Macintosh OS X – A Smooth Migration

Scott E. Hanselman
Ringling School of Art & Design
2700 N. Tamiami Trail
Sarasota, FL
(941) 359-7633
shanselm@ringling.edu

Mahmoud Pegah
Ringling School of Art & Design
2700 N. Tamiami Trail
Sarasota, FL
(941) 359-7633
mpegah@ringling.edu

ABSTRACT
The Ringling School of Art and Design is a fully accredited four year college of visual art and design with a student population of approximately 1000. The Ringling School has achieved national recognition for its large-scale integration of technology into collegiate visual art and design education and maintains a student to computer ratio of better than two to one. Due to the demand for computing power and the requirement for ease of use, we moved our instructional computer laboratories to the Mac OS X environment in the fall of 2002. OS X has the stability of a UNIX core and an easy to navigate Aqua graphical user interface. We have found that OS X is the best operating system for training future artists and designers. Moreover, at Ringling School, we can now use the Macintosh to run high-end graphics, nonlinear video editing, animation, multimedia, web production, and digital video applications rather than expensive UNIX workstations. As visual artists cross from painting on canvas to creating in the digital realm, OS X based solutions have made artistic expression in the digital realm easy and intuitive.

As with any new technology and large-scale deployment, IT managers will need to consider a number of factors as they migrate to Mac OS X. We would like to share valuable information with our colleagues, including tips and tricks we learned from our move to the OS X environment. Our paper will address issues such as instructional computer laboratory configuration, management tools, security, authentication, integration into an existing infrastructure, font management, application support, user training, printing, and peripheral support. We will also discuss the issues that arose and the lessons learned during and after the migration process.

Categories and Subject Descriptors
K.6.1 **[Project and People Management]:** Life cycle, Management techniques, Strategic information systems planning, Systems analysis and design, Systems development, Training

General Terms
Management, Documentation, Performance, Reliability, Security, Human Factors, Standardization.

SIGUCCS'03, September 21–24, 2003, San Antonio, Texas, USA.

Keywords
Macintosh OS X, Migration, Fonts, Network, NIS, NFS, SSH

1. INTRODUCTION
In the fall of 2002, Ringling School of Art and Design's Macintosh academic computer laboratories were migrated to Mac OS X version 10.1.5. There were a significant number of faculty and staff machines that were migrated as well, with the number of Mac OS X machines on campus totaling 325 out of 347 Macintosh computers or approximately 94%. As with any new technology and large-scale deployment, IT managers will need to consider a number of factors as they migrate to Mac OS X. We would like to share valuable information with our colleagues, including tips and tricks we learned from our move to the Mac OS X environment. We will also discuss the issues that arose and the lessons learned during and after the migration process.

2. MAC OS X, THE SYSTEM

2.1 Foundation
An important aspect in the implementation of any new system is the reception by it's audience. The new Aqua interface [5] helped to ensure that our users responded positively to its ease of use and pleasant new look. Providing a stable environment to Mac OS X is the underlying core of UNIX technology and open source project called Darwin [1]. One of the key components of this stability factor is memory management [2]. By providing each application or process in this multithreaded environment its own unique address space, Darwin is able to ensure that no one process encroaches upon another. This is a technology and reliability factor not present in Mac OS 9's cooperative multitasking, mono-threaded environment that could take down the entire system when one application crashed [4]. Darwin's foundation is the Mach kernel that provides pre-emptive multi-tasking functionality as well as modern virtual memory management [2].

2.2 Networking
Mac OS X provides superior networking through TCP/IP and supports numerous networking protocols such as LDAP, FTP, SSH, and SSL [6]. Mac OS X also provides many file services including NFS, AFP, WebDAV, and SMB, offered by a variety of platforms including Mac OS X Server, UNIX, Windows, and Linux making it the most versatile and compatible operating system offered today [7].

2.3 Security
While Mac OS X provides a robust framework for many services, security dictates that these services be enabled by an

administrative user and are disabled by default [8]. The root account is also disabled by default, providing an added layer of security [9]. The multi-user nature of Mac OS X, coupled with NetInfo user authentication, adds an additional layer of security in our open instructional laboratories, as access now requires a valid user account to enable log in. We also limit user access to a few of the applications including Terminal and other utilities as an added measure of security.

