

Active Directory Upgrade and Inventory Tracking Database

By

Ott Rose

u re.
f Scien
Bach
ience

Submitted to
the Faculty of the Information Engineering Technology Program
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Engineering Technology

ience

University of Cincinnati
College of Applied Science

March 2005

Active Directory Upgrade and Inventory Tracking Database

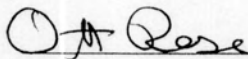
By

Ott Rose

Submitted to
the Faculty of the Information Engineering Technology Program
in Partial Fulfillment of the Requirements
for
the Degree of Bachelor of Science
in Information Engineering Technology

© Copyright 2005 Ott Rose

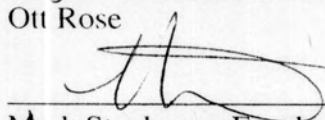
The author grants to the Information Engineering Technology Program permission to reproduce and distribute copies of this document in whole or in part.



Ott Rose

3/11/05

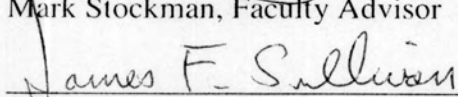
Date



Mark Stockman, Faculty Advisor

3/10/05

Date



James F, Sullivan, Department Head

11 March, 2005

Date

Acknowledgements

First, I would like to thank my wife for supporting me through college. I would like to thank my Senior Design adviser Mark Stockman for motivating me to push myself to learn and displaying confidence in me as well as helping me with technical related issues throughout the Senior Design process. I would also like to thank John Nyland for helping me with technical problems and motivating me to “own the knowledge”. Second I would like Jorin Edgerly, IT Analyst, at College Conservatory of Music. Without him this project would not have been possible. His confidence and trust allowed me to push my abilities and complete this project. Also I would like to thank Arron Rucker, IT Analyst, at the College of Design, Architecture, Art, and Planning for technical support with Mac OS X server and Active Directory Kerberos issues. I would like to thank my family for their encouragement throughout college. Also, I would like to thank Tony Goodman for introducing me to computers and displaying confidence in my ability to learn.

Table of Contents

Section	Page
Acknowledgements.....	iii
Table of Contents.....	iv
List of Figures.....	vi
Abstract.....	vii
 1. Statement of Problem.....	 1
2. Solution.....	2
2.1 User Profiles.....	4
2.1.1 Active Directory Integration.....	4
2.1.2 Inventory Tracker.....	5
2.2 Project Design Protocols.....	5
2.2.1 Active Directory Integration.....	5
2.2.2 Inventory Tracker Database.....	5
3. Deliverables.....	6
3.1 Active Directory Integration.....	6
3.2 Inventory Tracker Database.....	7
4. Design and Development.....	7
4.1 Time Line.....	7
4.1.1 Senior Design I Winter 2004.....	7
4.1.2 Senior Design II Summer 2004.....	8
4.1.3 Senior Design III Winter 2005.....	8
4.2 Budget.....	10
5. Proof of Design.....	10
5.1 Active Directory Integration.....	11
5.2 Inventory Tracker Database.....	15
6. Testing Procedures.....	22
7. Conclusions and Recommendations.....	24
7.1 Conclusions.....	24
7.2 Recommendations.....	24
Appendix.....	26
A. VBScripts.....	26
A 1. Creating Users.....	26
A 2. Creating Folders.....	28
A 3. Changing Folder Permissions.....	29
A 4. Creating Home Directories.....	29

B. Database Reports	
B 1. Inventory Report by RAM.....	31
B 2. Inventory Report by CPU Speed.....	31
B 3. Inventory Report by OS.....	32
Reference.....	34

List of Figures

Figure Number

Page

Figure 1 Student Labs.....	1
Figure 2 Budget Breakdown.....	10
Figure 3 PC Client Login.....	12
Figure 4 Mac Login Process.....	13
Figure 5 AD Architecture.....	14
Figure 6 Welcome Screen.....	16
Figure 7 Asset Utilization.....	17
Figure 8 Computers.....	18
Figure 9 Monitor.....	19
Figure 10 Place Computer.....	19
Figure 11 Printers.....	20
Figure 12 Software Details.....	20
Figure 13 Software Tracker.....	21
Figure 14 Relationship Diagram.....	21

Abstract

This project creates a cross platform single sign-on for students added to the usability, enhanced manageability, and increased security in the student computer labs. The inventory tracking database helps with computer upgrades, locating computers, and the availability of software. The single sign-on gives students the ability to move between both Mac and PC with their documents that are stored on a file server. The student computer labs are in a Windows domain which gives the IT staff the ability to manage PCs with group policies. The integration of a Mac server gives the IT staff the necessary tools needed to effectively manage Mac clients, which is known as preference management. A product of group policies and preference management is increased security. The inventory tracking database is available from remote locations. This is a must for auditing computers. Previous methods were slow and not practical. The ability to connect to the database remotely also aids in upgrading computers. Using the reporting features, the IT staff is able to cycle a computer to their desired locations to keep users with the fastest and newest computer possible. This document gives the details of the problems that the College Conservatory of Music had with student computer labs and tracking inventory as well as the solutions implemented with this project.

