CONF-900557--12

UCRL- JC-103311, Rev. 1
PREPRINT

Received by OSTI

JUN 29 1990

MTX Data Acquisition and Analysis Computer Network

D. N. Butner, T. A. Casper, M. D. Brown, M. Drlik,
W. H. Meyer, and J. M. Moller

Lawrence Livermore National Laboratory
Livermore, CA

Eighth Topical Conference on High-Temperature Plasma
Diagnostics

Hyannis, MA
May 6-10, 1990

June 1990

Lawrence Livermore National Laboratory

MTX Data Acquisition and Analysis Computer Network*,

D.N. Butner, T.A. Casper, M.D. Brown, M. Drlik, W.H. Meyer, and J.M. Moller

Lawrence Livermore National Laboratory, University of California,
Livermore, CA 94550

UCRL-JC--103311-Rev.1

DE90 013066

Abstract

For the MTX experiment, we use a network of computers for plasma diagnostic data acquisition and analysis. This multivendor network employs VMS, UNIX, and BASIC based computers connected in a local area Ethernet network. Some of the data is acquired directly into a VAX/VMS computer cluster over a fiber-optic serial CAMAC highway. Several HP-Unix workstations and HP-BASIC instrument control computers acquire and analyze data for the more data intensive or specialized diagnostics. The VAX/VMS system is used for global analysis of the data and serves as the central data archiving and retrieval manager. Shot synchronization and control of data flow are implemented by task-to-task message passing using our interprocess communication system. The system has been in operation during our initial MTX tokamak and FEL experiments; it has operated reliably with data rates typically in the range of 5 megabytes/shot without limiting the experimental shot rate.

*Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under contract number W-7405-ENG-48.

I. A General Purpose Data Acquisition System

We have developed a shot based, general purpose data acquisition and analysis system for experiments at LLNL. The major functions this system provides include collection of experimental data using a heterogeneous network of computers, distributed intershot processing over the network, automated, centralized archiving of all data, and easy access to data for users and programs. This data acquisition system is quite flexible with hardware and software configurations easily rearranged and distributed among the various computers as required.

The keys to our system's generality are its shot coordination scheme, its ability to communicate transparently among diverse computers and processes, and adherence to standardization where possible. The shot coordination scheme relies on the existence of software subsystems and a shot model that specifies a logical progression of states during acquisition and processing. The shot model is implemented by a supervisory program with diagnostic subsystem programs executing concurrently as a state machine. We achieved transparent communication by developing a portable Interprocess Communication System (IPCS) software package for passing messages and data between tasks that reside anywhere in a network of computers.[1]

The system's flexibility includes its ability to run on a heterogeneous network of computers. This flexibility provides for easy expansion and was achieved through the use of modular programming techniques, IPCS, and portable computer languages. Use of IPCS allows the external interface of a program to remain the same even when internal functional changes are made. Most of our programs are written in the C language, which allows us portability between VMS and UNIX computer systems. While our existing UNIX workstation inventory consists only of HPUX systems, much of our software has been run on SUN workstations and is readily portable to most UNIX based platforms.

One advantage of expandability is the ability to add new computers, diagnostics, and processing and to support multiple experiments. Reliance on commercial networking and the IPCS communication standard simplifies the integration of these new subsystems. Since we can move programs transparently around the network, our ability to add new computers to the network provides immediate use of the additional computational power. In addition, the system's flexibility and expandability allow us to distribute the intershot processing among the various computers in our network.

An important feature of the system is "transparent data access", which means that a program running anywhere in the computer network can access data stored anywhere in the network without knowing where the data are. To support transparent data access, we have a centralized data base that contains the current location information for all data whether on-line or off-line. We use a commercial data base manager to maintain the data location information. We have also integrated a commercial archiving system into the data transparency scheme. Transparent data access not only refers to the ability to find data anywhere, but also to the fact that subroutine calls to access the data and their arguments are invariant, regardless of location.

