

Technische Grundlagen zur Schaltsekunde

Martin Burnicki

Email: martin.burnicki@meinberg.de

Meinberg Funkuhren GmbH & Co. KG

Bad Pyrmont, Germany

<http://www.meinberg.de>

Mai 2015

Bis 1972 wurde die Länge einer Sekunde durch die Erdrotation bestimmt

- Bei Änderung der Erdrotation wurde die Sekunde länger oder kürzer
- Schlecht für technische Lösungen

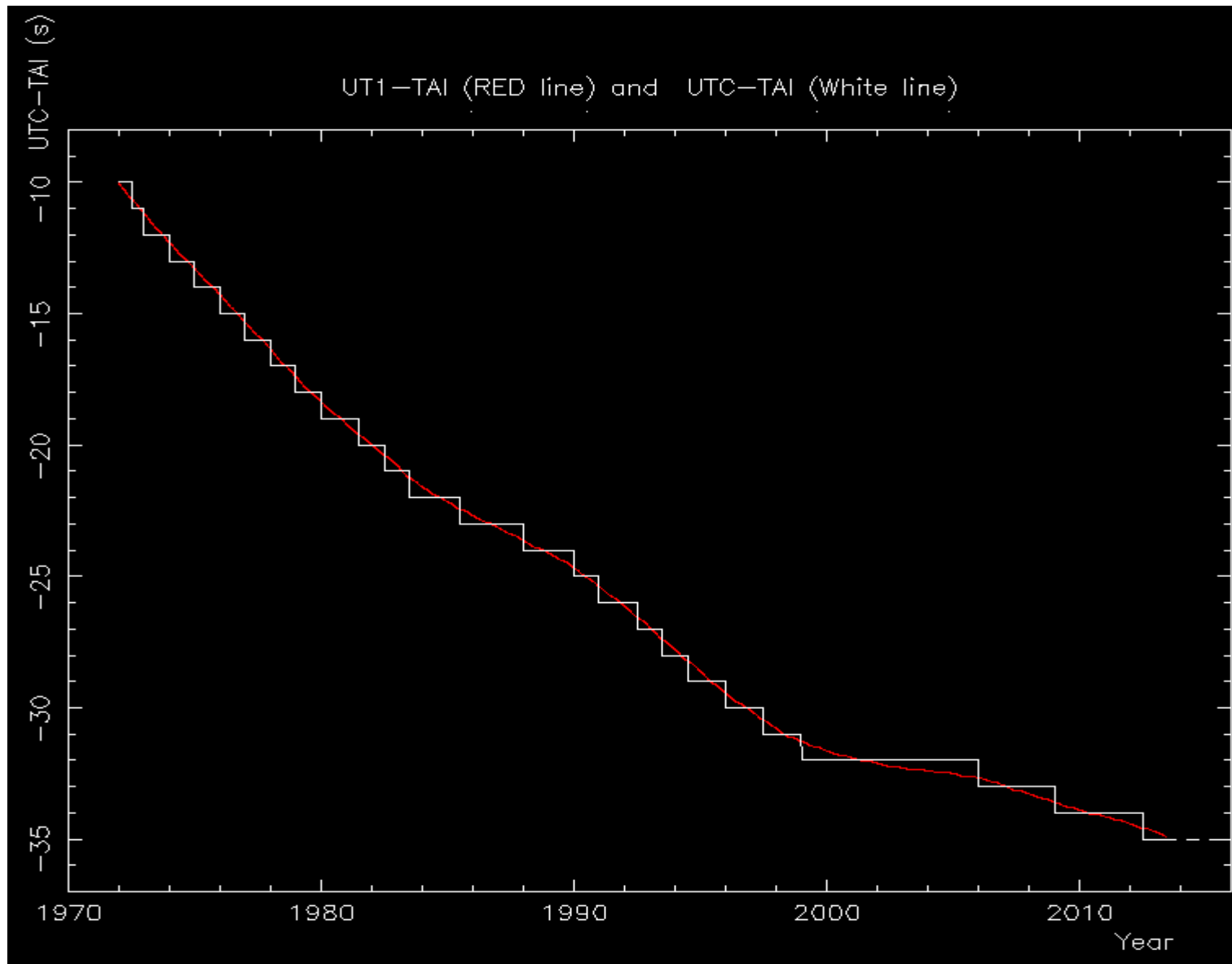
Seit 1972 wird die Länge der Sekunde durch Atomuhren bestimmt

