

Supporting Ubiquitous Computing Through Directory Enabled Technologies

Michael Richichi
Drew University
36 Madison Avenue
Madison, NJ 07940 USA
+1 973 408 3840
mrichich@drew.edu

Paul Coen
Drew University
36 Madison Avenue
Madison, NJ 07940 USA
+1 973 408 3840
pcoen@drew.edu

ABSTRACT

Drew has been providing computers to students since 1984. Many universities have ubiquitous computing programs where students receive a laptop computer as part of their educational package. These programs reduce the dependence on and management issues of traditional computer labs, and allow 24x7 computing access to every student at the University. Drew also provides Novell Directory Services (NDS) accounts to all of these students, and utilizes Novell ZENworks to customize software, personalize information, maintain machines, and reduce the number of visits to the help desk. This session will describe how we use the directory and other Novell software to provide a seamless computing experience for students, and how we achieve an easily maintainable, yet loosely managed environment. A combination of the retail ZENworks product and open source Web and LDAP tools enables us to leverage the ubiquitous computing environment and maximize economies of scale, permitting a far more in depth support program than would otherwise be possible in a student-supplied computing environment. There will be a discussion about leveraging a well-populated directory to provide personalization of the network experience for the end user as well. While the session will be most relevant to schools with ubiquitous computing programs, many of the concepts and ideas could be applied to a computing lab environment or to student-supplied computing models.

Keywords

Ubiquitous computing, directory services, ZENworks, eDirectory, LDAP, laptop programs, management.

1. INTRODUCTION

Drew was the first liberal arts college in the US to provide computers to all students, beginning in 1984. In 1988 the program began providing laptop computers; in 1992 the first notebooks were issued, and notebooks are currently the standard provided computer.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, republish, to post on servers, or to redistribute to lists requires prior specific permission and/or a fee.

SIGUCCS '01, October 17-20, 2001, Portland, Oregon, USA.

Copyright 2001ACM 1-58113-382-0/01/0010...\$5.00.

In 1988, Drew began what was called the “Knowledge Initiative” which consisted of a campus-wide email and information system, as well as an integrated voice/data telephone network. This provided 9600bps connections to dorms and offices. This network remained until 1997, due in part to its being marketed as a “network” at the time, and people not being fully informed of the benefits of an actual local area network. Network pilot projects had begun as early as 1992, however, and all faculty and staff were networked in 1996, with student networking completed two years later. The explosive growth of the Internet helped eventually fuel enthusiasm for the network.

There had also been an opinion among some technology staff and faculty that while Drew had made great progress in providing technology to the community, little was being done in teaching and guiding people in how to implement technology in the classroom. Drew began a faculty development program in 1993 to promote use of advanced technologies by faculty in the classroom, which has been successful. This model led towards a shift in thinking from technology infrastructure to outward-facing technology services as the chief mission of the Academic Technology department, also created in 1993.

Drew’s late entry into local area networking presented unique opportunities. Our original NOS during testing was DEC PathWorks 4.1, but when the decision to bring the network to the community was made, the decision was made to go with the then market leader, which was Novell NetWare. Novell’s newest version of NetWare was 4.1 and that was the basis of the new local area network services. This relatively unique choice enabled us to become involved in the concepts of directory-enabled networking at an “early adopter” status. We also were early in adopting dynamic networking technologies like DHCP.

2. CURRENT ENVIRONMENT

As of fall 2001, Drew has approximately 2200 FTE faculty, staff and students, and in the neighborhood of 3000 active NDS eDirectory accounts. All new people are given eDirectory accounts which control access to all network resources. All students’ accounts are in one organizational unit (OU), and faculty and staff are in OUs that roughly correspond to academic or administrative departments, or other functional groups. Three main NetWare 5.1 servers provide file and print services for the entire campus community, with a total of 300 GB of disk space available. A fourth NetWare server provides Web proxy services using Novell’s BorderManager product.

