Tips and Tricks for Mac Management

Mac OS X Leopard
May 2009

Table of Contents

Overview

This document presents some of the best practices and tips and tricks for managing Mac clients in a Leopard environment. The default environment for this document is an environment with Mac OS X client systems running the current version of Leopard (v10.5.x), Mac OS X Server running Leopard (v10.5.x), and a mixed wired and wireless infrastructure.

The document covers the following topics:

- Definition of client management for Mac OS X
- Setting up the server to provide management
- Setting up the client for network-based management
- Basics of Managed Client for Mac OS X (MCX)
- Details in MCX that enhance management
- Explanation of user accounts, mobility, and portable home directories
- Suggestions for additional ways to promote workflow management

As with any supplementary documentation, the material presented is designed to complement the Apple server product documentation that can be found online at http://www.apple.com/server/resources.

Defining Client Management for Mac OS X

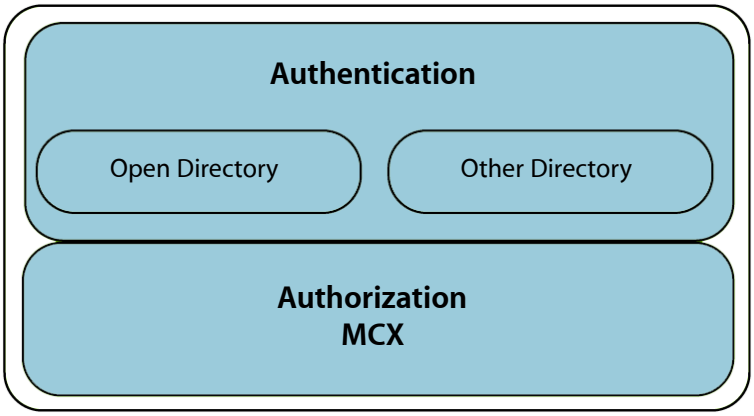
Being able to establish a stable user experience is the core definition of client management. Managed Client for Mac OS X, or MCX, is a subset of Open Directory, the Apple directory service. The policies set for client systems are stored within a directory as part of either a computer, group, or user record. Using centralized management to store management policies on a network database, system administrators can easily define the user experience for a large number of computers owned by the institution. MCX settings are actively cached onto the client computers, allowing the management settings to stick to the system when away from the network, a very useful practice in the growing use of digital learning environments.

Being part of a directory, more specifically an LDAP (lightweight directory access protocol) directory, MCX is considered as the follow-on portion of the user experience when accessing a client computer. The first thing a user generally has to do is authenticate to a directory, whether that directory is stored locally or on the network. This authentication portion of the directory contains, at a minimum, the user's name and password. Once the user has authenticated to the directory, the user's authorization, or *policy*, is checked to see what items that user actually has permission to use.

Directory Services—Authentication to Authorization

For a Leopard client, there are numerous methods to provide the necessary authentication and authorization databases. The three most common network directories are OpenLDAP (Apple's default on Mac OS X Server), Active Directory (Microsoft), and eDirectory (Novell). Although the entire process of login and policy management can easily be performed using Apple's directory services, some sites choose to use one of the other directories to provide user account information. They sometimes even extend their directory schema, or mappings, to include the MCX settings. This document will briefly discuss how a client may need to be configured to support a non-Apple directory focus. For detailed information about these foreign directories, see the *Mac OS X Server Open Directory Administration* guide available on the Apple website.

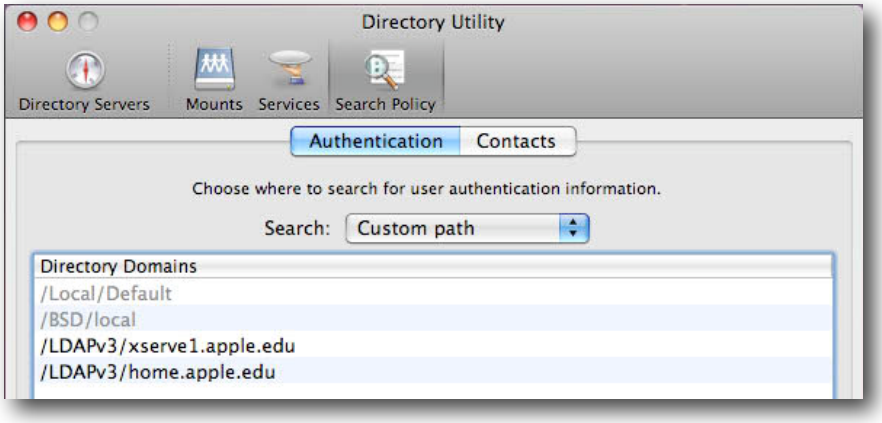
This is how the services fit together:



Directory Services

When a site deploys a non-Apple directory for authentication, such as eDirectory or Active Directory, your users' authentication information—names and passwords—are stored in one directory, with the authorization information (MCX) stored in Open Directory on a Mac OS X server. The process of using more than Apple's Open Directory environment is usually referred to as the "golden triangle," in which the client system and two differing directories provide the information needed for both login and management.

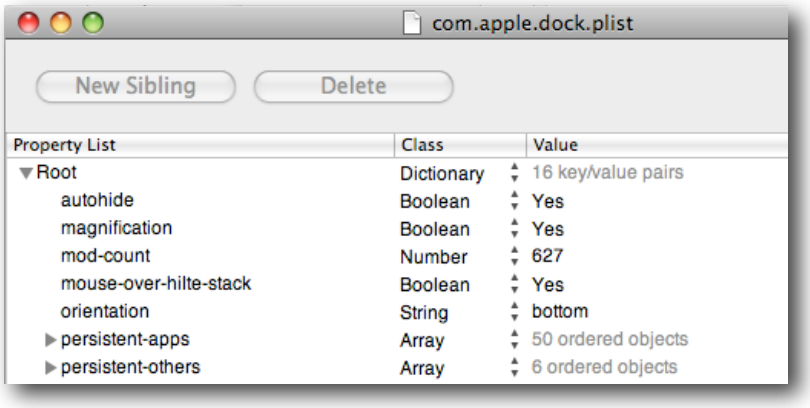
To set up binding for two directory services, the MCX server is listed first, then the authentication server. When the user logs in, the search path always passes through the MCX directory on its way to the user authentication. For example, here's the search path for a setup with "xserve1" carrying the MCX settings and the user accounts stored on "home":



Set the search order to access the MCX directory first, then the user data.

Managing Preferences

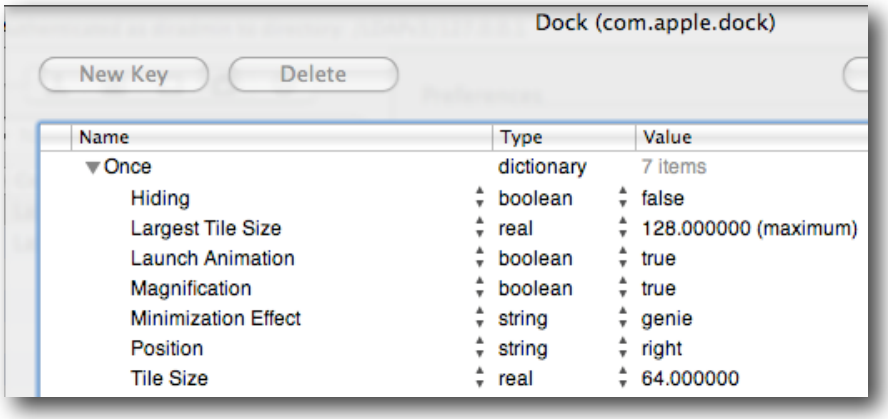
Managed client settings are nothing more than property values stored in the directory. Locally, you see the preferences stored in /Library/Preferences for the computer and in ~/Library/Preferences for a specific user. When a network directory is used to store these settings, they live inside a specific domain, such as a managed group (also referred to as a workgroup) or a computer group. The values, in both cases, are stored in XML format. Here are several ways to look at the same type of data, depending on where it is stored:



Local user's Dock settings viewed from Property List Editor

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>autohide</key>
    <true/>
    <key>magnification</key>
    <true/>
    <key>mod-count</key>
    <integer>627</integer>
    <key>mouse-over-hilte-stack</key>
    <true/>
    <key>orientation</key>
    <string>bottom</string>
    <key>persistent-apps</key>
    <array>
        <dict>
            <key>GUID</key>
            <integer>572026437</integer>
            <key>tile-data</key>
            <dict>
```

User's Dock settings viewed as raw XML



Network-managed Dock settings from inside Workgroup Manager

Attribute Name: dsAttrTypeStandard:MCXSettings

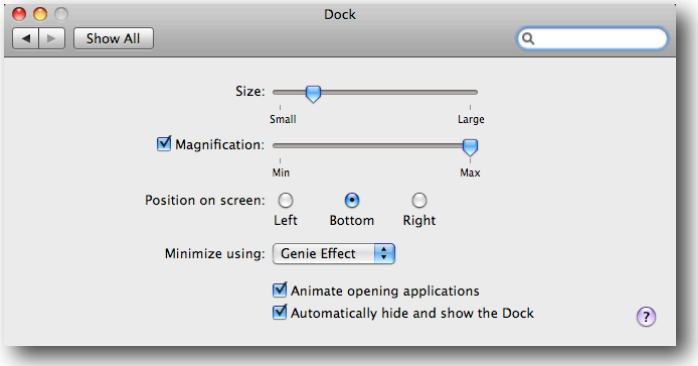
Text:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>mcx_application_data</key>
    <dict>
        <key>com.apple.dock</key>
        <dict>
            <key>Set-Once</key>
            <array>
                <dict>
                    <key>mcx_data_timestamp</key>
                    <date>2008-07-17T23:26:18Z</date>
                    <key>mcx_preference_settings</key>
                    <dict>
                        <key>autohide</key>
                        <false/>
                        <key>largesize</key>
                        <real>128</real>
                        <key>launchanim</key>
                        <true/>
                        <key>magnification</key>
                        <true/>
                        <key>mineffect</key>
                        <string>genie</string>
                        <key>orientation</key>
                        <string>right</string>
```

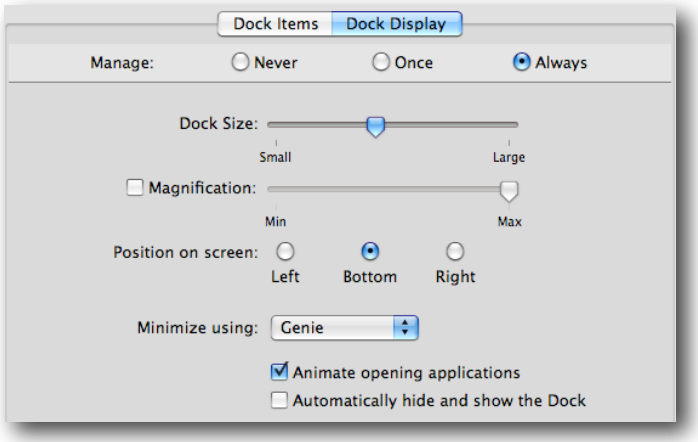
Network-managed Dock settings viewed raw using Inspector

Setting the values locally for a single computer makes sense, but setting the values as they will impact hundreds or thousands of users may not. Certain preferences can be set on a base system—what is referred to as *Client Zero*—and then duplicated to other computers for deployment. However, you might want to keep the base system pretty clean, then use managed preferences to establish the rules for the deployed systems based on who gets access to those computers.

The way most preferences are set in Workgroup Manager for a set of systems isn't that much different from the way those same values are set on a local computer. For example, here are the local and network Dock settings:



Local Dock settings



Network (Workgroup Manager) Dock settings

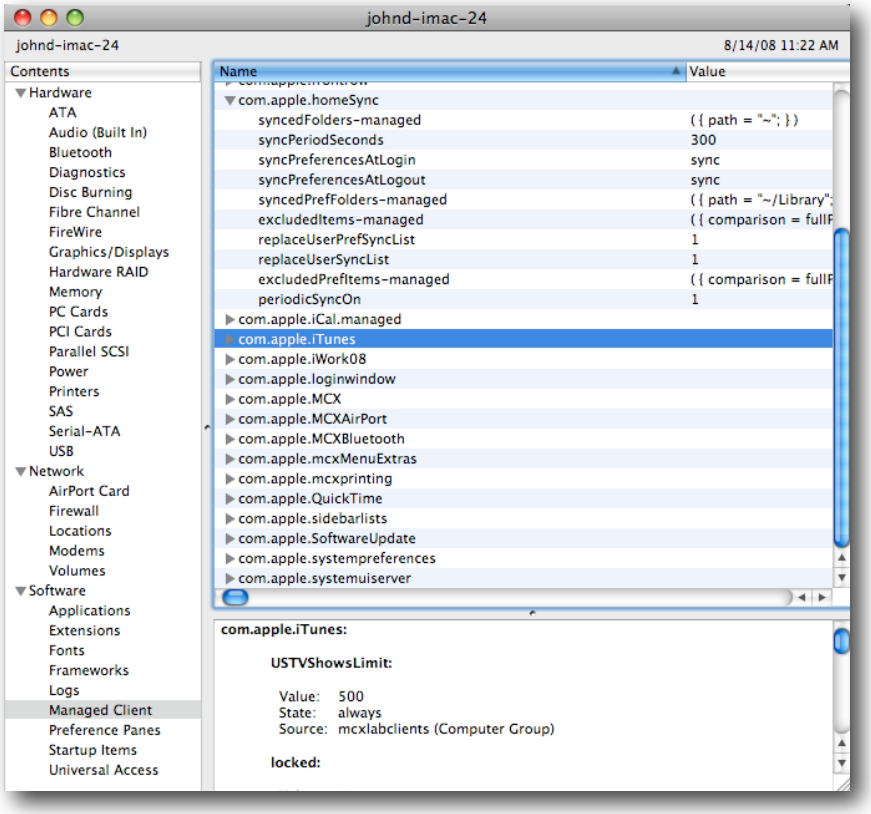
Preferences Interaction—the Rules of MCX

Before looking at the actual setup of managed preferences on a system, it's good to have a basic understanding of how all the parts interact. The preferences set within Workgroup Manager are stored in one of four domains: user, (work)group, computer, or computer group. Some values can only be set at the computer or computer group level, such as Energy Saver.

The preferences settings follow a specific hierarchy. This ordering determines which items win out in a "left, right, center" or "on/off" argument and establishes the order of listed items, such as in the Dock. The user account has the highest priority, the workgroup the lowest. Preferences that a user sets for himself or herself are stored inside the user's home directory. Preferences set at the directory level for a user are stored in /Library/Managed Preferences/<username>. All other preferences are stored in the cached "mcxsettings" in the local directory. The account order in which these preferences are obeyed is (highest to lowest): user, computer, computer group, workgroup (a user

group with managed settings). The MCX system takes all of the preferences settings and pulls them together using a tool called the *compositor*. This tool takes all the different preferences settings and pulls them into a single property list (*plist*).

Some of the preferences are set to load at the login window, such as the display of the login pane (list or name/password) and the login window message. The rest of the preferences are applied at login for a specific user. You can view the final preferences set in a number of different ways. One is with the Terminal command mcxquery, which allows you to dump the output of the compositor for any specific directory combination. Details on how this works can be found by typing "man mcxquery" in a Terminal window. The simplest way to view the compositor results for a specific user is to log in as that user and then open System Profiler (choose About This Mac from the Apple menu and then click the More Info button). Here is what a set of values looks like for one user logged into a managed client system:



Managed user's MCX settings in System Profiler

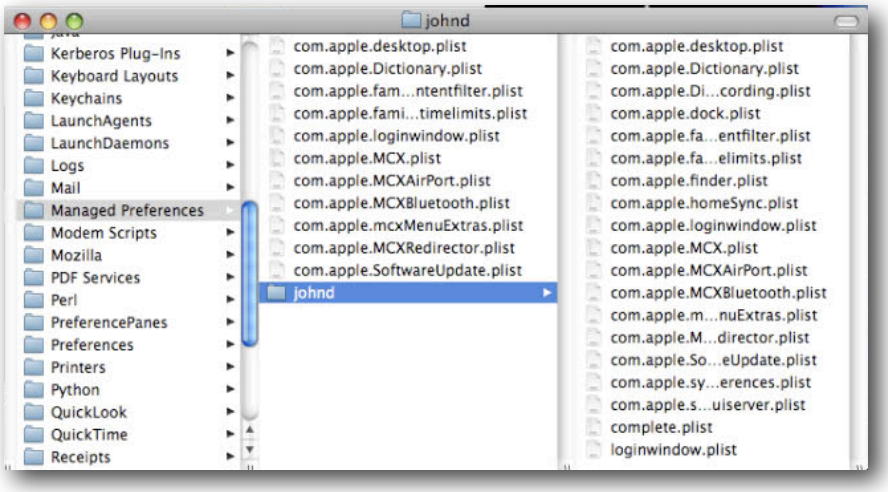
Note that the settings show not only exactly what was set, but also where each setting came from. For example, here the source of the iTunes setting is the computer group "mcxlabclients" and the setting is for Always. This is a great way to be able to compare what you thought you set in Workgroup Manager versus what the compositor finally ends up with for the end user.

The *hierarchy* followed by the compositor is defined by the different types of accounts that can hold preference settings. As the compositor works, it looks through the different settings and resolves them in one of two ways. First, it takes any direct conflicts, such as Dock (left)/Dock (right), and sets the value to that of the preference belonging to the highest account priority. Second, it takes any lists, such as Dock items or printers, and concatenates them into a single list, tossing out any duplicates and ordering the list into a sequence from the highest priority to lowest.

The *dominance* is how well the preference *sticks* or retains its settings. There are four settings, three of which are actual domains.

- The Always setting means that a preference is set by the administrator to be permanent. Settings made in this domain usually cannot be altered by the end user. The Always setting is often used for those settings that a user should not change, such as locking down items in the Dock that must be there all the time.
- The Often setting is the domain provided in the Details section of the preference editor. This domain allows the administrator to establish a setting, yet the user may be allowed to edit that preference during the session. With this setting, the user must have the ability and permission to access the setting and change it. At logout, the settings revert to the initial values set by the administrator. This domain is also the default location for non-standard settings, such as a preference file added outside of those in the graphical user interface. The Often setting is very useful in training when you might want users to experiment with settings, but have them revert to the presets at logout.
- The Once setting is the domain that an administrator uses to establish a starting point for a preference. The value has a timestamp attached to it. If the user is allowed to make changes to the setting, when the user does so, the timestamp changes. As long as the administrator doesn't touch that setting again, the user retains ownership of the setting. If the administrator changes the setting, the timestamp is now newer than that of the user, so the setting now displays the value set by the administrator, "once" again. Once is best used as a guideline for new users or as a template. A group of users in training may be exposed to a series of settings. If you use Once to define those same settings for their production computers, they will start off with the settings from training but will change them to suit their wants or needs.
- The fourth setting is Never, which does not mean unmanaged. It means that the setting is not defined at that level, so the defaults must be used.

The various plists used to define management settings are stored in two places. Values applied to the entire computer, regardless of user, will show up in /Library/Managed Preferences. The values that pertain to a specific user's experience at their login will show up in /Library/Managed Preferences/<username> and will include the individual settings as well as the final result file for the compositor to use. That file is titled "complete.plist."



Complete MCX plist set for a logged in user

Preference Enforcement and AUPs

Although it may seem like a really good idea to lock down the client systems, you should consider that the computer is there to help the end user accomplish assigned tasks. Try to set as few preferences as necessary to get the job done. Setting up too many restrictions can cause many users to spend an inordinate amount of time trying to get around the restrictions. If you need to enforce restrictions, start with an Acceptable Use Policy (AUP). No amount of technology will secure your client systems if you do not aggressively enforce the AUP.

Configuration Tips for Mac OS X Server

Setting up your environment for management involves some basic configuration of both the back end architecture and the client systems. This section deals with the basic settings to make sure your server or servers are properly configured to support MCX.

The services you need to provide will depend entirely on the size and scope of your deployment. This section covers those settings that are required but does not provide information about all aspects of the Mac OS X Server setup. Several ideas for extending the architecture for workflow and collaboration are included later in this document.

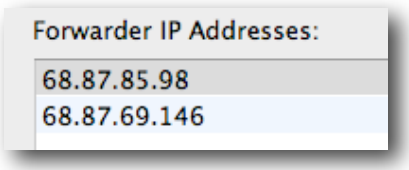
You need to provide both authentication and authorization. To support those two key components, you must provide the following services:

- DNS, to resolve server names for clients
- Network Time Protocol (NTP), for keeping everyone on the clock
- DHCP (or fixed IP services)
- Open Directory or some directory service
- AFP (preferred) or SMB, for home directories and workflow
- Other services as desired, such as Web, Software Update, iCal, and so on

DNS (Domain Name Services)

Every server should have a static IP address and have its name registered with a Domain Name Services (DNS) server. This allows the clients to locate the server and the directory services to store paths to settings in a clear, concise manner. Mac OS X, being UNIX, requires DNS, so don't bother trying to work around it. If your upstream ISP or IT staff provides you with a fixed address and name, you are all set. If they do so, make sure that the name and IP can be resolved from within the network you are operating on. If they can't provide this, you can set up your own DNS easily.

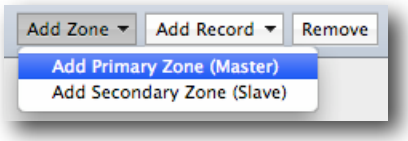
Server Admin has a completely new DNS interface for Leopard. When you first set up your server, you might have to enter an upstream server for your DNS setting, but you can change that after setup. It does not interfere with any DNS service being provided by your ISP. First, open Server Admin and connect as the local admin to your server. Select the DNS service (if you don't see it, add the service in the Services pane (click the Settings icon, then the Services button) when looking at the server itself. Select the Settings tool under "DNS." Add your upstream DNS settings to the Forwarder IP Addresses window:



Upstream DNS servers

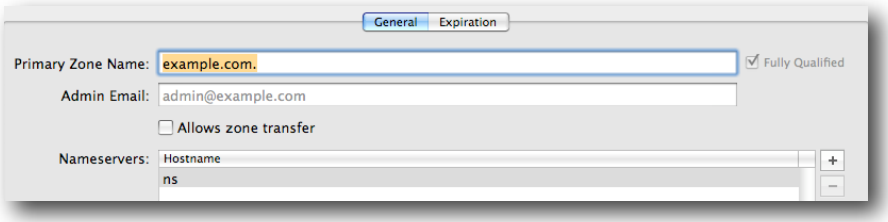
Note: When you make entries in Server Admin, always press Tab after each entry. This makes sure the value is written properly.

The forwarder addresses allow you to point your clients at your DNS server yet allows them to look up any Internet address. Next, select Zones and choose Add Primary Zone (Master) from the Add Zone pop-up menu.

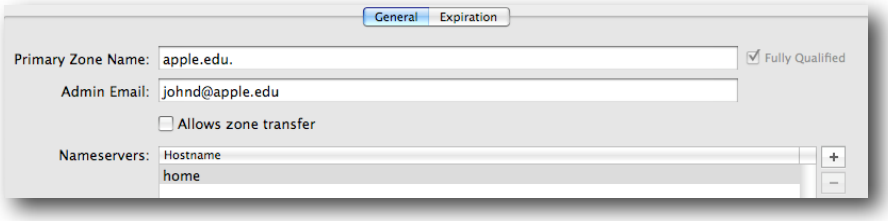


Choose Add Primary Zone (Master) from the Add Zone pop-up menu.

The initial values need to be replaced with your own settings:

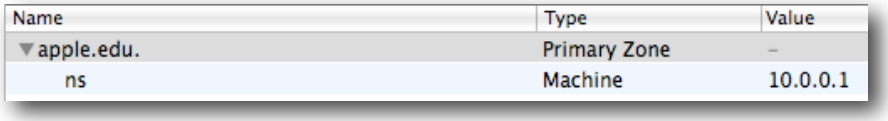


Change the example values to those of your actual server



… like this.

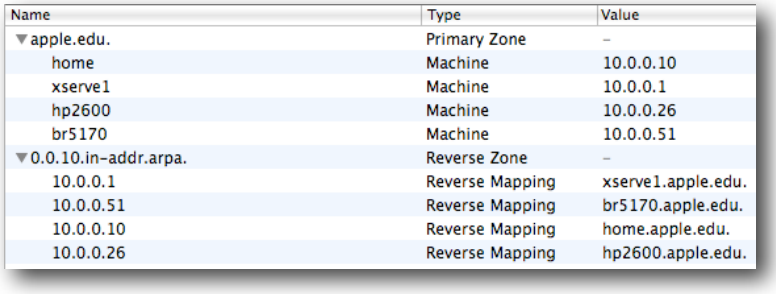
Next, select the new zone, click the disclosure triangle to edit it, and change the "ns" value to match your server.



Name	Type	Value
▼apple.edu.	Primary Zone	–
ns	Machine	10.0.0.1

Change the "ns" value.

Add any other "A" records that are needed. Note that the reverse settings are automatically applied. When you are done, you will have something roughly like this:



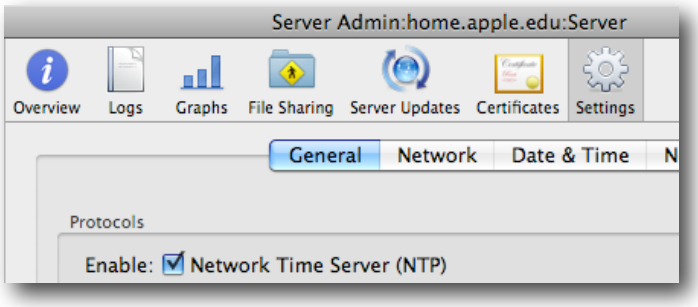
Name	Type	Value
▼apple.edu.	Primary Zone	–
home	Machine	10.0.0.10
xserve1	Machine	10.0.0.1
hp2600	Machine	10.0.0.26
br5170	Machine	10.0.0.51
▼0.0.10.in-addr.arpa.	Reverse Zone	–
10.0.0.1	Reverse Mapping	xserve1.apple.edu.
10.0.0.51	Reverse Mapping	br5170.apple.edu.
10.0.0.10	Reverse Mapping	home.apple.edu.
10.0.0.26	Reverse Mapping	hp2600.apple.edu.

