Dataless	Diskless	X-Terminal
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	18	27	34	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	584	597	605	0
After clicking in xterm (test3c)				
Frame Count	14	54	80	143
Total Delta Time	0.1017	8.0438	14.0621	13.5467
Mean Delta Time	0.00782308	0.15177	0.178001	0.0953993
Total Bytes	1988	7580	11128	24464
Mean Bytes	142	140.37	139.1	171.077
Mean NW Utilization Sniffer MB/sec. (%)	1.77071	1.95241	1.69625	0.780839
Mean NW Utilization Actual Bytes/sec.	19547.7	942.341	791.347	1805.9
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	14	50	68	0
Set Attributes	0	0	8	0
Root	0	0	0	0

D.3 Paragon Image Processing Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Lookup	0	0	0	0
Read Link	0	4	4	0
Read	0	0	0	0
Write Cache	0	0	0	0
Write	0	0	0	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	0	0	0	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	0	0	0	0
After application start (test3d)				
Frame Count	1056	1131	1152	3637
Total Delta Time	7.1817	7.7405	7.4845	37.905
Mean Delta Time	0.0068073	0.00685	0.00650261	0.0104249
Total Bytes	1164984	1222318	1226976	1209134
Mean Bytes	1103.2	1080.74	1065.08	332.454
Mean NW Utilization Sniffer MB/sec. (%)	38.6492	34.7123	35.8952	4.37534
Mean NW Utilization Actual Bytes/sec	162216	157912	163936	31899.1

	Datafull	Dataless	Diskless	X-Terminal
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	79	94	102	0
Set Attributes	0	0	0	0
Root	0	0	0	0
Lookup	32	36	38	0
Read Link	0	8	16	0
Read	205	219	225	0
Write Cache	0	0	0	0
Write	0	0	0	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	16	23	25	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	660	689	690	0
After application completion (test3e)				
Frame Count	855	957	1249	38986
Total Delta Time	288.884	329.049	374.962	464.601
Mean Delta Time	0.338271	0.344194	0.30045	0.011917

D.3 Paragon Image Processing Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Total Bytes	745862	766830	1088278	6508030
Mean Bytes	872.353	801.285	871.319	166.932
Mean NW Utilization Sniffer MB/sec. (%)	21.6643	20.2804	19.4906	2.4387
Mean NW Utilization Actual Bytes/sec.	2581.88	2330.44	2902.37	14007.783
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	140	192	214	0
Set Attributes	0	0	0	0
Root	0	0	0	0
Lookup	102	126	135	0
Read Link	0	6	18	0
Read	56	58	68	0
Write Cache	0	0	0	0
Write	91	102	155	0
Create	18	20	19	0
Remove	12	12	12	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0
Read Dir	6	6	6	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	403	409	587	0

	Datafull	Dataless	Diskless	X-Terminal
After Logout (test3f)				
Frame Count	112	209	306	739
Total Delta Time	11.4271	15.5701	21.9419	29.5556
Mean Delta Time	0.102947	0.0748562	0.0719407	0.0400482
Total Bytes	22588	43026	51680	171375
Mean Bytes	201.679	205.866	168.889	231.901
Mean NW Utilization Sniffer MB/sec. (%)	1.51705	2.39483	2.08484	2.50953
Mean NW Utilization Actual Bytes/sec.	1976.7	2763.37	2355.31	5798.39
NFS Procedure Totals				
Null	0	0	0	0
Get Attributes	104	148	190	0
Set Attributes	0	0	24	0
Root	0	0	0	0
Lookup	2	6	24	0
Read Link	0	42	52	0
Read	0	2	4	0
Write Cache	0	0	0	0
Write	2	2	8	0
Create	0	0	0	0
Remove	0	0	0	0
Rename	0	0	0	0
Link	0	0	0	0
Symbolic Link	0	0	0	0
Make Dir	0	0	0	0
Remove Dir	0	0	0	0

D.5 Paragon Image Processing Sniffer Data

	Datafull	Dataless	Diskless	X-Terminal
Read Dir	0	0	0	0
Stata File Sys	0	0	0	0
N Proc	0	0	0	0
UDP Continuation	4	9	4	0

D.3.1 Paragon Image Processing vmstat Data

Test	Datafull	Dataless	Diskless	X-Terminal
CLIENT RESULTS				
After Login				
swap ins	65	60	90	40
swap outs	0	0	0	0
pages swapped in	52	48	76	32
pages swapped out	60	56	80	76
page ins	7	7	6	0
page outs	0	0	0	0
pages paged in	11	11	10	0
pages paged out	0	0	0	0
After starting mwm				
swap ins	105	115	135	100
swap outs	0	0	0	0
pages swapped in	88	92	112	80
pages swapped out	96	108	120	104
page ins	122	122	121	0
page outs	0	0	0	0
pages paged in	240	240	239	0
pages paged out	0	0	0	0
After application completion				
swap ins	275	360	340	560
swap outs	0	0	0	0
pages swapped in	224	288	272	448
pages swapped out	236	304	284	492
page ins	292	297	299	48
page outs	0	7	35	35
pages paged in	576	586	588	98
pages paged out	0	119	70	324

D.3.1 Paragon Image Processing vmstat Data

Test	Datafull	Dataless	Diskless	X-Terminal
SERVER RESULTS				
After Login				
swap ins	35	35	45	15
swap outs	0	0	0	0
pages swapped in	28	28	36	28
pages swapped out	40	48	36	36
page ins	7	13	0	0
page outs	1	0	0	0
pages paged in	7	14	0	0
pages paged out	1	0	0	0
After starting mwm				
swap ins	75	110	120	80
swap outs	0	0	0	0
pages swapped in	60	88	96	64
pages swapped out	72	100	96	80
page ins	22	63	48	0
page outs	1	0	0	0
pages paged in	41	307	289	0
pages paged out	1	0	0	0
After clicking in xterm				
swap ins	120	135	165	130
swap outs	0	0	0	0
pages swapped in	96	108	132	104
pages swapped out	108	120	128	128
page ins	22	63	48	0
page outs	1	0	0	0
pages paged in	41	307	289	0
pages paged out	1	0	0	0

D.3.1 Paragon Image Processing vmstat Data

Test	Datafull	Dataless	Diskless	X-Terminal
After application start				
swap ins	155	210	240	185
swap outs	0	0	0	0
pages swapped in	124	168	192	148
pages swapped out	140	180	188	188
page ins	67	114	97	17
page outs	1	0	0	0
pages paged in	311	602	580	32
pages paged out	1	0	0	0
After application completion				
swap ins	305	385	415	500
swap outs	0	0	0	0
pages swapped in	244	308	332	400
pages swapped out	256	320	328	440
page ins	90	122	105	46
page outs	55	0	35	35
pages paged in	419	655	633	90
pages paged out	163	0	70	324
After Logout				
swap ins	345	420	450	625
swap outs	0	0	0	0
pages swapped in	276	336	360	504
pages swapped out	288	348	360	512
page ins	90	122	105	64
page outs	59	0	35	35
pages paged in	419	655	633	126
pages paged out	169	0	70	324