

**TINE TIME** 11.03.2011, S. Herb

I. Vulnerability of modern society to  
GPS malfunction ([www.newscientist.com](http://www.newscientist.com))

**San Diego Calif.** , summer 2007, a sunny afternoon :

- Problems in airport and harbor traffic control systems
- Cell phones stop working
- Bank ATMs stop serving up cash

**Reason:** a US Navy ship was generating radio interference as part of a (Navy) communications test, killed GPS reception

- Stationary GPS receivers are being widely used as precision time sources for coordinating distributed systems
- (for \$30 you can buy your own *small* GPS jammer box)

# TINE TIME

## II. What is the situation with Control System Machines ?

Network time source ([ntp.desy.de](http://ntp.desy.de), DCF77) is used to correct local clock errors

**Unix/Linux** machines use **NTP** (network time protocol)

- slow, adiabatic loop control by tweaking clock rate (but < 500 ppm)
- deviations at millisecond Level (once system has stabilized)

**Windows** machines use **WTS** (Windows time service)

- communicate with domain controller via NTP, but use simpler algorithms
- deviations mostly < 1sec but occasionally 5, 10, 20, ... seconds
- Microsoft says that WTS is **not** appropriate for 1-2 sec accuracy  
*and should not be used in time-critical environments such as Stock Exchanges*
- Non-adiabatic correction procedures ( observe dt/t up to 5%)

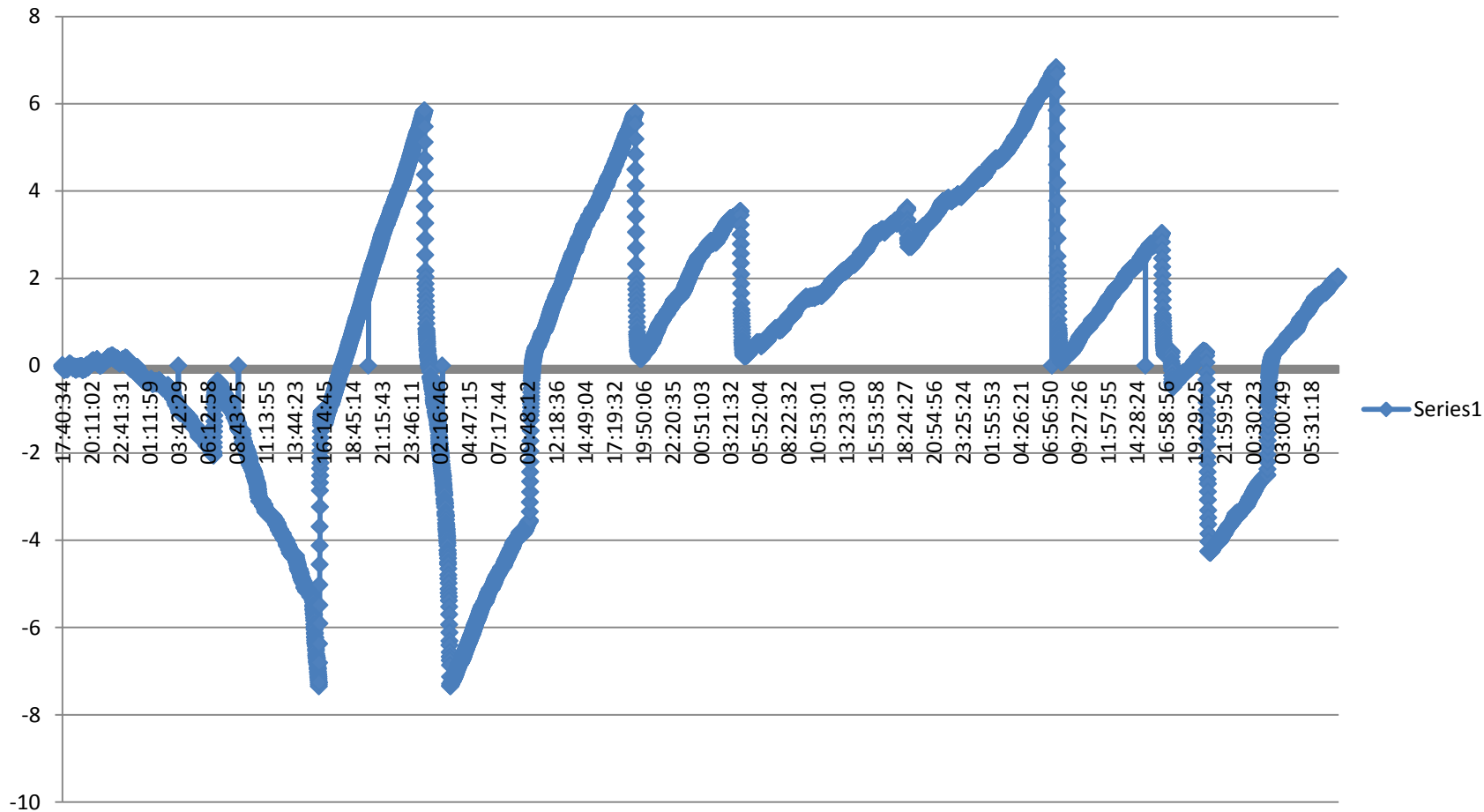
+ (Windows **and** Linux) occasional much larger errors ( e.g. Jan 1 1970 )

# AccXpPeR3D (Win XP PC) [ worse than average but not 'the worst' ]

**seconds** offset from ntp.desy.de over **~110 hour** period

drifts typically  $\frac{1}{2}$  sec/hour, fast corrections use  $dt/t \sim 1\%$

( data from command line `"w32tm /stripchart /dataonly /period:30 /computer:ntp.desy.de"` )