All the file servers have Novell ZENworks 3.0 installed, and the Novell Application launcher is used extensively to provide network-delivered applications to the community. Web services are provided by Red Hat Linux servers with Apache, and the `ncpfs` package is used to mount NetWare server volumes to present web pages.

The campus network is a hybrid Gigabit Ethernet/Fast Ethernet backbone with Cisco and 3Com switches, with most locations Fast Ethernet to the desktop. All residence hall rooms have one network port per pillow, and there are additional public access network ports in locations such as the student center, the library, and other public classroom spaces. In addition, the current entering students are receiving 802.11b wireless network cards, and wireless access points are being deployed in critical locations. There is a fully routed IP infrastructure (Drew has 14 non-contiguous Class C address blocks allocated,) and IPX and other protocols are being bridged at this time.

As mentioned earlier, all students receive a laptop computer as part of their educational package. In addition, all faculty and staff are assigned “Computer Initiative” computers, usually desktops, centrally funded by the University Technology organization. These computers are currently on a two to five year upgrade cycle, with computers often being deployed to more than one person during its usable lifetime, under a rotation program to make sure the heaviest technology needs are addressed. DHCP lease statistics indicate there are approximately 2000 active computers on the campus network during academic terms.

3. SERVICES PROVIDED

3.1 ZENworks Environment

3.1.1 Introduction

In 1996, Novell introduced the “Novell Application Launcher,” a tool which allowed for the directory-based distribution of program icons and software installations (snapshots of system changes) via NDS and a Windows utility. This eventually became the ZENworks for Desktops [1] product. It now includes workstation inventory and management, disk imaging, and other features. The application launcher utility is still there, accessible via an “Application Explorer” folder on the Windows desktop. It can also present its own application objects’ icons on the Windows desktop, Task Bar, Start Menu or Quick Launch toolbar allowing administrators to make applications easily accessible to users.

Naturally, the assignment of applications to end-users is managed through the directory. A general rule of thumb to remember when considering how ZENworks fits into the NDS directory structure is that the assignment or association of objects – such as applications objects or ZENworks policy objects – is inherited, flowing down the directory tree from the top. However, when the launcher or other workstation utility is querying the tree for the configuration contained in those policies, and which applications it should display, it reads up from the location of the user’s object in the tree upwards.

You can set any given NDS Organizational Unit as the topmost application launcher configuration point for its subordinate Organizational Units and objects. The launcher will only search upwards through to tree to that OU and then stop, only displaying the applications and using the settings it has found up to that

point. This allows you to prevent a given NDS OU and everything below it from inheriting an application object that the rest of the tree receives.

This has allowed us to easily distribute certain applications to faculty and staff, but not students – or vice-versa. Not only has this been important to us in terms of licensing, but also configuration. We manage our faculty and staff desktops more closely than the student notebook computers. So to distribute an identical application to the entire campus, we not only have to assign the application to the O=Drew, our topmost Organization object, but also to the “Stu” Organizational Unit under it, since it is configured as the top of the student application launcher configuration tree. If only certain departments need an application we assign it at the Organizational Unit level instead of at the Organization level above that, since each department on campus has its own OU.

One of the ZENworks management tools is a utility to view the list of what applications are available to a given user, and where they are assigned, or inherited from. This can be very useful in an environment with distributed management, where someone might not know where in the tree a certain application has been associated.

Policy objects (they can contain Novell client configurations for workstations, NT configuration settings, Windows 2000 group policies, printer assignments and so forth) are handled in a similar manner. There are “Search policies” which allow you to change the default policy search order for a given user or all objects in a given OU. We have rarely needed to use search policies.

Associating applications is only useful once you have something to deliver. The installation features of the application launcher are very flexible. There is a post-installation snapshot/change detection utility that scans the hard drive, the registry, and various .INI files on a sample system to create an installation template. Snapshot-distributed applications installations tend to be much faster than a traditional software installations, but they do have functional limitations. As a result, we sometimes use an application’s own setup executable, pointed to by an application object. This is usually for some complicated setup that requires logic that a snapshot-type utility cannot emulate. In this situation, we often pre-populate certain registry keys or configuration files with values (the username, or user’s full name, for example) obtained from NDS. ZENworks for Desktops also supports the distribution of MSI-based installations, which are becoming increasingly common.