This general solution to data acquisition and analysis has been applied to the Microwave Tokamak Experiment (MTX)[2]. It has also recently begun to support operations on the Ring Accelerator Experiment (RACE) and the High Field Test Facility (HFTF) at LLNL. We will now concentrate on details specific to the MTX implementation.

II. Acquisition System Configuration

The MTX Data Acquisition System uses Digital Equipment Corporation (DEC) VAX computers located in our User Service Center (USC) running the VMS operating system, Hewlett-Packard UNIX (HPUX) workstations and HP-BASIC control computers, as indicated in Figure 1. Use of these computers takes advantage of the much of the hardware investment made during the Tandem Mirror Experiment (TMX) program at LLNL[3].

The USC VAX cluster consists of three computers with redundant HSC hardware and approximately 6 gigabytes of disk storage. SMAUG, an 8200, is the primary VAX data acquisition computer and is connected by a serial fiber optical highway to the MTX CAMAC hardware in the experimental area, approximately a quarter of a mile from the USC. LLL, an 8600, is used for automated intershot processing and plotting and interactive analysis. In addition, LLL must support the standard USC tasks required for office operations and NERSC connection. LLV, an 11/780, is used to maintain experiment status displays and to meet the needs of the administrative staff. Except for restrictions based on hardware connections (such as CAMAC only being available on SMAUG), it is easy to run our data acquisition and data processing programs on any of the three computers.

The HP computers, located both in the office building and in the experimental area, are connected to the USC via a fiber-optic local area Ethernet (LAN). Nine HPUX workstations are presently connected directly to the LAN and communicate with the VAX computers via the Internet TCP/IP network protocol. These autonomous workstations typically have 8Mbytes of memory and 200Mbytes local disc storage for the operating system and data. Processes running on these computers can communicate directly with processes running on the VAX computers via IPCS. Data acquired into files on the UNIX computers are easily moved to the VAX computers using FTP or IPCS messages.

We presently have seven HP-BASIC controllers in the acquisition system. Due to their lack of networking capabilities, these computers cannot be connected directly to the network. Instead, they are connected to a 400Mbyte shared disk system called the Shared Resource Manager (SRM) for data and message storage. A dedicated HPUX computer, DALE, acts as a communication gateway between the HP-BASIC computers and the other computers using files created on the SRM.

The DEC terminal servers located in the office building and in the experimental area allow users in either place to connect to the VAX computers. The TELNET function provides a direct connection between UNIX and VMS computers with FTP providing file transfers.

In the development of the MTX system, we integrated software from various sources: commercially developed software, software developed at national laboratories, and software developed at universities. We have also relied heavily on standardization and modularity for flexibility.

We use the MDS[4] data acquisition software developed at MIT for our VAX/VMS acquisition subsystem. It acquires data from CAMAC modules using the ORNL drivers and stores the data in files. MDS allows a diagnostician to specify the types of modules used in a system, the pre-shot and post-shot commands for each module, and the files into which the data from each acquired signal are stored. MDS utilizes a signal naming convention that has been adopted as our defacto standard.

Our HP computers allow for more complex interactive control of diagnostics than does MDS. Acquisition in the HPUX workstations utilizes VIEW[5], an LLNL-developed image processing software package, for analysis and database tools with our own IEEE-488 based CAMAC acquisition software. It utilizes a command based interface with English language syntax to provide acquisition and processing procedures. HP-BASIC acquisition was implemented by standardizing, upgrading and extending (to accommodate IPCS messages) our earlier versions of HP-BASIC acquisition software.

Each of these acquisition subsystems, MDS, HPUX, and HP-BASIC, are fully integrated into the MTX data system via IPCS. Most intershot processing is completed with either Interactive Data Language (IDL, a commercial product) on the VAX or with VIEW on the HPUX. Special purpose software is required for native analysis on the HP-BASIC computers. However, a significant portion of the data produced on these computers is stored in MDS data files as part of the intershot processing. The data is then available for further analysis on the VAX cluster.