A properly configured DNS

Your final action will be to open System Preferences on your server and change the DNS settings for your network interface to point back to itself. This is needed to make sure all other services work properly. Because your server has the ability to forward any queries it can't answer, it will be a good local DNS for your users.

Network Time Server (NTP)

You can turn on NTP on your server, if you wish, to provide a clock check for your clients. If you don't do this, make sure this server as well as all clients are pointing to the same NTP server.



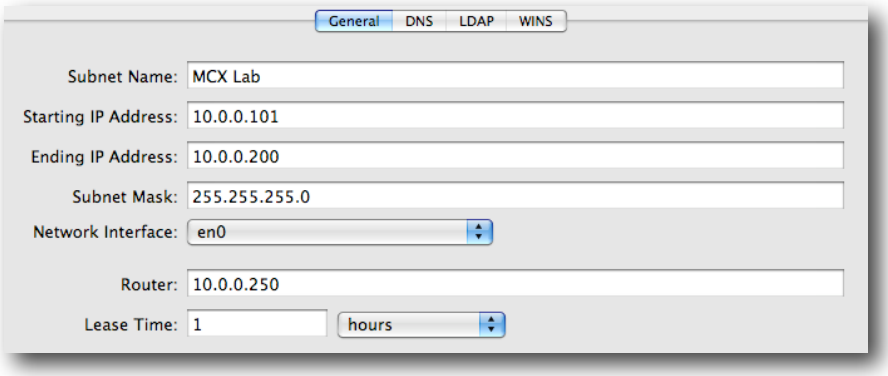
Enable NTP by checking the box.

NTP is so important because directory services use Kerberos for user authentication. This service demands that all clients have their clock within five minutes or less than that of the server. If the clocks are off, users cannot
log in.

DHCP (IP Services)

All networked clients require an IP address, and Mac OS X Server works as a good DHCP server. If you are already getting IP addresses from another device on the network, you don't need to worry about setting this up. If you do want to set it up, here are the steps involved.

In Server Admin, select your server, then the DHCP service. Select the Subnets tool and click the Add button (+) to create a new set.



Enter the settings so they support your network.

Do not create your own DHCP server if one is already running on your network. Go ahead and add the DNS entries, and leave the rest alone for now.

Open Directory

To provide centralized management policies, you must have a directory to hold those policies. If you are using Active Directory or eDirectory, you can extend the schema to add MCX settings. This subject is not covered here; for assistance, you may want to consult Apple Professional Services. This section addresses creating an Open Directory Master to contain the management settings used by all the client systems. If you are in a small enough environment, with less than 1,000 client systems, you can use the same server for your user accounts that you use for your management settings. This example describes setting up the authentication directory separate from the authorization directory. Doing it in this way allows you to learn to keep these items separate for planning purposes and lets you expand as needed when your installed base grows.

Suppose, for example, that you are required to use Active Directory for your user accounts. If you had no access to manage those accounts, the concept of adding MCX settings could be daunting. However, if you deploy a Mac OS X server to provide the policies to the Mac computers, you can easily work with that environment. This concept works even if you have a large Open Directory directory, such as a single district-wide personnel database and differing management needs in each school. You can deploy an Open Directory Master as the MCX server in each school to provide unique management for that environment, but still require the users to authenticate to the district database.

Setting Up the User Directory Server

If you are providing your own Open Directory Master for user authentication, with or without the MCX settings on this system, you need to follow these steps. If you are using a different directory for user accounts, you can skip to the next section.

First, you need to make sure your server has a valid fully qualified domain name (FQDN). Using Network Utility or another tool, check to make sure you can do forward and reverse lookups of your server both from itself and a client system.

Next, in Server Admin, connect to your server and select the Open Directory setting in the Sidebar (add it first in the Server/Settings/Services pane if necessary). In the General pane, click the Change button. Select Open Directory Master and add a password for the Directory Administrator (diradmin) account. For Master Domain Info you should see both a Kerberos realm and a search base. Confirm the settings.

Example settings for an Open Directory Master at setup

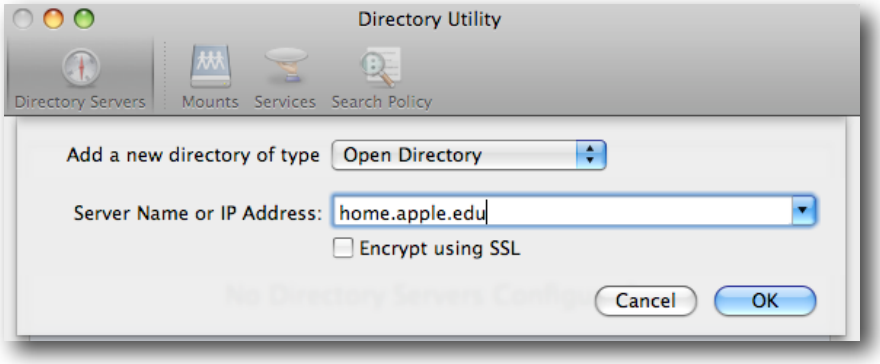
If you don't get settings similar to this, go back and make sure your DNS setup is working.

Do not use the previous setup if you will have an upstream directory server. Because that server would have Kerberos running on it, you must do the following extra steps before setting up your Open Directory Master.

Setting Up the Managed Client Directory Server

This setup would be used when the Open Directory server will be part of a larger directory environment. This can be used for an MCX server that is secondary to any other directory server; for example, using an Open Directory server for management or an Active Directory server that would contain the user records. For this document, an upstream Open Directory server is used that contains the user records instead of an Active Directory server.

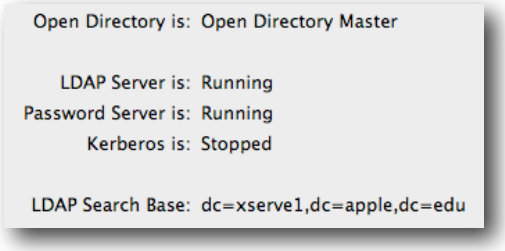
First, you must log into the MCX server as the local administrator and open Directory Utility. Unlock the window and click the Add button (+) and choose the type of directory your upstream server uses. In this example, Open Directory is chosen and "home.apple.edu" is the user account server.



Binding the MCX server to the Open Directory server with user accounts

You do not need to make this an authenticated bind unless you will also be using the MCX server as the wiki, blog, or collaboration server. Anonymous binding will work fine for now.

Now open Server Admin. In the Open Directory settings, make this server an Open Directory Master. The process is almost the same as in the section above, but you won't get any Kerberos information. Mac OS X Server senses that it is bound to a Kerberos server and won't activate the code on your MCX server that will create a Kerberos realm of its own. When you create the "diradmin" account, you need to change it to be different from the directory administrator account on its parent server. You can use something easy such as "MCX Admin" with a short name of "mcxadmin." Otherwise, you'll get authentication errors trying to connect. The final setup for this MCX server looks like the following:



The MCX server does not have Kerberos running.

Apple Filing Protocol (AFP)

The MCX server may not have any share points on it but it is still good to understand the general process. User account types and usage are discussed in "User Accounts—MAs, PHDs, and More." One planning criterion is making sure you do not overload any of your servers in such a way as to impact other services. If you have a server providing policies, it can handle some other tasks; however, one of those tasks should not be providing home directory support. If you are using dedicated network user accounts with network-based home directories, the load on the file server can be immense. If you are using mobile accounts with portable home directories, the load can still be quite heavy. The best practice for this is to have dedicated home directory servers.

This doesn't mean that you can't use the MCX server for other purposes. You can house common or group share points on this server as well as possibly support some of the new collaboration services, such as a wiki or blog. This section provides information about setting up basic file sharing to support all of these possibilities. You can then decide which ones to activate on your various servers.

Basic AFP Service

Basic AFP, or AppleShare, service in Leopard doesn't require a detailed explanation. You just turn it on. Guest access is off by default, so any share points are off limits to anyone but authenticated users. If you want to allow guests, select the Enable Guest Access checkbox in the Settings/Access pane.

Creating Share Points

In Leopard, the share point setup has moved from Workgroup Manager to Server Admin. This organization makes sense because this location is where all of the services are established. To create a share point, open Server Admin, connect to your server, and click the File Sharing icon in the toolbar. Using the Browse pane, you can locate or create the folder you wish to share. Just click the Share button and then save. Setting permissions and more advanced settings, such as access control lists (ACLs), are covered in the product documentation.

Automounts

With Leopard, the new AutoFS code is a highly resilient and reliable mechanism for mounting volumes. A best practice would be to make sure the number of automounts you configure per server is as small as possible. When you have many servers on the same network, the clients still have to keep track of all of them. However, with Leopard, the share points can mount and dismount as needed, without having to stay mounted at all times.

To set up a share point as an automount in Leopard, you use Server Admin to select the share point within the File Sharing tool, then select the Enable Automount checkbox. Choose the type of automount, usually User home folders, and then click OK. Authenticate as the directory administrator on that server (remember the MCX server that was set earlier is "mcxadmin," not "diradmin").

This completes the basic setup of the server. More information about these configurations is provided later in this document.

Configuration Tips for Mac OS X Clients

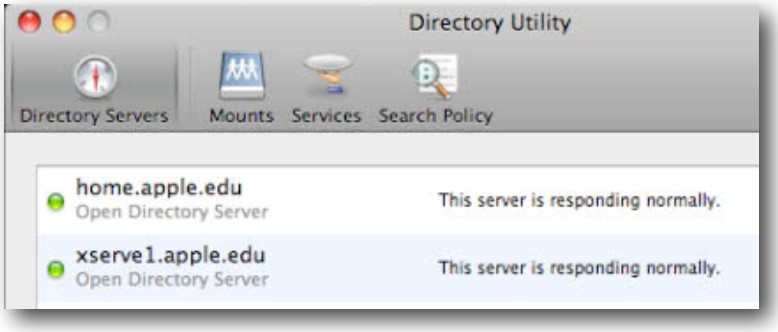
Setting up Mac OS X clients for MCX is pretty straightforward. All that you need for this process is a client system running Mac OS X v10.5.6 or later (highly recommended); also supported are v10.4.11 and v10.3.9). A few of the aspects of Tiger client behavior within the Leopard MCX world are discussed. However, Panther systems are not discussed other than to note that the basic management settings do work with Panther.

Clients should have all the appropriate software installed. If a system will be used by many people, install all the needed software and use MCX to control access to the software as needed. At least one client system should have the Server Admin software installed to allow this system to act as Client Zero for purposes of setting specialized MCX configurations.

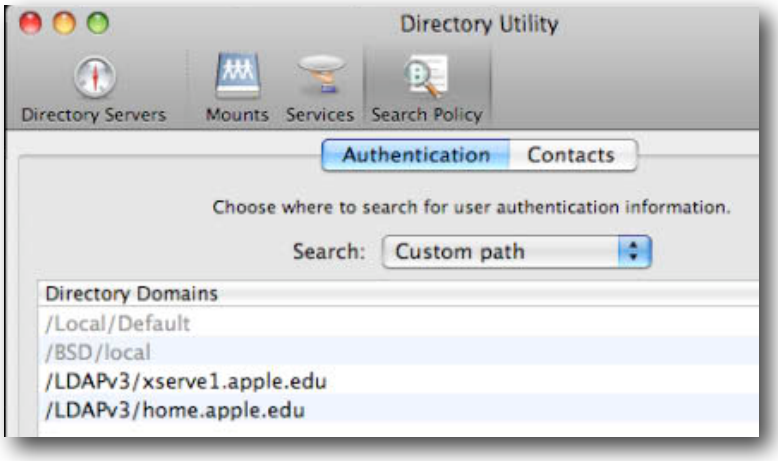
All clients should have a local admin account, hidden or not, so that basic maintenance can be performed on and off network. This is also the account you would log into to run Workgroup Manager and perform MCX setup. All clients need to have unique names, and remember to avoid special characters in the names.

The most difficult task that needs to be performed here is binding the client to the management server or servers. If you have only one server providing both user authentication and management authorization, a single bind is all you need. Dual binding is needed if you are attaching to a directory for access to user account information as well as to a MCX server for management settings. The tool for this is Directory Utility located in the /Applications/Utilities folder (different from the Directory tool, also located in the Utilities folder).

Log into the client as a local administrator and open the Directory Utility application. Unlock the settings and click the Add button (+). Enter the MCX server's domain information in the window, click OK, and wait for the dialog to go away. Click the Add button (+) again and enter the user directory server information. When you're done, check to make sure the search path for directory services is correct. What you want is for the system to search through the local directory first, then through the MCX server, and finally to the user directory. This insures that all MCX settings are picked up while trying to authenticate as a user. If you switched the order, the user may log in but have no management settings.



Directory Utility window



The list is alphabetical—but the search order is correct.

With this done, you can now set the MCX values for your client systems.

Basic MCX Configuration Setup

Setting up basic management policies for Mac OS X requires a few basic considerations: determining the level at which management will be applied, establishing baseline preference settings, and then testing the setup and tuning.

The first consideration is based on which domain, or domains, you want to use for management. The domains are user, group, computer, and computer group. With earlier versions of MCX, as well as Macintosh Manager for Mac OS 9, setting policies at the group level was often recommended. This was because workgroups were relied on heavily for both user management and for workflow. What this resulted in was users having to choose among several different workgroups at login to have access to a group folder and group settings. Users can still be required to log into specific workgroups, but Leopard now allows a more streamlined approach. With Leopard MCX, workgroup preference settings are combined by default into a single set of values. This means that instead of having to choose between the Math, Science, or Language Arts workgroups when logging in, a user can just authenticate and be taken directly to the desktop. All the settings for each of those workgroups are composited together, providing you with all the Dock items and a composite of all the other settings.

This is a beneficial change. Users had asked for a change because teachers and students would often have to log out and in again between classes or pick a specific workgroup for every class. With Leopard, a student, teacher, or other faculty member can log in as herself or himself and get a single work environment that encompasses all the person needs to do for that day. This is especially important as the number of 1 to 1 deployments increases with users who carry their own computers from class to class. They can log in once and be done with it for the day.

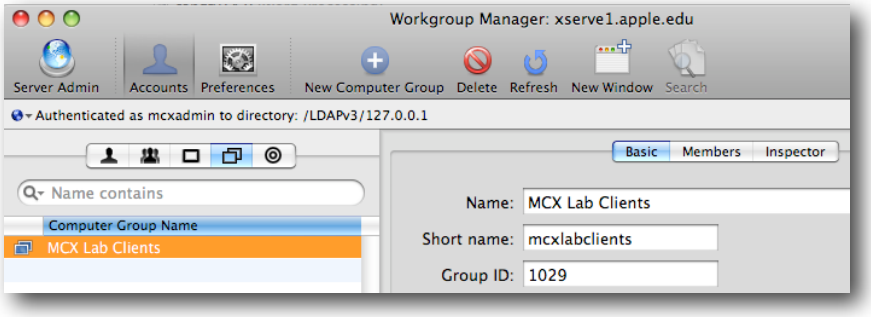
That said, you will want to set up the management scheme to best support this new capability yet allow for workflow and use of both Leopard and Tiger systems. To do this, the best practice is to use computer group level management most of the time, with some use of workgroup settings for unique circumstances. This works best because you can streamline the user experience and make it similar between Leopard and Tiger clients. Under the new group combination setting, Leopard users get a single user setup at the desktop, but Tiger users would still get the workgroup picker. If the management was more computer group based, both sets of the systems would respond in the same way.

So what are these different domains, and how do they relate? The user domain is the user account. Setting preferences from the network for a user is possible but is probably much too granular. If you set preferences for a specific user or mass set them for a group of users selected together, you will have to edit those preferences later one by one. Where individual preferences can work well is allowing special permissions for a user outside what that user's group or groups or computer group or groups allow. The user domain has the highest rank in the hierarchy, so settings made at this level override any made to the same preference at any of the other levels.

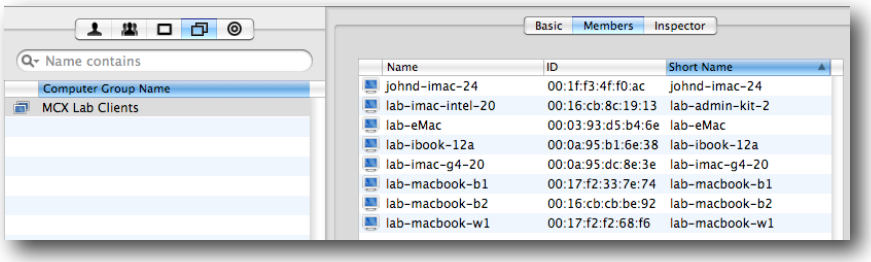
The computer domain is just that—the individual computer record. Just like a user record, a computer record can contain unique settings to vary it from the rest of the computers in any specific computer group. A special case is the Guest computer, which is defined like any computer bound to the directory server but is not specifically designated.

Note: In Leopard, the Guest computer account is no longer available by default as it was in Tiger. To enable and use the Guest computer account, in Workgroup Manager, choose Create Guest Computer from the Server menu.

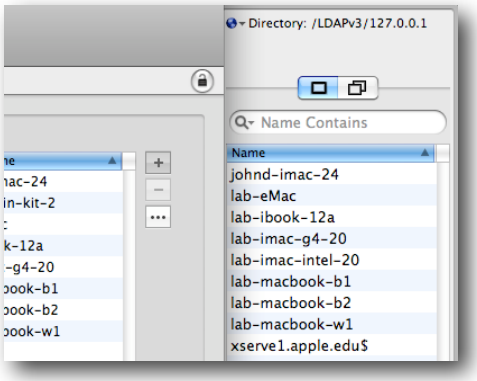
Computer groups are new to Leopard. Tiger used computer lists. A set of computers could belong to a single list, such as "teacher_computers." With Leopard, you can assign computers to as many groups as you need, the same as assigning users to different groups for various administrative and management needs. You can have a single system in the computer groups "teachers," "portables," "Middle School," and "Science" all at once. These groups may or may not have any management assigned. Computers that are bound to the directory can be added to any computer group. The computer is tracked by its Sharing name and the primary (onboard) MAC address. If a computer has more than one network interface, the computer record will contain multiple MAC addresses.



A computer group with…



many members, added from…



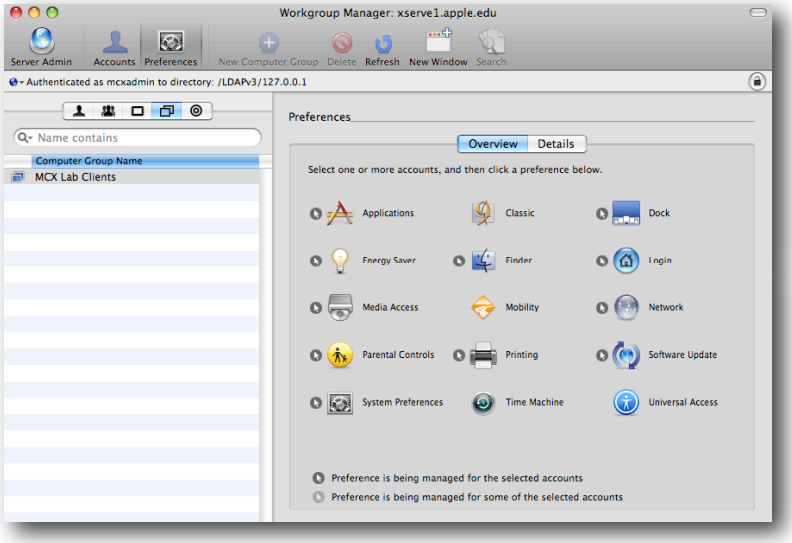
the list of bound computers.

Workgroups are user groups with preferences assigned. This group holds the lowest priority and should be used only when the settings must be unique for a specific set of users. Because you can create "group" folders that reside together in a common share point (for more information, see "Workflow and Collaboration Tips"), you don't need workgroups just to provide the standard hand-in folder any more. Where a workgroup might come in handy is when setting up a special workgroup for loading share points at login that aren't normally used. Then again, you could also set that same setting at computer group level—it depends on the usage and client deployment model.

Next, this section will look at the basic settings you need to employ to begin managing your workstations. You begin by logging into an administrator computer or client with server tools installed as the local administrator. Do *not* run Workgroup Manager from a server. It's not that it won't run, but the server isn't configured or loaded as a client so it will be much more difficult, if not impossible, to set many of the preferences. You would need to install all the applications, widgets, and other tools on the server just to be able to set management values. You also wouldn't be able to mount share points that you'll want to add to Dock or Login items later on.

Login Preferences

The first indication a user gets of being managed is at the login window. It is also a very good way of insuring that MCX settings have made it to the client. To set up Login preferences, select either the Guest computer account or one of the computer group accounts. Select the Preferences tool.



Setting preferences at computer group

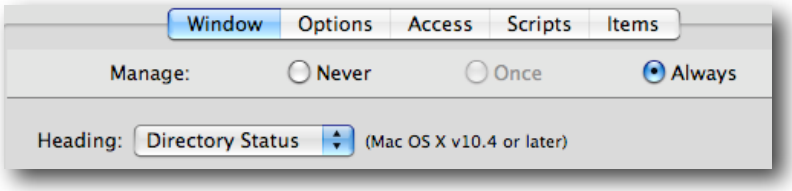
Now select the Login item. Five different sets of values can be added to the Login MCX domain: Window, Options, Access, Scripts, and Items. For the basic setup, a few of these are explored here, leaving the rest for later.

Login preference buttons

The Window pane is discussed first. It is here that you can set the view of the login window, add a message for the end user, and display system information.

Window

In the Window pane, select Always and choose Directory Status from the Heading pop-up menu. The Heading is also called AdminHostInfo in the database and is the set of values that is displayed in the login window underneath the large "Mac OS X." If you click these values, you will rotate through the set—Name, OS version, OS build, serial number, IP address, directory status, and time. The default for any system is the name of the computer. In this example, this is set to Directory Status so you can see at a glance if the client is properly bound to its directory server or servers.



Setting the login window to display directory status shows …

that you are bound to your servers correctly.

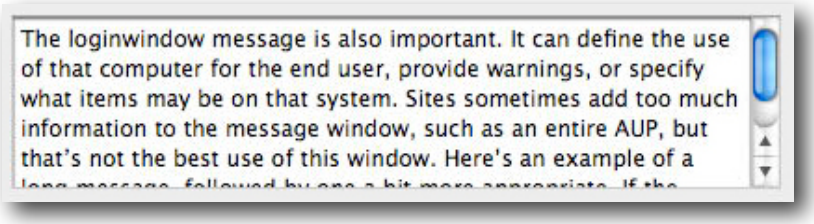
This colored circle usually starts out red at startup time, then turns green within 30 to 45 seconds (or faster). There are four possible values for this window:

One or more of your directory servers isn't reachable.

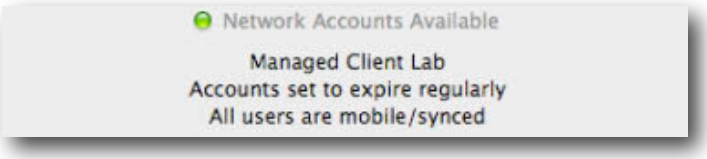
None of your directory servers is reachable.

The fourth value is "Network Login Required," a yellow tag, showing that your client is managed with 802.1x (Radius).

The login window message is also important. It can define the use of that computer for the end user, provide warnings, or specify what items may be on that system. Sites sometimes add too much information to the message window, such as an entire AUP, but that's not the best use of this window. Here's an example of a long message, followed by one a bit more appropriate. If the message is longer than the basic window, scroll bars appear.

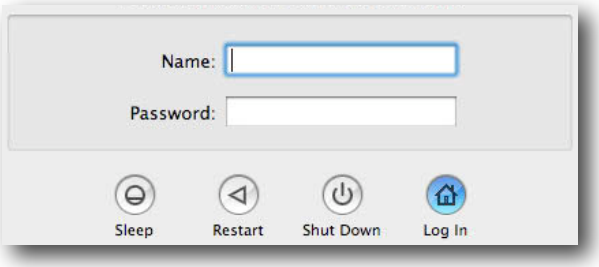


A bit much…



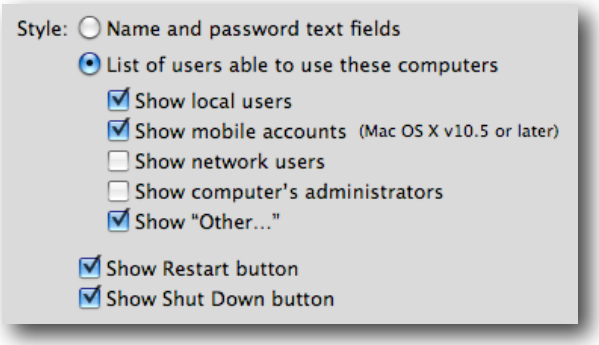
A more appropriate login window message

The final portion of the login window to define is the display style. You can choose to have users view just a name and password field set or display user names with a picture in a list. If you are using 802.1x authentication, you must use only the name and password set because the user list isn't cached until after the network connection is established. In some schools, or even just certain grades, having the student picture and name as part of the login window is a good idea. Adding a Picture field to the user directory with a path to a small image enlivens the user experience. If you have thousands of users, however, this could create a delay while the login window loads and also forces users to scroll through a long list. You can also choose whether or not to display the Restart or Shut Down button or both.



Different steps for the login window

The list view can be trimmed to display only local accounts, such as the Guest account and mobile accounts, and can hide administrators and network users. In this case, if a user logs in for the first time as a network user with a mobile account, he or she would use the Other selection. After that, the user's account would be visible in the window.