Application objects can also contain system requirements. The application launcher on the workstation evaluates them as it starts and queries NDS for the objects available to that user. Many of our distributions are conditional on the amount of free disk space, whether or not a registry key exists (or doesn’t) or has a certain value, internal program version numbers, and operating system versions. For example, our PC lab systems are stamped with registry keys we’ve added in order to exempt them from certain automatic software distributions. We also have slightly different snapshots for Win9x and Windows 2000 systems for many applications. Operating system updates are another obvious reason to use OS-specific requirements. The administrator can set applications (running an executable) or the delivery-only installation of a snapshot-generated template or a MSI file to

automatically run after the user has logged and the application launcher utility has loaded, so mandatory updates or installations can be delivered to users.

3.1.2 *The Benefits of Application Distribution: Time and Money*

We routinely use this type of application distribution to push out updates to our F-Prot anti-virus software and signature files. Major applications, such as Macromedia Dreamweaver or Director, Jasc Paint Shop Pro, and our WordPerfect Office installations, upgrades and services releases are delivered to faculty and staff this way. We deliver system updates and licensed academic packages to our students. Counting individual application objects – all packages, fixes and settings changes, we are delivering over 150 application objects to our users. This includes requirement-specific variations (for example, a smaller disk space footprint configuration of the WordPerfect suite for systems with smaller hard drives).

We are currently using VMware [2] -based virtual machines with bare bones Windows 95/98 and Windows 2000 configurations on undoable virtual drives to generate the snapshot-based installation templates and to perform some general installation testing. This has cut the time needed to add applications from a day or two of testing to (in some cases) an hour. Due to security concerns (the ability to force end-users to run arbitrary applications could be easily misused), our network administrators are responsible for any application objects.

3.1.3 *Examples*

In the spring of 1999, Drew migrated from our legacy e-mail system (Digital's All-in-1) to an IMAP-based solution using Netscape Communicator 4.5 as the standard client. We were able to build a customized Netscape 4.5 installer using Netscape's tools, and then augment it by having the ZENworks application launcher add values like the user's login name and e-mail address to the text-based Netscape configuration files. In addition, the ability to associate the application by NDS Organizational Unit meant that we could assign it to all of the departments in a given building on a certain date, and we could schedule support staff to be in that building on that day to handle problems with individual workstations. We migrated our 400+ faculty and staff in two weeks, starting in early April. Four or five support staff at a time (divided between administrative and academic buildings) were on hand each day. There were few problems, so their primary tasks proved to be making the users more comfortable with the process, and providing users some contact with the computing support staff. Once that was complete, the installation was distributed to the 900 or so students who could run the software (the seniors remained on the legacy system through the end of the year). Most of the students had migrated by the end of the semester in mid-May.

The distribution of all Y2K updates for our Windows 95 and 98 systems in December 1999 was even easier. One administrator chose the updates and configured the system requirements based primarily on operating system version. The applications were set to trigger in order (another ZENworks feature) and to require a reboot between each. It took under an hour to configure. As our users logged into the network, the updates automatically ran on their systems. The end-users only had to reboot their systems

when prompted, and no support staff were used in the field to install the updates.

We also managed a campus wide rollout of the KeyServer [3] software metering utility in 2000. In a matter of days we were able to start using KeyServer-metered applications, not only because we could distribute the workstation KeyAccess utility, but we were also able to deliver the keyed versions of program executables to the end-users. This again took a small amount of time on the part of one administrator, and placed no burden on our front-line support staff.