1. Statement of Problem

There are seven student labs at the College Conservatory of Music (CCM) (Figure 1.). The labs had generic accounts set up for the students. Students that are taking classes that require large storage, like digital video classes, were using the local hard drive for storage. A centralized storage location now allows students to work on bigger digital video projects and provide basic back up functions. Applications like Microsoft Word can be opened and edited on both platforms creating a need for a cross platform user account. There is an increasing need to have better security in the labs to prevent use by unauthorized personnel

Lab Name	Computer Type	Number of Computers
Music Education	Mac	6
Open	Mac and PC	26
Writing	Mac	20
Electronic Newsroom	PC	13
Digital Video	Mac	10
Jazz	Mac	10
Design	Mac and PC	10
Total		95

(Figure 1 Student Labs)

The inventory tracking database was poorly designed with low usability. There was a need to have an accurate and detailed database for auditing hardware and software. Previously, if a computer was not being used, it was in one of two locations. Both of

these locations are accessible by other individuals at CCM. If one of the used computers were stolen, it could take time to determine if it was actually stolen or simply not put in the database. The process of updating the database was slow and error prone. All four members of the IT staff needed to be able to access the database and run queries. In order for them to access the database they needed to call the IT department to see if it was being used. This was not efficient because the IT staff were not always available. This database did not have any software tracking functionality.

The IT staff needed to have a way to track software licenses. If a user requested an application to be installed, the IT staff had to complete a series of steps before telling the user they could or could not have the requested application. First, the number of licenses available needed to be checked. Second, they needed to track down how many were in use. This was done by asking the other IT staff members. Third, they called the user back and gave them the answer. This was difficult because the IT staff is not always together and the user was usually in a hurry to get the software.

2. Solution

After upgrading CCM's two standalone Windows 2003 servers to Active Directory (AD) and integrating a Mac OS X.3 server the CCM IT department fixed previous problems with student user accounts, disk space, and security. The most effective way to implement AD was to join the preexisting UC AD.

Some of the benefits of joining were a redundant AD database and the ease of configuration. AD replicates its database to all domain controllers in the UC forest providing a redundant AD structure. The College of Law and UCIT are in the AD forest this provides the redundant AD database. UCIT is in the process of having an off site,

copy of the AD database. UCIT has configured AD to receive up to date faculty, staff, and student account information from multiple database under UCIT's administration. Adding faculty, staff, and student accounts to CCM's AD is still being tested. Until UCIT has this feature completed CCM's IT department has manually create student accounts.

The Mac Server is connected to CCM's AD servers through Lightweight Directory Access Protocol (LDAP). This accommodates the Mac computers, through preference management. The authentication of users is done through AD. An example would be if a user has an account in CCM's AD that person can sit down at a Mac in one of the student labs and login. The user will be authenticated through the AD, and will see the appropriate settings, like a custom desktop coming from the Mac Server. This allows students that are taking digital video classes to store their large videos and give them the ability to move to different machines.

Most lab computers are now joined to the CCM domain. This allowed for improved security. The lab computers no longer have a generic login account allowing anyone to use the computers. Only faculty, staff, and student have accounts. The labs make up a small percent of the computers in CCM, but are time consuming to fix. The faculty and staff are staying on the Novell system until the IT department has enough resources to take on the extra load.

The limited capabilities of Microsoft Access to handle multiple users and new features that the IT department needs required changing the current database. The best solution for the current inventory tracking database was to completely redesign it. CCM has a SQL server already running. There are ample licenses available and the IT department is familiar with administering the SQL Server.

The new database can track computers, printers, and software. The IT department can track a computer or printer even when they are not given to faculty, staff, or students; this increased the accuracy of the database. The database also provides a record of all equipment that is sent to asset utilization which was non-existent in the previous database.

Software tracking is needed to insure that license agreements are not violated. The database has the ability to keep track of the licenses, the IT staff is now able to search for a specific application and see the total number of licenses available. There was no previous implementation of software tracking at CCM.

The client interface is Microsoft Access 2003. Access is installed on all computers at CCM. The faculty and staff that need to use the database are familiar with the Microsoft Office Suite and use other Access databases on a daily basis.

2.1 User Profiles

2.1.1 Active Directory Integration

There are two types of users that will be using the AD integration, the first is the administrators of the AD and the second is students. The administrators are maintaining and supporting the servers. The administrators are creating and managing user accounts and disk space. There was some training involved, specifically related to the integration. The students are using the disk space and received an e-mail with instructions on using their user name and password. CCM has 1,400 students in this winter quarter 2005. Their computer skills vary because some of them are new students, others are seniors or working on their masters degree. The e-mail contained only basic instructions, but the lab assistants are able to provide some additional support. Another option to help students is

to post common problems and solutions on the bulletin boards in the open labs and classroom labs.

2.1.2 Inventory Tracker

There are two categories of users for this application: administrative and auditor users. The administrative users are the IT staff. They will be updating and maintaining the database. These users are familiar with Access and are trained on SQL server administration. The auditor is the Assistant Dean. She is using the database in a read only fashion. She is familiar with Access and can direct any problems to the IT staff.

2.2 Project Design Protocols

2.2.1 Active Directory Integration

AD is the proprietary directory service for Windows 2000 and 2003. Windows 2003 was selected due to the increased ability to integrate other OSs. AD is installed on two of the servers making them domain controllers, (DCs). The file server is also running Windows 2003. This was the best option for cross platform file share because of ease of use and it saved money because the hardware was already available. Creating users was done using VB script (put in code). Using this script allowed the staff to create 1,300 users in less than two minutes. Students that enrolled after autumn are added manually.

2.2.2 Inventory Tracker Database

SQL is the ideal multi user database backend. SQL was in use for other databases at the school. This means that the staff will only need minimal training and some of the test was completed before starting this project. Access was the clear choice for the front end because of usability and availability. Access is very easy to use for the IT staff and they can train other users easily. The university has a site license for Access meaning all

PCs have it installed by default. This allows the staff to use Access from any PC. The combination of stored procedures and VBA is used to customize the application (put in code).

