Evolution to the Athena Workstation Model: An Overview of the Development Plan

by Jerome H. Saltzer

Project Athena will evolve to the model of computation described in section C of this Technical Plan by way of two intermediate phases.

The first intermediate phase (already accomplished at this writing) is deployment of off-the-shelf equipment available at the project outset from Athena's two industrial partners--minicomputer time-sharing systems and PC workstations--interconnected with a high-performance network. That equipment allows good approximations of various aspects of a workstation environment, and thus gives the M.I.T. community an early opportunity to develop applications for an environment much like the expected future one. It also provides experience in educational computing on a large scale. Phase I runs from 1984 through 1986.

The second intermediate phase, based on newer, Athena-specified equipment of the desired target configuration and capability, is a public workstation model. Supporting these public workstations are four new services: authentication, name resolution, mail drop, and storage. The only important difference between this intermediate phase and the final model is that the cost of available workstations is too high to permit student ownership, so they must instead be placed in public places. The technical consequence of this difference is that, except for experimental private installations, every workstation must be prepared to operate on behalf of any user. Phase II runs from 1986 through 1988.

The third, final phase of Athena is one in which a new generation of student-owned workstations begins to predominate, using the same (or more advanced versions of) services developed in phase II. Public workstations will continue to be deployed in laboratories, libraries, and departmental work areas, but private workstations will appear in fraternity and dormitory rooms. In addition, the range of applications and application-building tools will be continuously growing throughout the first two phases, and by the beginning of the third phase those tools should be more useful than before. Phase III begins at the scheduled end of Project Athena, in Fall, 1988. At that time, the operation of services and the continued deployment of workstations will become an ongoing responsibility of some permanent M.I.T. organization, probably part of M.I.T. Information Services.

This section of the Athena Technical Plan provides a high-level overview of the plan for evolution, sketching the range of things that must be developed and acquired. For each part of the Phase II and Phase III plans that requires significant development, a later section of the plan provides more detail.

This is an initial draft of the development plan. Some of the things described here have already been implemented, others have been agreed to but

not yet accomplished, while still others have been discussed little or not at all. Until this document reaches "approved" status, the reader should assume that individual details of the plan described here are questionable. In particular, until that time this document should not be cited as an authoritative source.

Certain underlying philosophies and approaches of Project Athena provide a set of ground rules under which to understand this plan. When the ground rules conflict with one another, their order here indicates roughly the priority in which to follow them:

1. The experiment that M.I.T. is undertaking in the use of ubiquitous networked workstations to improve education is the primary driver of Athena development projects, rather than technology opportunity or research opportunity. Development activities are appropriate only when there is a reasonable expectation that a faculty member teaching a class will want and use the result to improve education, or there is some other expected, beneficial effect on the education process.

2. The resulting system, including workstations, network, and services, must fit within education-sized budgets in competition with other worthy uses for the same funds. One can expect hardware technology advances to make computing equipment low-cost, but one must use design ingenuity to keep labor costs down. Thus the design of support and deployment systems must provide robust automatic operation with minimum intervention and with minimum expertise.

3. M.I.T. is not a software support organization; the development activities of Project Athena are temporary. Thus each development project must include a scenario under which the developed item evolves to acquire support from non-Athena sources or is eventually replaced with a vendor-supported version of the same function. In addition, to encourage both outside support and other workers to build software that is usable at M.I.T., our own developments should be designed as potential standards capable of being offered to others.

4. In order to minimize development risk and also to encourage export of Athena-developed ideas to the rest of the world, the several pieces of software developed by Project Athena should be modularly independent of one another. That is, unless technically infeasible it should be possible both to operate at M.I.T. and to export to other places any major subsystem or service independently of the others.

5. Because both the intellectual and monetary resources of Project Athena are limited when compared with the possible number of things that could be developed, we must minimize development by using software available from other sources whenever it is adequate.

1. PHASE I: THE OFF-THE-SHELF SYSTEM.

The equipment and software initially available from the two industrial partners is quite different, so in Phase I the plan builds upon two rather different sets of architectures, objectives, and opportunities. Installation of equipment is in clusters scattered across the campus, with each cluster

containing enough equipment to handle 20 to 40 active users.

1.1 HARDWARE AND SOFTWARE FROM DIGITAL EQUIPMENT CORPORATION.

The closest approximation to the intended workstation environment economically feasible on available Digital hardware at the time the project began was minicomputer time-sharing systems. Approximately 50 VAX 11/750 minicomputers have been installed, each with six to eight graphics-capable terminals. The Berkeley 4.2 (now 4.3) version of the UNIX [a trade mark of the American Telephone and Telegraph Company] operating system, and a collection of application packages obtained from various sources, are also part of this off-the-shelf environment.

This choice provides an opportunity to test the use of UNIX as a base for education applications and student use, and to develop service management tools and the eventual application programming environment.

1.2. HARDWARE AND SOFTWARE FROM IBM.

The closest approximation to the eventual workstation configuration available in the IBM catalog for Phase I was the PC/AT personal computer, with a Professional Graphics Display and 3COM Ethernet interface. Approximately 160 PC/AT's have been installed. The software environment consists of PC/DOS, plus a set of higher level languages and application packages chosen to simplify moving to an eventual UNIX environment. In addition, 105 PC/XT personal computers, mostly with Color Graphics Display, have been installed, primarily for use as laboratory experiment controllers and also as cluster print servers for the PC/AT's.

This choice provides an opportunity to test the use of individual, networked workstations, and obtain experience with workstation logistics management and software distribution. It also provides an early opportunity to use color, hardware support for 3-dimensional graphics, and the Graphical Kernel System graphics programming interface. Section Z1 of the Athena Technical Plan describes the AT workstation environment in detail.

1.3. OTHER HARDWARE

Each VAX 11/750 has disk drives with space for at least 650 Megabytes of storage. Ten of the 50 VAXes have an additional 600 Megabytes of storage and a tape drive, providing a place for backup of a cluster of up to seven other Vaxes. Each cluster is also equipped with one or more line printers and at least one medium capacity (20,000 pages per month) or small capacity (4000 pages per month) laser printer. Clusters of IBM PC/AT workstations have at least one PC/XT that manages a graphics dot matrix printer, accessible via network from the other workstations in that cluster.