3. IMAGE CREATION

3.1 Hardware

Any and all firmware updates needed to be applied while running Mac OS 9. Applications for deploying firmware updates would not run in the Mac OS X environment at the time of our migration. It is recommended that this factor be considered before installing Mac OS X.

3.2 Operating System

The operating system out of the box needed to be customized to fit our needs before it could be integrated into our infrastructure. We implemented several techniques locally in addition to utilizing best practices from others that had already accomplished what we needed to achieve. We chose the HFS+ file system due to the fact that most application installers do not recognize the UFS file system. We used a two-partition scheme providing a major portion of the hard disk as a local storage area for our users. Before we made any changes to customize Mac OS X for our environment, we made back up copies of files we were about to modify to ensure that we could return the system to a usable state in case of failure. Customizations included:

- Installed the system using a custom install to avoid unneeded languages and printer drivers that can be problematic
- Ran all version updates for Mac OS X including security updates and installed the latest updates for software applications
- Adopted a scheme similar to Marcel Bresnik's "Integrating Mac OS X in an NIS environment" [10].
- Configured NFS auto-mounting of user's server home spaces
- Modified many files including crontab for daily 7 a.m. reboot
- Changed SUID bits for many applications
- Disabled and/or hid several applications including mail, terminal, and Key Server
- Changed login window to reflect user name and password fields
- Customized Apple Remote Desktop for functionality
- Used LPR printing via IP
- Allowed for secure shell remote login via sshd_config

3.3 Tools

Disk images were created using Apple Software Restore (ASR) after the customization was complete. We used five custom- built images for distribution to our seven hardware configurations. The use of ASR to compress and uncompress these images made it a much more manageable process for distribution.

4. DISK IMAGE DISTRIBUTION

We distributed disk images using Apple Software Restore via firewire drives. Due to the inherent I/O bandwidth offered by the firewire drives, we could distribute disk images in a very efficient manner. Previously we used Assimilator to distribute our Mac OS 9 images which was a time consuming procedure requiring approximately 45 minutes for an image of 3GB. The use of ASR and the firewire drives cut the time dramatically and we were able to complete a machine in less than ten minutes from initial booting from CD, setting up the printer and Remote Desktop, to logging off the machine. The only drawback to this procedure was that it required Mac OS 9, so we needed to boot from a Mac OS 9 CD or boot from the Mac OS 9 system we installed on the firewire drive.

Image sizes varied but all surpassed the 2GB threshold and thus the limitations of ASR so we could not distribute them across the network. The process for images larger than 2GB is to create a multiple disk image. We found this method to be problematic so we resorted to the external firewire drives and were very pleased with the performance. We were fortunate that all of our hardware had a firewire interface.

Custom installations were required in many labs and included the applications Painter and three legacy applications Director, Fireworks, and Dimensions that ran in Classic mode, supported in only one lab. Painter was a drag and drop install from the firewire drives that helped to limit the number of images we needed to create and maintain.

5. TESTING PROCEDURES

5.1 Beta Test

We distributed two of our custom images to Mac G4 computers in two faculty offices early in the summer of 2002 to let our most demanding faculty members test the new working environment. The benefits of this testing procedure were twofold; we were able to expose some issues before they came to fruition in the labs and the faculty became well versed in what to expect from the new system. These faculty members became power users and provided another support mechanism for our other users. They became a first contact point at times and another information resource for their peers.

5.2 Institutional Technology Staff Members

Two Institutional Technology (IT) staff members have been using Mac OS X since its release in spring of 2001. The two authors of this paper have been using Mac OS X since its debut, installing beta versions as they became available to us. There were many occasions during our investigation and tweaking of the system that we rendered it unusable, requiring re-installation. This continual manipulation of Mac OS X led to a deep knowledge of the operating system.

5.3 Faculty Center Machines

One G4 workstation was placed in the faculty center in the late spring of 2002, providing unfettered access to the new system to all of our faculty members any time of the day or night. This exercise exposed the general faculty population to the new Mac OS X environment for testing on their time and at their own pace. It also provided a means to satisfy their curiosity and to educate themselves on the new system. We encouraged the faculty to familiarize themselves with Mac OS X and to communicate to us any questions or concerns.

5.4 IT Student Workers

Our student workers are well versed in the graphics application software and hardware providing us a valuable real world testing environment. We asked them to put the new system and applications to the test to help ensure that we could place as stable an image into the labs as possible. We brought every unique type of hardware to a central location for testing and thoroughly documented solutions as they were developed. This testing included every type of peripheral in the academic labs including slide and flatbed scanners, Wacom tablets, Zip drives, and audio/video hardware.