3. Deliverables

3.1 Active Directory Integration

1. AD upgrade and integration
 - Two Windows 2003 servers were upgraded to AD
 - One Mac OS X server was joined to AD and is running Open Directory (OD), and preference management for Mac clients.
2. Computer labs are joined to the CCM Domain
 - Digital Video lab
 - Open lab
 - Electronic Newsroom lab
 - Music Education lab
3. All students have storage space on the server
 - Each student has 100mb of disk space
4. The Mac server manages all Mac client preferences
 - Lab computers are limited on accessing OS critical files and applications, depending on the lab
5. Machine level group policy
 - Windows 2003 server DC uses group policy objects (GPOs) to secure PCs in labs
6. Cross platform sign on is implemented

- Single sign on both Mac and PC

3.2 Inventory tracker database

1. The old database was moved into new SQL database
 - Access database was exported and then imported into new SQL database
2. The new database has a software tracking feature
 - Forms accept new software packages
 - Forms allow tracking the number of licenses used for a given software package
3. The new database has user friendly forms
 - All forms are easy to read
 - All buttons are clearly marked
4. The new database has reports to make upgrading computers easier
 - The report feature of the database provides a easy way to upgrade computers
 - Users are able to search database by the asset tag number
 - Other reports come from the asset utilization table to provide information about computers, printers, and monitors that have to be sent to asset utilization

4. Design and Development

4.1 Time Line

4.1.1 Senior Design I Winter 2004

Weeks 1-5

- Research project

- Progress report I
- Problem/Area of Inquiry

Weeks 6-10

- Progress report II
- Proposal Draft
- Proposal Presentation

4.1.2 Senior Design II Summer 2004

Weeks 1-5

- July 12, 2004: Draft of project description and intended use and user profile
- July 26, 2004: Progress report I
- July 26, 2004: Complete all testing of AD
- July 30, 2004: Complete all testing of Mac server user preference management

Weeks 6-10

- August 2, 2004: Draft of design freeze
- August 16, 2004: Progress report II
- August 20, 2004: Complete file server configuration
- August 23, 2004: Final presentation

4.1.3 Senior Design III Winter 2005

Weeks 1-10

- January 3 -31, 2005: Train end users
- January 24, 2005 Progress Report I Due
- February 1 – 29,2005: Complete deliverables
- February 7, 2005 Abstract and First Draft of final report due

- February 14, 2005 Practice Final Presentation
- February 21, 2005 Progress Report Due
- February 28 - March 7 Final report and presentation

4.2 Budget

	Name	Bought From	Quantity	Costs
Software	OS X.3 Server	UCIT	1	\$100.00
	Windows 2003 Standard Server	Pomeroy	1	112.00
	Windows 2003 Standard Server	Pomeroy	1	607.00
	Windows SQL 2000 Server	Pomeroy	1	1,087.00
Hardware	Xserver G5	Apple	1	3,500.00
	Intel	Newegg.com/Pomeroy		2,995.00
	Compaq Proliant		1	4,015.00
	UPS	APC	2	350.00
Total				\$16,323.00

(Figure 2. Budget breakdown)

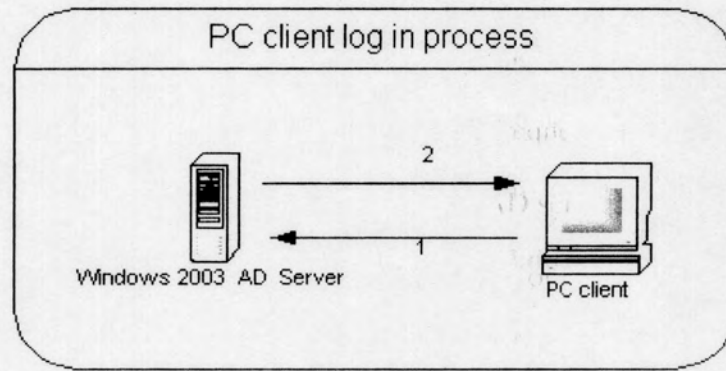
5. Proof of design

This section will have two subcategories: Active Directory Integration and Inventory Tracker Database. The Active Directory Integration covers the login process,

security, and network design. The Inventory Tracker Database section contains login process, screen shots of forms, and relationship diagram.

5.1 Active Directory Integration

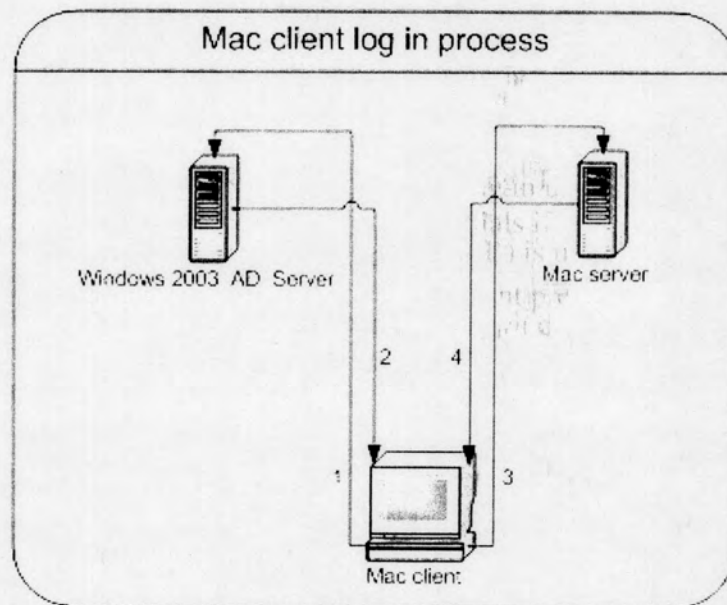
The login process is different depending on the clients. If a user is using a PC the user will see the standard login prompt. He/she will enter in the user name and password. The login request is sent to the AD server directly using Kerberos encryption. If the user has an active valid account the AD server will allow the user to log in. (Figure 3.)



