



Tech Info Library

DOS Compatibility Card: Using Timbuktu In Windows (2/95)

Article Created: 23 February 1995

* RESTRICTED: Apple Internal and Support Providers Only *
Not For General Public Release

TOPIC -----

When using Timbuktu on the Power Macintosh 6100/66 DOS Compatible, with both the Timbuktu managed and Timbuktu managing CPU running under the Mac OS, the Timbuktu display window performs as expected.

However, when I switch to MS-DOS/Windows, the Timbuktu managing CPU cannot see the DOS/Windows screen. The Timbuktu display window still shows the Mac OS.

Is this because of the dual monitor design on the 6100? Is there a hardware or software fix for this problem?

DISCUSSION -----

This is exactly what should happen. Think of Timbuktu as a client-server application pair, where the client is the active, connecting partner and the server machine is the machine presenting its display to the client. In your terminology the client is the managing machine and the server is the managed machine.

Timbuktu clients communicate with a partner copy running in a particular OS environment. There is no direct link or attachment to any hardware screen buffers. The client receives a screen image of the environment that the server partner is running in. In the DOS Compatible environment, when you connect to a Timbuktu partner running under the Mac OS, your managing client will always observe or control the partner copy running under the Mac OS, even when the physical screen is showing the Windows environment.

With one monitor attached, there are two logical screens, one for Mac OS and one for Windows being interchangeably displayed on one physical device. The hardware connections on the DOS Compatible Card permits two displays to be connected simultaneously, one showing the Windows environment and one showing the Mac OS. There is no shared video circuitry, both environments can be seen to updating simultaneously, because each environment has a separate microprocessor operating. The Command-Return hot key in this case, only switches keyboard and mouse input between both visible environments.

Solution

=====

Remember, that there are two physical machines here, a 486 and a PowerPC, each with a logical and maybe even a physical display for each. If the Windows side were running Timbuktu for Windows, then there would be a Timbuktu server process available for connections. From the original Timbuktu client machine, you could open a new client connection and connect to the Timbuktu Windows Server. There must be a Timbuktu Server in each environment to serve a display image to Timbuktu clients.

To be able to observe or control each separate environment with Timbuktu, you need a server copy of Timbuktu running in both the Mac OS and the Windows environment. You need two client connections, one to each server environment. These two client connections can be made from the same machine or two different machines. Even further, these client machines could be Macintosh computers running Timbuktu Pro or Windows computers running Timbuktu for an AppleTalk connection.

Network Limitations

=====

Based on the design limitations of multiple protocols over one physical Ethernet media, use of AppleTalk and TCP connectivity will have to be split between the two environments. Since Timbuktu for Windows only supports AppleTalk, the Macintosh side will have to be configured to use TCP and Timbuktu Pro. So turn AppleTalk off for the Macintosh side and on for the Windows side, install the AppleTalk protocol stack that comes with Timbuktu for Windows. Install and configure MacTCP on the Macintosh side.

Use Timbuktu Pro for this solution, since it supports both AppleTalk and TCP based connections. Once the network environment described in the previous paragraph is set up, along with a Timbuktu server running on each microprocessor environment, you are ready to establish connections. From a Timbuktu client, any Macintosh or Windows version, establish an AppleTalk-based connection to the Windows Timbuktu Server and with Timbuktu Pro establish a TCP-based connection to the Mac OS Timbuktu Server. Two Timbuktu clients windows are required to observe/control the two environments.

A disadvantage of this setup is that you've disabled AppleTalk usage on the Macintosh side. This isolates the Mac OS from using AppleTalk connections to AppleShare File Servers and Laser Printers. If Timbuktu for Windows supported TCP protocols, then this problem could be alleviated by configuring the Macintosh side to use AppleTalk for the Timbuktu connection and TCP on the Windows side for the Timbuktu connection.

There is another hypothetical solution. There are network protocol limitations because of the single Ethernet interface, shared across both the Mac OS and Windows environments. If there was a second Ethernet interface, each environment could have their own hardware interface to the network. There are several 3rd party SCSI Ethernet solutions available for the Macintosh. These devices do complicate the software situation by introducing additional network drivers for SCSI attached devices. There is a possibility that a SCSI Ethernet connection

and the onboard Ethernet could each be allocated between the two environments.

NOTE:

Apple has not tried any of these combinations. Farallon also said that they had not tested Timbuktu running under Windows in the Power Macintosh DOS Compatible environment. There may be unforeseen problems getting drivers configured to cooperate in this complex environment. With an Apple-supplied AppleTalk protocol stack for Windows coming, Farallon will probably be revising their product to exist in this environment. I would expect these problem to diminish as the products evolve, but it may take revisions of the Farallon product before Timbuktu runs easily in both environments.

Support Information Services
Copyright 1995, Apple Computer, Inc.

Keywords: kwindows

=====

This information is from the Apple Technical Information Library.

19960215 11:05:19.00

Tech Info Library Article Number: 17238