III. System Operation

We show a diagram of the data acquisition software in Figure 2. The bubbles represent processes or sets of processes running on various computers. Many of the processes may run unmodified on any of several computers; many processes run with no source code changes on either VAX computers or HPUX computers. Most of the interprocess connections shown in Figure 2 are implemented using IPCS. Control server programs provide access to commercial codes or to processes that do not know about IPCS. For example, MDSCONTROL communicates with MDS using MDS events (VMS locks). MDSCONTROL thus provides an IPCS interface to MDS for communication with processes running on other computers. HPCONTROL provides a similar function for the HP-BASIC computers and provides IPCS message and data file transfer via the SRM.

The supervisory program, SHOTSYNC, which implements the shot model state machine, provides the shot coordination interface between the various diagnostic subsystem control programs. HPCONTROL on DALE, MDSCONTROL on the VAX computers, and the acquisition processes on the HPUX computers translate SHOTSYNC's state transitions into actions for their respective subsystems. A supervisory data base program, DBSERVER, coordinates the collection of data location and shot configuration information.

SHOTSYNC accepts commands from the experiment leader via the SHOTLEADER user interface program. The shot supervisory program compiles a list of diagnostic subsystems, supervisory data bases, and various status-monitoring programs and coordinates their activities according to the state diagram. Much of this information is displayed by the SHOTLEADER program display as is shown in Figure 3. SHOTSYNC has no prior knowledge of what subsystems will be part of a shot. Instead, we use a dynamic check-in technique by which programs to be coordinated notify SHOTSYNC that they need to receive shot coordination messages. SHOTSYNC maintains a list of checked-in diagnostic subsystems, and each subsystem remains active until it dies or is checked out by the experiment leader or diagnostician. The diagnostic subsystem status information is visible in the display shown in Figure 3. For each shot, SHOTSYNC supplies the list of participants and other shot information as a package to the data base server at the times required by the state diagram. Thus we automatically record the actual shot configuration. Database utilities are available for subsequent interrogation to determine the experiment configuration and data location and availability for a given shot. A diagnostic subsystem or any other process that wants to synchronize its operations with the shot cycle must send a check-in message to SHOTSYNC and must respond as appropriate to SHOTSYNC's shot coordination messages.

The MASTERTIMER program provides the interface between SHOTSYNC and the MTX timing system; it senses the times at which system hardware events occur. It is the single interface between the data acquisition system and the hardware timing system. The SHOTLEADER status display, Figure 3, shows the existing state of acquisition and provides the necessary interface software for control as the data system proceeds through its shot model.

IV. Data Flow

The data subsystems (MDS, HPUX, and HP-BASIC), upon completion of data acquisition into their local data bases, notify DBSERVER of the files generated. DBSERVER provides the IPCS interface to an ACCENT-R database on the VAX containing all the shot status information. A generic store/retrieve programming interface provides the transparent data access. Data are referred to by logical name and there is a standard subroutine call interface. Nonlocal data are retrieved by interprocess communications with a database server, DATA_DAEMON. At present DATA_DAEMON knows only how to store into and retrieve data from MDS data bases. However, since DATA_DAEMON is an IPCS-using task, it can be reached from any of the networked computers. Presently, VIEW processes (on HPUX systems) utilize this method for efficient exchange of data with MDS in support of automated intershot processing tasks combining VMS and UNIX acquired data. For intershot processing in VMS using IDL procedures, UNIX data is typically first written to MDS with the help of DATA_DAEMON. Future versions will handle other data subsystems and will perform more complex searches, thus eliminating such unnecessary intermediate steps as the writing of data to MDS above.

