

Tech Note 144

How the Time Zone & Daylight Savings Time Affect InTouch 7.0 Historical Logging

All Tech Notes and KBCD documents and software are provided "as is" without warranty of any kind. See the [Terms of Use](#) for more information.

Topic#: 000794

Last Modified: November 27, 2000

This *Tech Note* shows how Wonderware® InTouch™ accounts for differences in the time zone and Daylight Savings Time (DST). In addition, you will learn how to adjust your computer for DST while maintaining data integrity.

The InTouch 7.0 Historical Logging creates a new Historical Log file each day at midnight based on the computer clock. In addition, the Historical Logging uses *Greenwich Mean Time* (GMT) as a reference for recording and retrieving data. Greenwich Mean Time is also known as *Universal Coordinated Time* (UCT).

The Historical engine checks the values of the tags that have log data checked for changes greater than the Log Deadband and then records those values to the log file. Each record contains the updated value of the tag as well as the time and date the value was updated.

Historical Logging

To describe historical logging, let's assume we have a tag, called **tag1**, which changes its value at 10:00:00 am (computer time) and uses the following examples.

If we are logging west of GMT, the time stamp for the logged value is calculated as follows:

```
Logged_TimeStamp = Computer_Time+(Offset - DST)
```

Where Logged_TimeStamp is the time stamp of the record written to the Historical Log file.

Example #1

The Time Zone is set for Pacific Standard Time (GMT-8) and DST is enabled.

```
Logged_TimeStamp = 10:00:00+(8 - 1) = 17:00:00
```

Example #2

The Time Zone is set for Pacific Standard Time (GMT-8) and DST is not enabled.

```
Logged_TimeStamp = 10:00:00+(8-0) = 18:00:00
```

If we have the same tag and values, but are logging east of GMT, then the time stamp for the logged value is calculated as follows:

```
Logged_TimeStamp = Computer_Time+(-Offset+DST)
```

Example #3

The Time Zone is set for Amsterdam, Berlin, Bern, Rome (GMT+1) and DST is enabled.

```
Logged_TimeStamp = 10:00:00+(-1 + 1) = 10:00:00
```

Example #4

The Time Zone is set for Monrovia, Casablanca (GMT0) and DST is *not* enabled because Casablanca does not observe DST.

```
Logged_TimeStamp = 10:00:00+(-0+0) = 10:00:00
```

In this example, the time stamp in the Historical Log file is the same as the computer time.

Note that GMT0-Monrovia, Casablanca is the ideal time zone setting.

Historical Retrieval

The retrieval engine also accounts for differences in the time zone and DST settings on the computer on which the retrieval occurs. In most cases, if the logging and retrieval were performed on the same computer or on computers with identical time zone settings, then the retrieved records will be correct.

Note If the time zone settings are different between the Logging computer and the Retrieval computer, then the record logged at 10:00:00 on the Logging computer may show at a different time on the Retrieval computer. In older versions of InTouch (InTouch 5.6, InTouch 6.0, InTouch 7.0 SP2 Patch03 and InTouch 7.1 Patch02), if the DST is checked, for certain Time Zones the data will be retrieved with a 2-hour offset. The solution to this problem is to set the time zone to one that does not observe DST, such as GMT0-Monrovia, Casablanca. Starting with InTouch 7.0 SP2 Patch04 and InTouch 7.1 Patch03 any GMT setting can be used. Make sure that there is no TZ variable in the autoexec.bat and no TZ environment variable in **Control Panel/System/Environment**.

Keep in mind that DST can affect the data retrieval, even when the logging and retrieval are performed on the same computer or on computers with identical time zone settings. Choosing a time zone that observes DST or enabling DST causes the retrieved data to be two hours off and, at the beginning of the day, the Historical Trend chart will show two hours of flat line or no data. This behavior is present when the Time Zone setting is of the type GMT+x (east of GMT) and DST is enabled.

For example, if the time zone on your computer is set to Amsterdam, Berlin, Bern, Rome (GMT+1) and DST is enabled, then the retrieved data will show a flat line at the beginning of the day and the current data will be two hours in the future.