Certain experiments that require specialized hardware are also underway or intended as part of Phase I. One cluster of VAX's uses water cooling, to explore the use of unairconditioned space. One VAX 11/785 provides an

opportunity to experiment with a high-performance computation server. Several PC/XT's are equipped with a Metrabyte data acquisition board and associated sensor/actuator links, for use in a laboratory experiment-controller configuration.

1.4. SOFTWARE DEVELOPED OR ACQUIRED DURING PHASE I

Phase I software development activities are concentrated in five areas: a window system, a user registration system, a laboratory support system, changes to UNIX, and configuration of several dedicated server systems. Even though Phase I is based primarily on a time-sharing system model, server systems are deployed for two reasons. First, some facilities are best provided this way even in a time-sharing environment. Second, early deployment of servers provides experience that will be helpful in designing the Phase II environment, which depends strongly on servers to support the workstations.

1.4.1 X: Athena Windows

The largest single software development project during phase I is a general-purpose window system that integrates with UNIX in such a way that all applications can run within it, either without knowledge that a window system is in operation or taking advantage of the additional user interface opportunity provided by such a system. The Athena window system, named "X", assumes the availability of a bit-map display device, mouse, and keyboard, and provides both a text and a graphics interface.

X is further organized to exploit the network environment, by placing a network link in the path between an application and the display. The consequence of this design is that one window can display output of an application running at one network site, while another window displays results of a second application running elsewhere, perhaps on a workstation of different manufacture.

The programming interface to X is relatively low-level, being intended as a target for graphics libraries rather than as a direct target for user application programs. One such library emulates the window management interface of the CMU Andrew system. Another library will provide a Virtual Device Interface (VDI) programming target, for use with Graphical Kernel System applications. A menu construction library provides one easy way for any application to make effective use of the mouse and display, using pop-up menus in the form of a deck of cards. Finally, a terminal emulator library provides to application programs that are unaware of the existence of a window system the illusion that an ordinary display terminal is handling their output. (Two kinds of terminals are emulated, the Digital VT-102 and, for graphics display, the Tektronix 4010.)

Because availability of a window system is such a fundamental addition to the basic tools of an operating system, it is part of the Athena Technical Plan to encourage its widespread adoption, at other universities and in standard products of manufacturers. Such adoption is desirable, because it could potentially increase dramatically the possibilities of import and export of interesting application systems.

The interface to the X window system is specified in the Athena document "Xlib--C language X interfaces," by Gettys, Newman, and Della Fera. A technical paper, in preparation, describes the design considerations and the internal architecture.

1.4.2. User registration data base

The primary system management tool developed during Phase I is a centralized user registration system. As normally operated, one adds users to a Berkeley UNIX system by logging in as a privileged user and running a maintenance program that asks a few questions, adds the user's identification and password to the password file, and creates a home directory for the user.

In the application of UNIX to Project Athena, there are 50 time-sharing systems and 3000 users. In order to control load and avoid the need to distribute all class-specific software to all 50 machines, specific classes are assigned to specific time-sharing systems. As a result, one student may have accounts on two or three different systems. In this environment, the normal UNIX registration approach is quite unwieldy. The user registration system provides a central way of managing this problem, starting with one time-sharing system that maintains a relational data base.

This data base contains, briefly, one entry for every registered user of Project Athena, containing the user's UNIX identification and password, and that user's association with any of several groups. A typical group might contain all of the members of one M.I.T. class. A second table in the data base lists on which of the 50 time sharing systems the members of each group should have accounts. A remote interface to the data base allows it to be examined and updated, with appropriate authorization, from any Project Athena time-sharing system.

Users get into the Athena registration table as follows: the M.I.T. registrar provides Project Athena each semester with a machine-readable list of all current M.I.T.-registered students; that list is also entered into the database. One or more of the user time-sharing systems runs on some terminals a user registration program that allows an individual student to select an unused UNIX user identification and password. Using the data base remote access interface, this program marks the student as Athena-registered.

Addition of a user to a group is done in a decentralized way by a group administrator. The head teaching assistant of a class, for example, might manage one group. Logging in to an Athena system, that teaching assistant obtains from the class a list of Athena-registered user id's, and updates the class group to include all those names. Again, the remote data base interface provides direct update of the data base on the central system.

An Athena administrator manages the association of groups to specific time-sharing systems, taking into account the load likely to be imposed by different class projects.

The final step in the management process is that four times each day a data base extraction program running on the central machine goes over these tables and generates from them an individually-tailored password file for each time-sharing system. Following that extraction, it then sends each of those time-sharing systems, via the network, its new password file.

To fully integrate this system into UNIX, several minor changes are required in UNIX commands. For example, the command to change a user's own password connects to the central data base and changes it there; the change takes effect throughout Athena only after the next user-machine update.

This description has been simplified; the full user registration system provides quite a number of other facilities, such as keeping track of where each user prefers to receive mail, and generating a new mail forwarding list early each morning. A separate document, titled "An Introduction to the Athena User Registration Data Base," (not yet written) provides a more detailed overview and pointers to fully detailed specifications.

1.4.3. Laboratory support system

A set of programs developed by a joint faculty project provides a way of using the special laboratory workstation configuration (the PC/XT described in the section on other hardware, above). That hardware includes a programmable data acquisition board which allows digital sampling of analog voltage levels in a laboratory setup. The programs provide two avenues of approach to the data acquisition hardware. First, a turnkey program provides a complete simulation of a multi-channel oscilloscope, with the option of printed output. Second, two sets of library calls, one for Fortran and and one for the C language, allow a student to construct special purpose programs to control sampling rates, set values of actuator outputs, collect and plot data, and send it to a printer. More details of this design are found in the "Introductory Users Guide for Project Athena Laboratory Computers."

1.4.4.