Currently, some highly processed data from the HP computers is stored in MDS files shortly after a shot occurs. The raw data from the HP computers is stored locally during operations. Later (i.e. overnight) it is compressed and moved to files on the VAX cluster via FTP for automatic archival. All raw data and some processed data from both the VAX computers and the HP computers are archived using a commercial archive product that provides for retrieval of data by filename.

Due to the parallel architecture of this system, the overall shot cycle time is determined by the slowest element. Presently, the dedicated VAX8200 running MDS requires approximately 1.5 minutes to acquire about 3 Mbytes of data per shot. An additional 2 Mbytes of processed data are received from workstations asynchronously. Since UNIX and BASIC workstations have (so far) remained dedicated to specific instruments, the actual cycle time utilized for acquisition is minimal with the majority of the intershot time used for analysis. For example, the FIR interferometry system requires about 15 seconds for acquisition of data from 15 channels at 8 kwords/channel. The line averaged density is calculated, displayed locally, and a copy sent to MDS within 30 seconds of the shot. Local data storage on disc and hard copy plots (used for data verification and searches) requires an additional 2 minutes. At the end of this 2.5 minute period, a "ready" signal is sent to the supervisory program thus allowing for another shot to start if all other systems are ready. This status information is displayed locally, Figure 3, from which any needed performance improvements are readily identified. About 2 minutes additional time is used for (interruptible) profile calculation and display. Any time-critical, uninterruptible work has been maintained at somewhat less than the cycle time determined by the cool-down time of the high field coils in MTX (> 3.5 minutes for tokamak operation at field strengths > 5T) . Any additional processing load is scheduled on the VAX8600 and is allowed to lag behind the current shot.

V. Summary and Future Work

Presently, our data volume is split about half from VMS and half from HP, most of which is HPUX. Many of the acquisition and processing tasks can run on either system with only special "commercial" applications confined to a particular computer. The modularity allows for smooth migration of selected portions of the system to other computers as might be desired to take advantage of the rapidly evolving workstation industry. The application of distributed data acquisition and analysis has proved quite useful for off-line diagnostic development and for integration of visitor's diagnostics into our acquisition system. The lack of single point catastrophic failures has provided a fairly robust system which can be operated (at reduced capacity) in the event of partial system failure. The reliance on commercial products, local area networking, and IPCS has given us a flexible and reliable system maintained by only a few people. We anticipate the possibility of evolution towards a full UNIX-based implementation in the future; this would, however, require replacement of several of our commercial products and of MDS with their UNIX equivalents.

We have designed extensions to the present system that will provide fully transparent access to all data by signal name (for compatibility with MDS), whether the data exists in a file on any computer or on an archive tape. The implementation depends heavily on the creation of data base data sets that define signal names and their related files and on data base server tasks that provide such information to any process running anywhere in the network. We designed our present system with these extensions in mind, thus providing for an evolutionary implementation. We are now concentrating on implementing additional utilities to facilitate the transparent data transfer, to provide more analysis tools, and to re-implement our journal file, a reduced resolution on-line database for multiple shot statistical analysis.