- Grundlage für die koordinierte Weltzeit UTC
- Sekunden haben immer die gleiche Länge
- Aber die Erdrotation variiert immer noch

Schaltsekunden zum Ausgleich werden durch den International Earth Rotation Service (IERS) festgelegt, um die UTC-Zeit an die Erdrotation anzupassen

- IERS Bulletin C wird zweimal pro Jahr veröffentlicht:
Schaltsekunde ja oder nein
- Prinzipiell könnten Schaltsekunden eingefügt oder herausgenommen werden
- Bisher wurden immer nur Schaltsekunden eingefügt

Schaltsekunden seit 1972



Quelle: <http://hpiers.obspm.fr/eop-pc/earthor/utc/leapsecond.html>

- Von 1972 bis in die 1990er Jahre wurden Schaltsekunden im Abstand von 1 Jahr oder 1 1/2 Jahren eingefügt
- Zwischen Dezember 1998 und Dezember 2005 gab es 7 Jahre ohne Schaltsekunde (!)
- Seitdem Schaltsekunden im Abstand von ca. 3 Jahren

INTERNATIONAL EARTH ROTATION AND REFERENCE SYSTEMS SERVICE (IERS)

SERVICE INTERNATIONAL DE LA ROTATION TERRESTRE ET DES SYSTEMES DE REFERENCE

SERVICE DE LA ROTATION TERRESTRE DE L'IERS

OBSERVATOIRE DE PARIS

61, Av. de l'Observatoire 75014 PARIS (France)

Tel. : 33 (0) 1 40 51 22 26

FAX : 33 (0) 1 40 51 22 91

e-mail : services.iers@obspm.fr

<http://hpiers.obspm.fr/eop-pc>

Paris, 5 January 2015

Bulletin C 49

To authorities responsible for the measurement and distribution of time

UTC TIME STEP
on the 1st of July 2015

A positive leap second will be introduced at the end of June 2015.

The sequence of dates of the UTC second markers will be:

2015 June 30, 23h 59m 59s

2015 June 30, 23h 59m 60s

2015 July 1, 0h 0m 0s

The difference between UTC and the International Atomic Time TAI is:

from 2012 July 1, 0h UTC, to 2015 July 1 0h UTC : UTC-TAI = - 35s

from 2015 July 1, 0h UTC, until further notice : UTC-TAI = - 36s

Leap seconds can be introduced in UTC at the end of the months of December or June, depending on the evolution of UT1-TAI. Bulletin C is mailed every six months, either to announce a time step in UTC or to confirm that there will be no time step at the next possible date.

Daniel Gambis

Head

Earth Orientation Center of IERS

Observatoire de Paris, France

Wichtig: Unterscheidung zwischen der **Verbreitung** der Ankündigung *an* die Endgeräte, und der **Verarbeitung** der Schaltsekunde *durch* die Endgeräte

Unterschiedliche Möglichkeiten, eine Schaltsekunden-Ankündigung zu *verbreiten*

- Funksignale, Zeitcodes, serielle Zeitlegramme
- Netzwerkprotokolle, Dateien

Unterschiedliche Möglichkeiten, eine Schaltsekunde zu *verarbeiten*

- Zeitsprung zu Beginn oder zum Ende der Schaltsekunde
- Weiche Nachführung der Zeit über ein bestimmtes Intervall

Geräte müssen *früh genug* die Schaltsekunden-Ankündigung erhalten

- Weitergabe der Information durch eine ganze *Kette* von Geräten und Protokollen

UTC spezifiziert, wie die Zeitzählung bei Schaltsekunden geschieht

- Auslassen der letzten Sekunde „59“ bei herausgenommener Schaltsekunde
- Einfügen einer Sekunde „60“ bei Einfügung einer Schaltsekunde