6. USER TRAINING

6.1 Faculty and Staff

In April of 2002 and again in the fall of 2002 in an attempt to familiarize our user base with the new operating environment, we designed and implemented training workshops of about one hour in length. We offered several opportunities to attend our workshops to ensure that everyone had a chance to be exposed to the new environment. We achieved a turnout of approximately 50% of all faculty and staff Macintosh users.

6.2 Students

In collaboration with our CORE Studio Program faculty, we designed and implemented training sessions for the entire freshman class of fall 2002. Over the course of three weeks we were able to introduce all of the incoming students to the Mac OS X environment. Furthermore, in these training sessions we addressed issues related to file management, ethical use of campus resources, our new web portal, and other related issues.

6.3 One-On-One Training

Some of our user base required hands on one-on-one training that we were happy to provide at their request. We wanted to make the transition as seamless as possible for our entire user community.

7. NFS SERVER HOME SPACES

By enabling automount and making our workstations trusted clients of our NFS server, we were able to mount the user's home space upon login. We placed a symbolic link in the user's desktop directory to their server home space which placed the default graphical representation icon of their home space on their desktop.

8. NIS USER AUTHENTICATION

The Network Information Service (NIS) [12] protocol allows multiple computers in a client-server computing environment to share administrative records such as user information. The administrative records are stored in a central database in the NIS server and made available to the NIS clients. Although NIS is a popular and mature technology, Apple Computer does not support NIS and it is disabled by default in Mac OS X. Mac OS X instead relies on NetInfo technology developed by NeXT Computer, or LDAP [11], to share administrative records. While NetInfo offers a good solution, it is supported only in the Mac OS X environment. It is not feasible to use NetInfo services in a heterogeneous infrastructure such as the typical academic environment with its clusters of Unix, Windows and Mac OS X systems. In our computing infrastructure, we were using NIS to share administrative records so using NIS to integrate Mac OS X into our computing infrastructure was attractive and feasible. As a result, we enabled and configured NIS on Mac OS X workstations. In our implementation, we are using Sun Solaris as the primary NIS server. For client configuration our solution is similar to the solution provided by Marcel Bresink [10].

9. ISSUES

9.1 Drivers

Some peripherals needed a work around and some were just not compatible with Mac OS X. None of our flatbed or slide scanners were supported in Mac OS X. We used the Vuescan [13] utility to provide a scanning solution. We looked at other utilities but they did not support our Polaroid slide scanners. In the spring of 2003, Twain drivers became available for some of our Epson large format scanners, so we were able to provide scanning in Twain compatible applications such as Photoshop. The introduction of the Twain drivers was welcomed in the labs and once again required documentation and communication that this scanning method was now available. We will be upgrading to firewire and USB scanners in the fall of 2003 including the Epson 3200 flatbed that provides for slide scanning , circumventing our need for Polaroid support. To overcome some of the difficulties at the departmental level, we provided three departments with OS 9 scanning stations. This solution worked well and enhanced our faculty and student workflow.

9.2 Third Party Applications

9.2.1 Installation

9.2.1.1 Final Cut Pro

Final Cut Pro (FCP) was installed in limited locations based on need and licensing restrictions. This was one of our specialized disk images. To eliminate the need for individual workstation installation we deployed the lab pack purchased from Apple. The serial number for the software associates itself with the hardware address of the computer executing Apple's internal software protection. The application searches a flat file composed of the hardware addresses of the machines authorized to run FCP. The lab pack instructions are very thorough and detailed.

9.2.1.2 Painter

As mentioned earlier in this paper, MetaCreations Painter was a drag and drop installation from the firewire drives used for disk image distribution. This installation method eliminated the need for an additional disk image and the use of the firewire drive made the process more efficient.

9.2.2 Functionality

9.2.2.1 Macromedia Dreamweaver

Dreamweaver produced an error upon launch stating that a required file could not be found. We found after some investigation that the file existed but was in lower case and the application was looking for the file in upper case. Converting several file names from lower to upper case resolved this issue.