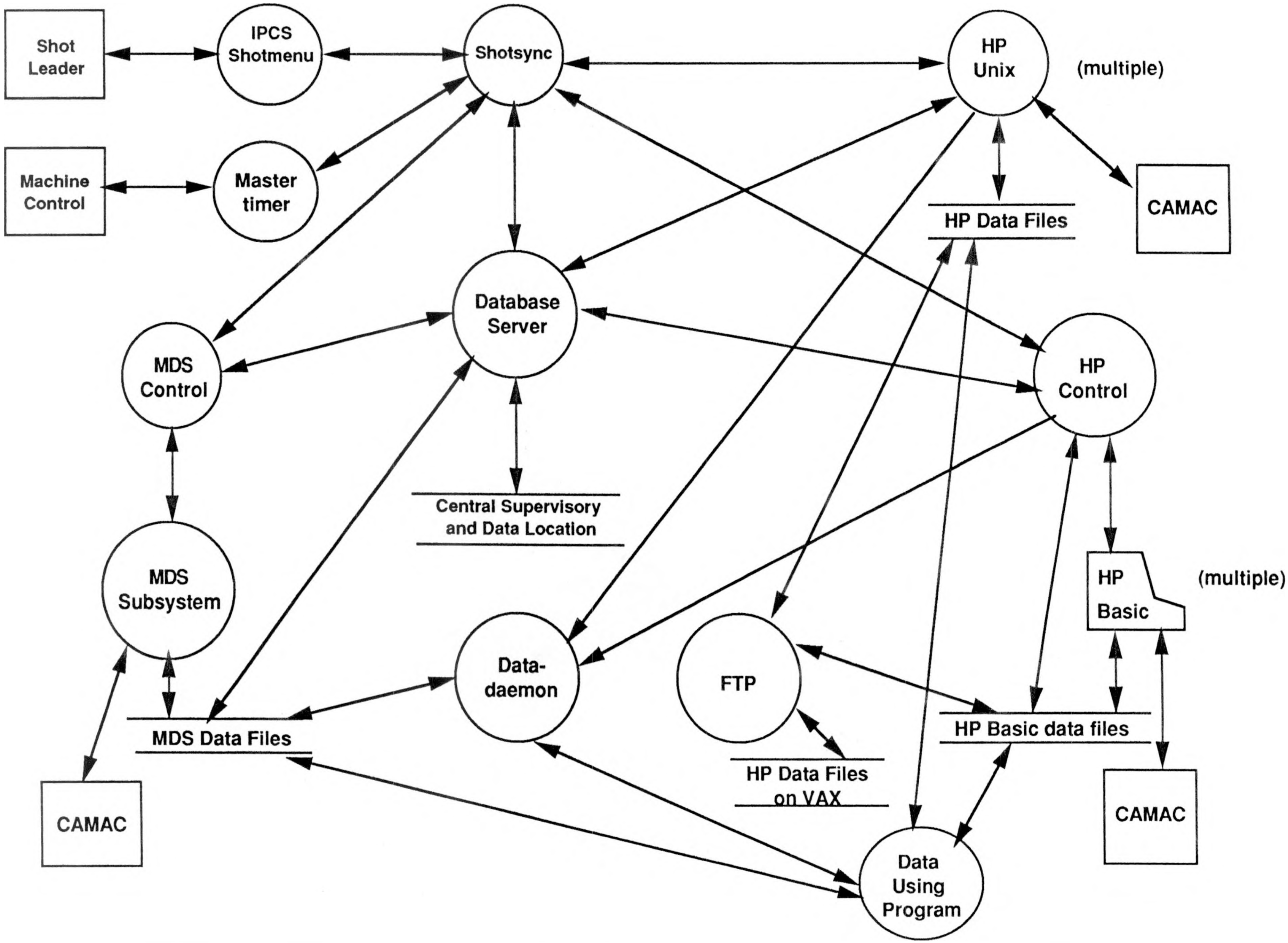
References

1. G.G. Preckshot and D.N. Butner, IEEE Trans. Nucl. Sci. **NS-34,** 858 (1987).
2. K.I. Thomassen, presented at the 14th Conference on Plasma Physics, St. Andrews, Scotland, 1-3 July 1987.
3. T.A. Casper, H. Bell, M. Brown, M. Gorvad, S. Jenkins, W. Meyer, J. Moller, and D. Perkins, Rev. Sci. Instrum. **57,** 1880 (1986).
4. T.W. Fredian and J.A. Stillerman, Rev. Sci. Instrum. **57,** 1907 (1986).
5. J.M. Brase, V.J. Miller, and M.A. Wieting, LLNL UCID-21368, March, 1988.