Style options

Options

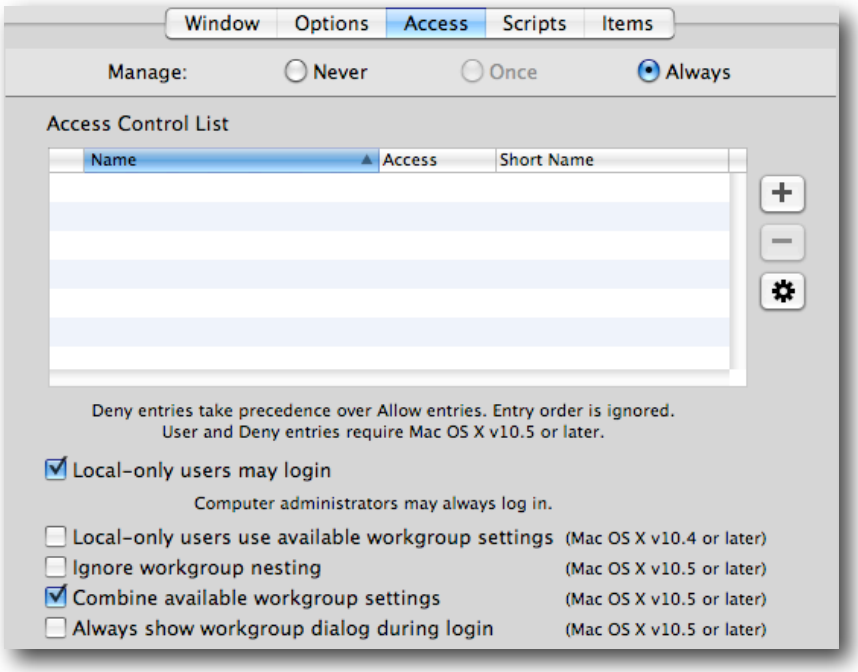
The Options pane contains settings that limit non-standard logins, force logouts, and determine if local administrators are managed or not. It also contains the flags for globally enabling guest and external accounts. These two topics are discussed in detail in the user account section.

Note the v10.5 or later tags.

You can force even local administrators to be managed. This might work well in a school in which some users have been allowed to be local administrators of school-owned computers in order to install software but are inexperienced otherwise. With this option selected, that user can remain a local administrator and can install software but not take other actions.

Access

For now, the Access settings can be left in their default mode. This is the pane where you force local users to adhere to network management settings, allow or ignore nesting, combine workgroup settings, and allow or suppress the workgroup picker. The difference between nesting and combining is that you can create a managed group with preferences, then place that workgroup inside another workgroup. Nesting allows all values to be taken into effect when the compositor runs. Not allowing nesting is the normal behavior for Tiger clients. Combining takes all of the workgroups at the same level and composites all of their settings together into a single experience. The Access Control List allows you to specify groups and users who are allowed and denied from logging in to that computer group. If you have several workgroups, you can place the few that should have access to those clients; only those group members will be able to log in, and only those workgroup settings will be combined.



The default settings for Options pane

Scripts

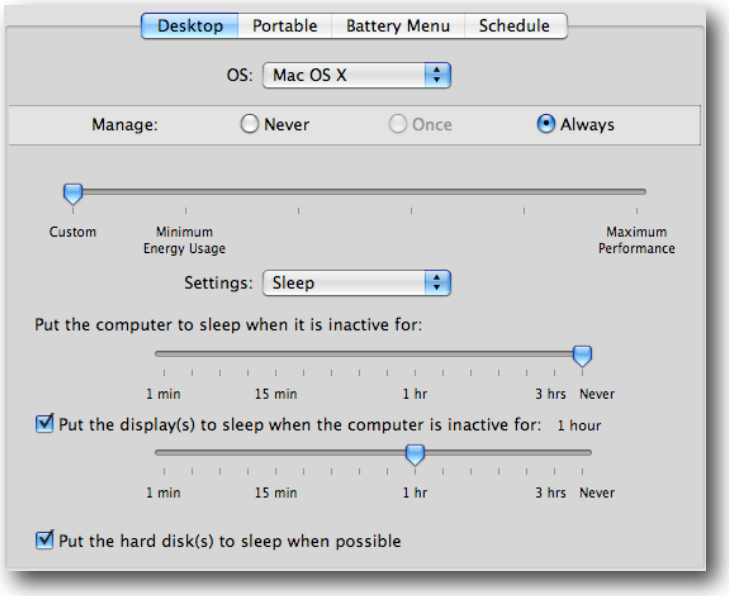
If you have a special action that needs to take place at login or some housekeeping to do, you can use the Scripts settings. The script gets stored in the directory, so you do not need to copy it onto your image, and you can change it as often as you need. More information on this is in the *User Management* guide. (http://images.apple.com/server/macosx/docs/User_Management_v10.5.mnl.pdf)

Items

This pane is discussed in "Workflow and Collaboration Tips." Here is where you add non-automount share points that need to be loaded. It also works well for having an application, document, or URL that needs to launch at login.

Energy Saver

These settings can only be done at computer or computer group. Energy Saver is one of the few MCX settings that takes effect at the login window. The settings affect all users. One benefit of the settings is that you can force the client systems to stay awake so you can reach the systems with tools like Apple Remote Desktop.

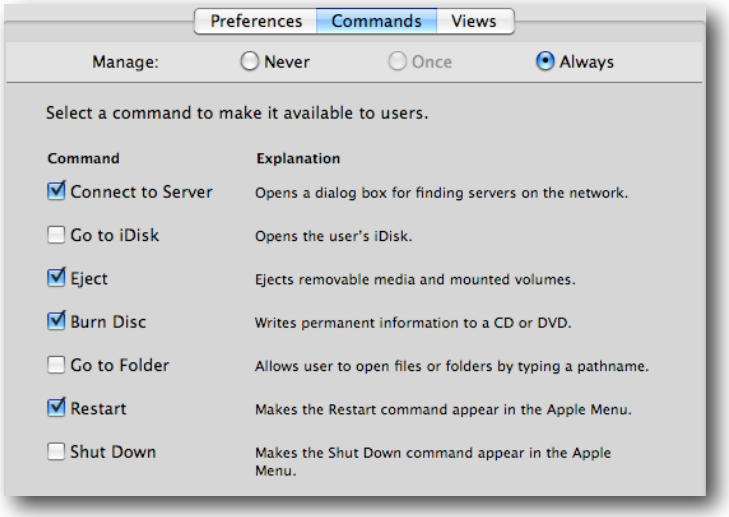


Defining Energy Saver settings

If you use the Schedule settings, try to stick to the sleep/wake setup versus the shut down/startup setup. Be aware that if you set portables to sleep, it shuts off the AirPort card, so reaching those systems afterward isn't easy.

Finder

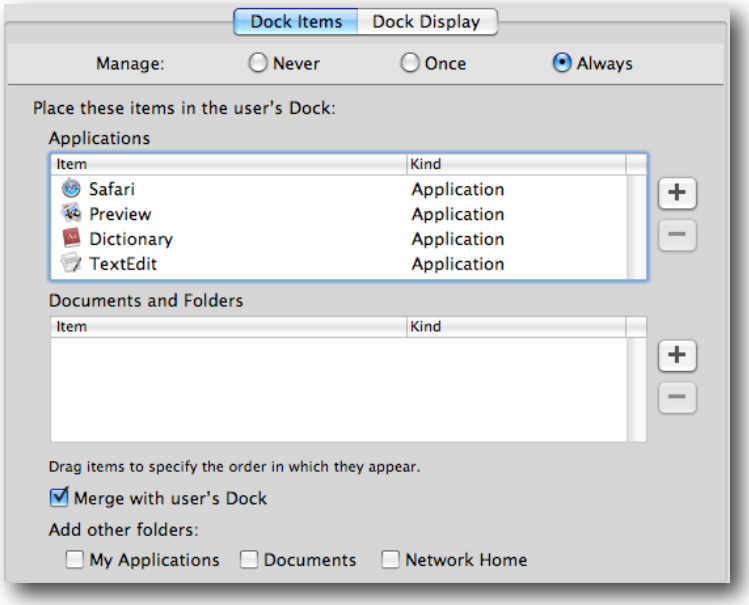
With the Finder, some of the most important settings are being able to force new Finder windows, set which Finder commands are allowed, and organize the desktop. For example, you can turn off the Go to Folder and Shut Down commands, stopping users from exploring places they shouldn't look and requiring them to log out versus accidentally shutting down.



Defining Finder command settings

Dock

The Dock can be set at several levels. Globally, you can assign items to the Dock that all users need to have yet also allow users to add and delete their own items.



Basic Dock settings with merge enabled

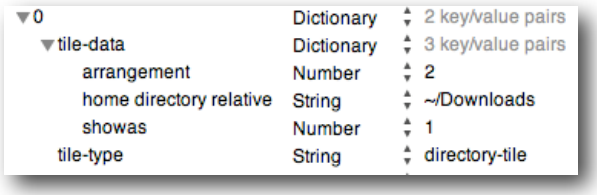
If you set the Dock to merge with the user's settings, you'll also get the system defaults. Those values are stored in the Dock application itself. Here's how to edit the defaults.

Log into a client system as admin or root. You can use TextWrangler (freeware) as admin, or use Property List Editor (from the DevTools) as root. Locate the default.plist—/System/Library/CoreServices/Dock.app/Contents/Resources/English.lproj/default.plist—and open it. TextWrangler works well to view the plist as XML:

```
<dict>
  <key>tile-data</key>
  <dict>
      <key>home directory relative</key>
      <string>~/Downloads</string>
      <key>arrangement</key>
      <integer>2</integer>
      <key>showas</key>
      <integer>1</integer>
  </dict>
  <key>tile-type</key>
  <string>directory-tile</string>
 </dict>
```

Viewing the plist with TextWrangler

Property List Editor lets you edit the keys a little more easily.



Viewing the plist with Property List Editor

Remove any of the items that you don't want to show up for every new user. Save the file and log out. The changes will remain until the next time the Dock gets updated.

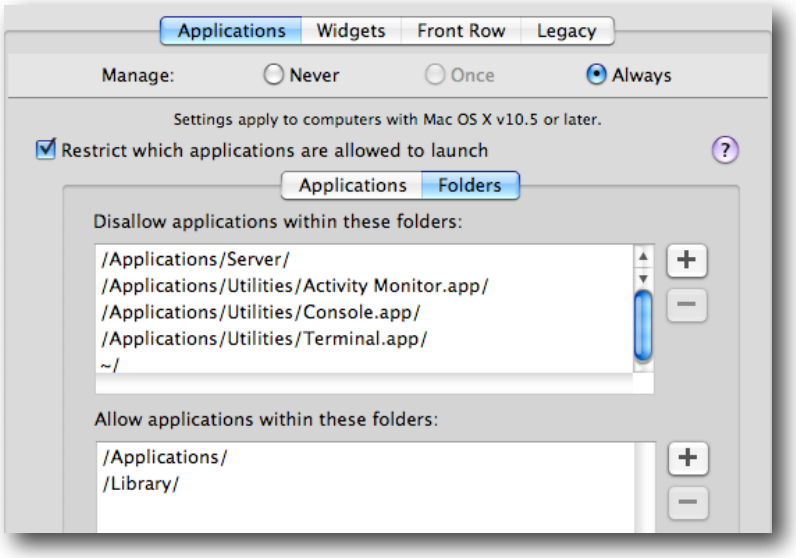
Applications

Management and control over applications has improved significantly since Tiger. In the past, a user could drag an application into his or her home directory and alter it to bypass restrictions, and schools had to totally disable Dashboard to stop users from running unapproved widgets. Leopard MCX has much better control.

Application Management

There are two different settings for application management. One is the ability to digitally sign applications to keep them from becoming altered. Although this is a great setting, it does not provide application restrictions. The setting does not have the ability to set "don't allow anything but signed applications to run" or something to that effect. This renders the signing ineffective as a control mechanism.

What works best for application management is path or folder restriction. You can set the locations where applications are allowed to run and locations that are forbidden. Here is an example:

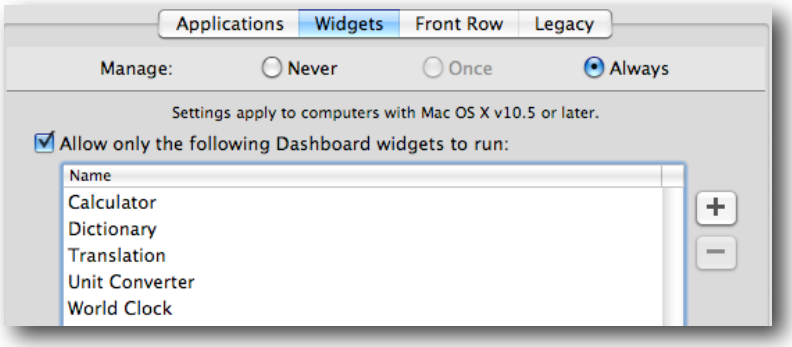


Setting what locations a user can run applications from

The idea here is that the default Applications folder is safe, and the /Library folder often contains sublaunched applications needed by mainstream ones. The user does not have permission to change either of these folders. You could further restrict /Library by designating just the Application Support folder. A few system administration applications, such as Sassafras' K2 tools, need to run as a user task from /Library itself. Note that the user's home directory is not allowed so even if users download an application to their home directory, they can't launch it. You could also deny use of Installer. Although the dialog says "Folders," what it really means is the path. This means that you can add the path directly to an application as denied. The rules work like a firewall. This means denies always win, and you can allow an application inside a denied folder except if you add the application to the Applications pane, it can be launched anywhere, including from inside the user's home directory.

Widgets

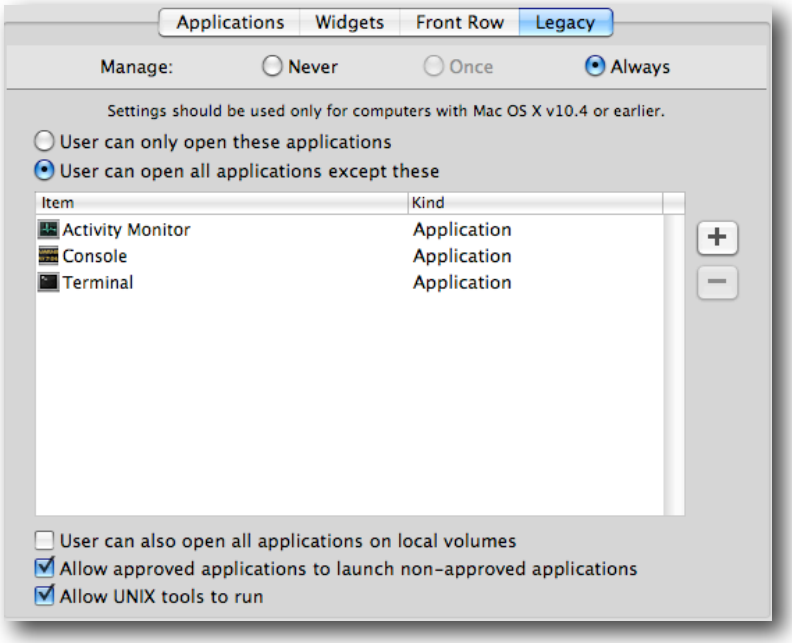
The reaction from many education customers to Dashboard and widgets was to turn them off. There is a better way to take advantage of the valuable widgets that are available. You can allow specific widgets to run and deny all others. Selecting the widgets you want users to have access to will automatically deny access to any others, including any inside the user's home directory.



Setting the allowed widgets

Legacy

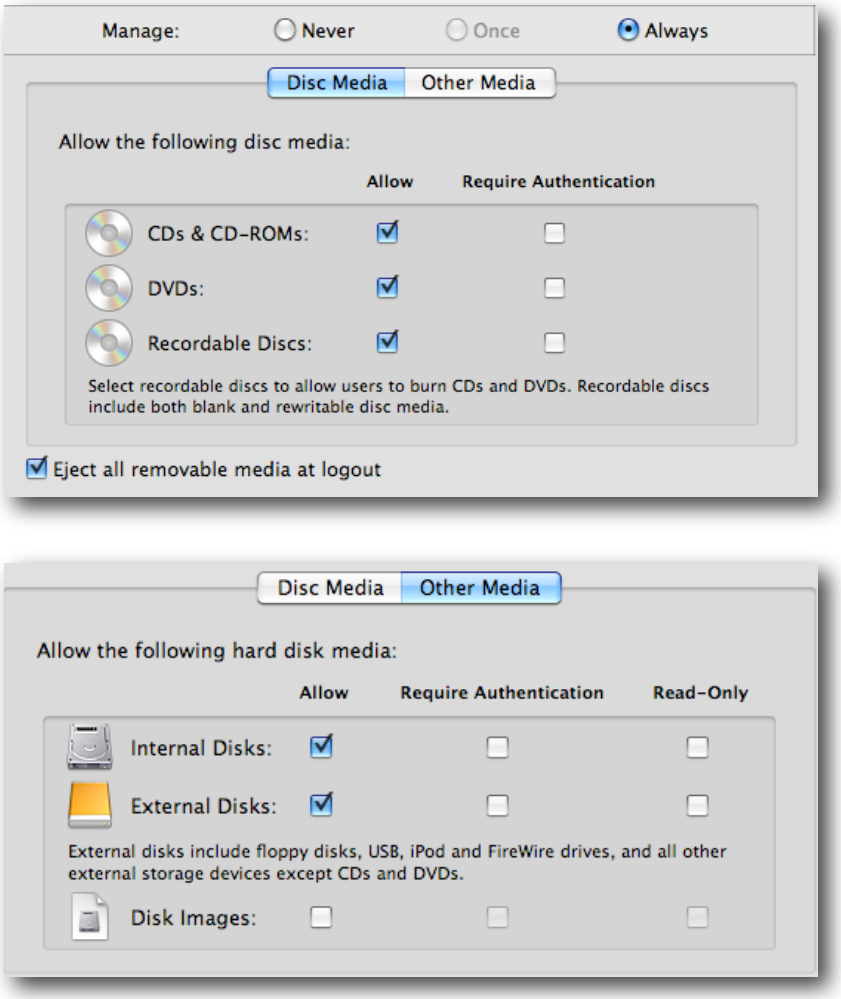
The controls here work as they did in Tiger. This is the only pane you can use to establish application restrictions for non-Leopard clients.



Tiger and Panther application restrictions must be set here.

Media Access

This preferences pane allows you to control access to read/write media such as CD-ROMs and recordable DVDs. You can also control access to external drives. One great item is the "Eject all removable media at logout" option. That setting alone should aid teachers who try to locate the CDs they handed out at the beginning of the class period.



Controlling access to disk images, external media, and so on

Network

The Network settings in Leopard allow for proxy settings and controlling access to Internet Sharing, AirPort, and Bluetooth. The settings that existed with Tiger to set the default web and mail applications are gone—because only Safari obeyed the settings. Be careful with the interfaces. Turning off AirPort for a group of desktop lab systems may make sense; accidentally doing that to a group of portables may be counter-productive.

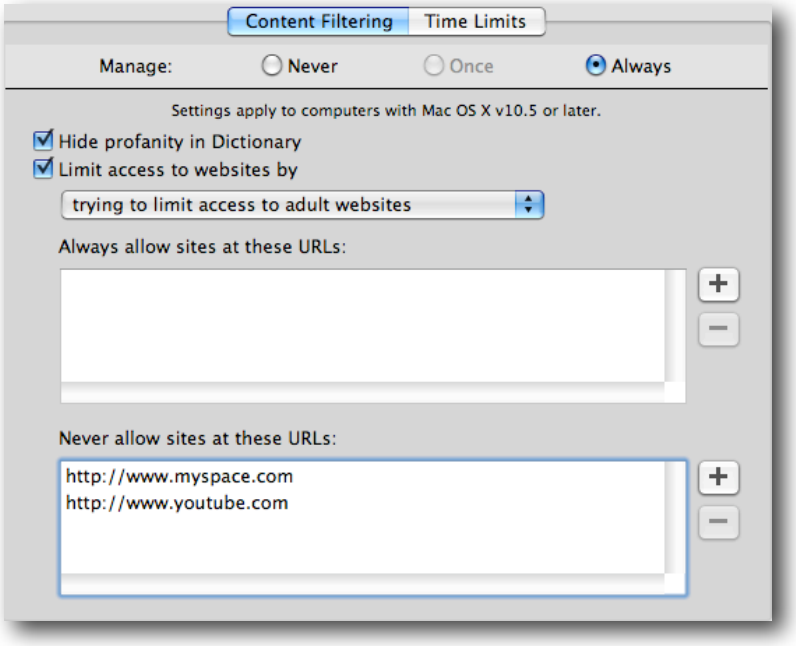
Setting global proxies and interface access

Mobility

Mobility settings are covered in "User Accounts—MAs, PHDs, and More."

Parental Controls

New with Leopard are the Parental Controls settings. These settings address the issue raised when a parent asks, "So, you're going to give my child a portable to bring home. How do you plan to keep her off the Internet at 2 a.m.?" Here are some of the key settings:

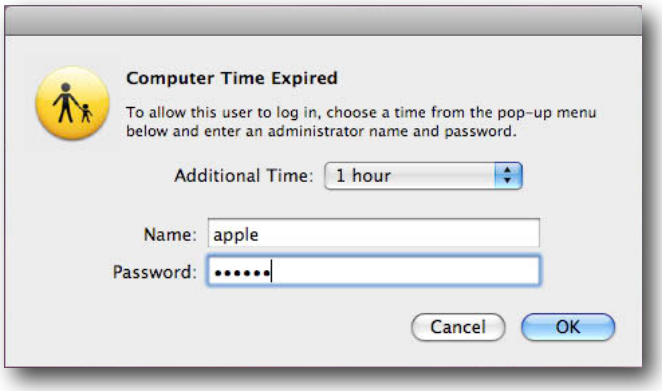


Content filtering and white/blacklisting

The filtering run by Parental Controls is a local proxy that uses the same kind of logic used for spam filtering. It has a huge database of what are "good" and what are "bad" things to find on the Internet and can determine what's acceptable with a high degree of certainty. It provides you with a mechanism to filter web content even when the users are outside the school network.

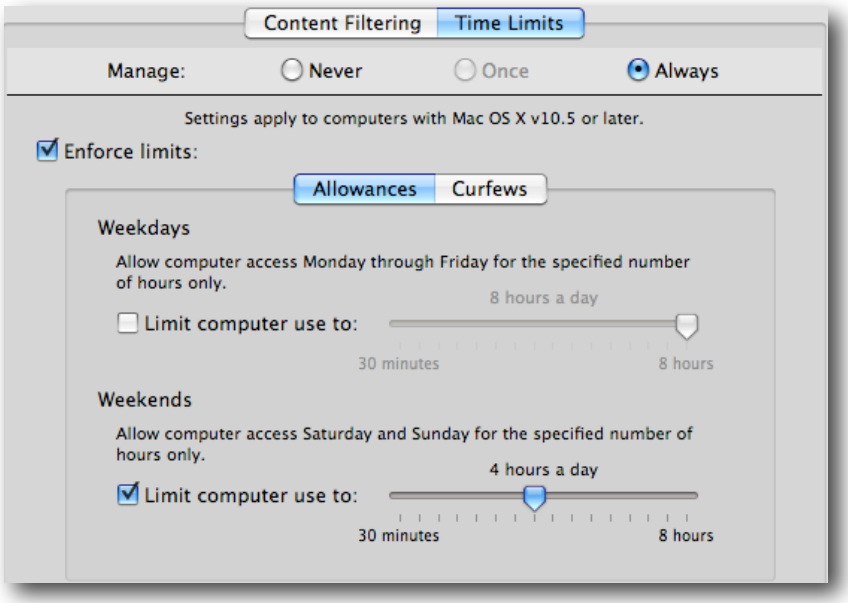
Note: Don't bother putting www.apple.com into the whitelist—it's hard coded in there already.

Setting time limits and curfews are the next most important items. You can specify a specific amount of time a user is allowed on the computer on the weekend; this may be something the parents want in the AUP for a 1 to 1. Being able to set a curfew resolves questions parents may ask about late night users. At the specified time, or if the usage limit has been reached, the user sees the following message:

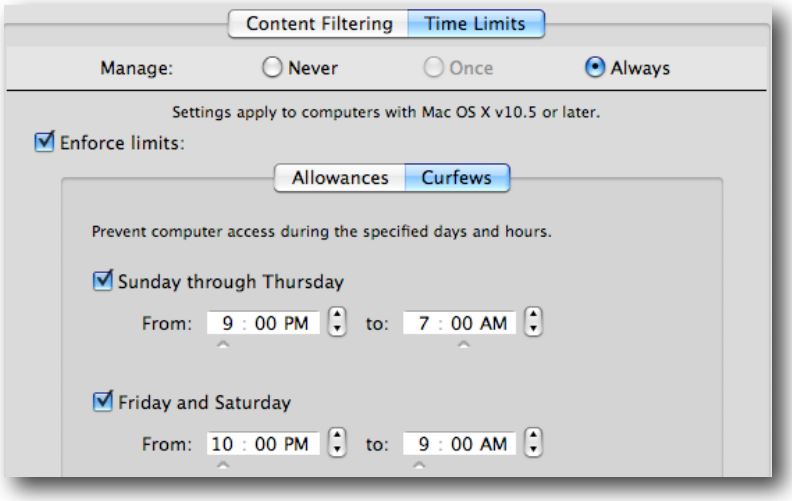


Time's up—Parental Controls

It takes a local administrator's account password to bypass the restriction. A series of warnings appears as the expiration time approaches.