Changes to and adaptations of UNIX 4.3 for Athena environment

As received from Berkeley, the 4.3 UNIX system required a number of modifications for use in the Athena environment. Here is a list of those changes.

a. For communication with the M.I.T. Artificial Intelligence community and Electrical Engineering and Computer Science department systems, installation of network protocol support for the Chaos protocol family.

b. To make software distribution to 50 machines as systematic as possible, rearrangements to minimize and isolate site-specific files.

c. Interfaces to the Athena user registration data base.

d. Kernel changes to allow installation of the remote virtual disk system.

e. Kernel changes to allow installation of the X window system.

f. Addition of lock driver for RTI Ingres.

g. For control of experiments, a change to allow real-time scheduling.

h. Addition of several Athena library directories:
/etc/athena
/usr/athena
/usr/lib/athena
/usr/unsupported

i. Rearrangement to allow building from sources in a single tree.

j. Bug fixes, especially in network code.

1.4.5. Servers deployed during phase I

1.4.5.1 Mail forwarding

Available mail handling software is organized around the assumption that most users read their mail on a few large time-sharing systems; they require the sender of mail to know the name of the time-sharing system and the name of the user on that time-sharing system. In a workstation environment this approach will not work, but even when using 50 mini-computer time-sharing systems it begins to fail; it is a significant nuisance for one user to need to know on which other machine a particular user reads mail.

To deal with this problem, a single VAX 11/750 (later, a MicroVAX II) running UNIX is dedicated for use as an Athena-wide mail forwarding service. All outgoing mail from any Athena time-sharing system automatically starts by being forwarded to this central forwarder for address analysis. The forwarder operates with reference to a large UNIX "alias" file that for each Athena user has an entry showing the preferred mail-reading site. This alias file, in turn, is updated daily from the central user registration data base. A user with accounts on two different systems can change the preferred mailreading site by running a command that updates the central data base; the change takes effect the next morning at 6:00 a.m. when the alias file is rebuilt from information in the central data base.

This service has an availability strategy tailored to the specific availability requirements of mail. It is important that long outages be avoided, but short outages, perhaps up to an hour, can be tolerated. In addition, there is little state information maintained in the forwarder--messages are normally passed along as fast as they arrive. When a failure of the forwarding machine occurs, up to half an hour can be devoted to repair; if that time is insufficient to complete the repair, a user time-sharing system in the same cluster is taken out of time-sharing service, and the function of mail forwarding transfers to the former user system until the original mail forwarding system is operational again. When, as planned, a MicroVAX II takes over operation of the service, the corresponding repair strategy will be the same, with the difference that if repair takes too long, the machine is simply replaced with a spare.

1.4.5.2 Domain name resolution

For the names of host computers, Project Athena uses the ARPANET domain naming system, and the UNIX systems operate an implementation of that system called "bind". Each Athena time-sharing system provides a network name resolution service. Those sites simply pass such requests on to one of five authoritative name servers, running on dedicated MicroVAX-I or MicroVAX-II machines. These servers, in turn, pass requests for resolutions of names outside of the M.I.T. naming domain to domain servers located elsewhere in the ARPANET.

1.4.5.3 Time-of-day

Every Project Athena UNIX time-sharing system provides time-of-day service. One system acts as the master clock, and several others as secondary masters. When they boot, all other systems set their clocks from the master clock, if available, or else they try to locate an operating secondary master. This service is used primarily by workstations being rebooted.

1.4.5.4 Remote virtual disk

A remote virtual disk system intended for Phase II deployment is used in an expermient in one cluster of IBM PC/AT workstations. Technical Plan section D2.3.1 describes the remote virtual disk system in detail, and section Z2 provides the detailed deployment plan for remote virtual disk in the cluster.

1.4.5.5 Printing

The remote line printer facilities (lpd) that are part of the Berkeley 4.3 version of UNIX have been adapted to permit network access from any time-sharing system or workstation to any printer located on another time-sharing system. A database known as the "cluster table" provides a site-independent table that permits site-dependent resolution of printer names into specific host names and local printer identification.

A pay-as-you-go service provides the ability to send files to an M.I.T. Information Services Xerox 9700 high-performance laser printer. The software operates by forwarding the file to be printed via network to the MIT-Multics computer system, which forwards it to the computer that manages the laser printer. The user picks up output at a central dispatching desk, where the charge is also paid.

1.4.6 Other software used during Phase I

Several software packages from third-party vendors provide a wide variety of application tools without need for further development. Project Athena has therefore acquired them under site license agreements. These are:

Operating system: Berkeley UNIX system distributions 4.2 and 4.3
Editor: (GNU) Emacs
Graphics interface: Graphical Kernel System (GKS)
Spread sheet: 2020
Text formatter: Scribe
Data base manager: RTI Ingres
Algebraic manipulation system: Macsyma
Scientific subroutine library: Numerical Algorithms Group (NAG)

Experiment support system: RS/1
Graphics support tools: Blox, Penplot
Printer stream conversion: Transcript
Languages: C, Fortran, and Lisp (Franz and Scheme)

The vendor-supplied packages listed above, as well as the ones developed by Project Athena itself, are the ones that the Project provides as "supported". This means that they are acquired with expectation of vendor support, local expertise is maintained on their use, and if problems are encountered, Athena staff will attempt to help resolve them. The supported packages are not the only ones to be found in the Athena libraries. Following a philosophy of "let a thousand flowers bloom," and an education value of promoting communication, the Athena standard system configuration also includes many other software packages, and a library specifically intended for adding contributions from the community.

1.5 SYSTEM ENGINEERING

The management of the configuration of a UNIX system (and to a lesser extent, a DOS system) is normally handled by a "wizard" who knows many details of the actual hardware and software configuration being used. Since the Phase I system at Project Athena consists of 50 UNIX time-sharing systems and 265 DOS systems, a systematic approach is necessary to avoid the need to hand-tailor every different machine's software and hardware configuration. The development plan identifies three distinct aspects to this systematic approach: release engineering, management of site-specific files, and daily update of driving tables. These three aspects correspond to the first three of four categories of files that are based on two properties: Whether or not they change frequently, and whether or not they are identical at every site:

1. Change is infrequent and all sites are identical. Release engineering is concerned with this category of files, which consist of the binary programs and data that constitute the system standard software.

