

Technical Note TN1151

Creating NetBoot Server-Friendly Applications

CONTENTS

[Introduction](#)[Discussion](#)[Summary](#)[References](#)[Change History](#)[Downloadables](#)

This Technote describes some guidelines that application developers should follow to ensure that their applications are able to work correctly in a NetBoot environment.

This Note is directed at all application developers.

Updated: [Jul 1 2000]

Introduction

In January 1999, Apple introduced Mac OS X Server and its NetBoot services which enable Power Macintosh and PowerBook computers to log in and boot directly from a server rather than a local hard drive. Since all NetBoot computers boot from a single system, NetBoot computers require no local configuration. Thus, setting up a network of Mac systems is fast and simple, and updating the server once automatically updates all Macs on the network. At the same time, NetBoot protects the System Folder and applications from corruption and includes management tools for setting system policies and controlling user access. NetBoot offers another important advantage: users can securely access their own applications, documents, and personal desktop preferences from any Mac on the network.

Because all software (including the operating system and the applications) in a NetBoot environment will reside and be launched from a server, applications must pay closer attention to how they launch and work with the System Folder. Software developers are urged to adapt their applications as necessary to the following guidelines in order for applications to run and behave correctly in this environment.

IMPORTANT:

By making your application NetBoot Server-friendly, you also increase the chances of it being compatible with Multiple Users on Mac OS 9 and higher, and vice versa. See Technote 1186 [How To Be a Good Multiple Users Citizen](#) for details.

[Back to top](#)

Discussion

The key to allowing your application to be "NetBoot Server-friendly" is to keep in mind that NetBoot clients will share System and Application folders loaded from the server. If you've written your application to properly use the Preferences folder and not assume that the application is launched from a writeable volume, your application will probably work fine.

Allow launching from a locked volume

To use disk space effectively, applications may be stored on a shared locked volume. Your application should be able to launch and run from a locked volume and store the user's personal information, such as preferences and data, on a different volume. At the minimum, check for and deal gracefully with the situation when the application file is locked.

The easiest way to test whether your application can be run from a locked volume is to copy your application to a file server (Personal File Sharing is perfectly adequate for this test), set the permissions of the application's folder to read-only, and then attempt to launch the application.

Allow multiple users to access the application simultaneously

In order to conserve disk space on the server, applications may be stored on a common/shared volume on the server.

You can check whether your application can be simultaneously launched by multiple users by installing it on a standard AppleShare server and launching your application across the network from two client computers.

To allow multiple users to launch your application simultaneously, you must set the `kIsShared` bit in the `fdFlags` of your application's Finder information (the `FInfo` record returned by `PBGetCatInfo`). If you're building with MPW, you can use the `SetFile` command to set this bit. If you're building with CodeWarrior, you must set this bit manually (you can do it with ResEdit) or in some other way that is appropriate for your build process.

Store all application preferences in the Preferences folder

Because a single "station" will be used by different people, each user's preferences will be retained and restored when she logs back in, either at the same machine or at a new one, via the Macintosh Manager software. In order for your application's user preferences to be retained, be sure to save them in the Preferences folder, as returned by `FindFolder` in response to the `kPreferencesFolderType` selector. Saving preferences here makes it much easier to administer the server, and allows the preferences to follow the user regardless of which Macintosh on the network she uses.

A good way to test whether your application handles this correctly is to run it in the Multiple Users environment supported by Mac OS 9 and above. If you create a "Limited" user and then log in as that user, you will find that any attempts to write preferences outside of the Preferences folder (as returned by `FindFolder` when called with the `kPreferencesFolderType` selector) will yield an error.

Allow users to specify the location of personal data

Since multiple users will be using the same application, it is important for applications allow the user to specify the location of their personalized documents and databases. Don't hardcode your application to store the user data in the same location as the application. This is particularly important for E-mail applications.

A good default location for user data files is the Documents folder, as returned by `FindFolder` in response to the `kDocumentsFolderType` selector.

Avoid storing the serial number in the application

Again, because multiple users may be using the same application, don't store the serial number in the application. If serialization of your application is necessary or desired, store the serial number within the Preferences folder so that different users can have "different" serial numbers while still using the same master application file.

Don't assume that `FindFolder(kSystemDisk, etc.)` will return -1 for the `refNum`.

Always use both the `vRefNum` and `dirID` returned by `FindFolder` to locate important folders.

Put application-specific files for the System Folder in the "Application Support" folder when you can.

If your application has files that get stored in the System Folder, like dictionaries, and they don't need to be stored in the "Extensions" folder, they should be placed in the "Application Support" folder.

Shared Libraries don't need to be in the Extensions folder; they can be in the folder with the application itself.

It's not necessary to copy a shared library used exclusively by one application into the Extensions folder; just leave it in the folder with the application.

[Back to top](#)

Summary

Enabling your application to function smoothly in a NetBoot Server environment is not that difficult. In fact, most applications today already do things properly and are "NetBoot Server-friendly." The things to keep in mind are to ensure that you can launch your application from a locked volume and have it run on multiple clients at the same time, and be aware of how your application stores files, including those that go into the System Folder, using the "Preferences" and "Application Support" folders when possible.

[Back to top](#)

References

Technote 1134 [The Preferences Problem](#)

Technote 1186 [How To Be a Good Multiple Users Citizen](#)

Networking and Mac OS X Server

[Back to top](#)

Change History

01-January-1999 Originally published.

01-July-2000 Updated in May 2000 to clarify various points and include references to Technote 1186 [How To Be a Good Multiple Users Citizen](#).

[Back to top](#)

Downloadables



Acrobat version of this Note (size in bytes, size in K)

[Download](#)

[Back to top](#)