Setting a usage time limit

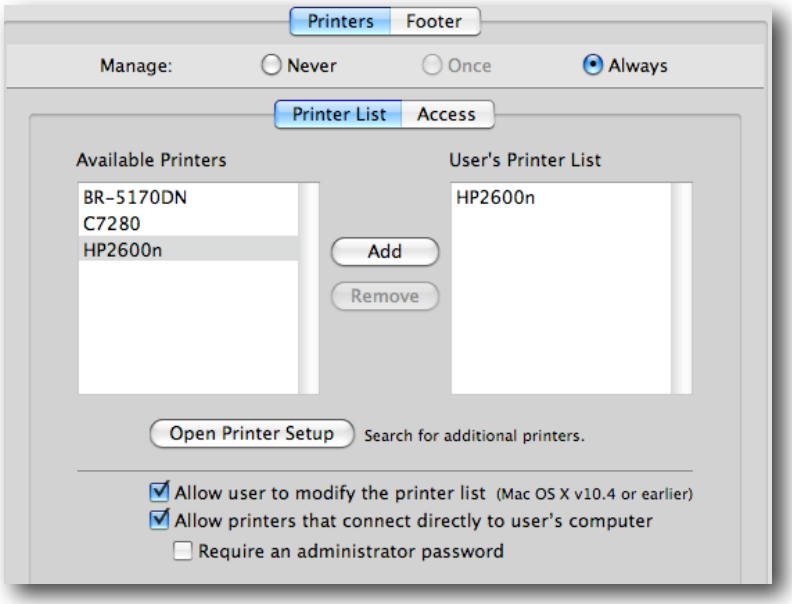


Curfews for your 1 to 1 users

Being able to implement a series of distinct controls on deployed portables and even on in-school systems is a good idea. You can set the lab computers to be unable to accept user logins after school, except for administrative work.

Printing

The print management has improved since Tiger with the addition of a setting that had disappeared with the end of Macintosh Manager. When you run Workgroup Manager from either an administrator or a client system, you will see a list of the printers captured by that system. If you will add many different printers from different areas, you will need to either run Workgroup Manager from each of those locations or capture all the printers on your administrator/client system.



Assigning printers to managed clients

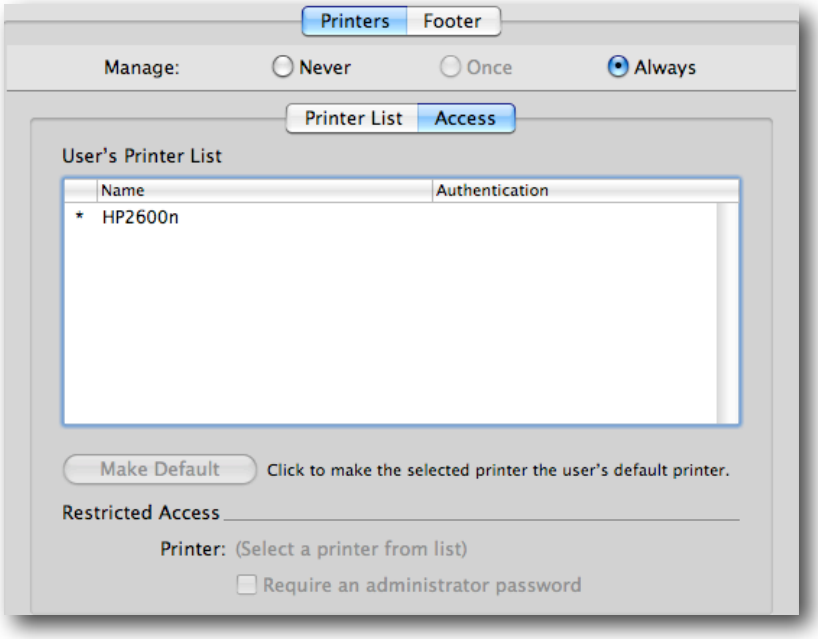
Note that the selection to allow users to modify the printer list applies to only v10.4 and earlier. This is due to a change in the Leopard printer System Preferences to require local administrator access to add and remove printers. If you would like your users to be able to add their own printers, you can make a change to a file on the client system. To accomplish this, you need to locate the "/etc/cups/cupsd.conf" file on your administrator system and open it with TextWrangler (or use Terminal and your favorite editor). Locate the line:

```
# All administration operations require an administrator to
authenticate...
```

Change the following lines to:

```
<Limit CUPS-Add-Modify-Printer CUPS-Delete-Printer CUPS-
Add-Modify-Class CUPS-Delete-Class CUPS-Set-Default>
    # AuthType Default
    # Require user @SYSTEM
    Require valid-user
    Order deny,allow
</Limit>
```

Save your changes and restart the computer. You can then use Apple Remote Desktop or any system management tool to deploy that new cupsd.conf file to all of your clients (or if you are ahead of the game, make it part of the image). The user still won't be able to add or remove printers in System Preferences, but they will be able to add a printer from the Print dialog.



Setting the default printer—plus authentication if needed

You can specify the default printer and choose to require an administrator password to print to a certain printer. Another important change is that you can now specify that all print jobs will have a footer attached with the user's information and a timestamp.

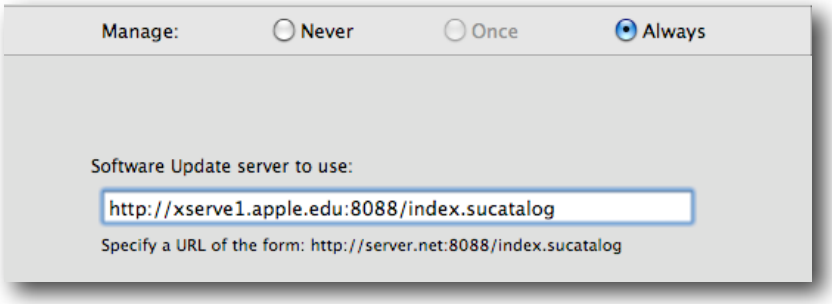
Adding a watermarked footer to print jobs

The footer is printed on top of anything else on the page. Because it is done as a watermark, if you have URL information, document footers, and so on, these are printed on top of that information. The watermark is printed at the bottom left of the page at the extreme limit of the printer's page limit. This may or may not be below any user-defined footers. Finally, if you set a footer and the user prints a photo, it may not be what you wanted on the picture.

A final note on printing: Any printers added by the users will show up for other users, so use caution in allowing users to attach their own printers.

Software Update

The MCX setup for Software Update binds the clients to a specific software update server, hopefully inside your network.

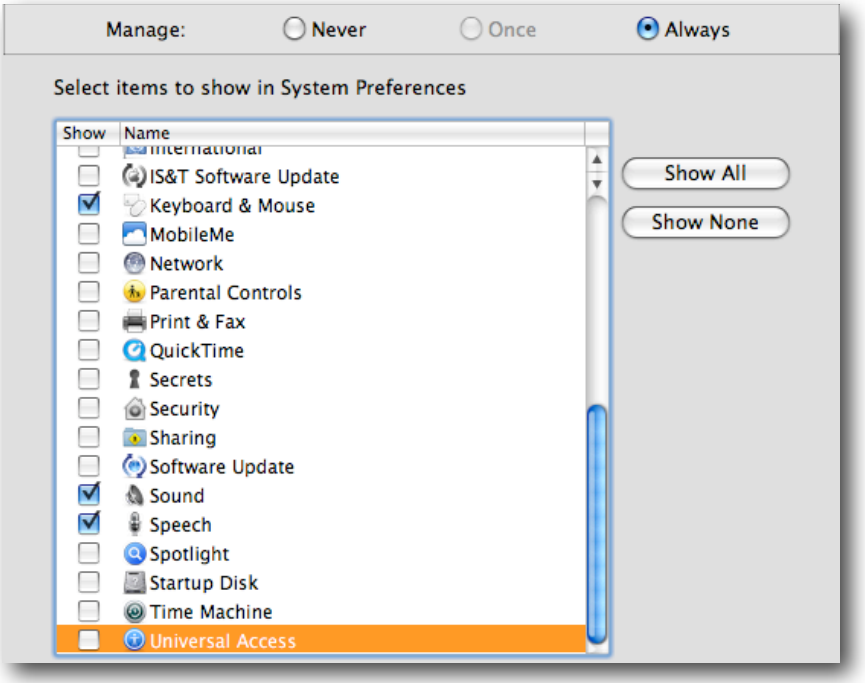


Pointing your clients to an internal Software Update server

This works for servers as well as client systems.

System Preferences

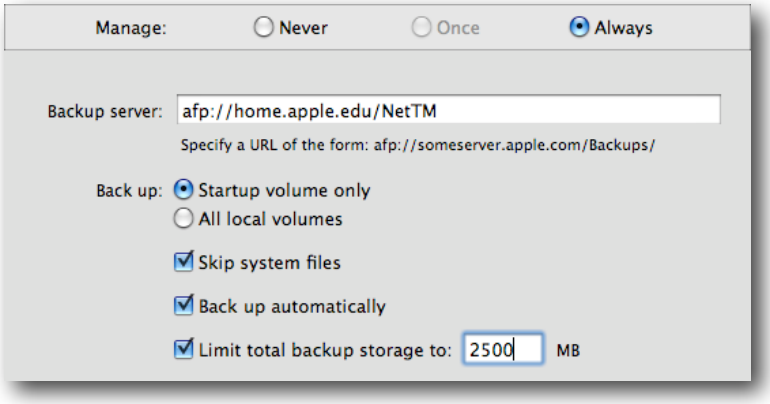
Establishing access to specific System Preferences has a couple of uses. First, you might want to just avoid the curiosity factor. While some System Preferences are locked to non-administrator users, they can view the settings. Second, if you use MCX to populate some System Preferences settings, such as the QuickTime Pro license, you may not want users to see what the settings actually are. For example, you may want to avoid them "borrowing" the license for use at home. Finally, even if you grant users access to a preference setting, they may not be able to edit the item due to administrator access requirements.



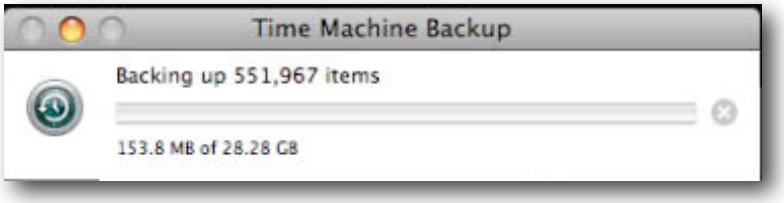
Managing System Preferences

Time Machine

Time Machine is a very interesting new capability in Leopard that has great potential, but is not without limitations in an educational environment. The first limitation is that the disk image that gets created at the Time Machine share point is named after the MAC address of the client. This means that a user could end up with backups stored on several images as they change computers, making restoration very difficult. The second limitation is that, although the Time Machine backup can be set to ignore System files, the first backup still copies the entire Applications folder from the client system into that user's backup—every user on that client gets a copy of the Applications folder. Until network TIme Machine is based entirely on the user account and not the computer, using Time Machine on a large deployment just isn't practical.



MCX settings for Time Machine

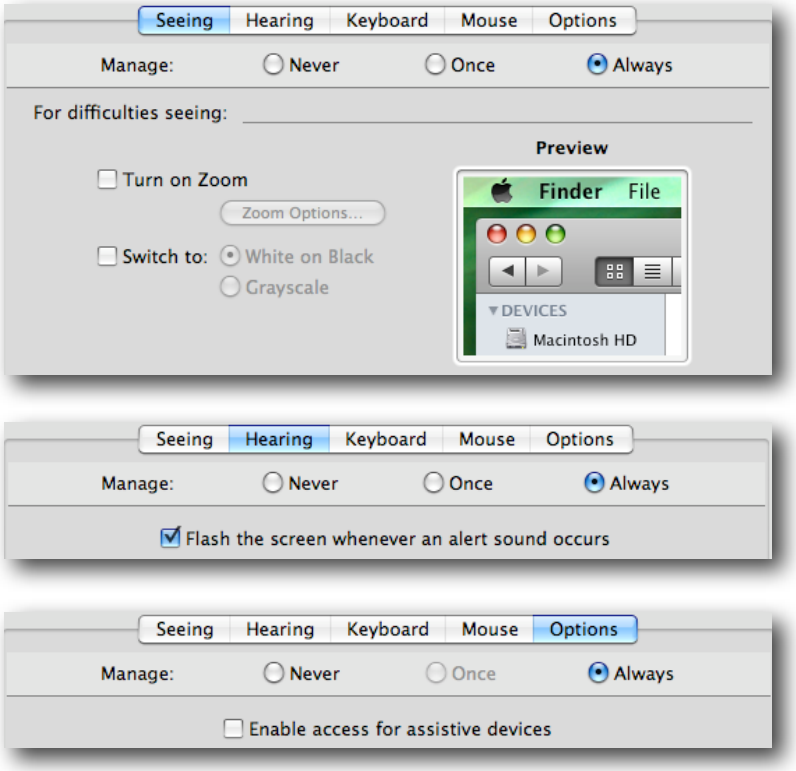


The user's first backup—his home is not 29GB.

Note that in this example, the user's Time Machine backup is trying to gather over half a million items. This user has 88MB of content in his home directory. The rest of that is the entire Applications folder being backed up.

Universal Access

In some circumstances, you may need to activate portions of the Universal Access System Preferences. Some external devices require the assistive device flag set. In a lab, you might want to set the screens to flash when a system beep is sounded. You can also turn on zoom as a default; however, with Leopard, all you need to do to zoom is hold down the Control key and use the trackpad or mouse ball to zoom in and out.



Some of the Universal Access MCX settings

MCX in Action—an Example of the Hierarchy

Now that the different MCX settings visible in the graphical user interface have been introduced, you can look at an example of these items in action. For this example, a series of Dock values will be created for two workgroups, a computer group, and two users. Then what each of the different users sees at login will be shown. For comparison, a series of screenshots for a Tiger client will also be included.

The Setup

For this example, the following values have been set for the Dock at each domain:

- Computer Group (MCX Lab Clients): Dock items are Safari, Preview, Dictionary, and TextEdit. The user's settings will be merged. Dock items are set to Always. Dock display is set to Once on the bottom (for easier screenshots).
- Workgroup (Creatives): Dock items are Chess and Address Book (Always).
- Workgroup (Math): Dock items are Directory and Grapher (Always).
- User (johnd): Dock item is iTunes (Always).
- User (thann): Dock item is Mail (Once).

Note that the "merge" setting also populates the Dock with the system defaults; the Dock is shown both before and after one of the users has deleted all the items they are allowed to remove.

The Results

The Leopard clients won't get a workgroup picker because the combine flag was set when the login window was set up (and it's on by default). The Tiger client, however, sees this:

Workgroup Picker portion of login window

Because of this, in Tiger, the "johnd" user must choose one of the workgroups. That user's Dock will contain only the items set there for that workgroup as well as items from the computer group and user MCX settings.

Tiger Dock for johnd in the Creatives workgroup

The Tiger Dock shows that only the items for the Creatives workgroup are available. The question mark instead of a valid icon is displayed because iMovie was not installed on this system. When johnd logs out and moves to a Leopard client, his first Dock looks like this:

johnd's Dock as a Leopard user—both workgroups' items are present

The items staged in the Dock from left to right are: computer group, first workgroup alphabetically, second workgroup, user. Items within the workgroup are staged alphabetically. What's also noticeable is that the System items are not present. They were set to merge with the user's items but because the user account didn't have the merge turned on, they aren't here.

If johnd had the merge flag set, this user's initial Dock at login looks like this:

johnd also has the Dock items from the System.

You can see where cleaning up the default Dock property list might be a really good idea. This is just one example of how the MCX hierarchy works. Testing out your settings can save many hours of troubleshooting later on.

Administrator Tips for MCX

These are a few tips for administrators for deploying the MCX settings.

Bypassing MCX Settings

When you set up your managed client environment, you can choose to allow local administrators to bypass MCX settings. You can also use that setting to refresh the MCX preferences if you don't want to wait for a restart or login/logout cycle. If the local administrator holds down the Option key at the login window when clicking the Log In button, this dialog appears:

Admin bypass dialog

If you select to remember the setting, you will bypass MCX settings until the next time you hold down the Option key at login. This is also the dialog you can use to refresh the management settings, which are covered next.

MCX and Cached Settings

Under Tiger, the system administrator could set MCX cache values. This has changed with Leopard. The MCX system works by caching the directory settings onto the client inside the local directory in two places. The first is in a /Computers record in the local directory. You can view this directory by opening Terminal and typing:

```
sudo dscl . -readall /Computers
```

This results in data like this (just a portion):

```
johnd-imac-24:~ johnd$ sudo dscl . -readall /Computers
dsAttrTypeNative:cached_groups:
 <?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
       <key>dsAttrTypeStandard:AppleMetaNodeLocation</key>
       <array>
              <string>/LDAPv3/xserve1.apple.edu</string>
       </array>
       <key>dsAttrTypeStandard:CopyTimestamp</key>
```

This information matches the information you can view in System Profiler for the Managed Preferences item. The rest of the locally cached information is stored in the Managed Preferences folder located at /Library/Managed Preferences. These values are updated on a very regular and consistent basis. The MCX cache is now refreshed:

- At boot time

- At each login
- At each logout
- At each restart
- At periodic intervals while sitting at the login window
- Each time there is a network transition (Ethernet to AirPort, for example)

The MCX cache is flagged as "dirty" as soon as it comes into existence. This means that the system always looks to refresh the cache at every possible circumstance. However, this does not mean that the cache is ever just deleted when the client cannot see the server. Users do not become "unmanaged" just because they left the network and do not suddenly become administrators. If you are experiencing problems with the MCX settings and think that the cache does not conform with what you set up, you need to perform a few basic troubleshooting steps.

First, open Workgroup Manager alongside System Profiler on a client system. Compare the values listed for MCX/Managed Client carefully. What is getting to the client should exactly match what you set up.

You can now flush the cache and restart the computer to force a complete rewrite with the following command (if you use Apple Remote Desktop, leave out sudo):

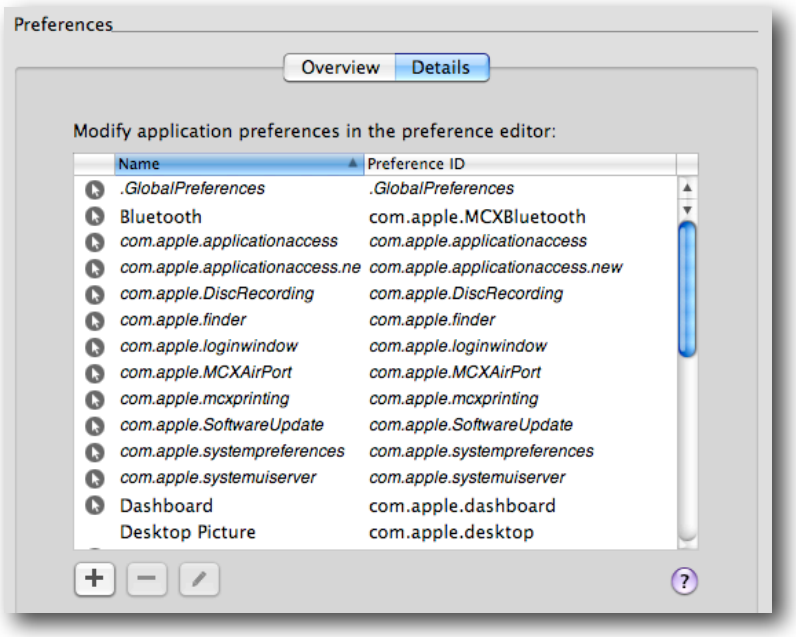
```
sudo dscl . -delete /Computers
sudo rm -rf /Library/Managed\ Preferences
```

Do *not* run these commands as part of a login script or you will end up with unmanaged users.

The next section addresses more complex MCX topics.

Advanced MCX Setup—Adding "Details"

Although using the Workgroup Manager graphical user interface to edit MCX settings is relatively simple, there are many times you need a setting or value that "just ought to be there." For that purpose, MCX provides the Details pane within Workgroup Manager Preferences.



Getting into the Details of MCX

All the settings created in the graphical user interface are visible here in property list format. This section of the settings is invaluable because it provides access to settings that extend currently available MCX graphical user interface settings. It also allows you to add settings that would never make it into the graphical user interface.

Preference Manifests and Other Hidden Settings

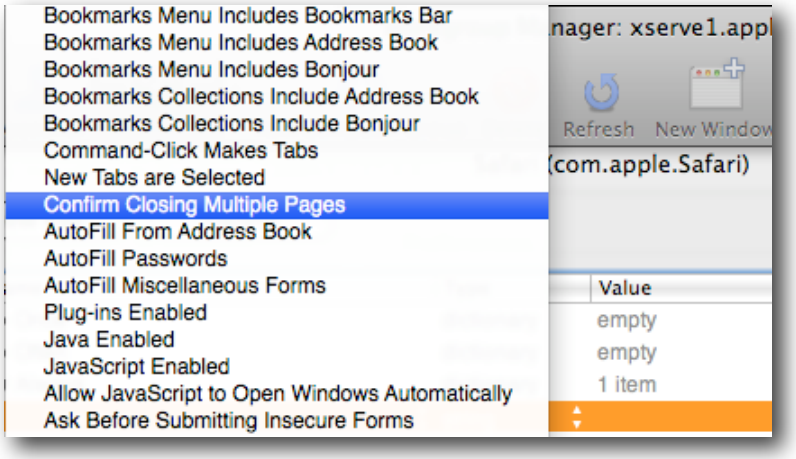
Many applications contain settings, or *keys*, that allow you to configure the user experience for yourselves. These settings can, in many cases, be stored in a directory so that they can be applied to many users. Several of these settings have already been explored in the previous section. Where MCX shines is in the ability to use the "other" keys that many applications have outside of System Preferences. The best case for these keys is a *preference manifest,* which is a database of values that the developer has imbedded into the application itself. This manifest lists the keys, and often their default values, that can be preset and are supported for inclusion in a directory.

Safari is a great example of the use of preference manifests. The application contains a large set of values that can be added to the MCX domain and will then apply to any user set to be managed with that domain. Many preferences contain a set of keys that are known to work with MCX from a network. By adding the preference file to the MCX domain, the settings will apply across all users. Setting the mouse is a good example of this.

Finally, there are many tweaks that can be performed on a system to enhance the user experience or to better manage the system. These settings are often altered with the "defaults" command. Many, if not most, of these type of adjustments can be made in the Details pane versus running a script or altering each computer individually with a command-line call.

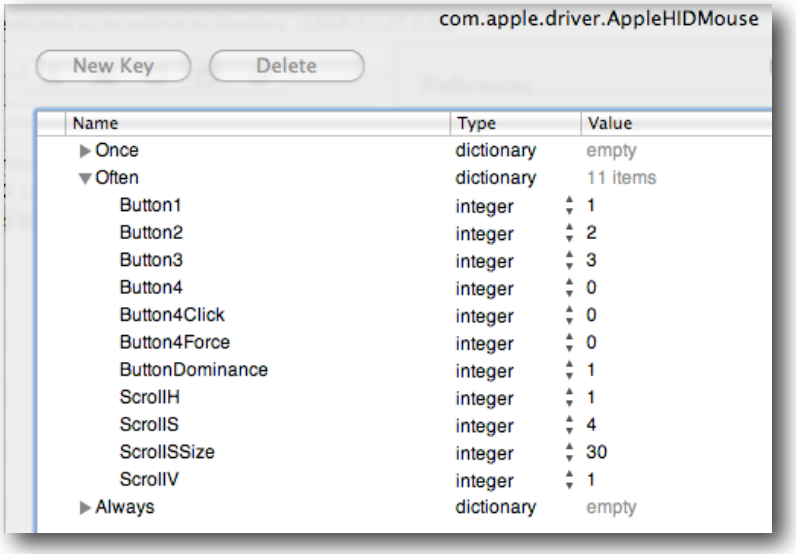
Here's a quick look at an example of each type of entry for Details.

The preference manifest for Safari is shown here:



Safari contains dozens of MCX aware keys in its manifest.

The keys in a preference file for Mighty Mouse are shown here:



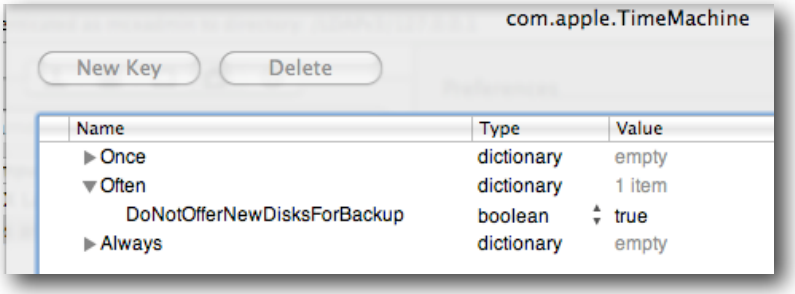
These keys allow the settings from the mouse to be the default for all users.

Turning a Default into an MCX Setting

If the messages from Time Machine are a problem for your users, you can have each user run the following command:

```
defaults write com.apple.TimeMachine
DoNotOfferNewDisksForBackup -bool YES
```

Alternatively, you could set the following information into a key in the MCX Details pane:



Setting Time Machine to stop reminding user at every disk mount

These were just a few examples. The next section will present some other tweaks so you can learn more about the Details pane.

Details Tricks

There are two ways to add settings to Details. The first is to configure a preference the way you expect it to behave, then add that preference to the Details section. For this to work, the preference file must be in plist or property list format. The second method is to locate the application itself and add it to the Details section. When this is done, the MCX code checks for the presence of a preference manifest first, then it looks for the preference file used by the application. It searches for the file inside the user's home directory in the path ~/Library/Preferences/<reverse domain>.<application> (such as com.apple.Calculator).

The preference file may contain many items that are either unique to a specific user or may be of no use in a managed setup. You can edit the imported plist to remove all but the keys you really plan to use.

A preference manifest looks empty at first glance. What it contains are the keys in a database. To employ these keys, you have to add each key to the domain. Look at the iTunes or Safari examples to see how this is done.

Here are just a few examples of the cool MCX settings that can be done with Details:

Mousing Around

The Mighty Mouse, by default, treats both sides of the front—the left and right buttons—as the left click location. If you set up a Mighty Mouse on your administrator system, you can import those values into Details to allow all users to use the same settings. The values can be brought into Always to force them, Often to test them, and Once to allow users to change them as they see fit.

Set up the Mouse System Preferences the way you want. In Details, click the Add button (+) and locate "com.apple.driver.AppleHIDMouse." Select the Always, Often, or Once domain to add the entries. Test the settings by logging into a managed client as a non-administrator user.