3.1.4 *Other ZENworks Features*

ZENworks also contains a service for creating dynamic local user accounts in Windows NT/2000, if the user has a ZENworks policy associated with him or her that allows it. Windows NT policies and Windows 2000 group policies can be distributed as well, again associated with users, Organization Units, or the topmost Organization object. This allows for NT/2000 workstation policy management without a Microsoft domain or Active Directory environment required. Also, applications can be set to run via a system service that allows for system-level access, allowing restricted users to install applications you have provided for them. As we move to Windows 2000 in our public labs and some other areas, this flexibility is allowing us to benefit from the strengths of the new workstation OS platform without having to change our NOS.

There is a workstation inventory tool to report and track workstation configurations and to report hardware/software inventories. Novell has extended this tool to allow it to interact with additional databases, including Oracle. We are looking at the possibility of integrating the ZENworks-based workstation inventory information into our existing internally-developed, Oracle-based computer inventory package. Again, directory-based policy management would allow us to inventory the University-owned faculty and staff systems, while not collecting configuration and software inventory information and from students.

A remote control application is included with ZENworks. It allows or disallows access to a workstation based on whether or not the user attempting to access the workstation remotely has the rights in NDS rights to do so, and provides policy-based mechanisms for controlling how much access the user has (to the interface, file system, how much warning the user on the workstation has, and so forth). This is much more manageable than simply controlling who has the password for a system using VNC or a similar tool. You can delegate access and remove access easily, simply by adding or removing rights in NDS.

The imaging utility included with ZENworks for Desktops 3 did not meet our needs at the time, so we have stayed with third-party solutions for now. While it allows for directory-based image associations and automatically registering the workstation with NDS after re-imaging, it lacked compression and required a Linux partition on the hard drive. Novell has addressed this in the new 3.2 release, which now includes image compression, has an add-on product that allows the imaging boot environment to be distributed via PXE, and allows for configuration scripts to be associated with workstations via policies as well. We plan on evaluating this in the fall, with an eye towards using it in our labs and other public areas. The idea of someone having a problem

with a classroom or lab system hitting F12 on startup for a network boot so that the malfunctioning workstation is automatically re-imaged is an attractive one.

3.2 Web Services

In 1998, Drew was providing space for personal web pages through our Campus Wide Information System (CWIS) running on an OpenVMS system. Departmental or group accounts were being handled by creating special OpenVMS accounts for those entities. Files had to be stored on the web server by using FTP or Kermit to copy files to the server directory.

Clearly a simpler solution was desirable. The Novell environment was already established and working well, so we investigated the ability to use the Novell Web Server product available at the time, and concluded it did not meet our needs. What we wanted was the ability to use the Apache web server for its ease of configuration and robustness.

A solution was devised to use Apache [4] on Red Hat Linux [5] as the web server, and use the ncpfs [6] package to mount NetWare volumes to the Linux system. A special Web gateway user mounts the NetWare filesystems to Linux, and “www” directories on user (F: drive) and departmental (G: drive) directories were created, and the web gateway user was given read-only rights to those files. The Apache mod_rewrite module is then used to map these NetWare spaces to simple names on virtual servers, and also allow for multiple directories to be served from one web host without complicated directory names—for example, the users.drew.edu web server actually serves user files from both the faculty/staff and student web servers, but provides a simple syntax of users.drew.edu/username for both, with the Apache server using file system checks to determine which server contains the user directory. This process is completely transparent to the end-user on the web.

Most importantly, for people in the community to publish Web pages, they simply save their files to a network drive, and the files are instantly available on the Web. The ability to leverage the existing LAN file sharing infrastructure led to an explosion of web page creation activity on campus, with now approximately 25% of all students, faculty, and staff having personal web pages, as well as nearly all University academic departments and a majority of administrative departments, as well as over 30 student groups. Off campus access to web page modification is provided through the Web File Access system we’ve created (see below) and in the future through secure WebDAV access.