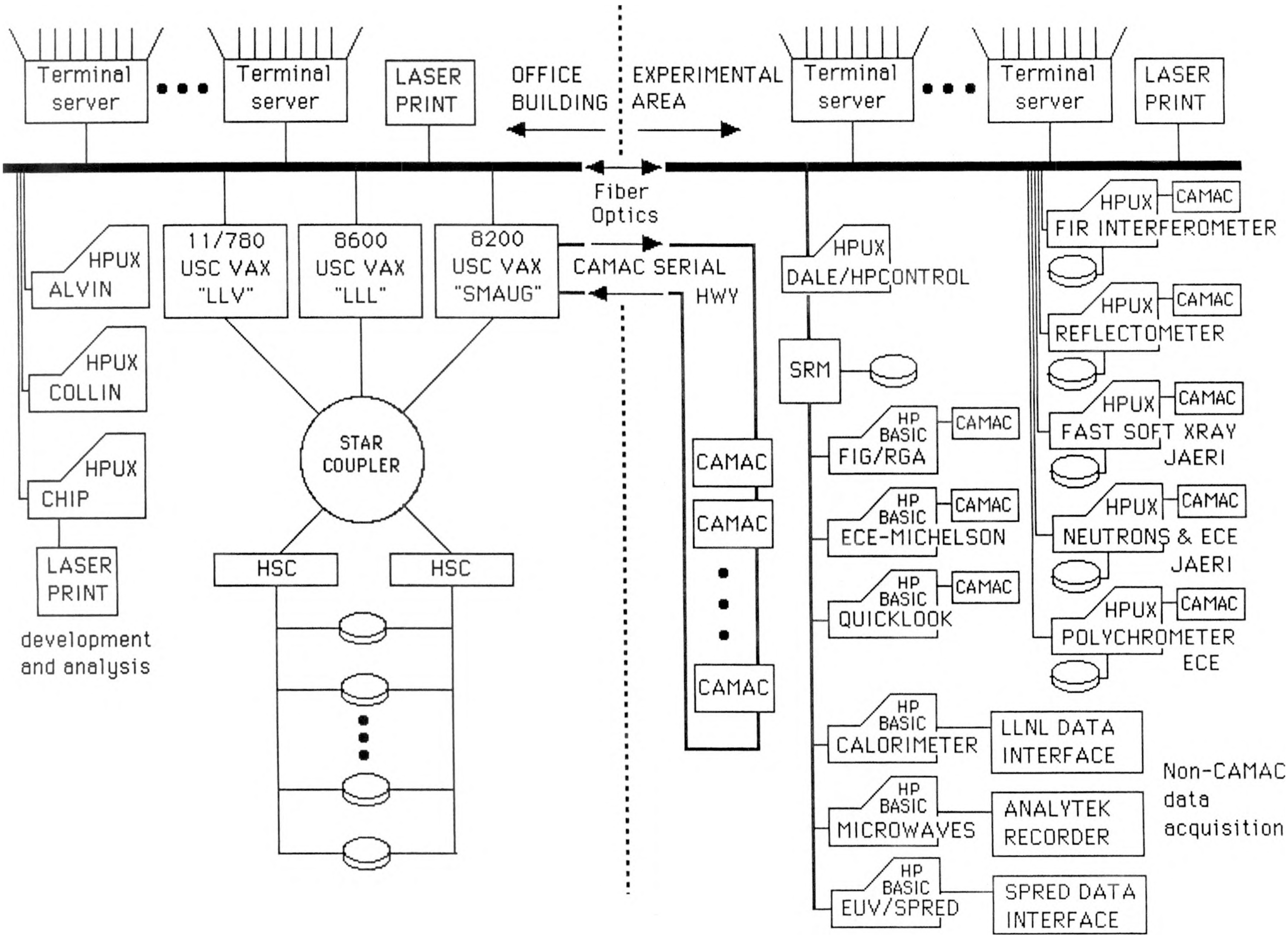
Figure Captions

Figure 1. Hardware configuration for the MTX data acquisition and analysis network.
Figure 2. Flow diagram of MTX data acquisition software; circles
represent tasks, arrows are communication/data messages, parallel lines indicate data files, and rectangles represent hardware (except possibly for the shot leader).
Figure 3. Typical state of the SHOTLEADER interface display; allowed interface functions appear in the left box while the right box shows the diagnostic configuration status (systems "checked in") and the present shot status for the MTX project.

```
--[ Shotleader ]-----------------------------------------------------------------
| Shot       Shot Plan      Logbook      Configure                       Quit     |
----------------------------------------------------------------------------------
  --[ Shot ]------ --[ Players ]-------------------------------------------------------
  |              | | PROJ: MTX                LOCAL TIME: 14:53:13   5/4         |
  | Start        | |  EXP: TF testing          SHOT TIME: 14:52:16   5/4         |
  | Hold         | |LEADR: Casper                   SHOT: 8214                   |
  | Continue     | |STATE: acquiring               EVENT: shot_over              |
  | Force        | |                                                             |
  | Abort        | |$$CASPER           Ready    LLL              Display         |
  |              | |ECE                BUSY     dale             Diagnostic      |
  | Leader       | |FFR                BUSY     simon            Diagnostic      |
  | Experiment   | |FIGR               BUSY     dale             Diagnostic      |
  ---------------- |FIR                BUSY     fir              Diagnostic      |
                   |JSX                BUSY     jaeri            Diagnostic      |