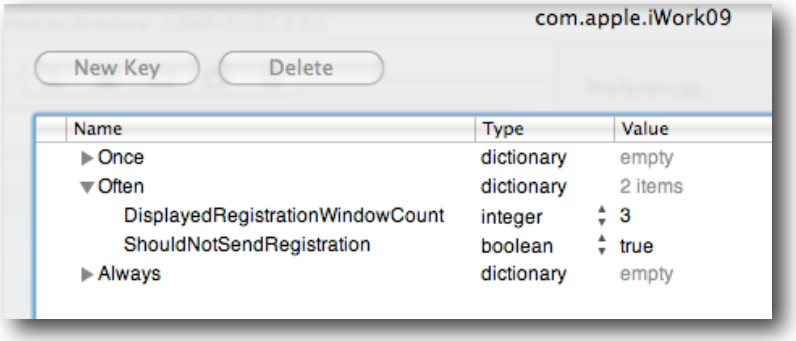
Note: You'll learn to use the different domains depending on the restriction you want to set, but you'll also find that some preference files only work when added to the Often domain. Usually, these are preferences that have no user accessible settings.

iWork

If you are deploying iWork, you might find that editing the settings can take a lot of time on a large number of computers. Using Details, you can grab all the settings at once and add them into your managed set.

Note: The tutorial window appears the first time the user opens the applications, but not after that.

Set up iWork '09 on your administrator system. Make sure you open each application at least three times to stop the tutorial windows from appearing at launch (unless you want the users to keep seeing them). In the Details pane, click the Add button (+) and locate "com.apple.iWork09.plist" in /Library/Preferences. Add the item to the Often domain.



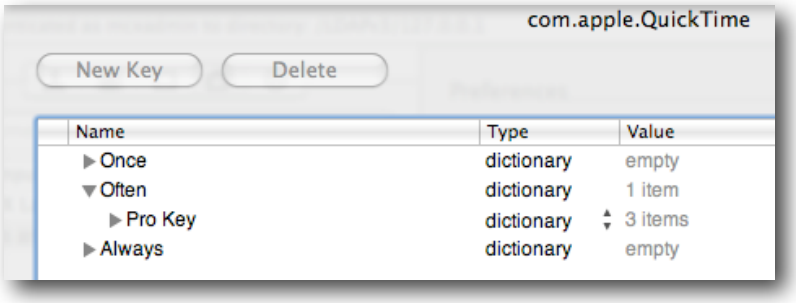
This should avoid the registration screen for each user.

Managed Client

In Leopard, a very large number of default manifest items have been built into the Managed Client application itself. See "The Managed Client.app Preference Manifest" later in this document.

QuickTime

Instead of spending the time to set the QuickTime Pro license on all of your systems, you can add the license to the computer group that has as members those systems you have licensed for QuickTime Pro. Just use Details to add the "com.apple.Quicktime.plist" to the Often domain for your computer group.

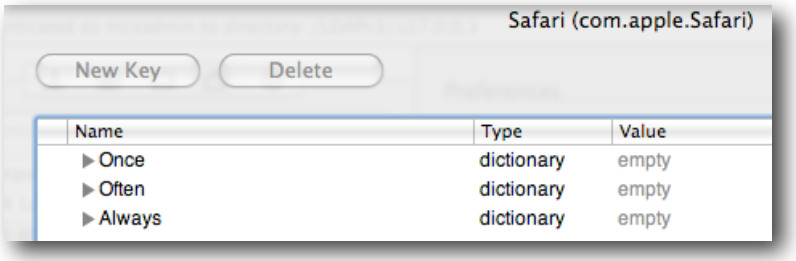


The QuickTime Pro key will automatically apply to all managed systems.

Tip: Make sure you hide the QuickTime System Preferences from the user's view after this to avoid someone accidentally borrowing the serial number for home use.

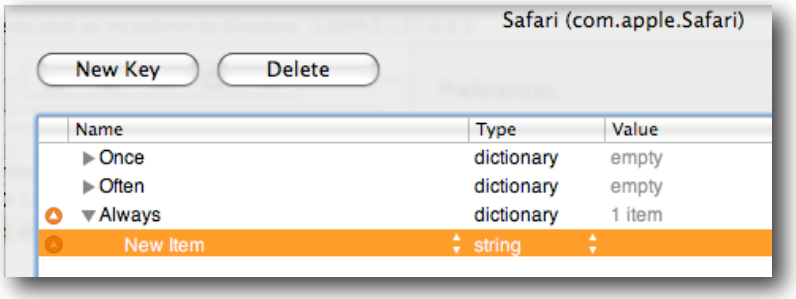
Safari

Safari contains the most extensive preference manifest of any Apple applications, followed by iTunes. Using Details, you can click the Add button (+) and add Safari itself. In this case, you can deselect the "Import my preferences for this application" checkbox. If you leave this checkbox selected, the settings would contain all the extra items from your administrator system's Safari preferences. This way, you start clean. The first thing you'll notice is that the domain looks empty:



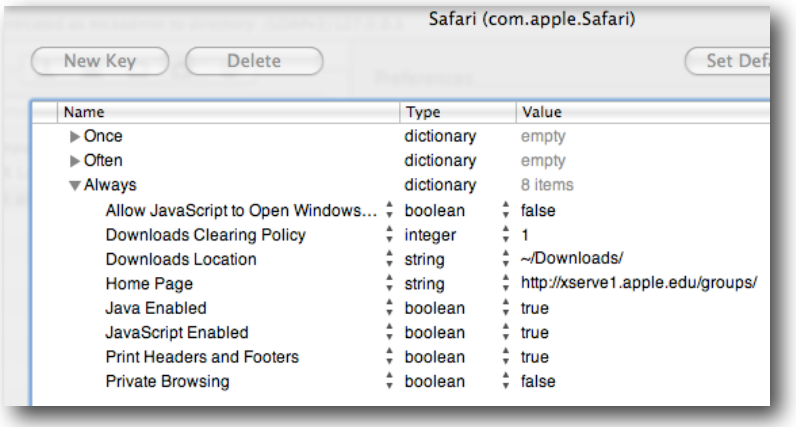
Only the manifest was imported—no values yet.

Then you begin adding keys to the settings. Start by selecting the Always domain and clicking its disclosure triangle. With Always selected, click the New Key button.



Adding a new key to the Safari settings

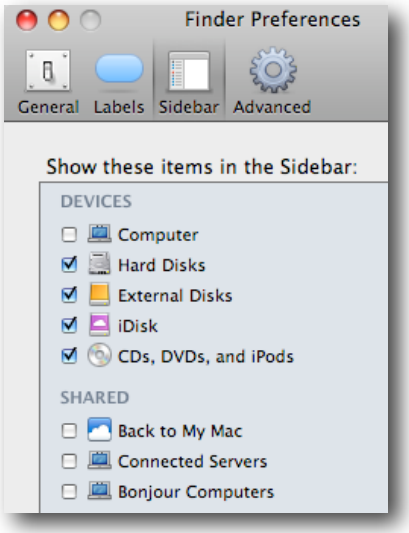
Next, click the new item to display a very large pop-up menu with available keys. Choose the Home Page item near the top of the list. You can now edit the Value field to enter the default website you want your managed users to get. Select the Always domain again and add a few more keys. You'll note that you have a lot of control over the behavior of Safari.



Private browsing now denied

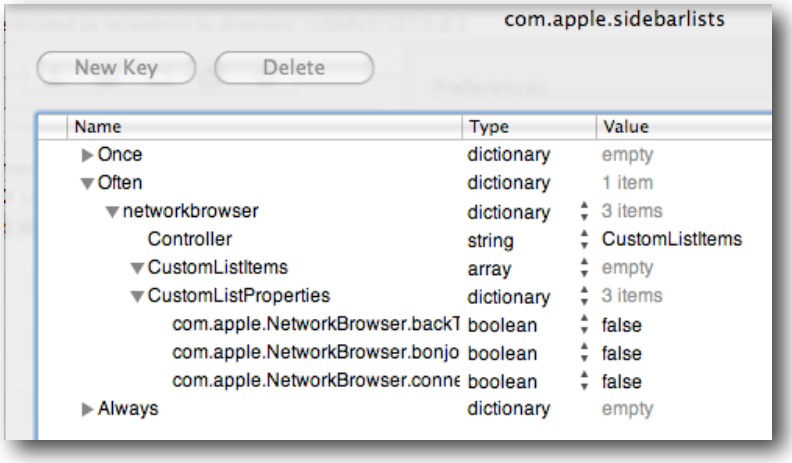
Sidebar

Many school sites have asked how to suppress the Shared section of the Sidebar. If you have many share points on the network as well as all the clients with the Remote Desktop agent active, many entries are displayed in that section. To suppress the Sidebar Shared section, you have start with your administrator system's Finder preferences. Select the Finder preferences on your system and deselect the checkboxes within the Sidebar settings.



Turn off all Shared settings.

Next, in Details, click the Add button (+) and locate the "com.apple.sidebarlists.plist" in ~/Library/Preferences. Add the item to the Often domain, then select the edit tool (the pencil). Remove most of the items listed, leaving the set shown here:



Sidebar MCX settings to turn off Shared items

You can leave the saved searches, if desired, but the other items should be removed. If you end up with odd items in your client's Sidebar, this is where to look for the problem.

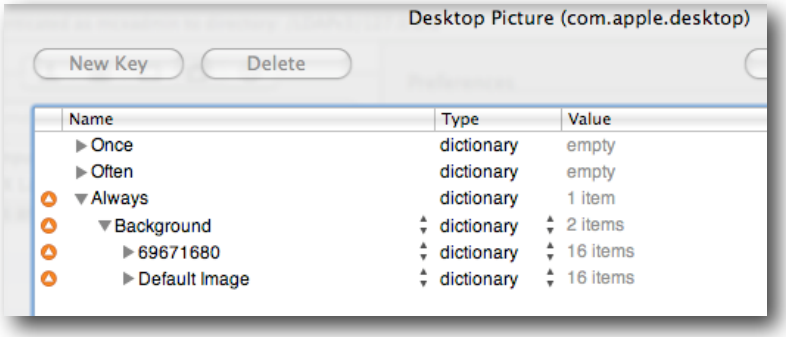
Note: If users turn these items back on in their own Finder preferences, the settings will revert at their next login. Placing the settings into the Always domain does not work at the time this document was written.

Desktop Picture

In some cases, you may want to lock down a user's desktop picture, for example, if your site rules do not allow users to use their own pictures. The problem has been that there are so many ways to set the desktop picture, and previously, it wouldn't stick. Details in Leopard helps get past that.

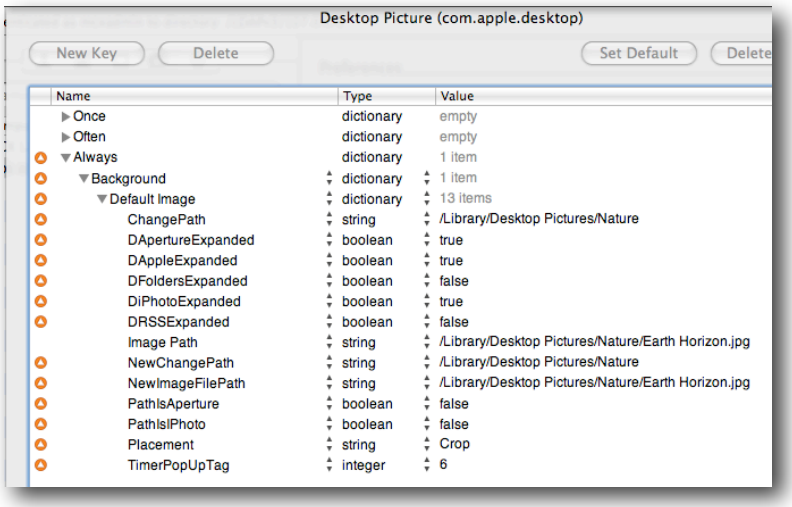
First, if you will assign a default desktop picture, make sure the image is stored on all clients in the same path. You do not store the actual picture in the directory, only the path to it. Hence, copy the image you want to use into the /Library/Desktop Pictures folder. Next, set it on your administrator system.

Now, in Details, click the Add button (+) and locate "com.apple.desktop" in your ~/Library/Preferences folder. Add the file into the Always domain (choosing the Often domain is not a good idea because the user will change it and then it will change back at every login). Select the edit tool and you'll see this:



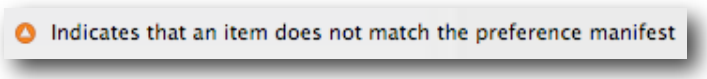
Desktop plist with all custom settings intact

The extra settings you see are the ones belonging to every monitor and display configuration that has ever been attached to your administrator system. Delete all but the Default Image key from the Background key. The final settings will look like this:



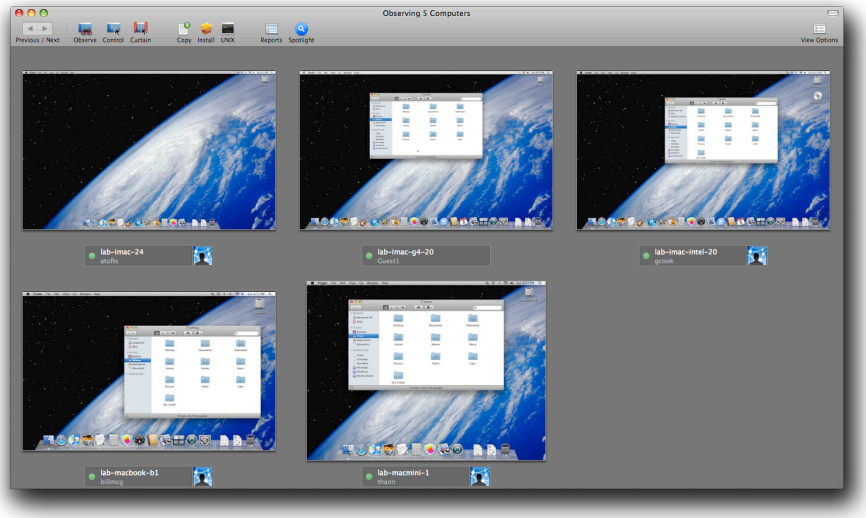
The final settings with empty settings deleted

The rest of the items are flagged to show that they are not part of a manifest:



You will see this orange triangle a lot as you edit preferences. It doesn't mean that the settings won't work, just that there is no manifest database item to correspond to that entry.

The users will not be able to change the desktop picture at this point. They'll try, but beyond seeing the preview image in System Preferences change, nothing else will work. Even iPhoto respects this setting in Leopard. All the users' desktops will now look like this:



Consistent desktop pictures

These are just some of the ideas for how you can use Details to enhance the MCX experience. Test out the concept for any applications or plist files you have.

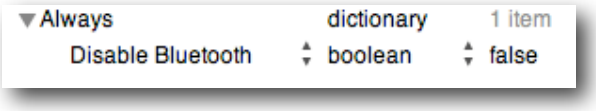
The Managed Client.app Preference Manifest

One of the most powerful portions of the Details setup in Leopard is hidden by default. This is the preference manifest set built into the Managed Client application itself. The Managed Client application resides in /System/Library/CoreServices and contains the code used to run the entire MCX process. The compositor is also located here, along with the code to run mobile accounts, portable home directories, and much more. The MCX team built a very large preference manifest list in here that is continually being refined and revised.

The manifest is made visible by going into Details, selecting the /System/Library/CoreServices/ManagedClient.app, and adding it to the set. The items show up as bold, gray items compared to the italicized items you already have. Some of the imports ask to replace ones you have already entered, such as Desktop. That's okay; the values you have already entered won't be disturbed. You will, however, be able to streamline some of your settings. Here's a walkthrough of the manifest items offered by the Managed Client application:

Bluetooth

This setting contains only one key—Disable Bluetooth (boolean).



Bluetooth setting

Dashboard

This manifest contains only one key—Disable Dashboard (boolean)—in respect for the Tiger systems that can't use widget management. If you did set widget management for your Leopard systems, the values you set show up here too.

Desktop Picture

This manifest contains keys to set the basic required information to establish a desktop picture:

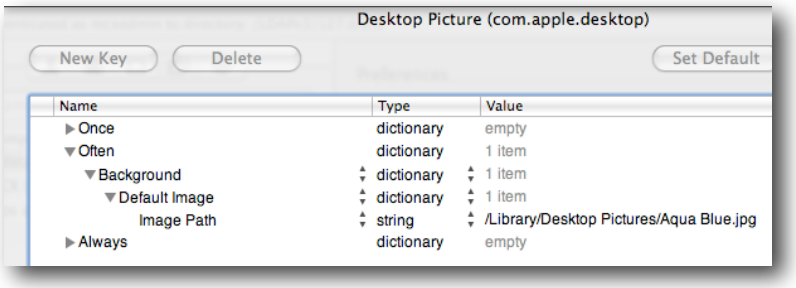
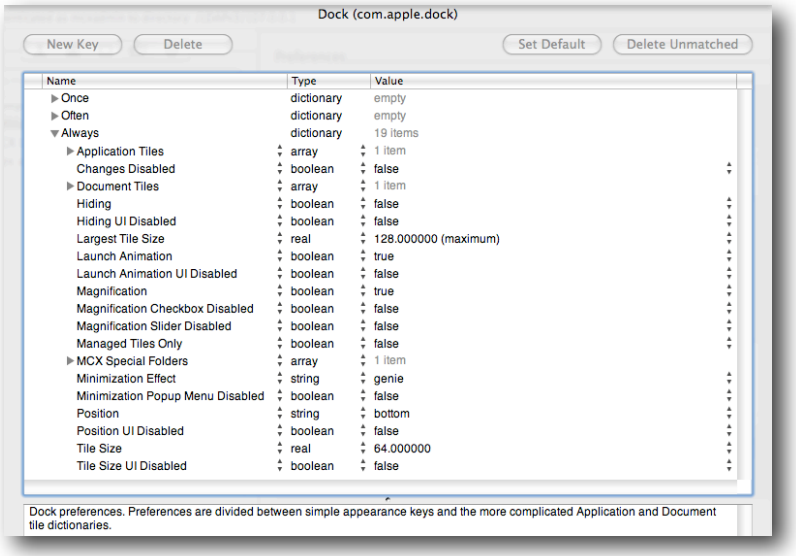


Image Path is the only required item.

Note that the manifest keys can only be created inside the Often domain here. If you move or import them, in the Always domain, they might not work. Test your settings often before locking them down.

Dock

The Dock manifest has a huge set of keys:



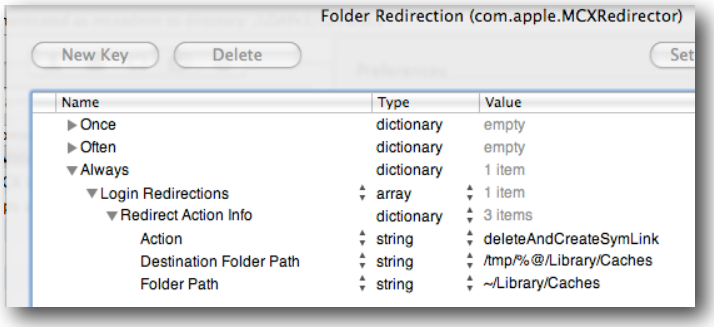
Note the definition window below the key settings.

Always check the definitions of each key as you add it. Some do not need to be used, and if not, their default setting will take effect.

Folder Redirection

This setting will help take some of the performance load off network home directory users. It is designed to force the current user's Cache folder into /tmp on the local computer. When you must use network home accounts, this can mean a huge reduction in network traffic. More information is included in "User Accounts—MAs, PHDs, and More."

To set this up, you open the setting to edit, select the Always domain, and add a new key. Choose Login Redirections from the pop-up menu. Now select that new subkey. Click to turn down the disclosure triangle, and add a new key to that key. This subkey will be called Redirect Action Info. When you open that key, you'll see that the defaults are already filled in for you:



The default Login Redirections item

The other actions are Logout Redirection and Other Redirections. One key point here is that you cannot create a redirection to force the user's home, or a subfolder, to an upstream or network location. These redirections take place before any mount points are available, so the redirection would fail.

Home Sync

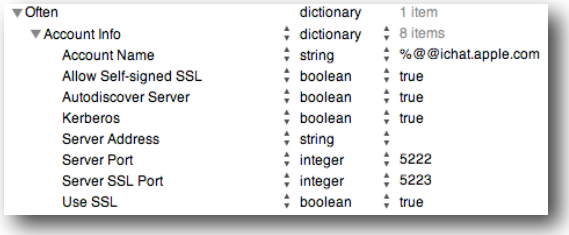
This setting gets its own discussion in "User Accounts—MAs, PHDs, and More."

iCal

The setting supports adding an Imported Accounts key with values. It is designed to preset iCal accounts for users.

iChat

As with the iCal setting, the iChat setting is designed to preload account info for local iChat server users.



iChat setting

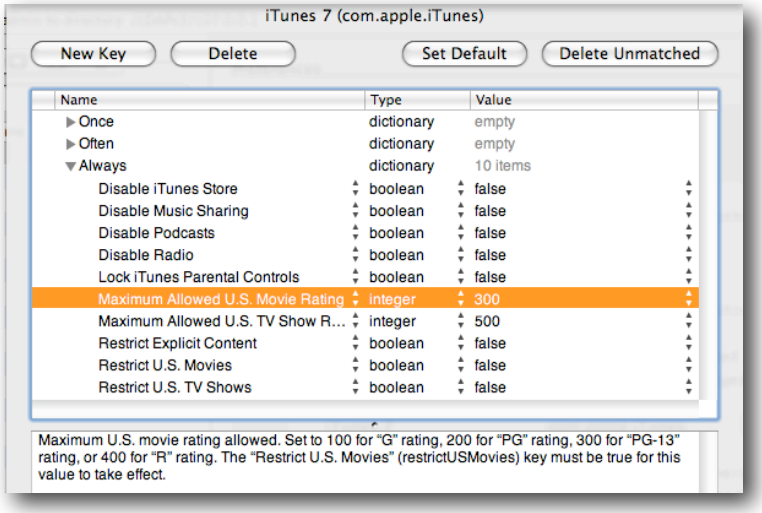
Internet Configuration

These are the settings and values that were removed from the graphical user interface for Leopard and generally work only on Safari and Mail. They are worth testing out.

Internet configuration

iTunes 7 and iTunes 8

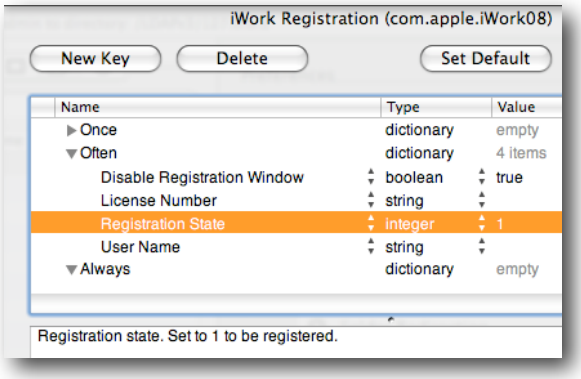
These are the settings that allow granular control over the entire suite of settings in iTunes. The manifest in Managed Client refers specifically to version 7, but iTunes 8 items can be added as well. Just import the current plist you are using.



Access to the iTunes Store, restricted movies, and so on

iWork Registration

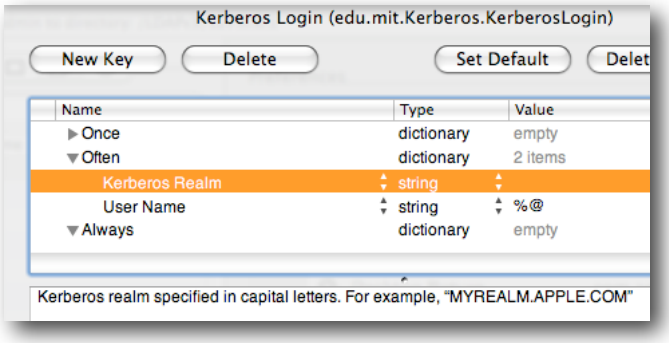
These are the basic settings to establish the registration keys for network managed systems. The iWork08 manifest is built into the Managed Client application. iWork09's plist can be added to the Details section.



Basic iWork08 registration information

Kerberos Login

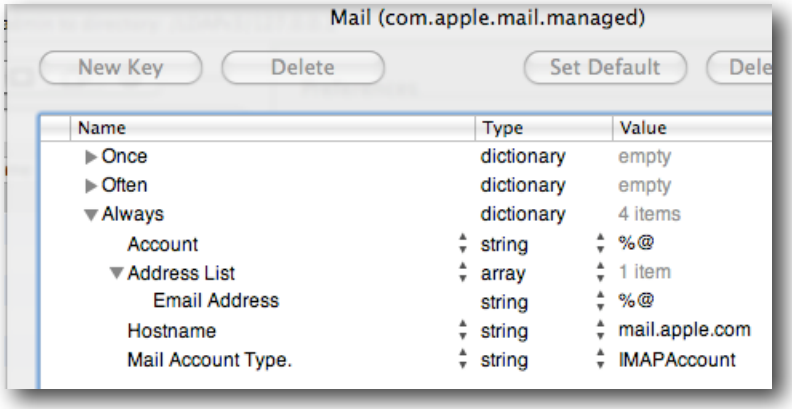
These are values to preset the Kerberos login.



The "%@" means the current logged in user.

Mail

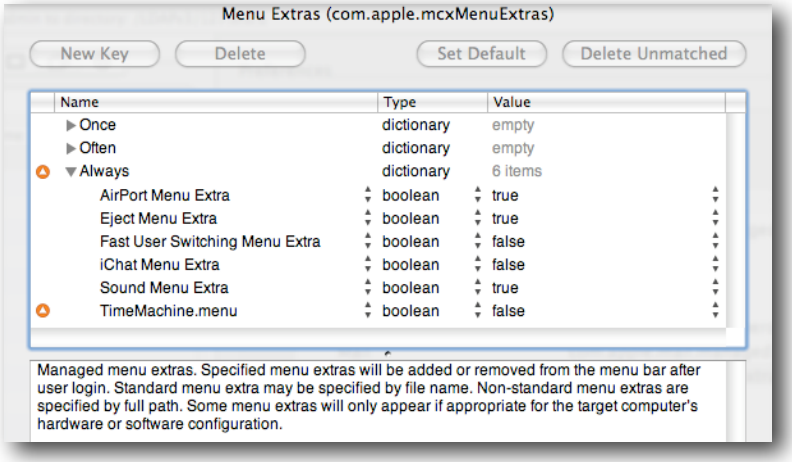
This manifest allows you to create preset values for the user's Mail account.