Zählweise der UTC-Zeit beim Einfügen einer Schaltsekunde:

```
2015-06-30 23:59:57
2015-06-30 23:59:58
2015-06-30 23:59:59
2015-06-30 23:59:60 <-- eingefügte Schaltsekunde
2015-07-01 00:00:00
2015-07-01 00:00:01
2015-07-01 00:00:02
```

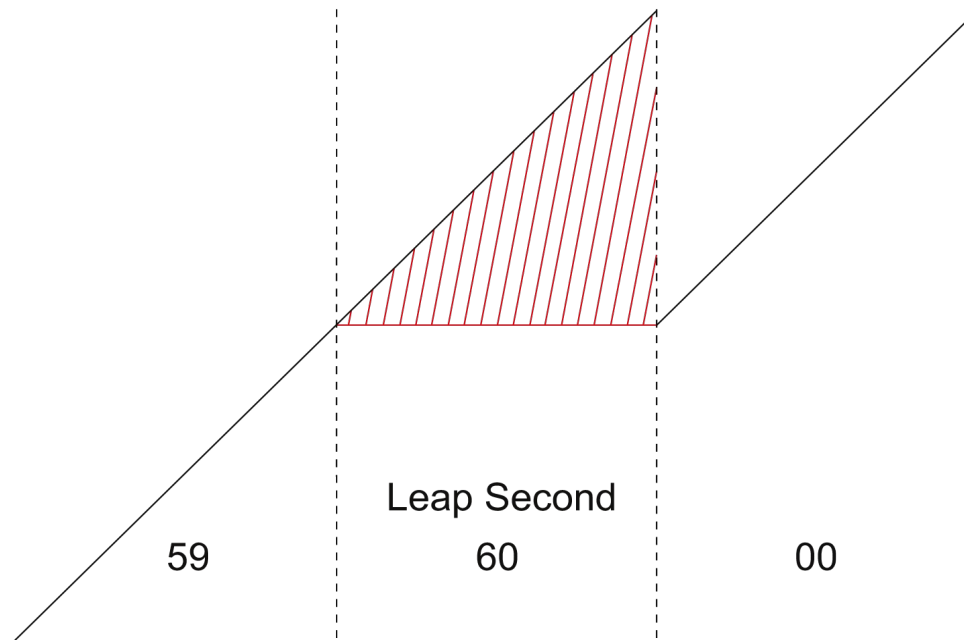
- Bei Normalisierung ist die Zeit zu Beginn und Ende der Schaltsekunde gleich
- Gleiche Zeitstempel der Systemzeit zu Beginn und Ende der Schaltsekunde (POSIX)
- Wie werden Zeiten mit Bruchteilen der Sekunde verarbeitet?

**Zeitstempel der Systemzeit kennt keine Sekunde 60,
sondern nur Sekunden seit einer Epoche**

Zeitstempel	Datum/Zeit (UTC)	Offset zu linearer Zeit [ms]	
D93DABFD.2A54001E	2015-06-30 23:59:57.165	-0.602	
D93DABFE.2AB85E4F	2015-06-30 23:59:58.166	-0.621	
D93DABFF.2B1FD406	2015-06-30 23:59:59.168	-0.646	<-- „59“, vor Schaltsekunde
D93DABFF.2B8DEDA5	2015-06-30 23:59:59.170	-1000.654	<-- „60“, Schaltsekunde
D93DAC00.2BF7A2CE	2015-07-01 00:00:00.171	-1000.665	
D93DAC01.2C610C6B	2015-07-01 00:00:01.173	-1000.692	
D93DAC02.2CC3E439	2015-07-01 00:00:02.174	-1000.718	
D93DAC03.2D281C20	2015-07-01 00:00:03.176	-1000.762	