2. Change is infrequent and each site is different. An example is the file that contains the network address of the system. Considerable analysis and reengineering has minimized the number of such files on Athena UNIX systems, and a site-specific file management system updates those that remain. (In the case of the DOS systems, the only file that is site-specific is the one that contains the network parameters.)

3. Change is frequent, for example daily, and each site is different. The driving tables, such as the password file for each time-sharing system, are in this category. So are all files created by individual users. The user registration data base system and its user interface, gadm, manage the frequently changing, site-specific, system-owned files from a central location. In the case of workstations, it is essential for operatability that there be no examples of files in this category other than the user's private data.

4. Change is frequent, and all sites are identical. This category seems to have no members, probably because changes that could affect every site in the same way require planning and coordination, and thus can't happen very frequently.

The basis for system release engineering is two dedicated machines each having enough disk storage to maintain both the binary system and, where available, sources. The first dedicated machine is a system engineering master that maintains the complete set of latest revision-controlled source modules and corresponding binary images. (In the case of vendor-supplied systems that come without source, only the binary images are available; they are handled through the same procedures.) All development is done by checking source modules out of this master system, modifying them, and then checking them back in through a submission process that verifies that the changes are the ones wanted. All source changes are tracked using the UNIX Revision Control System (RCS). Vendor-supplied binary systems go through the same submission process before installation as part of the system engineering master.

Periodically, a release is initiated by copying some or all of the contents of the engineering master system to a separate system release master machine. After the release master is verified to run correctly against all available tests [testing is not a strong point of the Phase I plan] the release master is taken to be the starting point from which all client machines in the field are updated.

Two kinds of releases, major and minor, are handled this way. In a minor release, a small set of modules is identified as having changed, and just those modules are transferred from the system engineering master to the release master, and from there to the field. Minor releases are scheduled to occur perhaps once per month, and are done by running command scripts that copy the appropriate modules from the release master to each system. In a major release, every module of the system is replaced. Major releases happen once or twice each year. Emergency bug fixes are typically done working backwards. That is, a particular machine in the field is first updated with a fixed module, and after verification that the problem is under control, the update is made to the release master machine and a minor release transfers the change to other systems in the field. The bug fix is also submitted to system engineering as a proposed permanent change. At that point is is reviewed to see whether it should be installed directly in the system engineering master or referred to development for an alternative change.

Site-specific files are updated using scripts similar to those of a minor release. The release master machine holds a directory for each site, in which is located a copy of each site-specific file for that site. Generally these copies are hand-tailored in the traditional way, but at one central location.

1.6 PHASE I NETWORK

By the end of Phase I, all Project Athena time-sharing systems and most Project Athena workstations will be linked as the largest single customer of a campus-wide high-bandwidth data communication network provided by the M.I.T. Telecommunications Office. This campus-wide network is based on a backbone network consisting of a ten megabit/second fiber token ring that in concept visits all those buildings that have network attachment requirements. The physical and logistical barriers to installing this backbone network prevent it from springing instantly into existence, so it is gradually spreading in response to specific demands both from Athena and other M.I.T. sources of data communication requirements.

Within each building that requires data communications, a local area network such as an Ethernet runs through the building to provide service to individual host computers and workstations. That local area network is in turn linked to the spine through a gateway, a small packet-forwarding computer. During phase I, three different generations of gateways are used. Initially, one of the time-sharing hosts on each net is pressed into service as a gateway, using the packet-routing code that comes with Berkeley UNIX. As rapidly as feasible, these time-sharing hosts are being replaced by MicroVAX-I (later to be upgraded to MicroVAX-II) computers. In the MicroVAX, the gateway software is initially a copy of Ultrix, using a version of the same packet-routing code as in the time-sharing hosts. In the third generation, still running in the MicroVAX, an M.I.T.-designed high-performance gateway package, known as the C gateway, replaces the Ultrix gateway. A cooperative project between the M.I.T. Telecommunications Office and the CODEX corporation is exploring implementation of a high-performance gateway for possible future use.

The other important aspect of network communications is the choice of protocols used. A single protocol suite, the Department of Defense standard TCP/IP protocol package, is implemented by all Athena workstations and computers. This choice is made to maximize the ability to communicate with other university organizations in the outside world, and because a good implementation of TCP/IP is available as part of the Berkeley UNIX system and for PC/DOS. At some time in the future, but only after seasoned UNIX implementations become available, a conversion to ISO standard protocols will probably be undertaken.

1.7 PHASE I SCHEDULE

Use of the Phase I Athena system by students began on a modest scale in September, 1984, and increased to large-scale use by 3000 students by January, 1986. This phase begins to taper down in September, 1986, when Phase II equipment installations first come on-line.

2. (PHASE II: THE PUBLIC WORKSTATION MODEL)

Significant coherence begins to take effect in Phase II. Although Digital and IBM provide workstations of different architectures, they are comparable in performance, function, and configuration, and a common operating system and library repertoire allows most applications to run on either one, simply by recompiling.

2.1 HARDWARE AND SOFTWARE FROM DIGITAL AND IBM.

The Digital VaxStation II and the IBM RISC Technology Personal Computer are used as workstations, both running Berkeley 4.3 UNIX systems on a kernel supported by the manufacturer, Ultrix in the case of Digital and 4.2A in the case of IBM. The earlier VAX 11/750 machines remain in place, but change their function to that of servers for the workstations.