Mail values

Menu Extras

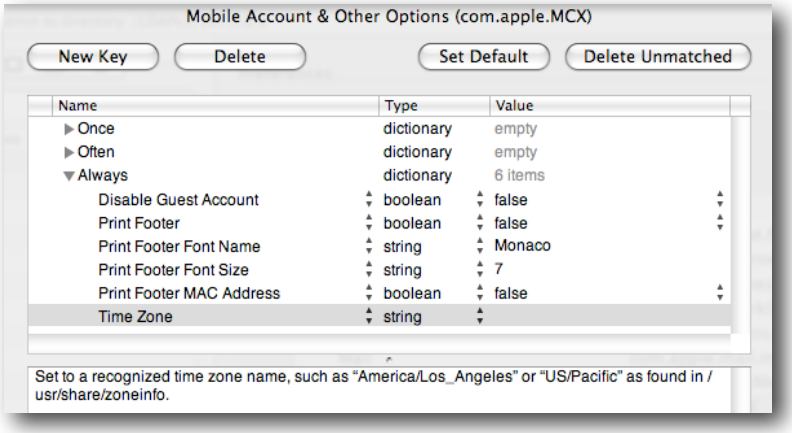
This setting allows you to add menu items or disable them. There are a lot of items in the manifest. For items not in the manifest, you can add the name or path, such as the Time Machine menu item shown here:



Menu Extras—the Time Machine item is disabled.

Mobile Account & Other Options

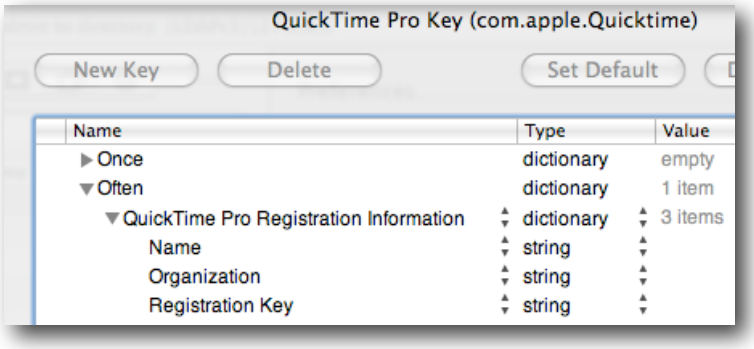
The values here support mobility plus other items tossed in. The mobile items are covered in "User Accounts—MAs, PHDs, and More"; the others are shown here:



Didn't like the choices for footer? Change them here.

QuickTime Pro Key

This is a simple way of adding the QuickTime Pro registration information from an administrator station to a managed client set.



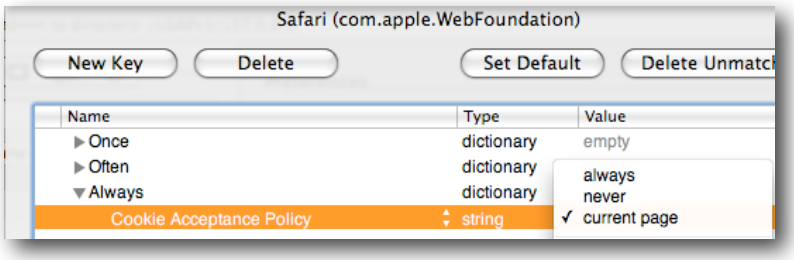
Make sure you have a multiple system or site license before adding this value.

Safari

This has been covered already in "Details Tricks." It's a whole lot of information. Safari has a huge preference manifest.

Safari (WebFoundation)

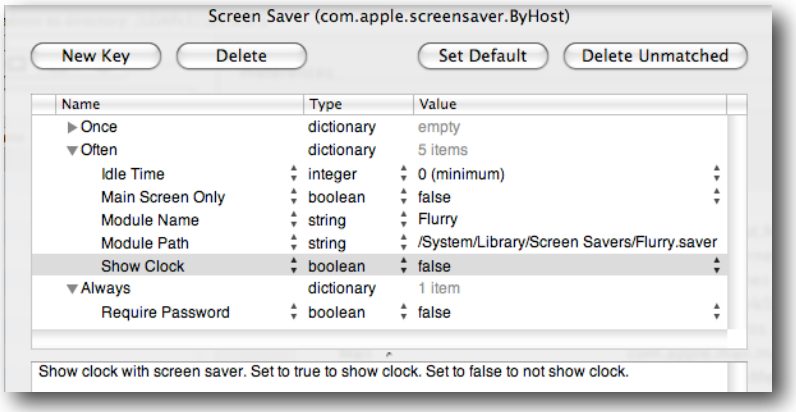
This setting contains the Safari cookie policy.



Cookie policy

Screen Saver

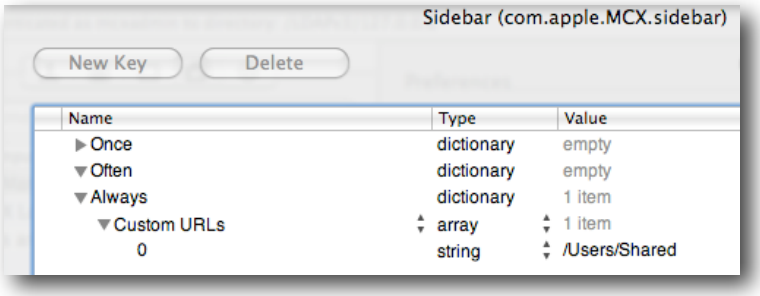
Use this setting to establish a common screen saver for all managed clients.



Note that the password key shows up only in the Always domain.

Sidebar

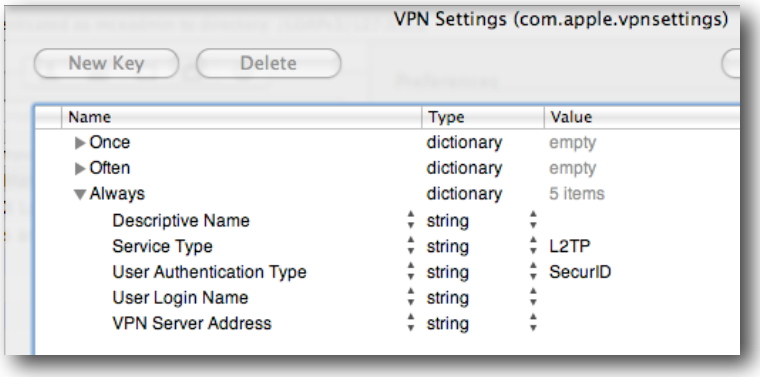
This setting is for adding custom URLs to the Sidebar. It is not the same as the com.apple.sidebarlists settings. This setting allows you to add the path to a file or webloc that is stored in a common space, such as "Users/Shared."



You can add the path to any file in common space.

VPN Settings

These are settings to preload the VPN values.



VPN settings

Several of the settings here are still experimental as of the time this guide was written. The MCX team added them to enhance management. In some cases, the settings have not been widely tested, so test your configurations well before deploying them.

User Accounts—MAs, PHDs, and More

There are three classes of user accounts—local, network, and mobile. With Leopard, information on these accounts is stored in either the local directory, the network directory, or in both places at the same time. Along with these accounts are two locations for the user's home directory, on the network or on a locally attached drive, usually the client system's boot drive. Keeping the combination of the user's directory information and the user's home directory data functional is the challenge. This section next looks at each class of account and where it is best employed.

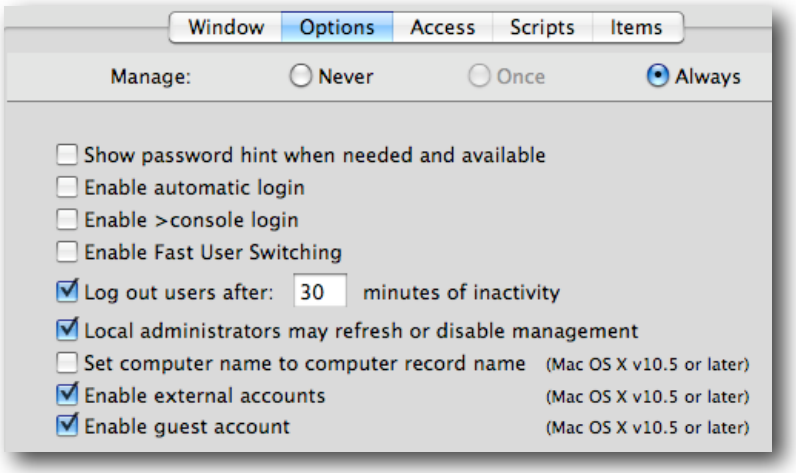
Local Accounts

Local accounts are just that—user accounts established on a local directory. The home directory for this class of account is stored in a path that can be reached without moving across the network. Local accounts have the benefit of being able to function when there is no network connection and are totally portable in that the computer can be relocated without any adverse impact on the account. Three types of local accounts exist: guest, non-administrator, and administrator.

Guest Account

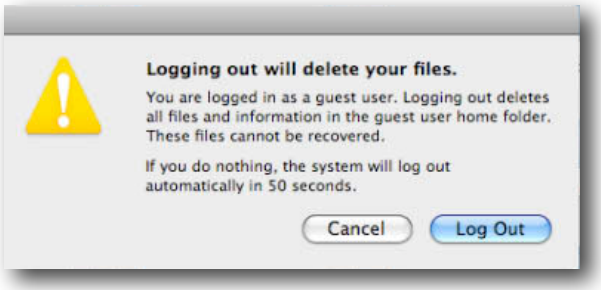
New to Leopard is the guest account. It was created in response to the need for an anonymous user account that can be used in circumstances where user tracking and logging isn't needed. Good examples of this would be a locked down visitor kiosk, a kindergarten computer, or other systems that are configured in such a way that an anonymous user logging in would provide more benefit than possible harm.

In MCX, the guest account is activated in the Login preference setting:



Globally activating guest account

The guest account is only functional on Leopard clients; this does not work with Tiger. When a user logs in as the guest account, a complete home directory is created for the user from scratch using the local home directory template. A key feature of the guest account is its temporary nature. At logout, the contents of the guest account's home directory are deleted. There is no way to go back into the system to recover anything accidentally left behind. If a guest account is being used for work that needs to be saved, the user needs to either save to an external device or mount a network share point and copy the relevant data to the network.

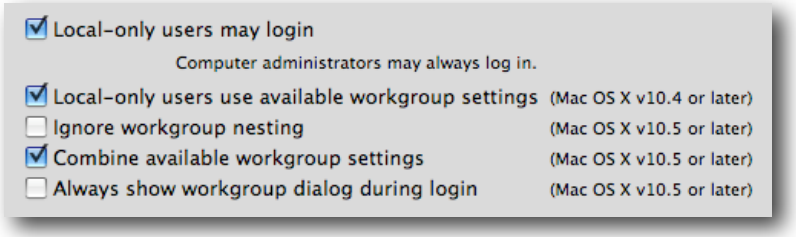


If you need your data, make sure you relocate it.

In the login window, the guest account either appears as a single icon labeled "Guest Account" or, in the name and password fields, you would type "guest1" with no password. The guest account is treated like any local account for management purposes.

Non-Administrator Local Account

On a client system, a non-administrator local account would be a local account with no administrative privileges. Examples of this are the generic "student" and "teacher" accounts, or the "maclab-11" account to go with the portable labeled "maclab-11" in the cart. This type of account is easy to set up on one to a few systems, but rapidly becomes an administrative hassle when trying to deploy and manage across hundreds, or even thousands, of client computers. MCX settings for this type of account are usually done at computer level and can use workgroup settings if the flag is set to force local users to respond to them.



Local user management set in Login/Access

Local user accounts are best used when network logging and tracking of users is not required and/or the client systems are often removed from the network.

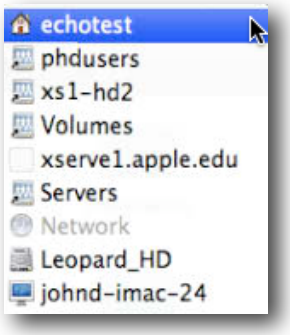
Local Administrator Account

A local administrative account is pretty much required on current systems. The account can be hidden (contains a UID less than 500) or just inaccessible to non-administrator users. The local administrator account is used to set up the Apple Remote Desktop agent or any other systems management tool, plus it is used for local maintenance. In MCX, the local administrator account can bypass MCX settings, normally, by holding down the Option key when logging in. The directory administrator can also force local administrators to respond to MCX settings in the Login/Options pane. Although with Tiger, you could add a local non-administrator account to the network administrator group, essentially making that user a local administrator, no such capability exists in Leopard.

Network Accounts

Network home directory accounts were created with Mac OS 9 for Macintosh Manager. The concept was for a network user to be able to log into a workstation, yet maintain his or her documents on a network share point. All of the user's preferences were copied to the local computer, and all the work the user did was stored in this single folder on the network. Mac OS X introduced a new structure in which everything for the user is stored in a network home directory—a folder containing not just the user's documents, but his or her preferences, music, movies, web caches, and everything else.

Although it is possible to tweak the network account setting to force the network user account's home directory to exist on a local drive, that's not the way suggested here. That would be better served by using the mobile account setup in the next section. A network user account consists of a network directory entry containing the user's identity, password, and any unique contact and management information for that account. The directory entry also includes the location information for that user's home directory. With Leopard (and Tiger), this home directory usually lives on an automount share point on a Mac OS X server. It can live on any network share point using AFP, SMB, or NFS to provide the network file sharing.



Network home location

Network accounts are extremely easy to set up and manage. All account information resides in a network directory; management settings and the home directory also reside on the network. The user account can, theoretically, log into any workstation bound to the directory server and be working in the user's home directory at any time. Changes made to the user account information, MCX settings, or home directory are immediately available to the user at next login. Another advantage is that the account could be used to back up a local guest or generic account. The users log in as Guest or Student, then mount their network home directory to save items that they create locally. The downside here is that users often get into the habit of just opening the items from the network store, enhancing the problems that are encountered using pure network accounts.

There is a saying "In theory, theory and practice are identical; in practice, they are not." Apple does not support network home directory accounts on networks less than 100MB/switched/wired. Many applications behave very badly with network homes, creating massive amounts of traffic to and from the home directory server just for basic needs. For example, just opening a web browser creates open log files and cache files with immense traffic before the user even gets to the first website. Using iLife with a network home directory could be problematic. Imagine trying to perform a video capture in iMovie to your home directory located on the network.

The best use for network accounts is in high-speed networks with low demand on the user's home directory. When network accounts are required, you should activate the Folder Redirection setting in MCX to keep the cache files local. While that may not totally resolve application traffic problems, it should reduce them significantly.

Mobile Accounts

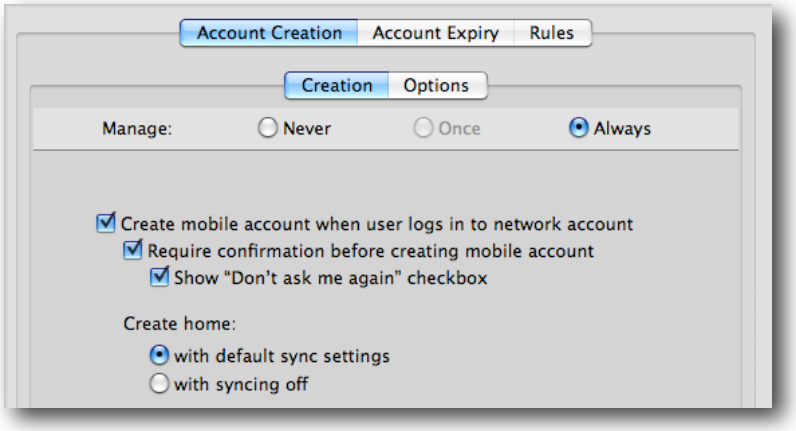
Combining the ease of management in network accounts with the performance and portability of local home directories, mobile user accounts are literally the best of both worlds. The idea is that your user account information is stored in a network directory and at login is cloned to the local directory on a client system. Your network home directory contents are cloned onto the local system, and you have the option of mirroring your work so that your network home directory and your local home directory always contain the same data. Leopard introduces several options to enhance mobile accounts, such as expiry, FileVault, external accounts, and granular sync settings. This section will discuss setting up mobility and how all the parts play together.

Practically all types of users can use a mobile account, such as administrators, secretaries, support staff, teachers, and other faculty who use the same computer every day. If people primarily use one computer all the time, they could have their account information managed on the network—passwords, policies, and collaboration settings—and have their home directory on the computer they go to or carry all the time. Of course, mobile accounts are perfect for 1 to 1 computers. Each user has a network managed account with all of his or her work on the computers being used all of the time. Even if a user switches between a couple of different computers during the day, being set up as a mobile account makes both productivity and management much easier.

Setting Mobility—Account Creation

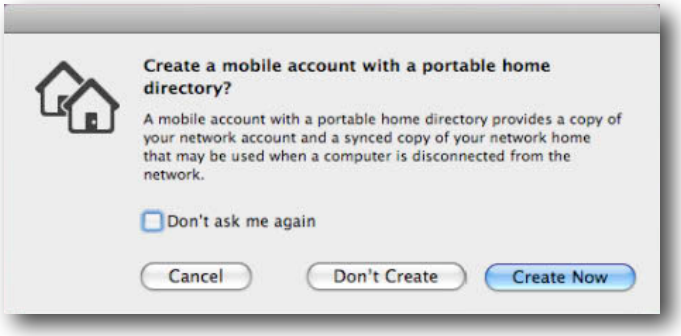
Mobility can be set at user, computer, or group level. Doing it at user level for testing is fine, but you'll find yourself going back to the user account off and on to adjust settings. That can become a bit bothersome. Setting mobility at group level works fine for pure Leopard environments, but Tiger systems see the mobility workgroup as a choice, not a requirement. Using the computer group as the focus for the mobility settings gives you both a more centralized focus and the ability to define which computers get mobile accounts while others may remain in use for network users. The use of computer groups or workgroups for mobility settings depends on the circumstances. You may want a 1 to 1 project set up where every computer in that set gets nothing but mobile account settings, yet your library computers may still support only network user accounts. Carefully setting up syncing and other mobility options can help you provide a well run environment.

To set mobility, log into an administrator system as local administrator, open Workgroup Manager, and create or locate your computer group. (If you will set mobility by group or user, the steps are the same, just the account type is different.)



The initial MCX Mobility window

Select the "Create mobile account when user logs in to network account" checkbox. It is recommended that you deselect the checkboxes for "Require confirmation before creating mobile account" and "Show 'Don't ask me again'" because these options give the end user the option of backing out of the setting. The following is the dialog that the end user sees at login—this may be too many choices for the user to make.



The Don't Create button causes the user to be logged in as a true network account and mounts the user's network home directory. The "Don't ask me again" checkbox flags the system to never try to create a mobile account. If the user selects only the first option, the next time the user logs in, the dialog appears again. If the user selects the checkbox, the dialog does not appear—and you'll need to figure out why so many of your users are complaining that they can't log into their computer when they go offline.

Whether or Not to Sync

The setting at the bottom of the Account Creation pane needs the most consideration, planning, and work. The default is to create a mobile account with Portable Home Directory (PHD) support. The other choice is to create the mobile account without any synchronization options. Which is best depends on the situation.

MA/PHDs Support RRtS

Mobile accounts with portable home directories support Rapid Return to Service. The idea is that a user logs into a client system and his or her network home is cloned completely to the local system. That user's account information is cached locally, allowing that user to log into the system both on and off the network. While the user is working on his or her computer, the filesync mechanism is maintaining a mirror of the data in the user's local home directory with the user's network home directory. If anything happens to the user's computer—the backpack gets tossed down the stairs, the portable gets dropped in the snow, the teacher leaves his MacBook on the roof of the car and drives off—that user can get a loaner/replacement from the tech support folks, log in, and have all of his or her data resynced from the mirror (the network home) in a matter of minutes.

Capability with a Cost

This capability does not come without a price. The infrastructure to support PHDs must be robust and requires a lot of storage. You can get an idea by looking at the size of your own home directory on a portable you or one of your colleagues is using. A student using all the tools offered in iLife to create projects for school can easily gather 6GB to 8GB of material in a matter of weeks. Teachers can gather 10GB to 20GB a semester. Trying to decide what is critical data is almost impossible for the IT staff. What is critical to an eighth grader is very different from what is critical to the network administrator. This topic is explored in more detail later in this document. The concept is solid though. What you are offering the end users is the security of knowing that their data always resides in at least two locations, and if anything happens to the computer the school lent them, they can get back up and running in a hurry.

What If I Don't Sync?

That said, there are reasons why you may not want to turn on sync. First, the computer may spend far more time off the network than on it. Having the user's account information synced, which is always done, may be enough. Second, if users work in an office where they can back up, archive, or sync their own material, not syncing may work for them. Third, if you have provided a MacBook to every student in your school and your infrastructure cannot support more than providing them with basic network filesharing, you may have to forego syncing. A current MacBook ships with a 120GB, 160GB, or 250GB hard drive. If you provided PHD support for a thousand users and covered only 10% of their possible home directory space, you may need to set up as much as 23 terabytes of storage on the network. (The computer may have 20GB to 40GB of system files and applications, but that still leaves 100GB to 200GB or more of free space.)

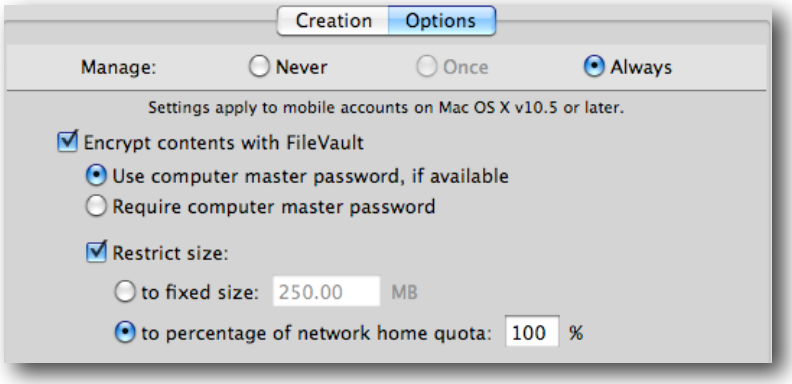
Here is a summary of the choices. If you turn on PHD sync, you can mirror your local working directory back to the network. From there, you might even add backup to the network storage giving you true survivability. If you decide not to sync, it is up to your users to provide a backup or mirror of their own data.

The next section continues this discussion by exploring the various options for tuning mobility.

Mobility Options

FileVault

Leopard mobility provides several powerful options. The first is FileVault. You can select to create a FileVault container for the user at first login, encrypting the contents of the user's home directory. The FileVault container is opened at login; if sync is enabled, the contents are mirrored to the network drive. At logout, FileVault recovers any extra space and closes itself up tight. If you have users who keep sensitive data on their school-owned computer and you want to maintain network manageability, you will want to activate this option. Another benefit of this option is that this setup allows you to define the size of the local home directory. You can think of this as a locally controlled quota.

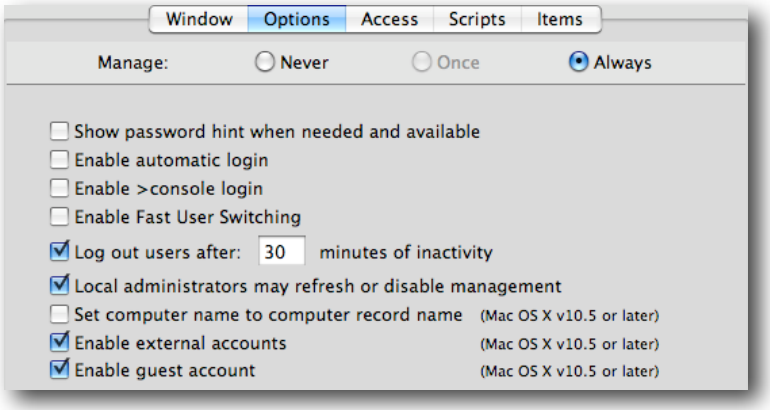


FileVault options for mobility

Home Folder Location—External Accounts

The home folder location option is the hidden gem of the mobility settings in Leopard. By default, the mobile account's home directory is created in the /Users folder on the boot drive of the client system. Using the home folder location setting, you push the home folder creation to a different partition, drive, or even an external device. This functionality is referred to as *external accounts* where the mobile account user logging into a managed client is not using the boot drive of that system for that user's local home directory location.

Setting this capability up requires two parts—enabling external accounts and selecting a home folder location. The first part is done from within the Login settings:



Note the "Enable external accounts" option.

This setting allows MCX to support the use of a non-startup location for the mobile account user.

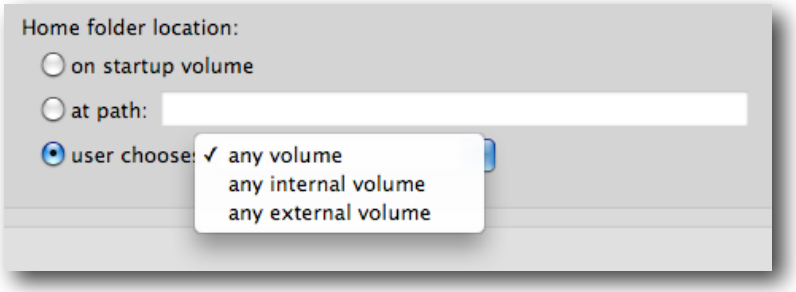
How does the use of external accounts affect the deployment of mobile accounts? Quite significantly. Here are a couple of examples:

Students use their USB keys as their home directory location for their mobile accounts. When they use one of the MacBook computers in the portable cart, they don't have to log in as a generic account or rely on PHD sync to get their home directory. They just plug in the USB key and log in. The account is managed across the network; their home directory is on the USB key. If they switch computers, it doesn't matter because their home travels with them. Benefits include an automatic quota limited to the size of the USB key, and if they have PHD sync turned on, their USB home is mirrored back to the network. This provides RRtS in case they drop the key in a puddle.