(Figure 3. PC Client Login)

1. Client sends login request to AD server
2. AD server will verify user credentials and allow login if user is valid

If the user is logging in via a Mac client the login screen is similar to a PC. He/she will enter in the user name and password. The login request is then sent to the Mac server. The Mac server forwards the request to the AD server which validates the user information and passes back to the Mac server a valid or invalid log in. (Figure 4.)



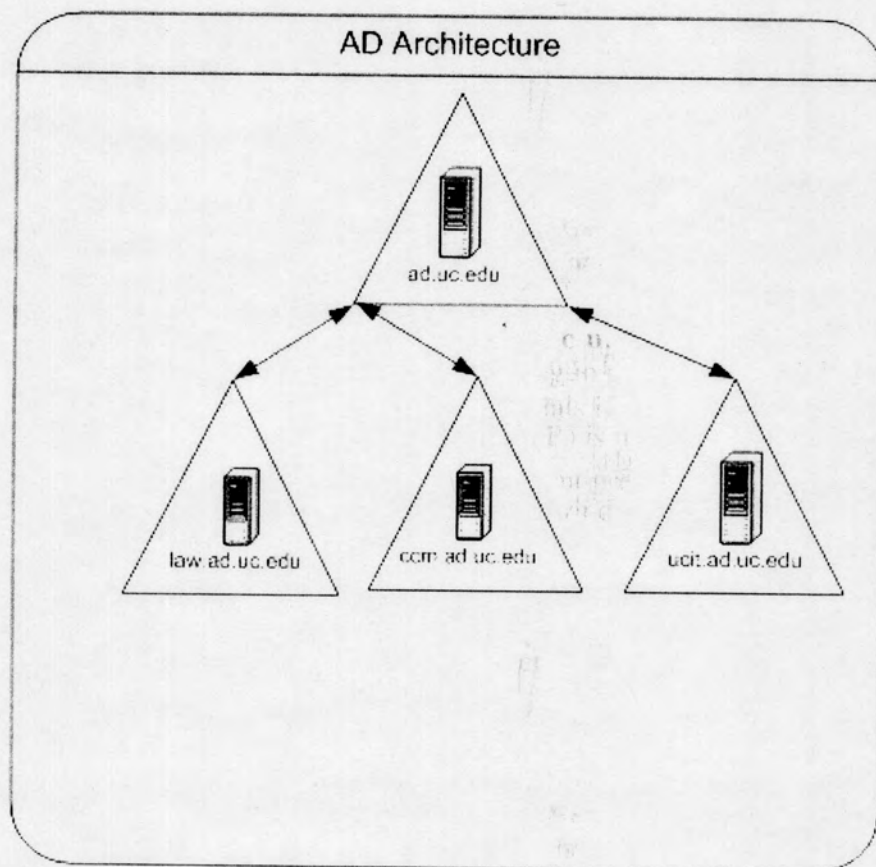
(Figure 4. Mac Login process)

1. Client starts login process by typing in username and password
2. AD server will verify user credentials if user is valid Mac OS starts login process.
3. Mac server gets request to implement preference management.
4. Mac server verifies client MAC address is valid and applies preferences

Installing AD on our Windows 2003 server enabled a security setting which prevented the current version of Mac OS, client and server, from connecting to a Windows 2003 server. This forced the IT staff to move file sharing to a platform that can be accessed by both Mac and PC clients. Turning off the security is possible, but not advisable. Windows 2003 server is used for file sharing. This is possible when the file server is not running AD. Server message block (SMB) is used to transfer files. SMB is supported by Windows 2000/XP and Mac clients 10.0 and later.

Network security was deficient at CCM. Joining lab computers to the AD domain increased security in several ways. The lab computers require a valid username and password, not allowing unauthorized individuals to use the lab computers. With the users activities being logged the CCM IT staff can track unauthorized access on the CCM domain.

The CCM domain is a child domain in the UCIT AD forest. There are two other child domains which provide redundancy for the CCM domain. (See Figure 5. AD Architecture)

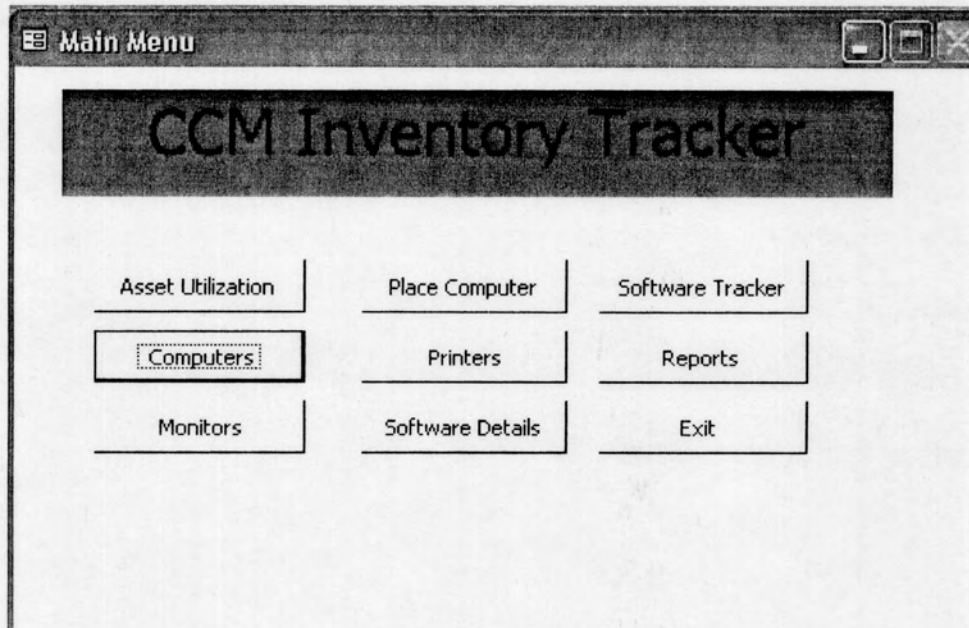


(Figure 5. AD Architecture)

5.2 Inventory Tracker Database

The Inventory Tracker Database login process contains two security measures. First, the user must have access rights to the shared folder on the server where the Access application is stored. To be granted access the user must have an active user account in the AD. Second, the user is prompted for his/her SQL account username and password. This account will not come from the AD, so the username will be different as well as his/her password.