Access control to web pages is provided in two major ways: The standard Apache location-based access controls can be used with the addition of htaccess files in the relevant directories, and many departments use this to restrict pages to sites within the drew.edu domain. However, the additional need to restrict things to certain NetWare users was desired. Philip Wilson of Drew wrote mod_auth_nds [7], which is an Apache authentication module that uses the ncpfs libraries to authenticate users based on their eDirectory credentials. Htaccess files can then be configured to allow access to specific NetWare users or containers, or simply any authenticated user. This system has allowed user-level access control of public web pages. mod_auth_nds is available under the GNU public license.

3.3 Roaming Access and the Proxy Server

In 1999 the decision was made to use Netscape Communicator as the campus standard email client, and Web browser. The Netscape configuration for desktop users was stored on the network in their F: drive space. Laptop users would have profiles stored on their hard drives in the default profile location. There was a need to provide for the ability to have this profile move seamlessly with users from computer to computer. The solution was to use Netscape Communicator’s Roaming Access feature, and the Apache module mod_roaming [8], along with mod_auth_nds to provide for Netscape roaming profile storage. Users who will be traveling from computer to computer and what their personal Netscape settings to travel with them can use Roaming Access. We also recommend that users use Roaming Access to configure copies of Netscape installed at home, so they do not need to retype settings and configurations.

Since our remote access model does not involve an on-site or static ISP service, there are services that other schools can provide just by regulating access to the dial-up pool that we needed to enable with Internet applications. One was the ability to access online journal and other information sites that the University Library subscribes to from off-campus. Most providers allow for this kind of access as long as it is access controlled to members of the University community. We are using the Novell BorderManager [9] product to provide a proxy server with authentication that provides access to these resources. A proxy autoconfiguration file [10] is provided at a URL that users can point their browser to. This file is actually a CGI script that generates the .pac file on the fly, based on the customer’s location. If they’re on campus, we prefer the proxy for access to all external web sites (hopefully allowing some requests to be provided by the cache), and if off, the .pac file will only use the proxy for the specific list of sites that require access from a Drew network address. These not only include off-campus subscription sites, but numerous other web sites that may be unavailable through the university packet filter, and sites with Apache access restrictions to drew.edu only. We have an accepted practice of putting such files in a “drewonly” directory, and the .pac file checks for that as part of the URL. We anticipate the need for access-controlled services available via the Internet to increase not only at Drew but at other universities as the adoption rates of cable modems and DSL lines increases.

3.4 ATTIC

See also “ATTIC: A Case Study of Directory Enabled Course Management” in these proceedings for more information.)

In 1998 as well, our departmental director had an idea of creating file system space for all courses offered in a given semester, and creating NDS group objects that control all access to the files in that directory to a given group. In fall of 1998, our first revision of ATTIC services was made. ATTIC services have evolved since then, and now include electronic reserves, course web pages, discussion groups, archived mailing lists, and a Web interface to ATTIC that provides for roster views, assignment check-in and check-out, and easy review of reserve readings.

The ATTIC system heavily leverages the eDirectory infrastructure, and the directory is the core of enabling access to ATTIC services. First, when students log in traditionally to the

LAN, they get a drive mapping (K:) to the root of the current term's course directory. They then see directories for each course they're currently enrolled in. Instructors can place files in these directories for student use, and can also open up directories for student access so files can be worked on collaboratively. Students have write-only rights to an "inbox" directory which can be used for online assignment hand-in. Instructors can put graded or corrected assignments in a corresponding "outbox" directory private to each student. All of this access is controlled via file system rights, and is modified nightly with registration changes received from our administrative computing system.

The ATTIC web interface is controlled by the Session Manager (see below) interface, and the scripts on those pages connect to the LDAP server built into eDirectory as the logged-in user, thus rights to objects in the directory are enforced on a per-user level. For instance, if a student has requested that their campus phone extension not be public information, a roster view will not show that information to other students, but a course instructor logging in to ATTIC would see it normally.

It is our opinion that the complexity of the access structures and the level of detail we can achieve in access control would be extremely difficult with another vendor's solution, and would have been impossible with the technology available to us in 1998. The fact that all students already had NDS accounts also made the idea of using NDS as an access control and central data store mechanism viable. ATTIC has been one of the most successful technology implementations at Drew since the original Computer Initiative, and has fully pervaded the institutional culture—students will actively complain to faculty if materials are not made available on their K: drives.