                   |MTX_DBSERVER       Ready    SMAUG            DBserver        |
                   |MTX_DBSERVER       Ready    SMAUG            Display         |
                   |MTX_DISPLAY_DMN    Ready    chip             Display         |
                   |MTX_MASTERTIMER    Ready    SMAUG            Diagnostic      |
                   |MTX_SMAUG_MDSCTL   BUSY     SMAUG            Diagnostic      |
                   |POLY               BUSY     dale             Diagnostic      |
                   |QCKL               BUSY     dale             Diagnostic      |
                   |SPRD               BUSY     dale             Diagnostic      |
                   -------------------------------------------------------------
```


Shot Leader
IPCS Shotmenu
Shotsync
HP Unix
(multiple)
Machine Control
Master timer
HP Data Files
CAMAC
Database Server
MDS Control
HP Control
Central Supervisory and Data Location
MDS Subsystem
HP Basic
(multiple)
Data-daemon
FTP
MDS Data Files
HP Basic data files
HP Data Files on VAX
CAMAC
CAMAC
Data Using Program

TAC 5/90 Fig.2



Terminal server
Terminal server
LASER PRINT
OFFICE BUILDING
EXPERIMENTAL AREA
Terminal server
Terminal server
LASER PRINT
Fiber Optics
HPUX
ALVIN
HPUX
COLLIN
HPUX
CHIP
LASER PRINT
development and analysis
11/780 USC VAX "LLV"
8600 USC VAX "LLL"
8200 USC VAX "SMAUG"
CAMAC SERIAL HWY
STAR COUPLER
HSC
HSC
CAMAC
CAMAC
CAMAC
HPUX
DALE/HPCONTROL
SRM
HP BASIC
FIG/RGA
CAMAC
HP BASIC
ECE-MICHELSON
CAMAC
HP BASIC
QUICKLOOK
CAMAC
HP BASIC
CALORIMETER
LLNL DATA INTERFACE
HP BASIC
MICROWAVES
ANALYTEK RECORDER
HP BASIC
EUV/SPRED
SPRED DATA INTERFACE
Non-CAMAC data acquisition
HPUX
CAMAC
FIR INTERFEROMETER
HPUX
CAMAC
REFLECTOMETER
HPUX
CAMAC
FAST SOFT XRAY JAERI
HPUX
CAMAC
NEUTRONS & ECE JAERI
HPUX
CAMAC
POLYCHROMETER ECE