This database contains eight forms. (See figures 6.-13.) The first form a user sees after being authenticated is the welcome screen. This screen allows the user to navigate through the application. All forms have different colors for their background except for the software details and software tracker forms. The two forms are closely related and the light blue allows the user to make a connection between the forms. The other colors were selected to make the forms more readable. All forms, except for the welcome screen, contain an icon which represents the expected content of the forms.



(Figure 6. Welcome Screen)

AssetUtilization

Asset Utilization

AssetID 15

Computer	Printer	Monitor
Make dell	Make hp	ScreenSize 21
Model gx400	Model laserjet 6	Make dell
RAM 256	SerialNumber 12345	Model pke3
CPU_Speed 800	AssetsNumber 123456	SerialNumber 32
OS 2000	P_O_Number 14254	AssetNumber 123
P_O_Number 123123	Cost \$1,300.00	Cost \$3,939.00
AssetNumber 123123	PurchaseDate 1/5/2003	PurchaseDate 1/1/2001
SerialNumber 123123		
Cost \$900.00		
PurchaseDate 1/3/2003		

Form Commands Exit Form Execute

1 of 6

(Figure 7. Asset Utilization)

Computer



Computers

Make

Model

AssetNumber

SerialNumber

RAM

CPU_Speed

Cost

P_O_Number

PurchaseDate

OS

Form Commands

of

(Figure 8. Computers)

Monitor

 **Monitors**

Make

Model

AssetNumber

ScreenSize

SerialNumber

Cost

P_O_Number

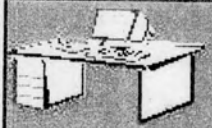
Purchase Date

Form Commands

of

(Figure 9. Monitor)

PlaceComputer

 **Place Computers**

PlacementID

Room

Building

ComputerAssetNumber

MonitorAssetNumber

PrinterAssetNumber

Form Commands

of

(Figure 10. Place Computer)

Printer

 **Printers**

Name:
 Model:
 Serial Number:
 Asset Number:
 Price:
 Quantity:
 Date:

Form Commands:

(Figure 11. Printers)

SoftwareDetails

 **Software Details**

Computer Asset Number:
 SoftwareID: 37

Form Commands:

of

(Figure 12. Software Details)

SoftwareTracker

Software Tracker

SoftwareID

Title

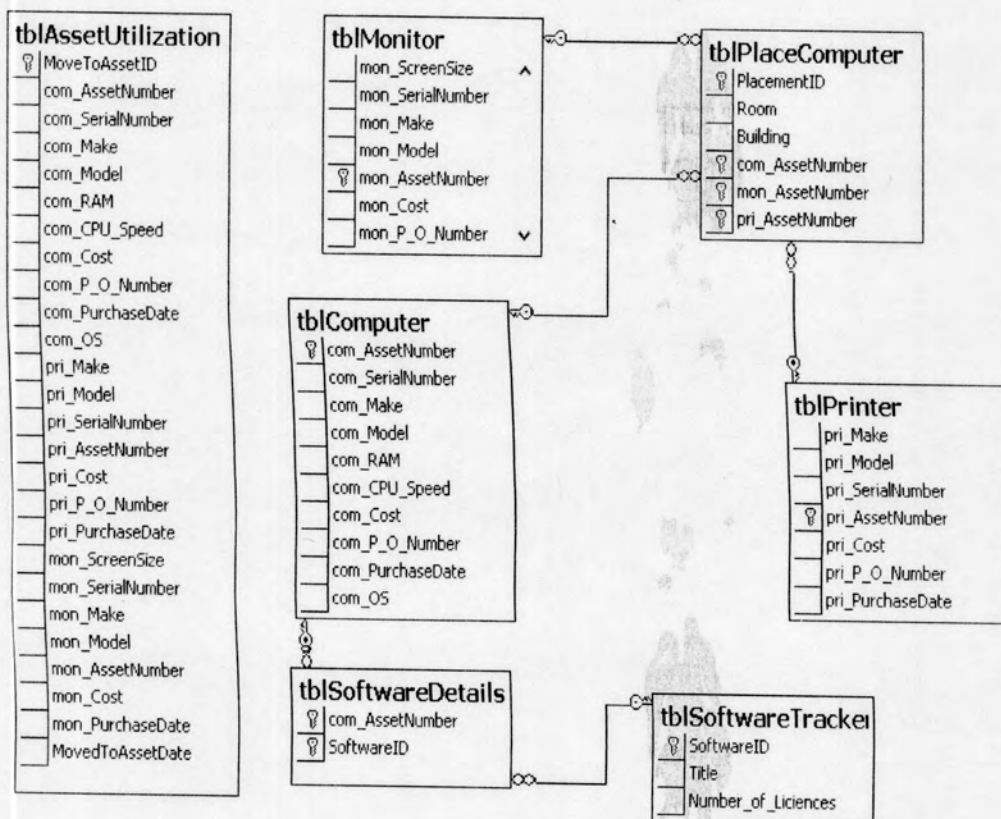
Number of licenses

Number used 6

Form Commands

of

(Figure 13. Software Tracker)



(Figure 14. Relationship Diagram)

6. Testing Procedures

The AD has been running for thirteen months. The IT Staff used it for applications storage, user backups during upgrades, and personal documents. One of our domain controls (DC) is running SQL server which is used by the students for a class. We did not see any problems that would prevent the AD from going into production. To provide extended testing the IT staff connected two labs to the AD for autumn quarter. That resulted in 39 computers which made the maximum number of concurrent connections between 39 and 43. This allowed for more testing to ensure stability for winter quarter. Starting winter quarter there was some problems with the integration which prevented joining other labs.