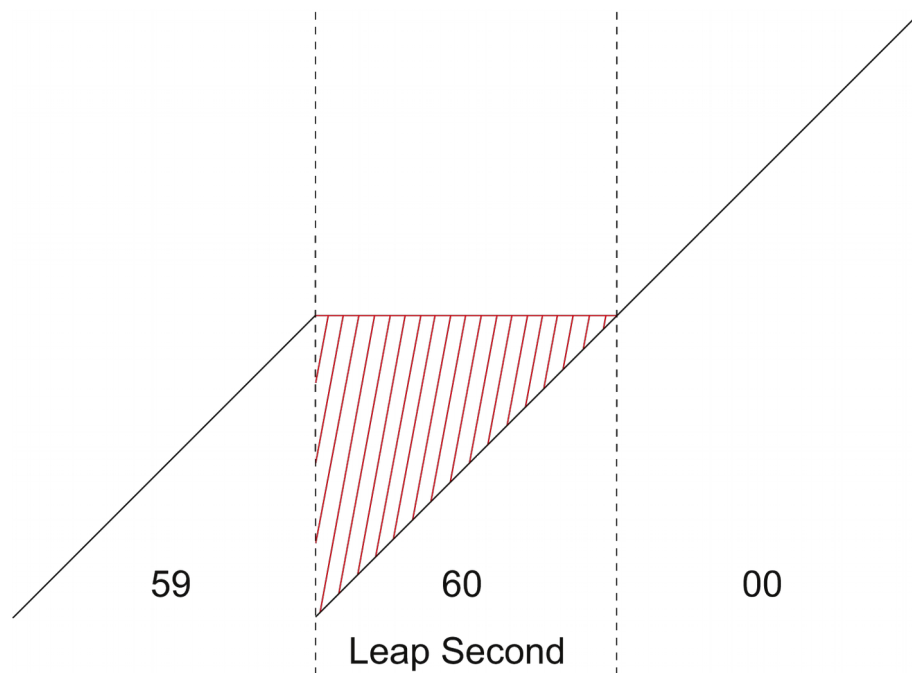
- ➔ Ohne Status auf Systemebene nicht erkennbar, ob Schaltsekunde oder nicht
- ➔ Konvertierung in Datum/Zeit liefert zweimal gleiches Ergebnis „59“

Zeitsprung um 1 s zurück *am Ende* der Schaltsekunde



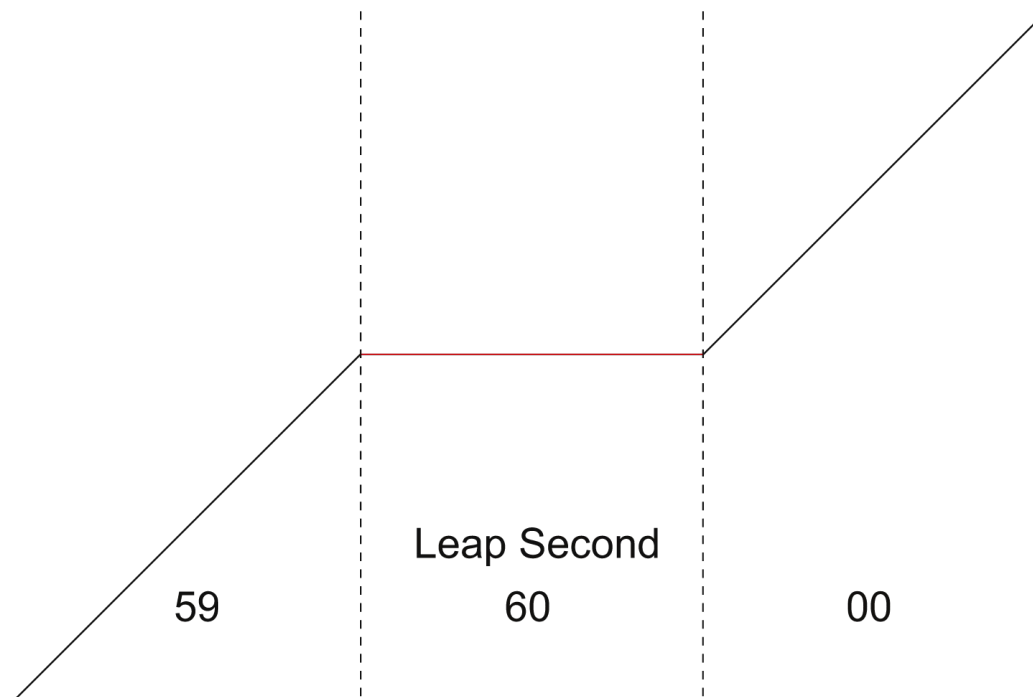
- Zeitverlauf ist nicht monoton
- Doppelte Zeitstempel *nach* der Schaltsekunde
- *Frühere* Zeitstempel für *spätere* Ereignisse
- Kein eindeutiger Beginn der neuen Minute / der Stunde / des UTC-Tages