The standard student workstation configuration is as follows:

Digital VaxStation II	IBM RT PC
1 Mips MicroVAX-II processor	2 Mips RISC processor
2 Mbytes memory	2 Mbytes memory
33 Mbyte disk	40 Mbyte disk
Ethernet interface	Token ring interface
19" 1000x1000 monochrome display	15" 1024x768 monochrome display
3-button mouse	2-button mouse
full, detachable keyboard	full, detachable keyboard
360K diskette	1.2 Mbyte diskette

These options are available:

70 Mbyte disk	70 Mbyte disk
large cabinet (3 disks)	large cabinet (3 disks)
streaming tape	streaming tape
CD rom reader	CD rom reader
19 " 1024x864 color display	14" 720x512 color display
	Ethernet interface
Any card for Q-bus	Any card for PC/AT bus

The 15" monochrome display is not available at the time of initial deliveries of the IBM RT PC, so a temporary 17" display with similar capabilities, provided by IBM Academic Information Systems Division, will be used in the interim.

2.2 OTHER HARDWARE

The only other hardware currently planned for Phase II is in the area of printer upgrades. The requirement in this area is for ubiquitous printing of both word processing and graphics output. The best current technology is the all-points-addressable laser printer, though durability is lower and cost is higher than really suitable for the education application. So as to avoid the need for all output generators to be modified every time a better printer arrives on the scene, all future printers should accept the PostScript standard format for printer streams. The Phase II plan is to evaluate printers as they are announced, and when better devices are identified, deploy them as rapidly as possible. At the current time, the Apple LaserWriter and the IBM 3812 printer are under review.

2.3 SOFTWARE DEVELOPED OR ACQUIRED DURING PHASE II.

2.3.1 Remote virtual disk and file systems

The education applications of Athena lead to two kinds of requirements for access to data beyond that on the individual workstation:

1. Libraries, often large, that are read (frequently or infrequently) and occasionally updated. Such libraries would contain system programs and

data such as font descriptions, application programs and data for particular M.I.T. subjects, and unusual libraries of data such as files of photographic images used by the School of Architecture, or census files. There is one primary reason for wanting to place such libraries in a central place: they can be updated without making a trip to every workstation. (A secondary reason is that they are too large to store at every workstation. The advent of videodisk technology could make this secondary reason less important. Another secondary reason is that for libraries that require it, access control can be implemented by requiring individual identification of users.)

2. Storage for personal files, with the ability to use them from any workstation. During Phase II, when the primary vehicle will be public workstations, this requirement is very clear. But even during phase III, with student-owned workstations, there will be times when a student uses a workstation in a laboratory or the library, and needs to leave the results of that use in a place where the privately-owned workstation can get at them later.

To begin with, a Remote Virtual Disk (RVD) system developed by the M.I.T. Laboratory for Computer Science will be upgraded and deployed to provide those two services. At the same time, since several projects at other sites are working on the development of higher-function remote file systems, those file systems will be evaluated to see how they are progressing in terms of transparency, user acceptance, performance, scalability, security, and manageability. If one of those efforts appears to be paying off, it will be imported and installed, to provide the convenience of shared writable files. Depending on performance capabilities of such a future remote file system, it may also pick up part or all of the handling of private files and shared libraries initially carried by the RVD system.

The Remote Virtual Disk system fits into UNIX in the same way as a removable magnetic disk device. The device driver on the workstation opens a conversation with a server that holds a Remote Virtual Disk pack of interest to this user. (In analogy with removable disk pack systems, the client is said to "spin up" the remote virtual disk pack.) From the point of view of the UNIX system, an ordinary file system located on this removable device is mounted into the user's file hierarchy. Thus, when a user's program reaches for a file, the UNIX file system operates as usual right through to the point where it has called on a disk device driver to read the appropriate records from the disk. The RVD device driver then forwards the request across the network to the server, which reads the records and forwards them back across the network.

There are two important properties of this design. First is the limitation that a remote disk pack may be operated in only one of two modes: completely private, which allows writing, or else shared but read only. Although limiting, these two modes correspond closely to the two major requirements outlined above. Second is a performance consideration: all of the computation associated with a file system (directory management, name lookup, access control, track allocation, etc.,) is performed by the workstation; the work of the server is minimized. Minimizing the work that a server must do for a client is important because one expects that it will be economically important to maximize the number of clients that use a given server.

The conventional use of RVD packs will be of two kinds, corresponding to the two requirements. Each student user will be assigned a small, private "locker" disk pack intended to be sufficiently large to hold current work, and usable from any Athena workstation. [In the initial implementation, a remote disk pack, once set up to work with one vendor's workstation, will be usable with only that vendor's workstation, because the UNIX file system is sensitive to the byte order of the underlying storage medium. The UNIX modification effort required to relax this restriction is still being evaluated.] Classes will be assigned virtual disk packs for use as libraries.

Updating of library disk packs is accomplished by allocating two disk packs to a library. At any given time, one of these packs is in service, possibly being used by several workstations. The library maintainer spins up the second pack in the private mode and creates on this second pack a complete copy of a new, updated library. When the new library is ready, a command exchanges only the names of the two disk packs, with the result that users requesting spin-up in the future will get the new library. Current users continue to operate with the old one until they spin it down, normally at the end of their session. This approach assures that any single session of use of a library obtains consistent results from beginning to end.

Assignment of user lockers to servers and allocation of storage space on RVD servers is done by a service management system, described below. Assignment of system and class libraries to servers is more infrequent, so is done by hand.

An in-depth description of the Remote Virtual Disk system and the upgrades required to make it operate in the Athena environment are found in plan section G3.1, and details of specific deployment plans for RVD are found in Deployment Plans Z2 and Z3.

2.3.2 Backup and Archive service

The foreseeable costs of disk storage, when compared with the funds available for education computing, do not permit the possibility that a central storage facility can hold all the files that a student would accumulate over an undergraduate career. For this reason, every workstation is equipped with a removable medium storage device--diskette or streaming tape--and the user is expected to manage storage beyond that of a small central locker by taking less-needed files offline. The student is expected to use this mechanism both for archive and also for backup of current, hard-to- reconstruct files, whether those files are stored in the hard disk of a personal workstation or on a private RVD locker.