The solution is to set the time zone to one that does not observe DST, such as **GMT0-Monrovia, Casablanca**. This causes the time stamp in the Historical Log file to be the same as the computer clock. In addition, DST is *not* enabled because GMT0-Monrovia, Casablanca does not observe DST.

Setting the Time Zone So That DST is Not Enabled

The methods for setting the time zone without enabling DST vary depending on your operating system, which are explained below

Setting the Time Zone for Windows 95

From the Windows Start menu, select Settings, then Control Panel. Double-click **Date/Time** from the Control Panel window. Or, click on the time display on the task bar. Click the **Time Zone** tab.

If the selected time zone observes DST, then the **Automatic Adjust Clock for Daylight Savings** check box is accessible. The Historical engine will log and retrieve with DST regardless if the Automatic Adjust Clock for Daylight Saving Time is checked or not.

To log and retrieve without DST, select a time zone matching your time offset that does not observe DST, or select GMT0-Monrovia, Casablanca. For example, if you are in the time zone GMT+2-Athens, Istanbul (which observes DST), select GMT+2-Harare, Pretoria which has the same time zone but does not observe DST.

Setting the Time Zone for Windows NT

From the Windows Start menu, select Settings, then Control Panel. Double-click **Date/Time** from the Control Panel window. Or, click on the time display on the task bar. Go to the **Time Zone** tab. If the selected time zone observes DST, then the **Automatic Adjust Clock for Daylight Savings** checkbox is accessible. If the Automatic Adjust Clock for Daylight Savings check box is selected, the Historical engine will log and retrieve with DST.

To log and retrieve without DST, select any time zone matching your time offset that does not observe DST, deselect the Automatic Adjust Clock for Daylight Savings checkbox, or select GMT0-Monrovia, Casablanca. For example if you are in the time zone GMT+2-Athens, Istanbul (which observes DST), select GMT+2-Harare, Pretoria, which has the same time zone but does not observe DST.

Note If you make any changes to the time zone, before starting WindowViewer make sure that you move (or delete) the log files ***.lgh** and ***.idx** for the day that you made the changes.

Frequently Asked Questions

How can I adjust the computer clock for DST if the Automatically Adjust Clock for Daylight Savings Changes checkbox is not selected?

Manually set the computer clock so that it observes the correct time under DST.

Or, use the appropriate InTouch script below:

Setting the clock forward in the Spring (For Windows 95 and Windows NT 4.0):

Using the SendKeys() function, create the following Condition QuickScript:

```
$Year == yyyy and $Month == 04 and $Day == dd and $Hour == 02 and
DaylightSavingsTime == 0;
```

Where: *yyyy* is the year; *dd* is the date of the time change; and *DaylightSavingsTime* is a user-defined memory discrete tagname indicating DST with an initial value of zero.

```
ON TRUE:
DaylightSavingsTime = 1;
StartApp "c:\Winnt\System32\control.exe Date/Time"; {you may need to change the
path to reflect where control.exe resides on your computer}
Flag=1; {flag is a user defined memory integer tag with initial value 0}
```

Create a Condition script: Flag==1

```
WHILE TRUE every 1000 Msec:
SendKeys "%(t)";
SendKeys "03";
SendKeys "{TAB}";
SendKeys "{TAB}";
SendKeys "{TAB}";
SendKeys "{ENTER}";
Flag=0;
```

Depending on the speed of your computer, the delay introduced by the second While True Condition script can be increased or decreased. This delay allows Control.exe to start before sending the Keys.

Setting the clock forward in the Spring (For Windows 95 only):

Create the following Condition script to change the time from the command line:

```
$Year == yyyy and $Month == 04 and $Day == dd and $Hour == 02 and
DaylightSavingsTime == 0 ;
ON TRUE:
DaylightSavingsTime = 1;
StartApp ("C:\COMMAND.COM /C TIME 03:00 ");
```

Setting the clock forward in the Spring (For Windows NT 4.0 only):

Using the WWDosCommand() function from the KBCD, create the following Condition script:

```
$Year == yyyy and $Month == 04 and $Day == dd and $Hour == 02 and
DaylightSavingsTime == 0 ;
ON TRUE:
DaylightSavingsTime = 1;
WWDosCommand ("time 03:00:00", "Invisible");
```