# TINE TIME

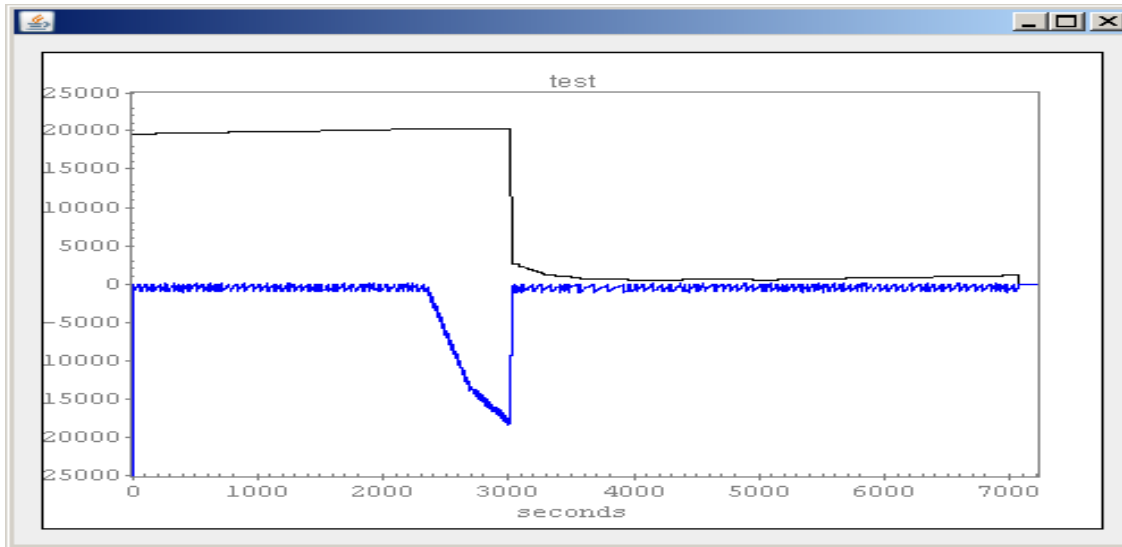
## TINE Timestamps are corrected for these effects

- TINE multicasts a 'global time' at 1 Hz ( derived from ntp.desy.de)
- This time is used by each server to calculate the difference between local system time and 'global' time.
- If the difference changes by more than 100 msec and is consistent for 6 consecutive packets, a new **OFFSET** is defined
- **TINE Timestamp** = local system time + **OFFSET**
- This avoids yet more (problematic) messing with local clock settings!

## But there are still problems, for example:

- For fast correction e.g.  $dt/t \geq 1\%$ , the consistency test fails, so the **OFFSET** is not updated, and the TINE timestamp drifts away from global time
- at correction end, the timestamp jumps back, perhaps by 5 (or 20) seconds
- clients reject packets with out-of-order timestamps

# TINE TIME



**AccXpCsTest02 :**

'fast correction'

**Black Line =**  
TINE OFFSET

**Blue Line =** deviation  
of TINE Timestamp  
from 'Global Time'

Here the Timestamp suddenly jumps forward. For the other case, that the Timestamp jumps 20 seconds into the past, the client would ignore packets for the next 20 seconds.

This appears to be happening with some of the Embedded XP PCs used for Undulator control at HASYLAB, for which the jumps are often as much as 70 seconds, resulting in problems for the clients

# TINE TIME

## PET3ID10.30 ( Embedded Win XP, from TINE Log File)

TIME	OFFSET	Largest Correction
08.03.11 <b>21:31</b> :14.143	204.612 sec	0.160539 sec
08.03.11 <b>22:31</b> :31.067	193.078 sec	0.177092 sec
08.03.11 <b>23:31</b> :38.971	181.577 sec	0.215824 sec
09.03.11 <b>00:32</b> :02.976	170.025 sec	0.177908 sec
09.03.11 <b>01:32</b> :21.381	158.49 sec	0.170748 sec
09.03.11 <b>02:32</b> :51.570	216.913 sec	<b>69.8853 sec</b>
09.03.11 <b>03:33</b> :00.949	205.405 sec	0.165559 sec
09.03.11 <b>04:33</b> :16.419	193.879 sec	0.173892 sec
09.03.11 <b>05:33</b> :45.673	182.308 sec	0.204539 sec
09.03.11 <b>06:33</b> :46.391	170.832 sec	0.183701 sec
09.03.11 <b>07:33</b> :49.139	159.343 sec	0.169457 sec
09.03.11 <b>08:33</b> :53.488	216.861 sec	<b>68.9449 sec</b>

# TINE TIME

## III. Possible Modifications, Improvements ?

- Ignore ? (it's working most of the time)
- Throw out a few sick PCs ? (or is it the software?)
- Reduce role of time stamps in TINE communications ?
- Modify Servers to track 5% time rate error (what % of problems is this?)
- Install 'real' NTP on Windows PCs ? (see below)
- Switch all the PCs to Linux ? (just a joke)

## (Provisional) Conclusions:

- The situation is not so bad (but not good for Undulator Embedded XP)
- 'real' NTP might not be an easy, 100% solution (situation is not clear)
- We **should not** relax our standards for the Timing and Timestamps
- We need to monitor the system to find the few really bad performers  
Check up on **YOUR** servers (for example with the FEC REMOTE program)

# Appendix: AccXpPer3D : 'real' NTP at work