For faculty and staff use, rather than trying to provide automatic backup of workstation and RVD-stored files to non-disk media, the following semi-automatic backup/archive service is provided: an authorized user can copy (using the UNIX remote copy command) files, groups of files, or directory hierarchies that are to be archived to a private directory on a centrally located UNIX file system. Using standard backup procedures, that UNIX system periodically transfers all files found there to off-line (initially tape, later optical) media, and then deletes them. [It is possible that this service may be elaborated in the future to return a receipt that can be used to simplify retrieval of all the files archived at one time.] If an archived file or directory needs to be retrieved, the user submits an electronic mail

request for retrieval. When the file or directory is found, the retriever places it in the user's directory at the central location ready for copying back to the workstation and sends an electronic mail notice of availability.

This semi-automatic system will become the only backup and archiving system for files, whether they are stored on RVD packs, in private workstations, or in a future distributed, shared file system. In addition, as mentioned earlier, it will be available only to authorized users, primarily faculty developers and staff members.

2.3.3 Kerberos authentication service

Use of private disk packs on a Remote Virtual Disk service, collection of mail from a Mail Drop service, and certain other services require that the user be reliably identified. For this purpose, a network authentication service, named Kerberos, will be designed, implemented, and deployed.

Kerberos operates at login time, when a user first begins a session with a workstation. As part of logging in, the user provides a name and a password to the login program. The login program sends a request to the Kerberos authentication service for a set of tickets that authorize use of various network services. Kerberos sends back those tickets, each doubly enciphered. The login program removes the first decipherment of the tickets using the previously supplied password as a deciphering key. When the user then decides to invoke a network service, the invoking program sends the appropriate ticket, as part of the request to the service for connection. The service removes the second decipherment of the ticket using a key that is private to that service (but that was known by the Kerberos server when it handed out the original ticket.)

Connecting together the original login with this network service invocation, inside the ticket is the name of the user who requested the ticket from Kerberos. With this name in hand, the service has a reliable indication of who is requesting the service and it can decide whether or not the particular request is acceptable.

Kerberos will be used to authenticate access to Remote Virtual Disk servers, mail drop servers, print servers, notification services, remote X window servers, and remote login services. It will not be used to control access to name or time servers. Kerberos integrates with an authorization and accounting service described in a later section.

This description of the operation of Kerberos has been somewhat simplified by omitting several necessary cross-checks and mechanisms to avoid spoofing and reusing old tickets. In addition, Kerberos is organized to allow multiple servers for additional reliability and availability, to cooperate with Kerberos servers administered by other organizations, and to be administered using the Athena service management system. A complete description of the Kerberos authentication service is found in Technical Plan section G2.3.

2.3.4 Information display support

The area of information display is only partly planned. Here is that part.

The anchor points for information display are the X window system as the standard interface to display devices, and PostScript as the standard interface to printers. These two base tools underly all text and graphics output plans for Project Athena. The X window system as developed in Phase I has been pushed far enough in its development that the highest priority now is to gain experience in its use rather than to add architectural innovations in Phase II. For this reason, the only planned development of X is incremental changes that are discovered to be necessary to support various higher level systems. All text formatters and other text output systems will have as a target the PostScript stream format, either directly or through intermediate translators such as Transcript.

As an intermediate target, the Virtual Device Interface standard will be supported with call interfaces for the major languages, and "device drivers" that translate from VDI to X and PostScript will be developed using GraphCap, a device driver development system from Visual Engineering Corporation.

At a higher level, a common implementation of the Graphical Kernel System, also from Visual Engineering, provides a standard way of using two-dimensional vector graphics. Except for experiments, extension to three-dimensional graphics will await availability of implementations of the standards in the area.

At the same higher level, Phase II will see deployment of several alternative interface construction systems, on the basis that at the present time user requirements are not clear enough to allow choosing one in preference to all others. Two such interface generators that will probably appear are the Andrew Base Editor developed at Carnegie-Mellon University, and the C-language version of the Apple MAC Toolkit, developed at Brown University. Another such higher level interface construction system that will be deployed is the pair of chart-generating programs named ProChart and C-Chart, from Visual Engineering. An image editing system named Director, which would require porting to the UNIX environment is a candidate for experimental use. Finally, the BLOX system will remain available as a rapid interface generator for application graphics.

A user interface task force is investigating the possibilities of a more aggressive development plan in the area of information display.

2.3.5 Workstation Notification System and Protocol

One of the additional communication systems needed for a networked workstation environment is a method of sending a brief, one-way message to a current workstation user. This system is needed primarily for messages from system services to individual users, such as "You have mail," "The file server you are using will shut down in five minutes," or "Printing of your file is now complete." One interuser UNIX service, "talk", which permits two users to connect their keyboards and displays in a kind of simulated telex conversation, needs a workstation notification system in its rendezvous procedure.

The Athena workstation notification system consists of two parts, a standard protocol for locating a user and sending a message, and a workstation server package that responds to that protocol, usually by opening a window and displaying the message.

The workstation notification protocol uses the IP Universal Datagram Protocol (UDP) as its basis. The client of this protocol first inquires of a name server where to locate the target of its message. It then sends a notification message as a single packet and awaits an acknowledgement; it repeats a few times if necessary. The meaning of an acknowledgement if that a server at the workstation has accepted the message and will make an effort to get it through to the user. A negative acknowledgment is also possible; its meaning is that the user has specifically asked not to receive such notifications. There is a provision in the protocol for a message type, which can be used by the client in filtering notifications.

The workstation server process, as suggested, filters incoming notifications based on both their type and their source address, and then directs them either to a window (creating one if necessary) or to a file.

2.3.6 Workstation mail

Since a student may use a different workstation each day, delivery of electronic mail is best handled by providing a central "post office box" which the user checks, via network, upon logging in.

To decentralize storage allocation decisions as much as possible, each student will receive a standard-size post office box on a mail server, and will be expected to pick up mail often enough that the box does not fill up. A post-box size of 256K bytes would allow a server with one 400 Mbyte disk to handle about 1600 students. Four such servers (or possibly two each with two disk drives, if cpu performance turns out not to be a bottleneck) would be adequate to serve the undergraduate population. Mail arriving at an overflowing mailbox is returned to the sender with an error message; if the sender's mailbox is also full, the mail is forwarded to a dead letter box for operator intervention.