9.2.2.2 Macromedia Flash MX

Flash MX's publish feature was non-functional in Mac OS X. We received an error message that a particular template could not

be found. It was discovered that this not only affected network users but local users as well. If a local user existed when Flash MX was installed, the publish feature did not work. When a new local user was added after the installation of Flash MX, the publish feature functioned properly. We experimented as time permitted with the creation date of network user accounts and did not find a solution along that path. The solution for publish feature functionality was to force Flash MX to launch in Classic mode and then the feature worked as expected.

9.2.2.3 DVD Studio Pro

After installing Apple's DVD Studio Pro it was discovered that serialization of the application did not hold for our network users. The serial number was placed in a local file in the administrative user's home directory. In order to get the application to function for our users we needed to somehow link this serial number to their NFS mounted home directory. We facilitated this by copying the serial number file into a readable directory at the root level of the computer and placed a hidden symbolic link to that file in the user's home directory.

9.2.2.4 Ableton Live

This application exhibited the same symptoms as DVD Studio Pro and was resolved in the same manner.

9.2.2.5 Vuescan

This application exhibited the same symptoms as DVD Studio Pro and Ableton Live and was resolved in the same manner.

9.2.2.6 MetaCreations Painter

Painter once again proved to be a challenge in that the user needed write privileges in an area in which they were not allowed to write. We rectified this situation by changing permissions and allowing them to write to this area but setting the sticky bit so they could not remove anything.

9.2.2.7 Legacy Applications

Classic was supported in one lab to facilitate the use of three legacy applications, Director, Dimensions, and Fireworks.

9.3 Scanning

Scanning was one of the biggest issues we faced with our migration. The problem with scanning, as with the migration to Mac OS X, was that it was unfamiliar to our user base. Our users were familiar with scanning using Twain compatible software such as Photoshop. The only means available to support all of the different models of scanners we had on campus was the Vuescan utility, which as mentioned earlier had its own issues. One of the hurdles to overcome with this utility was relaying the information regarding the use of Vuescan to our users. We provided documentation via hard copies taped to the scanning stations themselves and PDF files posted on our web site. The default location to save files with Vuescan is the user's documents folder. This in and of itself is not a big issue except that the application would remember the previous user's home space as the default saving place and the current user needed to reset this. Scanning was so problematic that in three instances we deployed Mac OS 9 stations strictly for scanning. These scanning stations included Photoshop, the drivers for the scanners, and a Chooser so users could upload the scans to their home space. This created more issues due to the fact that an AFP share is different than the NFS mounted home space in Mac OS X. We created further documentation advising them where to save their scans so that they could be retrieved once they logged into a Mac OS X workstation.

9.4 Fonts

We wanted to leverage the feature that shared fonts load into memory at boot time. This aspect would allow us to avoid the high cost of font management software while providing complete functionality. To facilitate this, fonts were installed in the normal location of /Library/Fonts so that all users had access to them. We then made modifications to the /etc/rc and /etc/rc.common files so that after the fonts loaded into memory they would be renamed as a dot file to hide them from the Finder view. This method allowed for the protection of our fonts from the GUI and we knew that fonts could not be successfully copied via the command line. This method did not, however, protect the fonts from Adobe's scheme of preflight packaging available in InDesign. Once we learned that this feature was being abused, we found that we were able to modify the application package to disable this function. This feature was designed with good intentions, but as with most good intentions, exploitation is possible as well.

9.5 Default Environment

In order to provide a consistent initial work environment, it was imperative to create and distribute a default user template for initial login. We created a customized Dock and desktop with applications and links in place to enhance the user experience and create a starting point for their own customization of the environment. We accomplished this task by creating a pristine user template and placed this template consisting of a Library, preferences, and symbolic links for the application serialization mentioned earlier in every user's home directory. The size of the default template was 15MB.

9.6 Printing

9.6.1 Hardware

Earlier we mentioned the issue with some drivers not being available to support all of our hardware which included three Epson 3000 color networked printers. Since these printers were not supported we had to come up with a temporary solution to provide for color printing in these locations. The solution deployed was a local Epson 2200 that provided high quality prints much like the Epson 3000. We installed the Epson 2200 printers on one workstation per location, as they are not network printers. Our users were understanding that we could not provide a network printing solution and appreciated the access to this high quality color printer.

9.6.2 Software

There were two applications that had printing problems. The resolution to InDesign and Illustrator printing issues was to create a PDF and send that file to the printer using Acrobat. There are two ways to create PDFs, one from the applications themselves and the other is through the Print Center. This is believed to have been a known issue and was rectified with an update.