The Mac integration had reached a point where serious testing could begin. Testing the Mac server came from using one of the features called preference management. This allowed the IT staff to customize and secure the Mac client environment. Starting in autumn quarter the open lab computers were managed. File sharing is being implemented on a Windows 2003 member server. This was tested on a single Mac and PC before putting it in to production. The PCs have their My Documents folder redirected through a GP. This folder is accessible on a Mac by the folder that is placed in the doc at login. This policy was tested on the single Mac and PC before applying the GP to the student OU.

One of the requirements that needed to be met before moving forward with joining other labs was 99% up time, excluding restarts for software updates. With two AD servers there should not be an instance where students were not able to login to the

lab computers. Based on the first three months the AD was running, both servers should not go down at the same time. This did not hold true. There were some problems with Kerberos that prevented the Mac clients from logging in consistently. After fixing the time issue, another problem arose when one of the DCs was physically moved due to heat related issues. This resulted in a week of random reboots by that DC. Reinstalling the OS fixed the problem. Even though both DCs did not go down at the same time, some logins would time out. The only uncontrolled force that could cause both servers to be down at the same time is a power outage. Both servers are running on uninterruptible power supply (UPS) which will allow for safe shutdown and faster boot up to minimizing downtime. The Mac server needed to meet similar requirements to move forward in testing maintaining a 99% uptime, excluding restarts for software updates. The Mac server did implement preference management 100% of the time. The one problem with the preference management was some preferences did not take affect. This is a bug with the server OS. One example is the preference to shut down the computers at night. This only works on three or four of the clients. During the autumn quarter the IT Staff tested the file sharing for stability and security with a single client using multiple AD accounts. The result showed the staff that more testing was needed. Permissions on the share folder did not secure the share to meet the specification that CCM wanted. Winter quarter provided the time needed to fix the security issues. File share is function and implemented.

The Inventory Tracking Database was tested by people with diverse computer knowledge. The testing was done by entering various values for all fields. The stored procedures were tested by using the buttons on the Access forms to call the procedure.

The reports were tested by running various reports and looking for errors or inconsistent results.

Data validation is done on both the server and client sides. The client validates all data on the forms. The stored procedures that execute without data being passed to it validates the required information needed to complete the procedure.

The database does not have any runtime errors resulting in the application exiting abruptly, locking up, or giving a meaningless error message. All run time errors that were found during testing were handled by returning a meaningful message to the user. All logic and syntax errors should be discovered and fixed during testing and finishing the code portion of the database.

7. Conclusions and Recommendations

7.1 Conclusions

The need for cross platform single sign-on was the focus for this project. Using the latest server and client OSs allowed the integration to go more smoothly than previous releases for the server and client OSs. The increased security and manageability that is a product of the integration will help the IT staff support the student labs and add to the student's experiences at CCM. Having an accurate database to track software and computers will also help the IT staff support the computers at CCM. Using SQL and Access will provide a stable database for several years.

7.2 Recommendations

Configuring the Mac clients became frustrating several times in this project. This was expected because integrating OSs is complex. The bigger issue with this project was the lack of resources on specifically intergrading Mac OS X with Windows 2003. A basic

understanding of Kerberos is a must as well as OD. Some other issues were the increased security that the Windows 2003 server has by default. This resulted in many weeks of tracking down the problem with Mac connectivity that was non existent once AD was installed. Trouble shooting login failures was near impossible because the Mac clients do not return any error messages. Improvement for this portion of the project should be focused on increased usability for students. One nice feature would be the implementation for user home directories that could be used on a PC. For example, the user would be able to see the internet favorites on both Mac and PC. This is currently not possible and may only be doable with custom Apple Scripts and VB scripts. Time constants on this project prevented me from thoroughly researching this feature. The students could also benefit from an online password reset. For video editing classrooms high end server hardware is required to allow for full functionality for server based storage for editing video. CCM does not currently have the budget for this hardware.

The Inventory Tracker Database required more VBA knowledge that preliminary research showed. Understanding T-SQL is a must for SQL database programming. Understanding stored procedures and how to return errors was a stumbling bock. One improvement for this application would be a web based interface. This would make accessing the database even easier for the staff. This could also lead to a mobile application. For example, create an interface for a palm that the staff could carry around and sync up wirelessly. Automating the auditing process for computers with some type of script that puts data into the database would save a tremendous amount of time and money.