3.5 Web File Access

While our LAN-based network services were very popular, and our campus is primarily (>90%) residential, there was still a need to make services available from off-campus. This is clearly true in the case of file access—there are commuter students (especially graduate and theological students) who need access to their files from off-campus, and faculty and staff who may wish to or at times are required to access work from home or off-campus. Also, although the Windows 9x/NT/2000 platform is the only desktop platform supported by Academic Technology, there was a desire to provide access from other clients if customers wished to use them.

Many institutions solve this problem by providing FTP access to file stores. FTP was considered an unacceptable solution to us due to the lack of security (specifically plaintext passwords) available with most FTP implementations, as well as scalability and robustness issues with available FTP servers for the NetWare environment.

Our solution was to take what we'd learned with ncdfs and implement a secure, Web-based file access system. Webfiles was written by Philip Wilson (pwilson@drew.edu) and Mike Richichi as a solution to this issue. Webfiles runs on an SSL-enabled web server. It asks for your NetWare username and password, and then uses those credentials to temporarily mount user file systems to the Linux system in a protected directory, and uses Perl scripts to parse the directory structures and present a Web based view to the users. Users can upload and download files, delete and

rename files, create and delete directories, and generally do any filesystem operations on their NetWare file stores, as well as access publicly available files on other file stores.

There are some limitations with the Web File Access system. Since we're dealing with simple HTTP, files can only be manipulated one at a time, there is no provision for uploading or downloading multiple files. There is also currently no way to modify file system trustees, although it is being considered for a future revision. Also, the solution requires a Web browser interface and precludes the ability to use native programs that could possibly provide a more file-based model to clients. We hope to ameliorate these issues by providing SSL-enabled WebDAV-based access to the NetWare file systems, either with a home-grown product using Apache mod_dav, or modifying a Novell or other vendor's offering to meet our needs. The availability of WebDAV clients for most client operating systems, and the semantics of the protocol will provide for secure file access that is more comfortable for users. The HTTP interface to the filesystem will continue to be offered.

3.6 Session Manager

The proliferation of Web-based services at Drew, and the desire to make all of them available by eDirectory authentication led to the desire to enable a single sign-on based solution for access to all Drew web resources. The Session Manager product was created to meet this need.

Session Manager is a Python script that maintains a database of active connections, and a Perl authentication handler for Apache. Servers that we want to Session Manager enable are provided a shared key that they use to access the Session Manager server, and directories are enabled with the authentication handler. Session Manager was written by Philip Wilson and Erik Larsson (elarsson@drew.edu).

When a client first attempts to access a Session Manager-enabled page, they will be redirected to session.drew.edu, an SSL-enabled server that prompts for their user name and password. The Session Manager server will then generate a one-time session ID, and present a cookie to the user with that cookie and some identity information, with a cryptographic checksum as well to prevent tampering of the Session Manager key. This cookie will be valid for all sites in the drew.edu domain. The Session Manager server will then redirect the user to the originally requested page, and access will be granted based on the session cookie presented to the browser. Cookies are expired after 10 minutes of inactivity, after which the user must authenticate again to regain access.

In addition, CGI programs on Session-Manager enabled web servers can query the Session Manager server for the user's cached authentication credentials (stored in encrypted form in the Session Manager database) and use those credentials to perform secondary authentication to other resources, thus allowing for single authentication to Web resources. Session Manager is currently the authentication handler for Web File Access, ATTIC, and internal departmental tools. Session Manager means that, for instance, students connect to the ATTIC web page (attic.drew.edu) and when a link to download a file via Webfiles is clicked, the Session Manager authentication automatically grants the user access to their files. Our current inventory/trouble

tracking solution is an extension of a solution presented by Drew at SIGUCCS 2000 [11]. This system allows customer access to their hardware information, as well as their personal trouble ticket history by Session Manager-enabled access to the inventory/trouble database. All access is still managed by eDirectory, passwords are not passing unencrypted over the wild Internet, and Session Manager to web server communications are also encrypted, as well as the back end LDAP authentication that Session Manager uses to actually verify the user identity.