The MH mail handling system that comes with Berkeley 4.3 UNIX provides a basic implementation of a post office protocol that works roughly as described above. That system will be developed for use in Phase II; other user mail interface systems may also be modified to use that protocol, as interest indicates.

The mail system is a client of the workstation notification protocol; when a new piece of electronic mail arrives, the mail server will use that protocol to send the message "you have mail" to the user if the name service has a record showing that user is currently active at a workstation.

The phase I strategy of passing all mail through a single forwarding server named "Athena" will be continued, so as to avoid any need for mail senders inside or outside of Project Athena to have to know the names of the several computers that act as post offices, and to provide a central point of processing for Athena-provided mailing lists.

2.3.7 Lisp

The Athena Lisp plan is based on two dialects of Lisp, Common Lisp and Scheme. The preferred dialect of Lisp for major application development at Project Athena is Common Lisp, because of its acceptance as a standard within the M.I.T. Artificial Intelligence Laboratory, a group on whose expertise the local Lisp community builds. A second, related dialect, Scheme, is widely known by M.I.T. undergraduates because of its use as a vehicle in the introductory computer science subject in the Department of Electrical Engineering and Computer Science. Both Common Lisp and Scheme use static scoping of variables and a single name space, two features that distinguish them from the other major LISP dialects.

There are several vendors working on Common Lisp implementations, none of which have yet achieved a high enough probability of success to make specific plans credible, though the number of different projects underway makes it inappropriate for Athena to develop yet another. As Common Lisp implementations become available, Athena will evaluate them, with the intent of acquiring the first reasonably high-performance, fully functional implementations that become available for Athena workstations.

As an interim plan, on VAX workstations, the Franz Lisp dialect, using dynamic scoping and two name spaces, can be used for application development by faculty projects who understand the differences and are prepared to make the future effort required to switch to Common Lisp when it becomes available. In addition, it is expected that some version of Franz Lisp will remain usable indefinitely, because it is the basis for the Macsyma system, as described in the next section.

The Laboratory for Computer Science has, with partial support from Project Athena, produced a C-language interpreter for the Scheme dialect. There is also a developing compiler that may evolve to be equally useful. Project Athena will use these two tools, and will continue to support this evolving pair of implementations.

2.3.8 Macsyma service

MACSYMA is a Lisp-based symbolic manipulation program developed at the M.I.T. Laboratory for Computer Science, and currently supported by Symbolics, Incorporated. The capabilities of MACSYMA are great, but extensive use of them requires a more substantial amount of memory and computation power than is available on a workstation, and also more memory and computation power than Project Athena can easily afford to provide on a university-wide scale.

As a compromise, Athena provides the MACSYMA system, via remote virtual disk, for use on any workstation. For modest-size problems, it is expected to operate in that environment with good performance; the student can decide when the problem has grown too large and decide whether to abandon the problem or to look for another, larger-scale environment instead. Because of its size, a single, centrally-located remote virtual disk pack is dedicated to MACSYMA.

Initially, MACSYMA is available only on the Digital VAXStation 2 workstation. It is expected to become available on the IBM RT PC at the time that the vendor-supplied Franz Lisp arrives for that workstation.

2.3.9 Improved text processing and formatting system

A low-priority project is to change from Scribe to a text processing and formatting system that provides display while editing of an approximation of the final printed appearance of a document. There are several third-party vendors developing such packages. As they appear, Athena will evaluate their capabilities and price and consider the possibility of doing such a replacement.

Three desiderata in choice of a replacement for the word processing system are:

1. In the M.I.T. environment the handling of equations is important.
2. Compatibility with X, the Athena Window system, is required.
3. An output interface to the PostScript printer stream format is required, to simplify the installation of new printers.

2.3.10 Software development or acquisition requiring further planning

There are several other areas in which development or acquisition during Phase II is required, but at this writing not enough planning has taken place to be able to describe a coherent approach. Later updates of this section will address these topics. Here is a list of those areas:

Service management system

Operations support

Name service

Workstation initialization

Authorization/accounting service

Application development tools

Projection, impromptu use, demonstration portability

Connections to other M.I.T. systems (libraries, registrar, etc.)

Statistics analysis package

Information browsing system

Forum/conversation system

Interactive Video Disk system (image and archive uses)

Laboratory support system (Switch to RT and VS-2, abandon RS/1?)

2.4 PHASE II SYSTEM ENGINEERING

The process of system engineering up to the point of release is unchanged from that of Phase I, except that multiple binary versions of all software are kept, one for the architecture of each of the vendors. Where sources are available, to the extent feasible a single source copy (containing compile-time switches for different vendor architectures) is kept in a single source tree.

Phase II system software release is divided into three areas: the software that runs on individual workstations, the software that is stored on servers but that runs on individual workstations, and the software that runs on servers.

Because the number of workstations is to be quite large, distribution of software to workstations is designed in such a way as to minimize its frequency. To this end, as many as possible of the files that make up the standard software distribution are not located on the workstation itself, but instead are in libraries stored in the Remote Virtual Disk system. Only the operating system kernel and a few command programs necessary for system initialization and bootload are stored in the workstation.

Both initial installation and update of a workstation are accomplished by bringing an installation diskette to the workstation. Contained on that installation diskette is a miniature operating system that retrieves from a Remote Virtual Disk a complete copy of the software that should be installed on the workstation. When this installation process is completed, the workstation is in a completely "clean" state, containing no evidence of any previous software or user files.

A planned experimental improvement in workstation update mechanics is to prepare a version of that installation program that can be stored in the workstation file system, avoiding the need to carry a diskette to the workstation to accomplish updates. A further improvement is to have the logout command inquire of a central workstation update service whether or not this workstation is due for an operating system release, and if so trigger the update program automatically. This improvement will be tried on a small scale to see how effective it is and what hazards it exposes.