Appendix A

VB Scripts

A 1. Creating Users

```
'+++++
'University Of Cincinnati College Conservatory of Music
'
'Date: June 8, 2004
'
'Description:
'    This script will create users in Active Directory under the Student OU.
'+++++

intNumUsers = 2           'Number of users to create
strParentDN = "dc=test,dc=ad,dc=com"
'-----end configuration-----

'configuring excel
dim objExcel, objFile, strExcelPath, objSheet
dim strUserFirstName, strUserLastName, intStudnetID, strMiddleName
dim strUserMail, strUserName, BirthDate, strSsnLength, strLastFourCharSSN
dim strPass, i, c

set objExcel = CreateObject("Excel.Application")
strExcelPath = "H:\users\jorin.xls"
objExcel.Workbooks.open strExcelPath
set objSheet = objExcel.ActiveWorkbook.Worksheets(1)

'taken from ADS_USER_FLAG_ENUM
Const ADS_UF_NORMAL_ACCOUNT = 514

set objParent = GetObject("LDAP:// dc=domainname, dc=com")

'start row count
c = 1
do Until objSheet.Cells(c,1).Value = ""
    c = c+1
loop
c = c - 1
'end row count

for i = 1 to c
```

```

'assigning variables
    strUserName = Trim(objSheet.Cells(i,7).Value)

    strUserFirstName = Trim(objSheet.Cells(i,3).Value)

    strUserLastName = Trim(objSheet.Cells(i,2).Value)
    strStudentID = Trim(objSheet.Cells(i,1).Value)
    strMiddleName = Trim(objSheet.Cells(i,4).Value)
    BirthDate = Trim(objSheet.Cells(i,5).Value)
    strUserMail = Trim(objSheet.Cells(i,6).Value)

'creating password with the following format MMDD + last 4 of SSN
    strLastFourCharSSN = Right(strStudentID,4)
    strPass = BirthDate & strLastFourCharSSN
'end assigning variables

    Set objUser = objParent.Create("user", "cn=" & strUserName)
    objUser.Put "sAMAccountName", strUserName
    objUser.Put "userPrincipalName", strUserName & "@ccm.ad.uc.edu"
    objUser.Put "givenName", strUserFirstName
    objUser.Put "sn", strUserLastName
    'objUser.Put "displayName", strUserFirstName strUserLastName

    objUser.Put "userAccountControl", ADS_UF_NORMAL_ACCOUNT

    'objUser.Put "homeDirectory", "\\ccm-qrcqeswrsr\homedir\" &
objUser.Get("sAMAccountName")
    'objUser.Put "homeDrive", "s:"
    objUser.Put "mail", strUserMail

    objUser.SetInfo
    objUser.SetPassword(strPass)
    'objUser.AccountDisable=FALSE
    'objUser.Put "pwdLastSet", 0
    objUser.SetInfo
    'WScript.Echo "Created" & strUser
next
'WScript.Echo ""
WScript.Echo "Created " & c & " user"
objExcel.Workbooks.close 'exit excel

'Destory variables
set objExcel =          Nothing

```

set objFile =	Nothing
set strExcelPath =	Nothing
set objSheet =	Nothing
set strUserFirstName =	Nothing
set strUserLastName =	Nothing
set intStudnetID =	Nothing
set strMiddleName =	Nothing
set strUserMail =	Nothing
set strUserName =	Nothing
set BirthDate =	Nothing
set strSsnLength =	Nothing
set strLastFourCharSSN =	Nothing
set strPass =	Nothing
set i =	Nothing
set c =	Nothing

A 2. Creating Folders

```

'+++++
This script will create folders
This script must be ran on file server
' ParentFolder must be set correctly
'+++++

dim username
dim WshShell
Set WshShell = WScript.CreateObject("WScript.Shell")

set objParent = GetObject("LDAP:// dc=domainname, dc=com ")
objParent.Filter = Array("user")

ParentFolder = "E:\MyDocRedirect\"
set objShell = CreateObject("Shell.Application")
set objFolder = objShell.NameSpace(ParentFolder)

for each objUser in objParent

    objFolder.NewFolder objUser.Get("sAMAccountName")

```

```
objUser.SetInfo
```

```
Next
```

```
WScript.Echo "Done"
```

```
set username = nothing
```

```
set WshShell = nothing
```

A 3. Changing Folder Permissions

```
'+++++
++
'script will set permissions on user folder
'root and user folder must be created before running this script
'+++++
```

```
dim username
```

```
dim WshShell
```

```
Set WshShell = WScript.CreateObject("WScript.Shell")
```

```
set objParent = GetObject("LDAP:// dc=domainname, dc=com ")
```

```
objParent.Filter = Array("user")
```

```
for each objUser in objParent
```

```
username = objUser.Get("sAMAccountName")
```

```
Call WshShell.Run ("cacls " & "E:\MyDocRedirect\" & username & " /e /g " & username &
":F", HIDE_WINDOW, WAIT_ON_RETURN)
```

```
Next
```

```
WScript.Echo "Done"
```

```
set username = nothing
```

```
set WshShell = nothing
```

A 4. Creating Home Directories

```
'+++++
+++++
'Source: Active Directory Cookbook, O'Reilly
```

'Edited by: Ott Rose
'Date: January 8, 2005
,

'Description:

' This code sets the home drive of all users under a container
' to be on a file server where the share name is the same as the user's
' sAMAccountName.

'+++++
+++++

```
set objParent = GetObject("LDAP://dc=domainname, dc=com")  
objParent.Filter = Array("user")
```

```
for each objUser in objParent
```

```
    'Wscript.Echo "Modifying " & objUser.Get("sAMAccountName")
```

```
        if objUser.HomeDrive <> "" then
```

```
            'WScript.Echo "already has Home Dir"
```

```
        else
```

```
            objUser.HomeDirectory = "\\ccm1.ccm.ad.uc.edu\MyDocRedirect\" & _  
            objUser.Get("sAMAccountName")
```

```
            objUser.HomeDrive = "U:"
```

```
            'WScript.Echo "Added home dir for "& user
```

```
        end if
```

```
    objUser.SetInfo
```

```
next
```

```
WScript.Echo "Done with creating Home Directories"
```