Now you can take a look at an example of how the external accounts feature could be used in an educational setting. In this example, teachers in a digital learning environment have their primary mobile home on the MacBook Pro they were issued. The teacher goes to the new high-end video lab, all decked out with new 24-inch iMac computers. Instead of just logging into the iMac computer and relying on PHD sync to clone all of their information, they reboot their MacBook Pro in target disk mode (holding down the T key at startup), making their portables now external FireWire drives and effectively an external account. The teacher goes to the iMac in the front of the room with all the AV/projection cables attached, plugs in her MacBook Pro to the FireWire cable and MagSafe power adapter provided by the tech support team, and logs in. Her home directory is still on her MacBook Pro, so she can get to work right away. The students do the same. After class, they can log out and carry away with them all the work they did because it never went anywhere but into the designated home directory on the MacBook Pro.

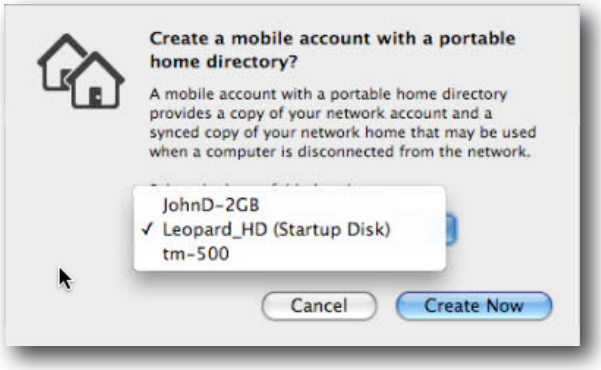
Note: Some of this will not work with the latest generation of MacBook computers (October 2008) because they do not support target disk mode.

Setting this up in MCX is a matter of choosing the initial home folder location:



Home folder location

Choosing the setting location is important. For schools where every student has been assigned his or her own computer, you would want to choose either "any volume" or "any internal volume" (or leave it set to "on startup volume"). To force the users to choose an external device initially, choose "any external volume," and if you are using a system with multiple partitions, you can set the path to the partition. This would come in handy, for example, if you created a boot partition with the System and Applications, then wanted the user's home folder to be on the second partition. Here's an example of an open setting where the user can choose:



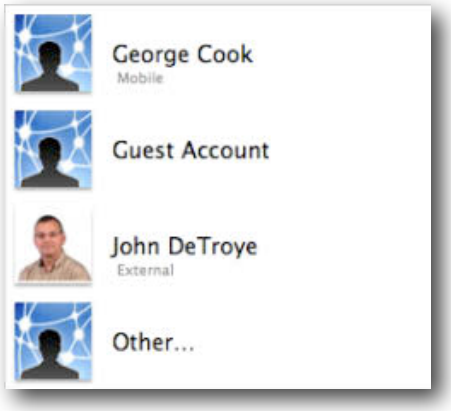
User chooses startup, FireWire, or USB drive.

Once logged in, the user sees a normal desktop and other settings. The key is where the user's home directory is now located:

Mobile account with external home directory

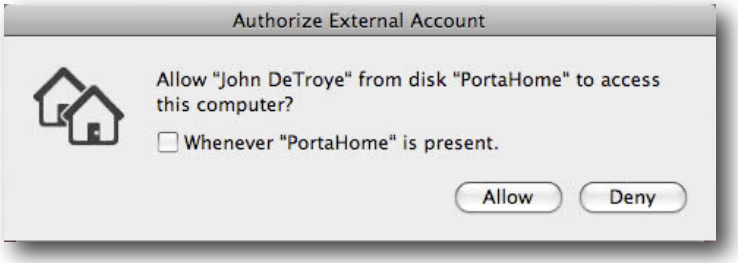
External Account Behaviors

The basic behavior of an external account is to log in and see the user's home folder as normal. If you view the login window as a list, you'll see the account as the following:



External account at login

The external device can be unplugged without error at the login window. If you go to another bound computer and plug in the device, the icon of the user reappears in list view; otherwise there is no indication the device is plugged in. Behavior changes if you go to an unbound system; for example, your home computer. If your home system is at the login window, this dialog appears:



External account device attached to unbound computer

You must then authenticate as a local administrator to allow login to continue. The benefit of this is that you can continue to use your external account's mobile home directory as your current working directory.

Note: If you had a Windows computer, you'd get nothing—the external device *must* be formatted HFS+ for this to work. This also means that those generic USB keys students get for school will need to be reformatted to work as external account devices.

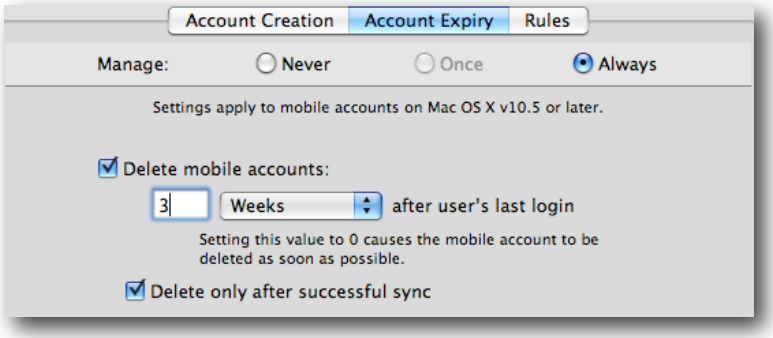
If you plug the device into a computer already logged in, the drive shows up as a normal external storage device. The external account's home will be on the device in /Users/<username>. Unless you enable FileVault protection, your external device will be open for access, so be careful with your data.

Account Expiry

One side effect of working with mobile accounts is that each time a user logs into a client system, a cached copy of that user's account is created on that computer as well as a local home directory. In an environment where everyone has their own computer this can work really well. In an open lab or portable cart setup you could end up with many instances of the same user having logged into random computers over time, leaving little cloned copies of themselves all over. Luckily there are two solutions to this problem. The first one was invented by Greek scholars about 2500 years ago—the seating chart.

Ensuring the students use the same computer each time they come to the lab, or scheduling and assigning specific users to specific portables, keeps the number of orphaned mobile accounts at a minimum.

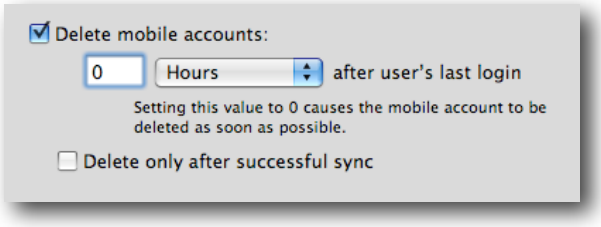
The second solution is account expiry. This setting is designed to age a mobile account. When it has reached the expiration date, the account is automatically deleted along with the local home directory at the login event by another user.



Mobile account expiration settings

Figuring out what the expiration setting should be is important. If you set the interval too short, you'll force users to go through a complete re-sync at the next login. A good practice would be to find out what the longest break is during the year, including flu season, then double that number. Taking the expiration to the extreme, you can set the value to zero and turn off the "Delete only after successful sync" setting. That done, your mobile accounts are deleted at logout. This could come in handy in non-sync setups where you want your users to have the benefit of running as mobile users with local homes, yet do not want the systems to have any leftovers after use. Users would have to store any files they want to keep on another device, use a workflow share point with hand-in folders, or in the case of lower grades, print their work before logging out.

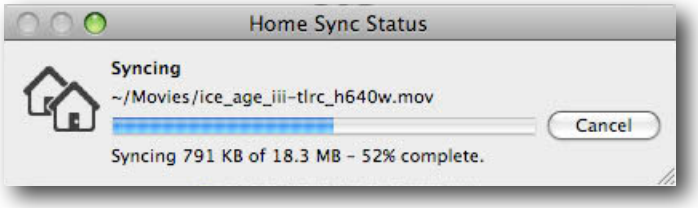
Turning on the expiration setting for 1 to 1 computers with their home directory on the startup disk is not a good idea. External accounts with the home directory on an external, removable device are not affected by the expiration settings.



Immediate elimination of mobile accounts

Rules for Portable Home Directory Sync

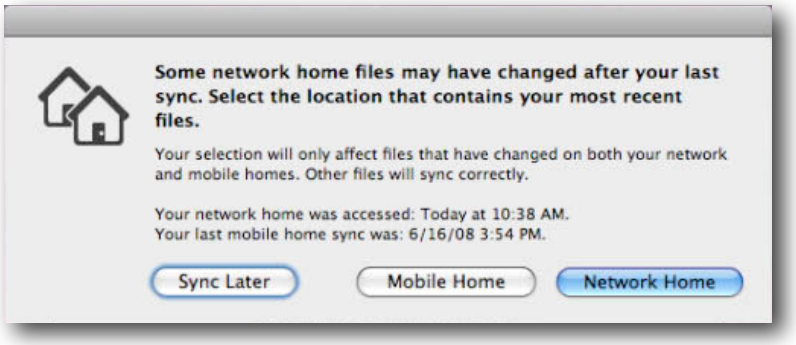
Setting up PHD sync can be as simple or as complicated as you want to make it. The idea is very basic—at first login, the entire contents of the user's network home directory plus the contents of any local home directory with that user's name are merged. If there is no network home directory yet, it is created using the network home server's home directory template, then cloned to the local drive. While the user is logged in, at a defined interval, the entire contents of that user's working directory (the local one) is synced with the network home, minus the user's Library folder and a few other active files. At logout, the entire contents of the local home is synced, minus a few designated files.



First time login sync brings everything together.

Subsequent logins sync only the user's Library folder and the Microsoft User Data folder, if it exists. While logged in, the user's entire home folder, minus the Library is synced. Only files that change, are created, or are deleted, will be synced. This is the default behavior, and if the network infrastructure is up to speed, it works very well.

Conflicts might happen. If you add items to a user's network home directory when the user is offline, the next time the user logs in, a conflict dialog will appear. In Leopard, you get a warning when you last touched your network home versus your local (mobile) home:

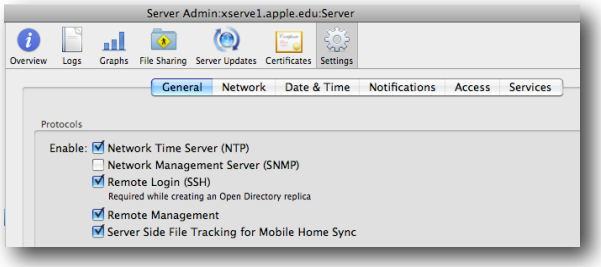


New Sync conflict dialog

At least now, you can check and say "Hey, the IT folks did say they were adding things to our homes. I should click Network Home."

Server Side Sync

PHD sync capability has significantly improved with Leopard. In addition to a whole new file sync mechanism, the server now supports "server side tracking" for mobile home sync. This can seriously improve PHD sync performance for Leopard clients. With Tiger, when PHD sync kicked in, the local home directory was watched over by a FileSync database that kept track of all file system changes. When the sync interval hit, this database reported the files that had changed, been added, or deleted. At that point, FileSync had to scan the entire network home directory for that user, file by file, looking for the files that matched the changes reported by the database. If you had a few hundred files, you wouldn't notice. Change that to a few hundred thousand files and things got really slow.



Server side file tracking for PHD sync

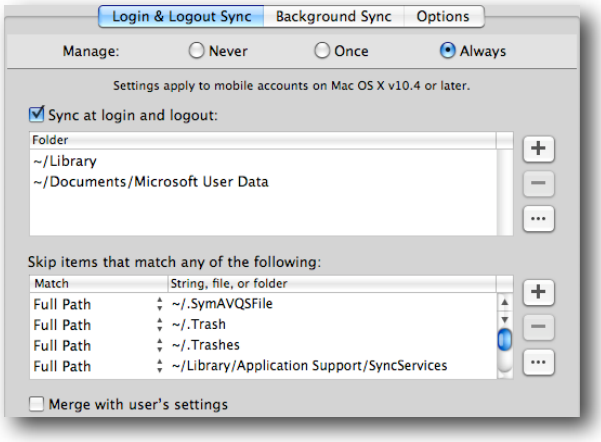
Leopard allows you to set the same FileSync database on the user's network home directory. When file system changes take place, a FileSync database is maintained at both ends. The PHD sync process now consists of two databases being compared. It no longer matters how many files you have in your home directory.

For this to work for your mobile accounts, all users must have a valid shell assigned in the network directory; using /dev/null does not work.

User login shell set to allow server side PHD sync

Tuning Login/Logout Sync

The default set for login and logout sync looks like this:



Default Login & Logout Sync settings

If you turn on these settings, you can override the default sync settings. There are two primary scenarios for using PHDs—24/7 users and 9 to 5 users. The first case are the students/faculty members who have been provided with a system for full-time use. They take the computer home, use it all day, weekends, and during non-school days. Those people should have the least load placed on them when returning to the network for performance reasons, and they always treat their mobile account's local home directory as their primary storage location. If such a person then came back to school from a long weekend and had uploaded a few hundred MBs of podcasts, worked on an iMovie project, and updated his or her iPhoto collection (plus the movie trailers, and so on), the user's initial login at school would consist of a lot of waiting for a very long sync period.

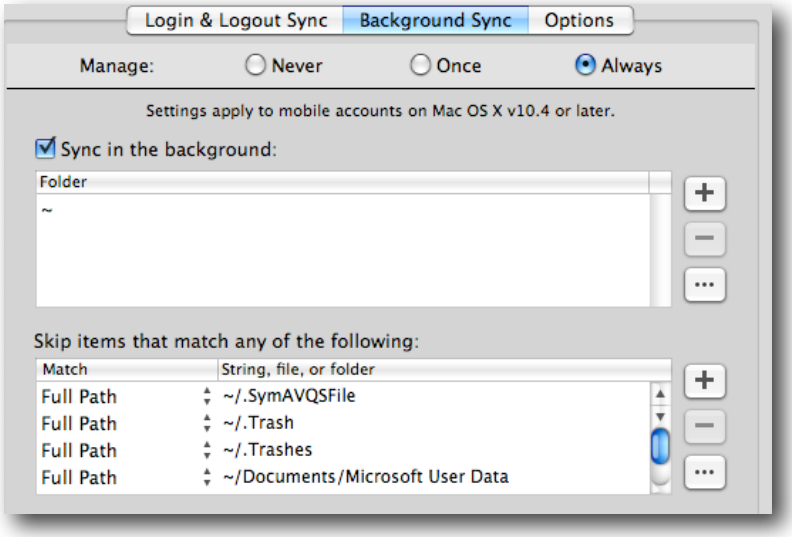
Because these users already have access to the primary copy of their home directory, you would want to get them logged in as fast as possible, then let the offline additions to their home directory sync in the background.

Users who work in the office all day, then head out at 5 p.m. can have even more open settings. For these users, you can set their login/logout sync to be "~" because they haven't added anything to their systems overnight. If you add some files to their network homes during off hours, those files will then sync at login.

One file that does need to be added to the login/logout sync set, if you use the defaults as a starting point, is the iTunes library. It's also a good idea to add either the iPhoto library itself or the Pictures folder to the login/logout set. This is because the sync process returns an error when a user is running iPhoto and background sync tries to mirror the picture library. If you leave the iTunes and iPhoto libraries set to sync in the background, you should warn users to quit those applications and choose Sync Home Now from the menu bar.

Tuning Background Sync

Background sync needs very little tuning. You might want to add the iPhoto Library or the Pictures folder to the Skip set to avoid the issue noted above.

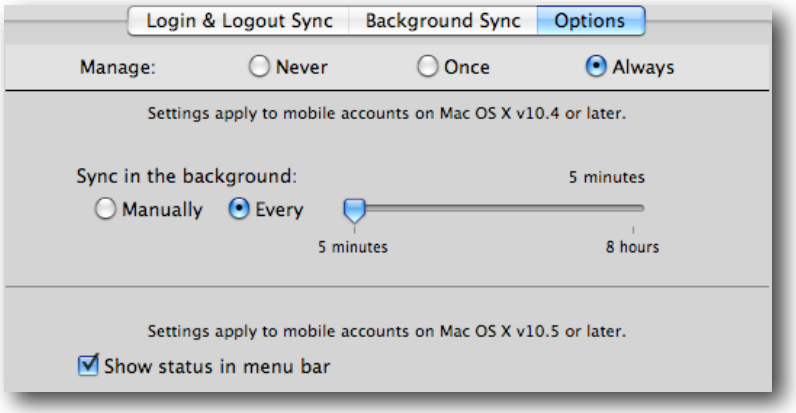


Note that the Microsoft User Data folder is not synced in the background.

One thing of note is the exclusion of the Microsoft User Data folder. This is due to the behavior of Entourage. Because file sync acts on files whose modified timestamps have changed, it would constantly be trying to mirror the Entourage mail database. Because the Entourage database is one large flat file, every time you check mail or select the Entourage application window, the file shows that it changed. This would result in a constant state of being mirrored. If you are not using Entourage, you could delete that item from the Skip set.

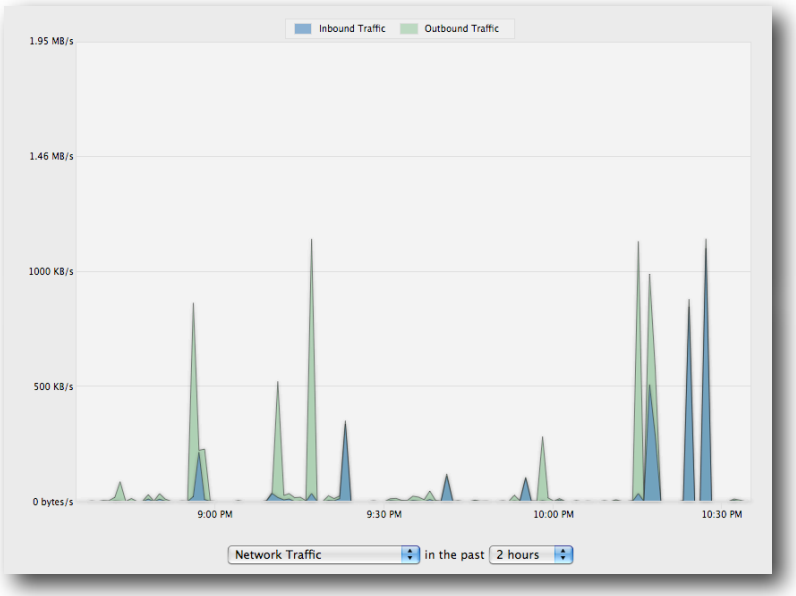
Tuning the Timing (Options)

The default interval for PHD sync is 20 minutes, and that may be too long.



Keep the sync interval short.

Using the default time may cause excessive network traffic when large numbers of clients build up many changes between each interval. Several sites set their sync interval to manual. A good solution is to set the interval as short as possible for your network—a good value is in the 9 to 11 minute range. Leaving it set to manual can work if the users are trained to invoke it often, perhaps in an AV lab where they don't want PHD sync to interfere with video capture. Where the problem pops up is when the users wait until just before the bell rings, then rush to sync. The traffic would be a tsunami of filesharing activity. Setting the sync interval to be shorter allows less work for the FileSync process because fewer changes take place between each interval. What you would see if you were monitoring the network traffic is a series of smaller, shorter network traffic peaks. The chance of overwhelming your available bandwidth is much less.



Short sync intervals keep the traffic down in the long run.

Other Options—Getting Restrictive

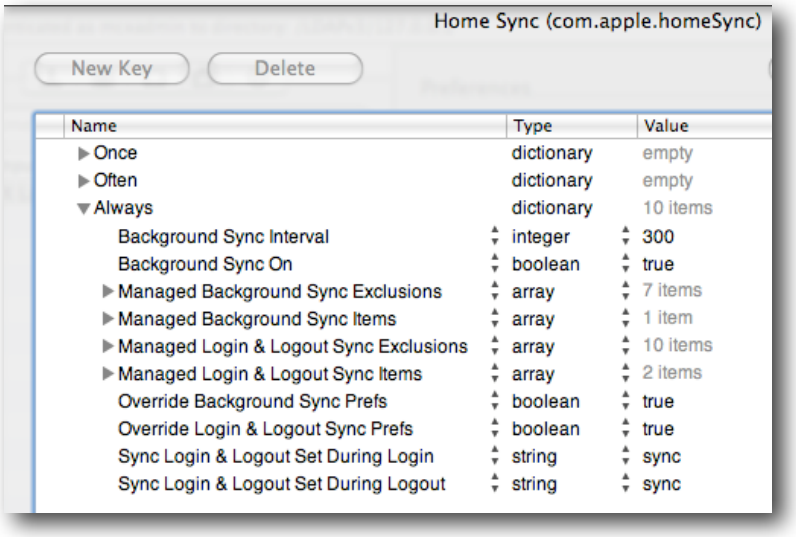
Many IT shops decide to manage PHD sync by shutting it down to just syncing the user's Documents folder or skipping all MP3 files. Doing this is not recommended for a couple of reasons. First, if you don't know what the data is or where the users store it, making a call like that can result in lost data—voiding the concept of Rapid Return to Service (RRtS). Second, if you try to filter by file type, say .mp3s, you may be skipping critical podcasts or an audio interview a student did that is imbedded in a Keynote presentation. That would cause the Keynote presentation to be rendered worthless. Remember, many files today are bundles, with dozens of smaller files inside.

Finally, trying to force the users to store their critical items in the Documents folder because all you'll agree to mirror is that folder only results in the user putting everything into that folder, including their iTunes library. If it comes down to a matter of network storage space, you might be better off setting up FileVault containers with a specified size or using external accounts on USB keys issued by the school. Then again, you could always add more online storage.

What is mission critical to an eighth grader or a teacher often has no bearing on what the IT support staff thinks is critical. Support the mission—support the end user. Select a file synchronization scheme that balances performance with the need for data redundancy and availability.

Digging Deeper—Details and Mobility

Many of the required settings for mobility are exposed in the graphical user interface. However, there are also just as many settings buried in the Details pane. These settings are hidden on purpose. Some really serious problems can be created if you play with these without careful consideration. That said, this next section examines the available mobility and syncing options.

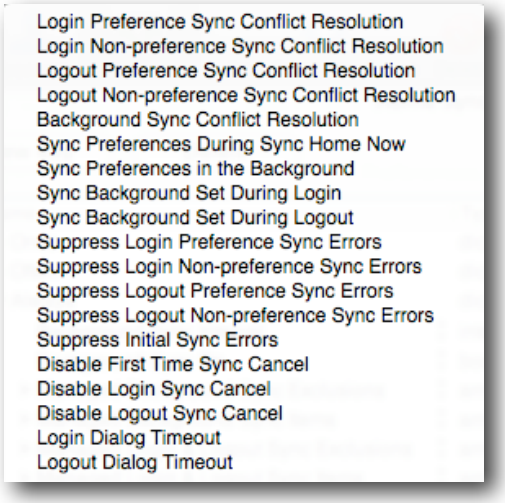


The "visible"sync options from underneath

These are the items that you can see from above. A few settings could be tweaked, specifically, the last two listed. The options there are "sync," "not sync," and "automatic." You can set the sync option to skip sync at login and only sync at logout, or vice versa. The Override settings mean to override the local settings made in System Preferences by that user. This would be if you allowed the user to set his or her own sync preferences. By overriding them, you are forcing the use of the MCX settings from the network directory.

Hidden Sync Preferences

The settings that are hidden are listed below. These are very dangerous settings to play with. It is highly recommended that you test any adjustments to the defaults with a non-production set of systems.



Hidden sync preferences (keys)

Note: Playing hard and fast with these settings may result in data loss for your users. Guesses about which location is right for your users will be wrong half the time. It's better to teach them to pay attention to where they put their files, and report really odd behavior to the system administrator.

Here is a cautious look at each of these settings with the qualifier that employing these keys is not supported in any way outside of the default values. The descriptions of the actions have been taken directly from the hints in Workgroup Manager.

Login Preference Sync Conflict Resolution

This setting affects syncing ~/Library at login.

- Use the "mobileHomeWins" setting to merge homes and have the local (mobile) home win conflicts
- Use "mobileHomeCopy" to copy the local home to the network home.
- Use "automatic" or "networkHomeWins" to merge homes and have the network home win conflicts.
- Use "networkHomeCopy" to copy the network home to the local home.

Login Non-preference Sync Conflict Resolution

This setting affects syncing everything besides ~/Library at login.

- Use the "showConflictDialogs" to show dialogs when conflicts occur.
- Use "mobileHomeWins" to merge homes and have the local (mobile) home win conflicts.
- Use "mobileHomeCopy" to copy the local home to the network home.Use "automatic" or "networkHomeWins" to merge homes and have the network home win conflicts.
- Use "networkHomeCopy" to copy the network home to the local home.

Logout Sync Keys

These behave the same as the Login keys. There is a consistency in the pattern here…

Background Sync Conflict Resolution

This setting affects syncing everything besides ~/Library in the background.

- Use the "automatic" or "showConflictDialogs" setting to show dialogs when conflicts occur.
- Use "mobileHomeWins" to merge homes and have the local (mobile) home win conflicts.
- Use "mobileHomeCopy" to copy the local home to the network home. Use "networkHomeWins" to merge homes and have the network home win conflicts.
- Use "networkHomeCopy" to copy the network home to the local home.

Sync Preferences During Sync Home Now

This setting allows you to sync ~/Library during a Sync Home Now sync. Set to "automatic" for the best choice, "sync" to sync preferences during Sync Home Now, or "dontSync" to not sync preferences during Sync Home Now.