Zeitsprung um 1 s zurück zu *Beginn* der Schaltsekunde



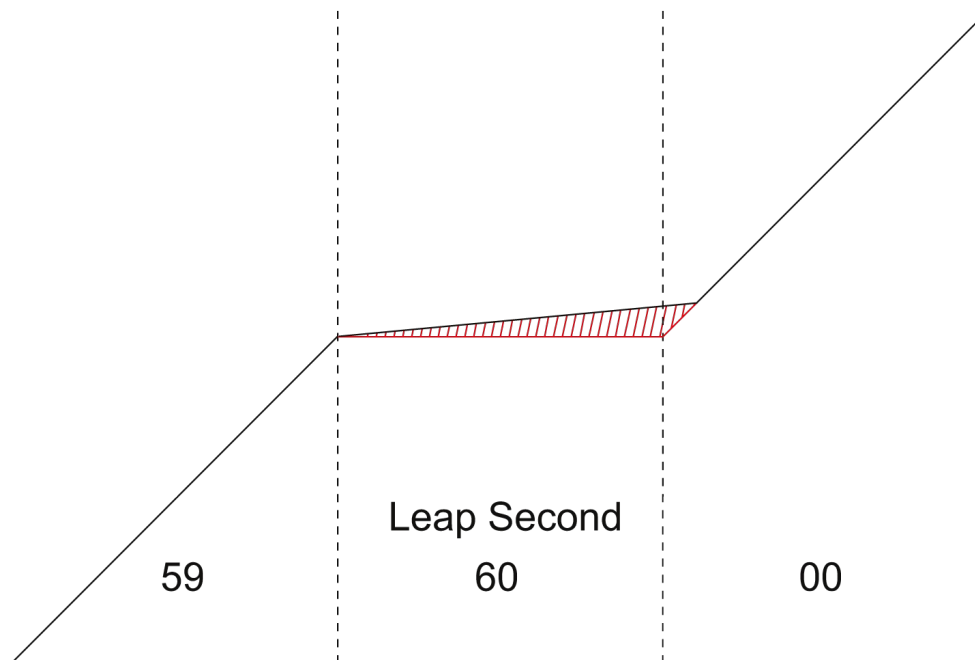
- Zeitverlauf ist nicht monoton
- Doppelte Zeitstempel *während* der Schaltsekunde
- *Frühere* Zeitstempel für *spätere* Ereignisse
- Eindeutiger und exakter Beginn der neuen Minute / der Stunde / des UTC-Tages