Appendix B

Database Reports

B 1. Inventory Report by RAM

Inventory Report

OS	Make	Room	Buildin	Model	RAM	CPU	Asset
98	UCPC	1490	Corbett	Custom	0	0	NoAssetTag0
unknown	unknown	3101	Emery	unknown	0	0	unknown 5
unknown	unknown	3101	Emery	unknown	0	0	unknown 6
unknown	unknown	3101	Emery	unknown	0	0	unknown 7
unknown	unknown	3101	Emery	unknown	0	0	unknown 8
unknown	unknown	2720	Corbett	unknown	0	0	unknown 9
unknown	unknown	182	MEM	unknown	0	0	unknown 13
Linux	Silicon Gr	182	MEM	O2	0	0	0313460
unknown	unknown	3101	Emery	unknown	0	0	unknown 16
7	Apple	2140	Emery	Quadra 660AV	16	33	NoAssetTag6
7.5	unknown	5231	Emery	7600	16	120	0329104
8.1	Apple	2200	Emery	6100/60	30	60	0299888
8.5.1	Apple	3755	Corbett	G3	32	300	0319851
95	UCPC	4230	Emery	UCPC	32	133	0310571
8.0	Apple	1371	Corbett	7300	32	200	0329103
98	UCPC	4599	Corbett	P75	32	120	0308494
8.6	Apple	5229	Emery	8500/180	64	180	0329148
8.1	Apple	321	DVAC	8600/200	64	200	0314424
8.1	Apple	445	MEM	7600/132	64	200	0329096

Saturday, March 05, 2005

B 2. Inventory Report by CPU Speed

Inventory Report

OS	Make	Room	Buildin	Model	RAM	CPU	Asset
10.1	Apple	5243	Emery	G4	256	800	0334784

10.1	Apple	5241	Emery	G4	256	800	0335979
10.1	Apple	5237	Emery	G4	256	800	0335977
10.3	Apple	3713	Corbett	G4	512	800	0335978
10.1	Apple	2510	Corbett	G4	256	800	0336548
10.1	Apple	2372	Corbett	G4	256	800	0336547
10.1	Apple	2430	Corbett	G4	256	800	0336544
9.2.2	Apple	2460	Corbett	G4	256	800	0336552
10.1	Apple	2440	Corbett	G4	256	800	0336545
10.1	Apple	1360	Corbett	G4	256	800	0336550
10.1	Apple	243	MEM	G4	256	800	0336546
10.2.8	Apple	325	DVAC	G4	256	800	0336549
10.1	Apple	5245	Emery	G4	256	800	0335980
2000	Dell	3110A	Emery	Optiplex GX200	256	800	0328643
10.1	Apple	5240B	Emery	G4	256	800	0336555
9.2.2	Apple	5207	Emery	G4	256	800	0336551
10.1	Apple	4218	Emery	G4	256	800	0335479
10.1	Apple	389	MEM	G4	256	800	0336553
10.2	Apple	389	MEM	G4	256	800	0336554

Saturday, March 05, 2005

B 3. Inventory Report by OS

Inventory Report

OS	Make	Room	Buildin	Model	RAM	CPU	Asset
2000	Dell	4760	Corbett	Optiplex GX1	256	1400	0321514
2000	Dell	2445	Corbett	Optiplex GX260	256	2400	0339306
2000	Dell	3820D	Corbett	Optiplex GX400	256	1700	0333978
2000	Dell	3885A	Corbett	Optiplex GX1	256	1400	0321046
2000	UCPC	4770	Corbett	p200	512	1800	0319038
2000	Dell	4764	Corbett	Optiplex GX1	256	1400	0321509
2000	Dell	4750	Corbett	Optiplex GX240	256	1800	0337009
2000	Dell	3820B	Corbett	Optiplex GX260	256	2400	0339303
2000	Dell	3885	Corbett	Optiplex GX1	256	1400	0322646
2000	Dell	3885	Corbett	Optiplex GX1	256	1400	0322644

2000	Dell	3885	Corbett	Optiplex GX1	256	1400	0322645
2000	Dell	3820A	Corbett	Optiplex GX260	256	2400	0339304
2000	Dell	2445	Corbett	Optiplex GX400	256	1700	0333981
2000	UCPC	3715	Corbett	UCPC	256	1000	0314849
2000	UCPC	3717	Corbett	P200	256	1300	0317050
2000	Dell	3860C	Corbett	Optiplex GX1	128	1400	0322655
2000	Dell	326	MEM	Optiplex GX240	256	1700	0334790
2000	Dell	3235	Emery	Optiplex GX240	256	1700	0335196
2000	Dell	3235	Emery	Optiplex GX260	256	2400	0339307

Saturday, March 05, 2005

References

1. Albrecht ,Ralf ., and Natascha Nicol. *Microsoft Access Projects with Microsoft SQL Server*. <http://safari.informit.com/0735610029>. January 16, 2002
2. Allen, Robbie. *Active Directory Cookbook*. <http://safari.informit.com/0596004648>. September 2003.
3. Bankaitis, Denise., and Juliana Aldous, eds. *Microsft Office XP developer's Guide*. Microsoft Press. 2001
4. Baron, Andy., and Ken Getz, Paul Litwin. *Access Cookbook*. <http://safari.informit.com/0596000847>. February 2002
5. Boswell, William. *Inside Windows® Server 2003*. <http://safari.informit.com/0735711585>. April 11, 2003
6. Carter, Gerald. *LDAP System Administration*. O'REILY: 2003
7. Dewson, Robin. *Beginning SQL Server 2000 Programming*. Apress 2001
8. Jepson, Brian and Ernest E. Rothman. *Mac OS X for Unix Geeks*. <http://safari.informit.com/0596003560>. September 2002
9. Minasi,Mark,., and Christa Anderson, Drain M. Smith, Doug Toombs. *Windows 2000 Server*. Sybex. 2001
10. Progue, David. *Mac OS X The Missing Manual*. O'REILLY. 2002