9.7 File Corruption

9.7.1 User Home Space

9.7.1.1 Incomplete Downloads

Incomplete downloads produced files that were incomplete and therefore corrupt. These incomplete or interrupted downloads were caused by many issues but the end result was that a corrupted file existed, usually in the user's home space. Depending upon the location of the file the complications would vary. Files existing on the desktops that were corrupt would cause the machine to hang indefinitely at login. The remedy was to remove all items from the desktop and place them in a new directory on the home space so the user could log in. Once the user could log in, we accessed the new directory with the Terminal application and removed the offending file. The users generally remembered what the name of the file was which simplified the process. If they did not remember the file name, it was time consuming to ferret out the offending file depending upon the number of files present.

9.7.1.2 Application Crashes

Application crashes, including scanning, would sometimes produce a file that was corrupt. The fix for this depended upon the file location as well. We found that most times, the users were working from or saving to their desktop.

9.7.1.3 Quota Encroachment

Quota encroachment corrupted the directory in which the offending file that put the user over their quota was stored. A temporary fix for quota encroachment was to increase the home space allotment for a brief period allowing the user time to log in and clean up or archive their files. After the allotted time period, their original quota allocation was re-instated. Complete corruption of the user's Library was also a common effect of users breaching their quota. This issue required that the default Library be re-established. One item of note here is that user Internet favorites can be retrieved from the old Library after the new Library is installed.

9.7.1.4 User Preference Files

User preference files can become corrupt for any number of reasons, usually affecting application functionality. We found that when an application behaves abnormally, removing the preference files and or directories associated with that application from the user's home space cured the problem. One example of this is a user getting an error message that Photoshop cannot be launched because the user does not have access privileges. By removing the one file and the one directory that pertains to Photoshop, the user is able to open the application normally.

9.7.2 Mac OS X System Files

There are two files we determined that became corrupt and may or may not be particular to our custom installation and system tweaking. The /etc/hostconfig file would on occasion become corrupt for unknown reasons and render the computer useless, exhibiting the symptom of a plain blue screen. The other file that became corrupted once again for unknown reasons was the local NetInfo database file. The remedy for both of these issues was to insert a new pristine file in the old file's location and reboot the computer. Several methods can be employed to facilitate this fix but all require a bootable device other than the internal hard drive.

9.8 Document Platform Portability

We ran into the issue of document platform portability as will most people undertaking this endeavor. This is not so much an issue with platform compatibility as it is with file saving conventions. In Mac OS 9, files are saved in two pieces called "forks", which are referred to as data and resource forks. When these files are moved to NFS as well as other file systems, the resource fork can be lost. "With Mac OS X, there is a mechanism called "Apple Double" that allows the system to work with disk formats that do not have a forked file feature, such as remote NFS, SMB, WebDAV directories, or local UFS volumes" [3]. The solution for the loss of the resource fork, which associates the file with its appropriate application, was discovered by trial and error. Files that were archived or burned to CD in Mac OS 9 appeared in Mac OS X intact and there was no problem with these files. The issue arose when those same files were burned to CD or archived in Mac OS X or accessed via the user's NFS mounted home directories. The Mac OS X archived files would have generic icons and would have no application associated with them. When you double clicked on these files you had to choose the application used to open it. This was not a straightforward task for some users so it's best to follow the procedure of archiving user's files in Mac OS 9. One more item of note, if the suggested method of file archival is not followed and the user does not remember what type of files they are, the question then becomes what application should be used to open it.

9.9 Security

9.9.1 Mac OS X System

By nature, a Macintosh computer can be booted from a number of devices, including CDs and external storage devices containing a bootable volume. Open firmware password protects the machine from being booted from anything other than the device chosen in the system preferences startup disk pane. This firmware feature must be installed to ensure the integrity of the security of the machine. If someone has a Mac OS X installation CD, they can obtain root access after booting from the CD and changing the admin or root password. If they boot from an external drive or other bootable CD such as a Mac OS 9 CD, they can bypass any protection in place via permissions or the font protection mentioned earlier. If a machine were booted from a device other than the one specified the local rc files would not load and our fonts would not be protected. We also limit access to a few of the applications to avoid unnecessary network traffic. Applications that come as part of Mac OS X that we changed the permissions on were mentioned earlier and were also renamed with a dot prefix to hide them from the Finder view.