Zeitverlauf 1 s lang „anhalten“



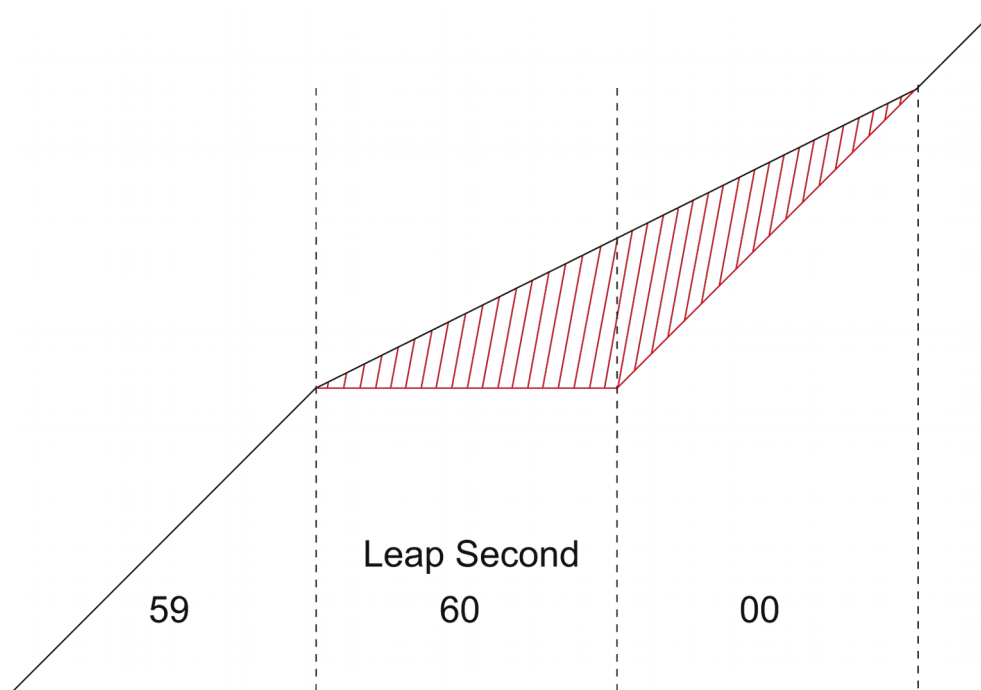
- Zeitverlauf ist nicht *streng* monoton
- Gleiche Zeitstempel während der Schaltsekunde
- *Gleiche* Zeitstempel für aufeinanderfolgende Ereignisse
- Kein eindeutiger Beginn der neuen Minute / der Stunde / des UTC-Tages

Zeit während der Schaltsekunde kleinstmöglich weiterzählen



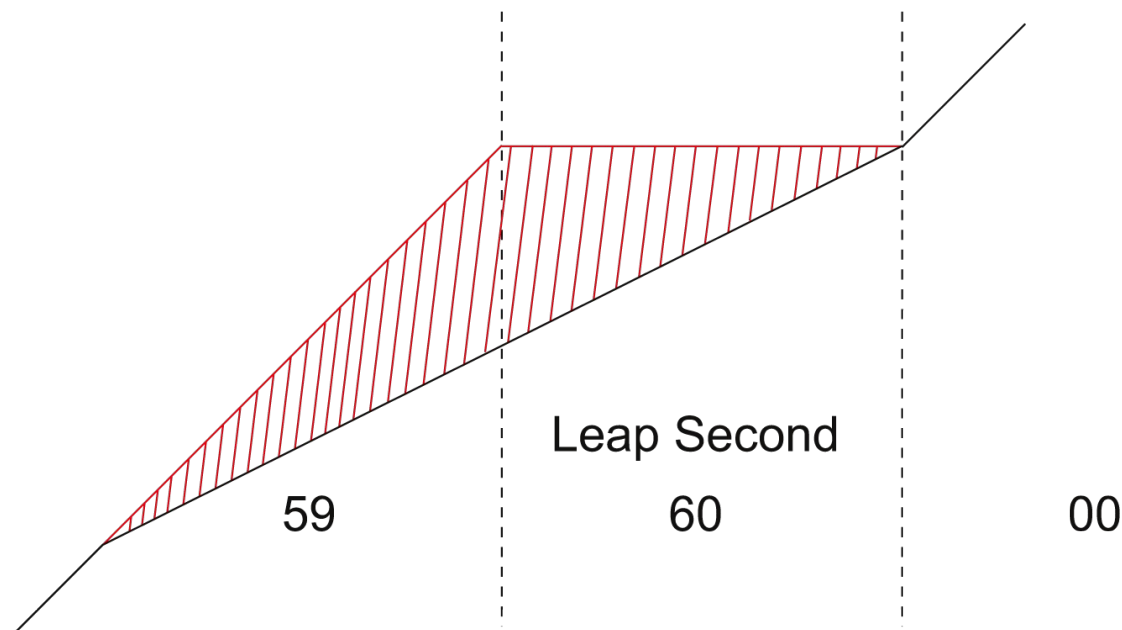
- Systemzeit wird jedes mal minimal erhöht, wenn ein Programm die Zeit liest
- Vorgeschlagen von David L. Mills, dem Erfinder von NTP
- Zeit verläuft streng monoton
- Keine doppelten Zeitstempel durch die Schaltsekunde
- Eindeutiger und fast exakter Beginn der neuen Minute / der Stunde / des UTC-Tages