Session Manager also meets one other critical need, and that is the ability to change your eDirectory password from off campus. Passwords are currently set to expire at Drew every 180 days, and often would expire while a user was off-campus, thus locking them out of Roaming Access and the proxy server.

3.7 LDAP Access to Services

As of Novell NDS 8, there has been a fully LDAP v3 compliant LDAP server available for access to NDS (now eDirectory) [12]. This has allowed us to tie any remaining systems and authentication methods to use single authentication through LDAP.

For example, our Usenet News server runs INN 2.3, and uses LDAP to verify the identity of users of our news server before allowing access. This allows us to provide access to our news server from off-campus while providing some protection from the server being used as a spam gateway, both by restricting access, and allowing us to track the individual users posting messages from our server. LDAP is used extensively in ATTIC to provide views of directory data, and will be used as the basis for directory access for other Web based data services. The combination of eDirectory and LDAP access from common CGI scripting languages (Perl, Python, and PHP are all used for services on campus) is extremely powerful, and allows us to use “best-of-breed” software, whether open source or commercially provided, to provide services to end-users [13].

eDirectory’s support of LDAP has also allowed us to automate many processes with simple scripting on a Linux system. For example, initial student accounts are created from a data file from our administrative system. A simple script sets up the eDirectory accounts with the correct settings, and creates directories with ncpfs. All ATTIC directory service and file system structures are created with Perl scripts that use LDAP to create and modify NDS structures. LDAP access is used by our Administrative Computing department to verify user identities if they have lost their PIN to access the administrative student information system. Basically, LDAP is available to any application we’d like to implement on campus, and we now require that any new packages being brought in support using LDAP to authenticate users at the very least, and ideally support using LDAP to store configuration data that the software may need. We will not advocate a solution that requires a separate store of users and authentication information, as the ability to centralize such information is fundamental to the environment we have created.

4. FUTURE CONSIDERATIONS

Our current solution provides a rich, well-integrated set of services to the user community. In the future, with the fundamentals of the directory being always available, we intend to support more services with the directory in the future.

We are heavily examining Novell Portal Services to provide a university portal, integrating some of the work we’ve done with ATTIC with new information services we haven’t thought of yet. Novell Portal Services ideally meets the requirements for new applications in our environment. It is fully LDAP v3 compliant, does not require a separate authentication infrastructure, and is easily extendable using standard LDAP rules as well as Java Server Pages, and can consume content from other Web sources. We hope to be able to implement a rich portal in the next 6 to 12 months.

Drew has recently made an institutional decision to standardize on Microsoft Office instead of the Corel WordPerfect Suite currently in use. We will be using ZENworks, in conjunction with the distribution tools included in the Office XP Resource Kit, to configure and distribute this package to our faculty and staff. While we anticipate the need for a good deal of training and end-user support for the new applications themselves, we will not be assigning a significant number of support staff to handle the actual installation on workstations.

Although we are a heavy Novell shop, we fully anticipate the installation of Active Directory servers and systems. Novell’s Account Management product for Windows 2000 will allow us to integrate Active Directory into the existing eDirectory environment, with the same single authentication standards we are currently using. We are already integrating Windows NT systems with NDS for NT, which provides a seamless environment for those departments requiring access to Windows-based servers.

One limitation of our Web page solution is the inability for end-users to run scripts. We’re planning out secure scripting solutions for end-users. The requirements here are that end-users cannot access critical system files, or each other’s files, while having read-write access to their own Web directories, and possibly a database to store data. Ideally scripts would not have to be reviewed for security issues every time they are created. Our initial project is an ATTIC service to provide for surveys and voting booths for courses. If this project is successful the framework will be able to be extended to provide general end-user scripts.