Update of the majority of the files making up the UNIX command system and libraries used by workstations is accomplished by creating a new remote virtual disk pack containing the new system. When the release manager has prepared the new system pack he or she renames the RVD packs. Current users will continue to use the old library. Whenever a user logs out from a workstation that workstation spins down all remote disk packs, and the next user of that workstation will spin up the new library pack, because it now carries the library name.

Control of the software configuration of servers is handled using the same system used for control of operating system kernel software for workstations: upon operator-initiated shutdown and logout, the server machine runs an update program that draws in a complete new copy of its software from a network (RVD) installation pack specially tailored for that server type. Server machines will be configured with just the minimum number of files and

command programs required to operate their service. System maintenance that might be more conveniently done using a full library of UNIX tools will be accomplished by logging in to the minimally-configured server and manually spinning up a Remote Virtual Disk pack containing the same standard user command and library system provided for workstations.

2.5 PHASE II NETWORK

In Phase II, the M.I.T. campus network will be expanded, using the same basic architecture as in Phase I, and Project Athena will continue to be the largest customer of the M.I.T. Telecommunications Office. As part of a replacement of the on-campus telephone system, a new set of optical fibers for use by the data communication network will reach every M.I.T. building, including dormitories and on-campus fraternities. Those optical fibers will be used to extend the backbone token ring to all places that Athena workstations are located.

Several different local area network configurations will be used in different Athena workstation clusters. These include ordinary Ethernet, thin-wire Ethernet, and the IBM token ring.

High-bandwidth connections between workstations located in off-campus fraternities and the on-campus network represent a communication problem that currently seems to have no reasonably-priced solutions. To overcome this handicap the following configuration will be tried, and if successful extended to all fraternities: A local area network within the fraternity will link workstations there with a large-configuration workstation in the fraternity that is set up as a local disk, name, mail, and authentication server. A 9600 bits/second leased line links that server with a gateway, located at M.I.T., to the on-campus network. This configuration, if used with care, should provide mail and other modest-volume communication between the fraternity and the campus. During phase II, Athena will explore the possibility that the link in the direction from the campus to the fraternity can be upgraded to 2 or 3 Megabits/second by using an Instructional Television Fixed Service channel dedicated to data communications.

The M.I.T. campus network is connected, both directly and indirectly, to a variety of inter-organization networks, including the ARPANET, Bitnet, Usenet, CSnet, NSFnet, and Mailnet. Some of these networks place restrictions on who may use them or for what purpose. It is expected that some kind of control will have to be developed to allow conformance with those restrictions, but the mechanics (indeed the requirements) have not yet been identified. Because maximizing opportunities for communication advances the education objectives both of Project Athena and M.I.T., it is the project's intent to minimize both the need for and application of such controls.

2.6 PHASE II SCHEDULE

Use of Phase II workstations by students begins on a modest scale in September, 1986, and increases to full-scale use of 1500 workstations by June, 1988, a few months before the scheduled end of the initial charter of Project Athena. Full-scale use of the Phase II system will continue well into Phase III.

3. PHASE III: THE PRIVATE WORKSTATION MODEL

While Phases I and II constitute the formal part of Project Athena, this plan includes a transition to the post-Athena environment, called Phase III. This next phase is based on the assumption that workstation prices drop to the point that student ownership is feasible. The specific workstations for Phase III are assumed to have essentially the same architectural properties and performance as the Phase II workstations, but are high-volume, low-priced versions yet to be engineered.

There are several scenarios for the transition from Phase II to Phase III. Most of them involve some form of gradual changeover, in which some students acquire privately owned workstations, while others continue to use the public workstations installed in Phase II.

The primary technical difference between Phase II and Phase III is the need for a network that pervasively covers all Institute living areas so that students can place their workstations in their own rooms. Two major developments, both related to the installation of a new campus-wide telephone system, will help to provide this pervasive coverage. First, in dormitory locations, the telephone changeover requires that each dormitory room receive new telephone wires. As part of that installation at least two extra twisted pairs for data communication will be included in each room. At the phone closet these pairs will be connected into a local area network for room-resident workstations using the IBM token ring and, potentially, other other twisted-pair local area networks that come available by Phase III.

The second development associated with the changeover to the new telephone system,is that the M.I.T. telecommunications office expects to install an X.25 gateway between the new telephone (#5 ESS) switch and the campus data network. This gateway will permit a workstation located adjacent to an ISDN telephone anywhere on campus to open a link to the campus data network at 16 kilobits/second initially, and depending on telephone company developments, may expand to 64, 128, or 144 kilobits/second. When that development is available, Project Athena will make use of it as an additional linking mechanism for isolated workstations located, for example, in a laboratory that is not adjacent to any other local area network. The same technology could in principle be used to reach workstations located in student apartments in Cambridge or elsewhere in the Boston area. Whether or not the price and availability of ISDN service to residential customers will make that an attractive approach at that time remains to be seen.

For phase III, the Athena services introduced in Phase II will be expanded to handle 5000 users. Although no extensions of those services are planned, it is likely that some features intended for Phase II may actually be implemented only by the beginning of Phase III.

Finally, the nature of library and system file arrangements for privately-owned workstations will be substantially different from those of the public workstations, for two reasons. First, there is no requirement that a privately owned workstation be prepared to operate on behalf of any user. Second, it is important that a privately-owned workstation be set up in such a way that a useful collection of functions is available even when the

workstation is not network-attached, because some students will live in private apartments, and many students will take their workstations home over the summer. At the same time, Phase III workstations that continue to be provided in public areas will continue to operate under the older set of constraints. The exact rearrangements in storage configuration appropriate for private workstations cannot be planned at this time; the experience of Phase II operation and the disk storage sizes affordable for private workstations must be available to determine that configuration.

The logistics of workstation deployment during phase III will be substantially different from the methods of phase II, because private ownership of a workstation creates a party interested in unboxing, setting up, and other related operations that in phase II were accomplished entirely by central staff. The major< development effort for phase III is the creation of a customer setup kit that allows a new, possibly computer-naive student to take home a collection of boxes from the Microcomputer store and confidently set up a running Athena system complete with libraries and network support.

file: plan/text/d