A few are skipped now to show this one:

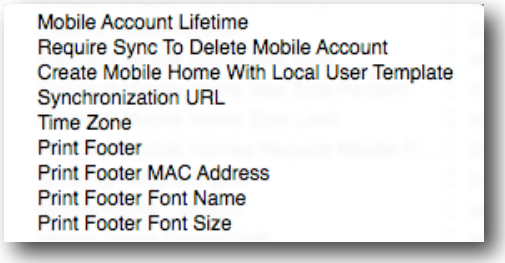
Suppress Login Preference Sync Errors

Set this to true to suppress error dialogs during login sync of ~/Library.

The "suppression" keys can result in problems for users where an actual problem between two files may exist. While some errors get reported and are not really problems, if you suppress the conflict dialogs, you may create a bigger problem.

Hidden Mobile Account Keys

These are settings that come from the Mobile Account & Other Options (com.apple.MCX) settings and can enhance performance of the mobile account or set additional restrictions. Any of the Options not set in the graphical user interface show up here as unused keys. In this picture, the Expiry keys are unused:



Unused mobile account and Other keys

The first two listed wouldn't be here if Expiry settings had been enabled, so those won't be looked at here. The last four are the values used in the Printing preference that has already been covered. The important ones are the middle three.

Create Mobile Home With Local User Template

When creating a mobile account, create the local home folder using the computer's user template. This allows you to pre-populate the home template of your master image, then use that as the source of mobile account home directories versus pulling the template from the network server. The end result is that the network home directory and the local template are combined. The user has a local home directory consisting of the contents of both templates. What this allows you to do is keep the network template very light—that is, empty—and put all the needed items into a local template that is cloned as part of the image.

Synchronization URL

This is the URL of the network home used for home sync. It is only settable for mobile account creation. The string "%@" will be substituted with the user record name before use. Example: afp://myserver.apple.com/Users/BuildingA/%@.

Time Zone

This is set to a recognized time zone name, such as "America/Los_Angeles" or "US/Pacific" as found in /usr/share/zoneinfo.

FileSync Troubleshooting

Keep in mind that PHD sync is mirroring, not backup. So the first troubleshooting tip is to make sure the user didn't throw something away expecting "the copy on the server to be there." Files deleted at either end are deleted at the next sync interval.

If you need to try to track down sync problems, the client log lives at ~/Library/Logs/FileSyncAgent/FileSyncAgent-verbose.log. For PHDs with server side tracking, the logon server lives at ~/Library/Logs/FileSync-server/FileSync-server-verbose.log.

To reset file sync, there are two methods. The "soft" one is to log onto the client as a local administrator and delete the mobile user account and home directory from within System Preferences. The user can log in again and have his or her mobile account and home directory resynced from the network.

The "hard" reset involves the "soft" steps plus:

- On the server, delete the user's ~/.FileSync folder.
- On the server, delete the user's ~/Library/FileSync folder.

Workflow and Collaboration Tips

Before Mac OS X, one of the biggest reasons schools used Macintosh Manager and set up management by workgroup was to take advantage of the group folder functionality. Having a share point mount at login for the user with a hand-in folder and one or more folders for class materials is a great way to encourage workflow. The idea of workflow isn't new; it's been used for many, many years. The advantage Managed Client under Leopard has is that you can now create a single workflow for many users simultaneously, without the requirement for multiple workgroups for the user to choose among. Add to that the new collaboration tools within Leopard to provide a true ability to share and distribute knowledge, and you are one step closer to the goal of meeting the standards asked of students by the world—*be able to work together in teams, solve complex problems, and present your solutions clearly and concisely in both written and oral form.*

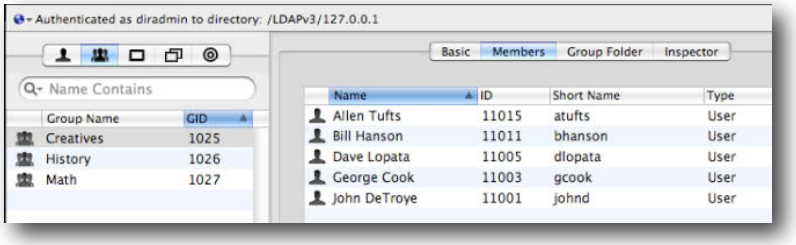
This section describes setting up an environment where the students and faculty can share information using common storage, yet allow for specific access to information by group as needed.

Setting Up the Workflow

The parts needed to make this work are a set of user groups reflecting classes or activities for the students and staff, a common share point with folders depicting the same set of groups, and the management settings to bring it all to the user's desktop. A workflow server would be the server, or servers, that contain common access share points for school use. By using "common" share points with subfolders, you no longer need to create dedicated "group folders."

Creating Groups, Not Workgroups

Instead of creating a series of workgroups with group folders for users to log into, you create only UNIX groups on your "workflow" server. These groups contain users from your primary directory server. The groups for this example are History, Creatives, and Math. In Workgroup Manager, connected to your workgroup server as a directory admin, you create these three groups and populate them.



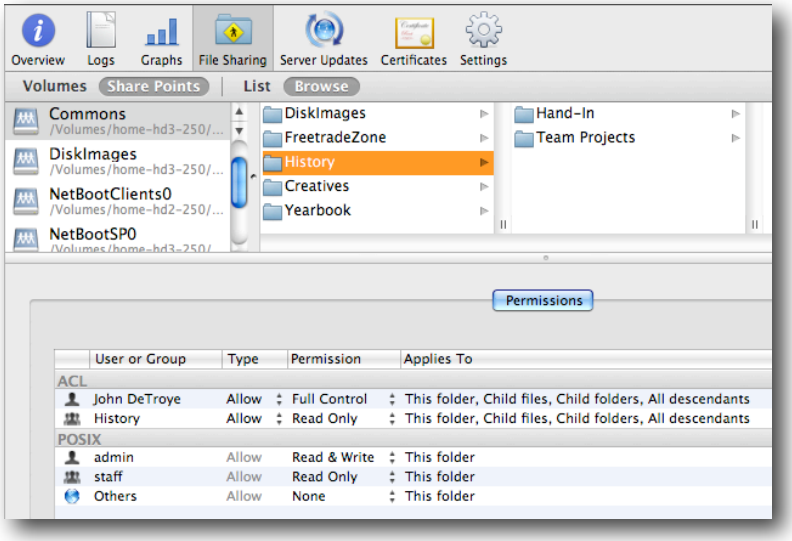
Three groups created on the workflow server

Now you move to Server Admin to finish this setup.

Building the "Commons"

In Server Admin, you select the workflow server, then the File Sharing tool. You then create a Commons folder on one of the drive volumes. Within that folder, you create subfolders for each of the groups. Keeping ahead of the game, you can also create subfolders for each group folder for Hand-In and other tasks as needed by the faculty member who will own the folder. You can also just show the owner of the folder how to manage it himself or herself, passing on the responsibility of keeping the folder in working order to the owner.

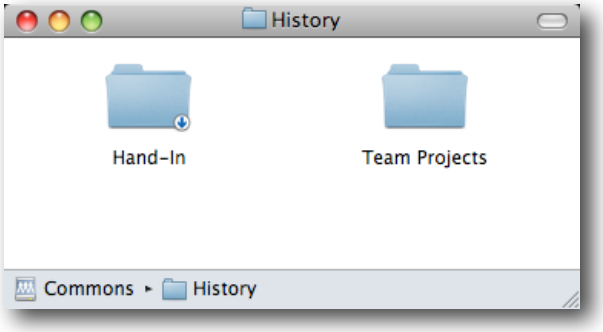
Next, you select one of the new group folders and add its owner and group into the ACL section of the Server Admin File Sharing pane.



Owner gets Full Control, group gets Read Only, Others get None

Next, you need to select the Hand-In folder inside History and turn it into an actual hand-in folder—the write only folder—by setting the basic permissions to "write only" for Others.

Repeat these steps with the other folders. Note that the only folder that is actually "shared" is the Commons folder. This provides a really streamlined method of providing your workflow environment. Test the setup by mounting the share point as various users onto your administrator system. Access to the folders and subfolders should match the ACLs you set.



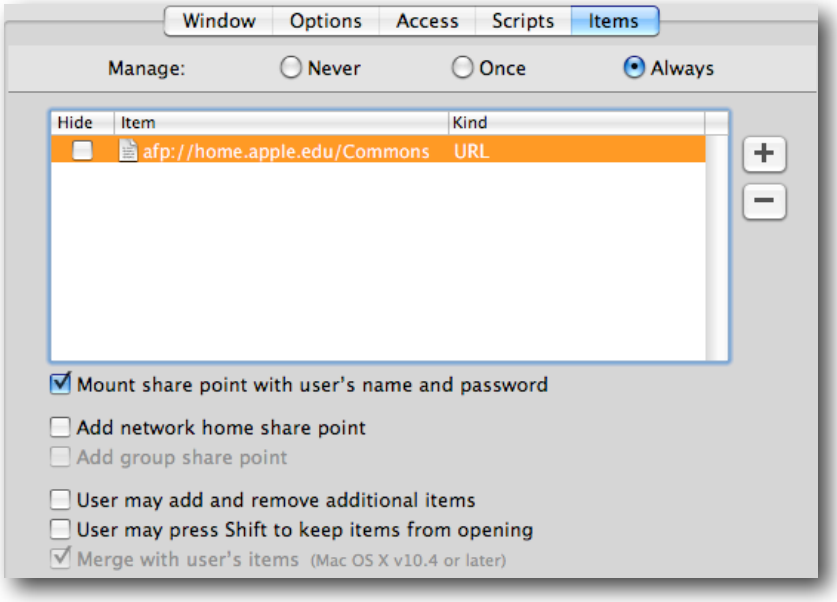
History folder from user perspective

Note that the Hand-In folder looks like one with the small arrow in the corner. If you had logged in as the user "johnd," you'd have full access to all folders. You can experiment with the various permissions settings to achieve exactly the effect you want. The Team Projects folder, for example, is set to allow all group members read/write access, but denies them delete permissions. This keeps people from accidentally trashing someone else's work. They can ask to have items deleted when necessary.

ACLs and MCX Together

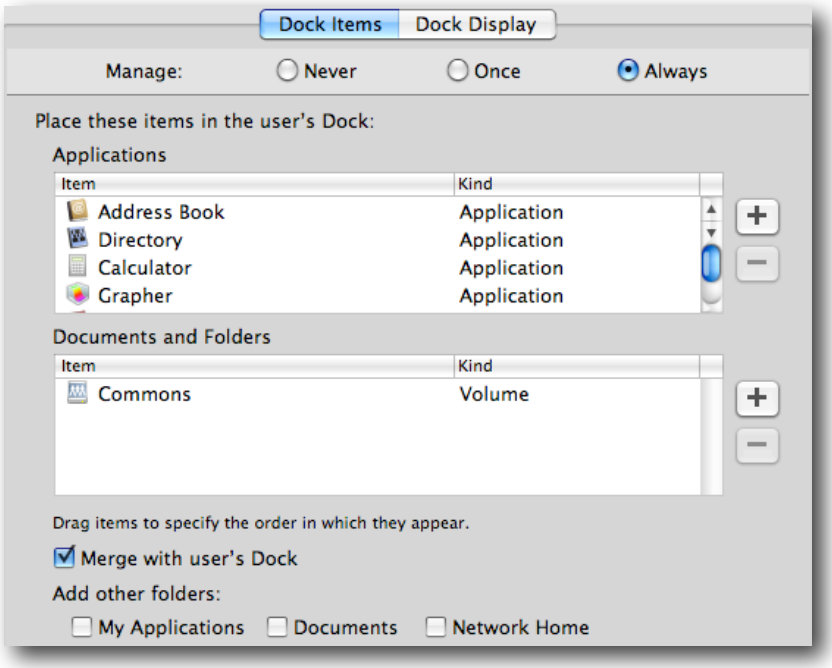
To make this workflow complete, you need to have the Commons folder show up when the users log in. First, on an administrator system, you mount the Commons share point on the desktop. Second, you launch Workgroup Manager and authenticate to your MCX server, then select the managed computer group you set up.

In Preferences/Login, select the Items pane, and select Always, then drag the image of Commons from the desktop into the window. Highlight the share point item, then select the "Mount share point with user's name and password" checkbox.



Commons will be mounted at login with the current user's credentials.

In case the user ejects the volume, you also add the share point to the Dock.



Commons is now a Dock item for all users.

Note: If you had only added the share point to the Dock, at login, the user would have seen a question mark in the Dock. This is because the Dock item is only a URL. At login, the Dock setting is checked by the Dock to resolve all the components. The URL would show a share point, but the Dock wouldn't see it—hence the question mark.

You can use this trick to create and mount share points with databases used by educational applications, such as Type to Learn, in order to premount the database for the application. If you do work with network databases for users, make sure you set the ACL on the folder to allow the user to write but not delete. Otherwise, the user could either toss the database in the Trash or not be able to save any test results.

Collaboration Tools—A Simple Beginning

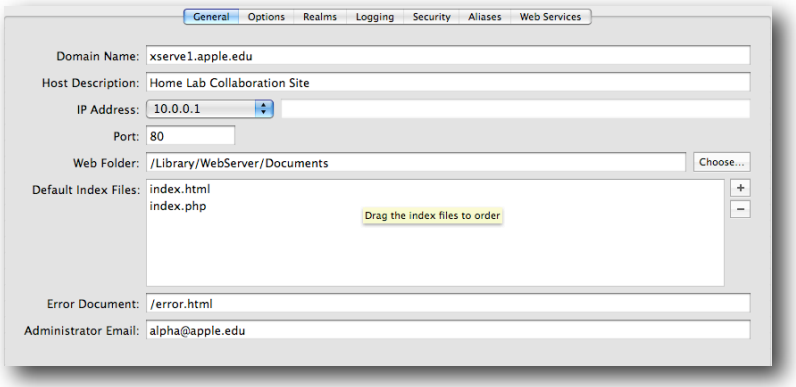
Leopard introduced a whole suite of collaboration tools for group or team work. Each of the tools has a use in the educational workflow. This quick look is only a taste—the world of collaboration services, podcasts, and so on, is huge and well worth exploring.

Who has not spent some time browsing Wikipedia? A teacher who is interested in educational technology would appreciate having his or her own wiki. Imagine not having to post and repost all the class materials or having the ability to let students help flesh out a class project online. This section looks at setting up the bare minimum you would need to start this process. Imagine a language arts teacher being able to post a reading assignment, then having the students comment on the assignment in a class blog? Think about an internal activities calendar run by the various groups at school.

Server Setup for Collaboration

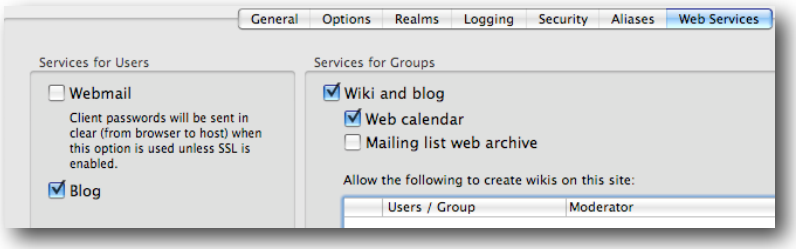
First, if your wiki server won't be on the Open Directory Master, you must use an authenticated bind to attach your server to the directory. Do that using Directory Utility**.** Using Server Admin, select the Web service. Select the Web Services pane in Settings. Choose a template for your wiki site or sites.

Next, select the Sites tool and select the current site "*." Edit the entries so that your site information is fixed versus blank and select the Enable checkbox.



Cleaning up the entries for your wiki site

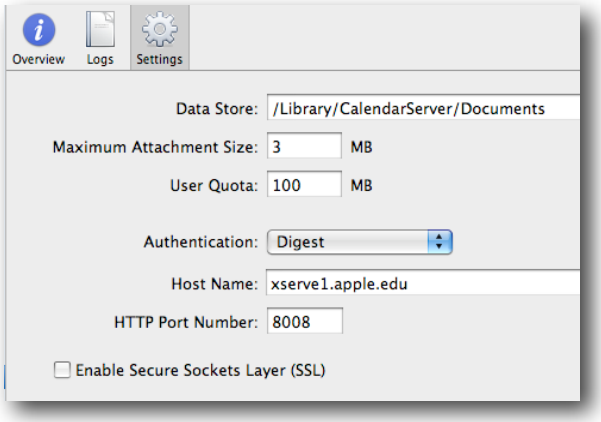
In the Options pane, make sure the Performance Cache is not enabled. Select the Web Services pane and turn on the collaboration items you will use.



Establishing collaboration services

Next, locate the iCal service and turn it on.

It's a good idea to set the authentication to Digest at this time.

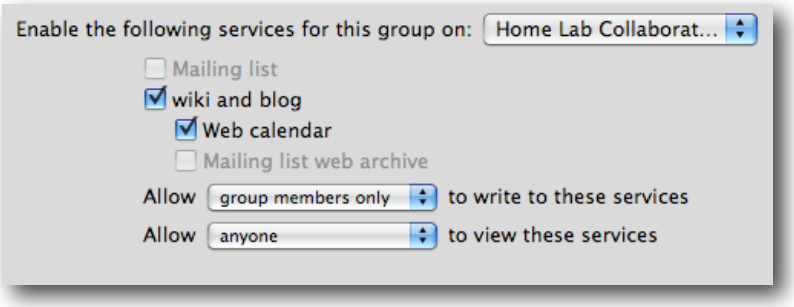


iCal service settings—choose Digest for authentication.

The services are using the boot drive of the server for their storage for now. You might want to change that before going into production, or just leave it this way for now.

Setting the Groups to Use the Collaboration Services

In Workgroup Manager, locate the group or groups for which you want to activate collaboration. In the Accounts/Basic pane, enable services as in this example:

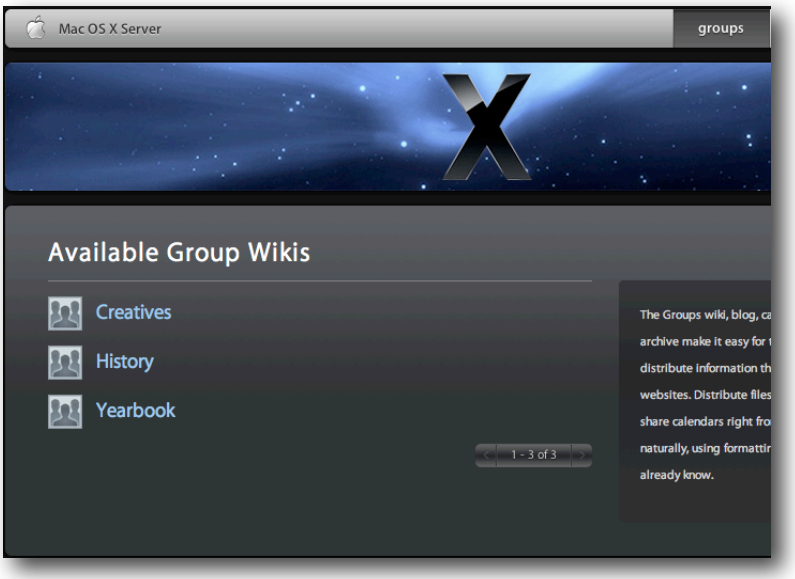


Enabling collaboration services for a group

When you have added the settings for your groups, go back to Server Admin and stop/restart the Web services.

Testing It

With everything up and running now, open Safari (or your favorite web browser), and enter the URL to your new collaboration server as follows: http://<webserver.domain.name>/Groups. You'll see something like this:



The group page of your new collaboration server

From this point on, you can click, select, edit, update, and enhance as much as you want. You'll find the links to the blogs and calendars inside each group's wiki page. Have fun, and for more information, check out the documentation on collaboration services.

Additional Tips and Tricks for Management

Here are a few tips and tricks that have been gathered from the collective knowledge and experience of Apple Education field engineers and from our customers.

Home Directory Templates

There are two locations for the templates used to create user's home directories. One is located on the client system in /System/Library/User Template; the other is on the home directory server in the same relative location. Root access is required to edit the template manually, or you can use the command line. If you open the template for editing, you'll see two key folders, "English.lproj" and "Non_localized." You can add items, such as a pre-populated iTunes library or an iPhoto set, to the template so that new users get these items when their accounts are created.

Local users get their home directory from the local template. Network users and mobile accounts, by default, get their home directory from the home directory server's template. You can use the Details section in Workgroup Manager to change the location where the mobile account pulls their home from to the local drive. This may make setting up users in large deployments much faster by populating the local home from the local system versus pulling all the information across the network at first sync.

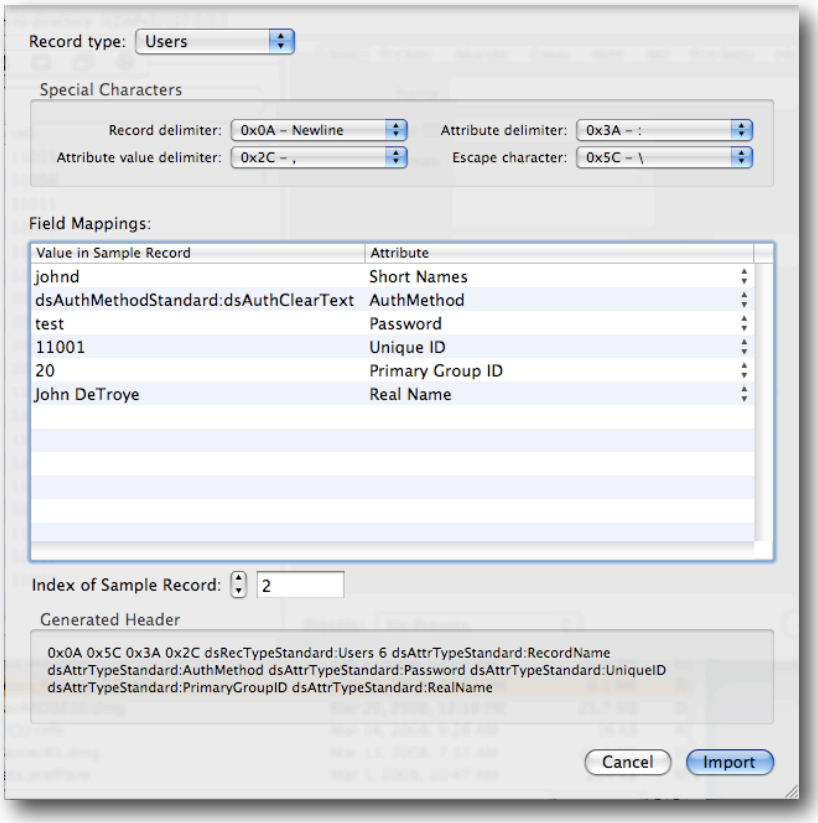
Importing Users

You may not know that Workgroup Manager supports text file imports of user records, and that it has since Tiger came out. Just create a text file with the record values you need, save it with UNIX line endings (this is easiest—you can use Mac line endings but need to know the hex code), then select Import in Workgroup Manager. You can use Passenger from MacinMind Software to format the import file first, but the raw method can work too. You need to add a special field to be able to import passwords.

This mechanism supports importing computer records, computer groups, groups, users—the whole ball of wax.

Here's an example set of records and the import window:

```
johnd:dsAuthMethodStandard\:dsAuthClearText:test:
11001:20:John DeTroye
billmcg:dsAuthMethodStandard\:dsAuthClearText:test:
11002:20:Bill McGlasson
gcook:dsAuthMethodStandard\:dsAuthClearText:test:
11003:20:George Cook
abriegel:dsAuthMethodStandard\:dsAuthClearText:test:
11004:20:Armin Briegel
dlopata:dsAuthMethodStandard\:dsAuthClearText:test:
11005:20:Dave Lopata
tweyer:dsAuthMethodStandard\:dsAuthClearText:test:
11006:20:Tom Weyer
wyeun:dsAuthMethodStandard\:dsAuthClearText:test:
11007:20:Warner Yuen
```

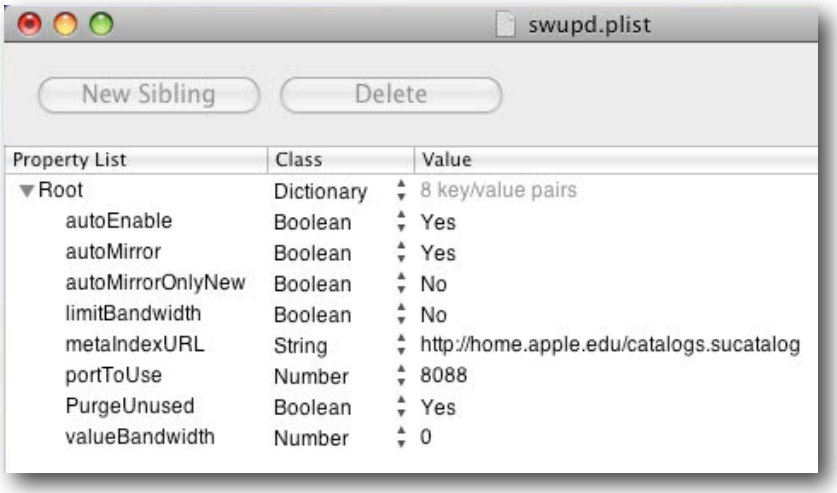


Workgroup Manager import function

On the topic of imports, don't forget to use Presets with Workgroup Manager. You can create a single account and then fill in all the common items that users' peers will need. Save that account as a preset, then select that preset when importing the rest of the accounts. This is a fast way to activate collaboration services, Mail, and other key settings for a lot of users at once.

Software Update Server (Cascading Too)

Setting up an internal Software Update server (SUS) on your private school network may save you many hassles. What is even better is setting one up at the district office, then setting the other Software Update servers in the schools to get their information from the master server at the district. This can be done by setting each lower server to cascade from the upper one. To set this up, log into the lower level (downstream) server as admin and edit /etc/swupd/swupd.plist. Set the metaIndexURL to point to your upstream server. The URL needs to end with "<domain>/catalogs.sucatalog" instead of the "index.sucatalog" used in MCX for the clients. You can then point the clients to the closest SuS, but know that all servers will get the same information.



Setting a downstream SuS to talk upstream

That's it for the time being. There are probably dozens of things that haven't been covered. Please make sure you review the product documentation and contact Apple Support for problems.