It is inevitable that Netscape Communicator will at some point not be supportable as a Web browser on many client platforms. Netscape 6 or Mozilla do not have the ability to do roaming access the way that Communicator does. We will need to either implement a profile management system for our end-users, move to a purely Web-based email system, or simply state that users are on their own for configuring their email and browser settings from different computers. Even using Microsoft roaming profiles will not be an adequate solution for many off-campus or non-Windows users.

Session Manager will be enhanced to support multiple redundant Session Manager servers, as well as cleaning up the Session Manager to web server protocol interaction, and possibly providing an abstraction layer which will provide access to any LDAP v3 compliant server for authentication. It is an eventual goal to make Session Manager a product for release, possibly as an open source project.

Obviously, we anticipate upgrading our NetWare servers to new versions of NetWare when available, as well as new versions of

eDirectory, and will continue to enhance ATTIC and other directory-enabled services. We will also continue to implement new services with a pure LDAP implementation wherever possible, thus not locking us in to a single vendor for our directory and integration services, and allowing for easy replacement of the back end directory if necessary or desirable in the future.

5. CONCLUSIONS

The early decision to standardize on a directory service for our local-area network implementation not only has enabled a rich computing environment, but has also enabled the implementers of the system to think in terms of directory-enabled applications and services. We came into the local-area networking game with no baggage of bindery- or domain-based user authentication systems. We also, unlike many larger universities, did not already have another campus-wide authentication system in place like Kerberos or DCE. Our acceptance of NDS as a campus-wide authentication and identity management platform permitted us to start from scratch providing directory-enabled campus computing solutions like ATTIC, ZENworks, and Session Manager enabled web pages. By leveraging our campus NDS structure through ZENworks for Desktops, we have managed to distribute applications and updates to our campus PC's with little need to devote staff to the mechanical task of performing installations for end-users. Not only does this mean that an end-user can have immediate access to a licensed application without needing to schedule a support visit, but it also frees our support staff to devote time to supporting the actual use of the application by that end-user. Our reliance on the fundamental principles of working with the directory has enabled a rich, well-integrated set of service offerings for both traditional LAN-based desktop users and purely web-browser based clients. Our philosophy of implementation and adherence to the fundamentals of the directory will serve us well in the future as more vendors and software authors provide directory-enabled solutions.

6. REFERENCES

- [1] ZENworks for Desktops, <http://www.novell.com/products/zenworks>.
- [2] VMWare, <http://www.vmware.com/>.
- [3] KeyServer, Sassafras Software, <http://www.sassafras.com/>.
- [4] The Apache Web Server, <http://www.apache.org/>.
- [5] Red Hat Linux, <http://www.redhat.org/>.
- [6] NetWare Core Protocol Filesystem for Linux, <http://platan.vc.cvut.cz/~vana/ncpfs.html>.
- [7] mod_auth_nds, http://users.drew.edu/pwilson/mod_auth_nds/.
- [8] mod_roaming—Roaming Access Module for Apache, http://www.klomp.org/mod_roaming
- [9] Novell BorderManager Enterprise Edition, <http://www.novell.com/products/bordermanager/>.
- [10] “Proxy Client Autoconfig File Format,” <http://home.netscape.com/eng/mozilla/2.0/relnotes/demo/proxy-live.html>, (March, 1996)
- [11] Saul, J., Black, E., and Larsson, E., “helpdesk.drew.edu: Home Growing a Helpdesk Solution using Open-Source Technology,” *SIGUCCS 2000 Proceedings*, Association for Computing Machinery, New York, New York (October, 2000)
- [12] Harrison, R., Sermersheim, J., and Trottier, S., “Novell’s LDAP Developer’s Guide,” Novell Press, Provo, UT (2000)
- [13] Richichi, M., “Using Perl, Python and PHP to access eDirectory 8.5 using LDAP”, *Novell AppNotes*, Provo, UT (August, 2001)