9.9.2 Application Software Protection

Key Server was the security feature of choice when faced with no longer having MacPrefect [14] to rely on. Every application was keyed, providing a security feature that required the client workstations to access our key server before the applications would launch. This is a very important feature that we implemented which not only provided for the protection of copyright infringement but also provided tools to monitor, track, and prevent certain applications from launching. In the past we simply monitored the use of non-standard applications but may in the near future due to various issues including network bandwidth usage limit peer to peer software by including them in the key

server's deny list for execution. We keyed most applications with a few exceptions. Apple software such as FCP and DVD Studio Pro could not be keyed but an ample amount of protection was inherent in those two applications.

9.10 Maintenance

To facilitate maintenance we used remote tools such as Apple Remote Desktop and enabled SSH login to a limited user group via modification of the /etc/sshd_config file. By using the "AllowGroups" feature you can simply change group associations as it becomes necessary as opposed to enabling "AllowUser" which would require the modification of each machine's sshd_config file. These tools allowed us to perform maintenance, troubleshoot, install applications, and copy data on a large scale to numerous machines simultaneously.

10. WHAT LIES AHEAD

- We will be upgrading our academic Macintosh computer laboratories to Jaguar 10.2.x in the fall of 2003. We are impressed with its new features and better performance. One of the authors has been using Jaguar for a few months on his desktop and it has a noticeably faster feel.
- With the advent of Jaguar and its support of the LDAP version 3 protocol, we will be implementing LDAP user authentication this fall to replace NIS.
- We will be installing color laser network printers that are supported in Jaguar in our academic laboratories.
- We will be replacing all non-OS X compatible SCSI scanners with new firewire or USB scanners that are Twain compliant.
- We are looking forward to serving our applications and fonts from a Mac OS X Server. The benefits from these two abilities will greatly increase our support capabilities and the users will surely enjoy the instant software updates this method will afford. In the past we would have been able to serve only about half of our applications due to installation and functionality issues. We are anticipating that font and application serving will provide a challenging environment for us this summer.
- Our images will be distributed using Net Install this fall. Installing a system by holding down the "option" or "n" key or simply choosing an image in the startup disk pane of system preferences is much more appealing than carrying around firewire drives.
- We will also be using our Mac OS X Server as a print server for the entire campus implementing CUPS.

11. ACKNOWLEDGMENTS

Mac OS X migration at The Ringling School of Art and Design was a team effort involving many people other than the authors of this paper. In particular we would like to acknowledge the technical support and significant contributions made by Glen Shere, Karissa Miller, Luis Hernandez, Divyangi Anchan, Ted Duguay, Omar Mendez, Nathan Kennedy, David Rocamora, Octavio Perez, and Andrew Connelly. We would also like to thank Apple Computer for their continual support in our efforts to provide our campus community with the best in computing resources.

12. REFERENCES

[1] ADC April 2001, Darwin: Mac OS X's Core OS http://developer.apple.com/darwin/

[2] Apple Mac OS X Technologies Darwin. http://www.apple.com/macosx/technologies/darwin.html

[3] Apple Care Knowledge Base. Mac OS X: Apple Double Format Creates File Name With the Prefix '._' http://docs.info.apple.com/article.html?artnum=106510

[4] Apple Mac OS X Features. http://www.apple.com/macosx/features/compare.html

[5] Apple Mac OS X System Architecture. http://developer.apple.com/macosx/architecture/index

[6] Apple Mac OS X Technologies Internet. http://www.apple.com/macosx/technologies/internet.html

[7] Apple Mac OS X Technologies Networking. http://www.apple.com/macosx/technologies/networking.html

[8] Apple Mac OS X Technologies Security. http://www.apple.com/macosx/technologies/security.html

[9] Apple Mac OS X v10.2 Technologies Security. http://www.apple.com/macosx/pdfs/Security_TB.pdf

[10] Bresink, Marcel. Integrating Mac OS X in an NIS environment. June 20, 2002 (v 1.83). http://www.bresink.de/osx/nis.html

[11] Carter, Gerald. LDAP System Administration. March 2003, O'Reilly & Associates, Inc.

[12] Eisler, M., Labiaga, R., and Stern, H. Managing NFS and NIS, 2nd Edition, July 2001, O'Reilly & Associates, Inc.

[13] Hamrick-Software. VueScan Scanner Software. http://www.hamrick.com/vsm.html

[14] Hi-Resolution Systems. http://www.hi-resolution.com/products_prefect_overview.html