Setting the clock back in the Fall (For Windows 95 and Windows NT 4.0):

Using the SendKeys() function, create the following Condition script:

```
$Year == yyyy and $Month == 10 and $Day == dd and $Hour == 02 and
```

```
DaylightSavingsTime == 0;
ON TRUE:
DaylightSavingsTime = 1;
StartApp "c:\Winnt\System32\control.exe Date/Time"; {you may need to change the
path to reflect where the control.exe resides on your computer}
Flag=1; {flag is a user defined memory integer tag with initial value 0}
```

Create a Condition script: Flag==1

```
WHILE TRUE every 1000 Msec:
SendKeys "%(t)";
SendKeys "01";
SendKeys "{TAB}";
SendKeys "{TAB}";
SendKeys "{TAB}";
SendKeys "{ENTER}";
Flag=0;
```

Depending on your computer speed, the delay introduced by the second While True condition script can be increased or decreased. This delay allows Control.exe to start before sending the Keys.

Setting the clock back in the Fall (For Windows 95 only):

Create the following Condition script to change the time from the command line:

```
$Year == yyyy and $Month == 10 and $Day == dd and $Hour == 02 and
DaylightSavingsTime == 0 ;
ON TRUE:
DaylightSavingsTime = 1;
StartApp("C:\COMMAND.COM /C TIME 01:00 ");
```

Setting the clock back in the Fall (For Windows NT 4.0 only):

Using the WWDosCommand() function from the KBCD, create the following Condition QuickScript:

```
$Year == yyyy and $Month == 10 and $Day == dd and $Hour == 02 and
DaylightSavingsTime == 0;
ON TRUE:
DaylightSavingsTime = 1;
WWDosCommand ("time 01:00:00", "Invisible");
```

Note Whenever a computer clock is set back, one hour of data will be lost. To prevent this loss, stop the Historical Logging, move the existing log files to a different directory, then restart the historical logging.

To prevent data loss when you set your computer clock back one hour, use the following example:

Create the following Condition script:

```
$Year == yyyy and $Month == 10 and $Day == dd and $Hour == 02 and
DaylightSavingsTime == 0 ;
```

```
ON TRUE:
DaylightSavingsTime = 1;
$HistoricalLogging = 0; {this will stop historical logging}
StartApp "c:\Winnt\System32\control.exe Date/Time"; {you may need to change the
path to reflect where the control.exe resides on your system}
Flag = 1; {flag is a user defined memory integer tag with initial value 0}
```

Create a Condition script: Flag == 1

```
WHILE TRUE every 1000 Msec:
SendKeys "%(t)";
SendKeys "01";
SendKeys "{TAB}";
SendKeys "{TAB}";
SendKeys "{TAB}";
SendKeys "{ENTER}";
FileMove("C:\history\9910dd00.lgh", "C:\backup","monitor");
Flag = 2;
```

Create a Condition script: Flag == 2

```
WHILE TRUE every 5000 Msec:
FileMove ("C:\history\9910dd00.idx", "C:\backup","monitor");
Flag=3;
```

Create a Condition script: Flag == 3

```
WHILE TRUE every 5000 Msec:
$Historicallogging=1;
Flag=0;
```

In the above example, **C:\history** is the directory where the log files reside and **C:\ backup** is the directory where the log files will be moved. Files **9910dd00.lgh** and **9910dd00.idx** are the ones to move. Simply replace the *dd* in the log file name with the correct day.

The delay of 5000Msec in the Condition scripts on Flag==2 and Flag==3 can be increased or decreased, depending on the size of the log files and on the speed of your computer. Allow sufficient time for the files to be moved to another location before restarting Historical logging.

D. Tulbure

The *Tech Note* is published occasionally by Wonderware Technical Support. Publisher: Invensys Systems, Inc., 26561 Rancho Parkway South, Lake Forest, CA 92630. There is also technical information on our software products at [Wonderware Technical Support](#)

For technical support questions, send an e-mail to support@wonderware.com.



[back to top](#)

©2010 Invensys Systems, Inc. All rights reserved. No part of the material protected by this copyright may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, broadcasting, or by any information storage and retrieval system, without permission in writing from Invensys Systems, Inc.