



Windows NT/2000, Citrix MetaFrame, Time Machine and Time Zone Processing

<u>1. Introduction</u>	1
<u>2. Problem Details</u>	1
<u>3. The Time Machine Solution</u>	2
<u>4. Summary</u>	2

1. Introduction

As the popularity of Microsoft’s Windows NT and 2000 operating systems with Terminal Services continues to grow, companies are implementing larger servers with hundreds of users in multiple locations. This relatively new phenomenon generally uses the popular Citrix MetaFrame software for load balancing and other enhanced services. Together, this allows users in any location worldwide to access an application environment. They also provide the ability for administrators to manage the environment remotely.¹

Unfortunately, there is one significant local time zone drawback to this implementation, which has created problems and headaches for users. Briefly, when a user executes an application, the application retrieves the current date and time from the Microsoft Terminal Server. The result is that users on the East Coast will see the same date and time as users on the West Coast, often causing considerable errors.

To address these concerns, Solution-Soft has developed Time Machine to provide accurate time zone processing. Time Machine is a date and time simulation tool that allows the system administrator to define the correct time zone for specific users. This allows the user to see their own date and time in all their applications where needed.

2. Problem Details

Users accessing applications running on Terminal Server have often wanted to “see” the time for their current locale. However, Microsoft NT/2000 users can not have a time different then that of the operating system. This has forced enterprises spanning multiple time zones to install and maintain additional machines solely to provide proper times to users. What happens if the correct date or time is not available to the end user?

For many users, not having the correct time is initially only an inconvenience; they “learn” to make a mental adjustment from the server time when they receive email, insert dates in their documents, or read their workgroup schedule. However, just one mistake in reading a memo can result in arriving late to a strategic meeting, or missing a call. This problem is exacerbated when automated software or interaction with others is added. For instance an automated request to a distribution warehouse would have the wrong deliver-by time.

¹ The purpose of this white paper is not to describe how Windows NT/2000 and Citrix MetaFrame work for these environments. For more information, please contact our partners Microsoft and Citrix. For information on MetaFrame see http://events.citrix.com/mf2000/docs/MF_DS2.pdf or go to <http://www.microsoft.com/ntserver/terminalserver/exec/casestudy/circuitcity.asp> to review a case study at Circuit City.

A new and growing arena where this issue is much more than just an inconvenience is with Application Service Providers (ASP). Often, users leasing time on the host machines are not aware that the date and time they will use is based on the server until they have begun using the service. This is amplified by the fact that the ASP's customers may connect from anywhere. An ASP with the ability to deliver "local" time has an advantage over others in that they can provide one machine to service customers in any area of the world. This removes not only the possibility of errors caused by users who don't understand the time difference, but also removes many complaints received by the ASP.

3. *The Time Machine Solution*

There are a number of different methods for implementing Time Machine. However, through our work with Citrix and Microsoft, the best method determined is to allow Time Machine to be enabled at the user-level instead of at the administrator-level. The system administrator can often facilitate this by creating batch files that are placed in the users startup folders. By implementing the following batch files, when a user logs on to the Terminal Server, they will see their own date and time, regardless of the system's date and time.

The first batch file automatically runs when the user logs into the server. While the details of Time Machine can be reviewed through the online help or manual, one can see how easy Time Machine is to implement with this batch file. The command line `tmuser -a -h 3` is all that is necessary. The `-a` option "adds" the clock for the user, and the `-h #` option defines how many hours the clock adjustment is. In the following example, the user's clock is moved ahead three hours. Using `-3` moves the user's clock backwards three hours.

```
REM Time Machine startup file
REM This file adds a time zone clock
REM three hours ahead of the system clock for the user
tmuser -a -h 3
REM endfile
```

As the user completes their session and logs off the Terminal Server, their virtual clock can be removed from the Terminal Server. Again, a simple process – the log off script – will remove the clock. As might be expected, the `-d` option deletes the user's time zone clock from the server.

```
REM Time Machine log-off file
REM This file removes the time zone clock when the user logs off
tmuser -d
REM endfile
```

Logoff scripts are a function of Windows 2000. Testing of Time Machine on NT4.0 has shown that to maintain virtual clocks for 20,000 users requires approximately 10megs of memory. Thus, removing users virtual clocks on log off should only be necessary on machines that are running at near peak capacity of system resources.

4. *Summary*

Now when users need to run their applications they can merely set their time at login as an offset from the server's time. By doing this users can utilize the correct date and time in their spreadsheets, word documents or database client programs. Powerful servers can be configured to run multiple instances of databases, with each database running under a different time zone to serve users in various geographic locations. Lastly, no interoffice communication need ever be time-translated.