Weiche, lineare Nachführung über 2 Sekunden *zu Beginn* der Schaltsekunde



- Kann verwendet werden, wenn die Systemzeit keine Schaltsekunden unterstützt
- Wurde implementiert in NTP für Windows (vor NTP v4.2.8)
- Workaround zur Schaltsekunden-Verarbeitung unter Windows
- Keine doppelten Zeitstempel durch die Schaltsekunde
- Minimaler Fehler, aber kein korrekter Beginn der neuen Minute / der Stunde / des UTC-Tages

Weiche, lineare Nachführung über 2 Sekunden *vor Ende* der Schaltsekunde



- Kann verwendet werden, wenn die Systemzeit keine Schaltsekunden unterstützt
- Wurde implementiert in NTP v4.2.8 für Windows
- Besserer Workaround zur Schaltsekunden-Verarbeitung unter Windows
- Keine doppelten Zeitstempel durch die Schaltsekunde
- Minimaler Fehler und korrekter Beginn der neuen Minute / der Stunde / des UTC-Tages

Vorschlag „UTC-SLS“

- UTC mit „Smeared Leap Seconds“
- Vorgeschlagen von Markus G. Kuhn, Universität Cambridge
- Systemzeit wird langsam angepasst während der letzten 1000 s des Tages
- Muss im Betriebssystem-Kernel implementiert werden
- Schaltsekunden-Ankündigung muss früh genug erfolgen, damit die Anpassung rechtzeitig beginnen kann
- Die so nachgeführte Zeit weist während der Nachführung einen signifikanten Unterschied zu UTC auf
- Eindeutiger und exakter Beginn der neuen Minute / der Stunde / des UTC-Tages
- Kein System bekannt, was dies implementiert

„Google Leap Smear“

- Von Google implementiert nach Problemen mit der Schaltsekunde 2005
- Modifizierte NTP-Server senden „korrigierte“ Zeit an NTP-Clients
- Zeitkorrektur erfolgt sehr langsam und weich (Cosinus-Funktion)
- NTP-Clients folgen der „korrigierten“ Zeit von den NTP-Servern
- Modifikation nur auf den Servern erforderlich, nicht auf den Clients
- Daher leicht zu implementieren für eine große Anzahl von Clients
- Kann nur im privaten Netzwerk verwendet werden
- Systemzeit mit signifikantem Offset zu UTC während des „Smear-Intervalls“
- Eindeutiger und exakter Beginn der neuen Minute / der Stunde / des UTC-Tages

GPS-Satellitensystem

- GPS-Zeit ist linear, mit konstantem Offset zu TAI, unterscheidet sich von UTC
- Zeitzählung in Wochen modulo 1024 sowie Wochensekunde
- Schaltsekundenankündigung durch die Satelliten ca. 6 Monate vorher
- Datensatz enthält Wochennummer und Tagesnummer der Schaltsekunde sowie UTC-Offset vor und nach der Schaltsekunde. Flexibel, denn
 - ✓ gibt den exakten Zeitpunkt der Schaltsekunde an
 - ✓ kann positive und negative Schaltsekunden ankündigen
 - ✓ kann theoretisch sogar mehrfache Schaltsekunden ankündigen
- Potentiell problematisch ist, dass die übertragene Wochennummer nur einen Zeitpunkt im Bereich +/- 127 Wochen angeben kann und sonst mehrdeutig ist
- Wie gibt der GPS-Empfänger die Schaltsekunden-Ankündigung an nachgeschaltete Geräte weiter?

Langwellensender DCF-77

- Sendet die gesetzliche Zeit für Deutschland
- Es ist nur eine Ankündigung für **positive** Schaltsekunden definiert
- Ankündigung negativer Schaltsekunden nicht möglich / nicht spezifiziert
- Schaltsekunden-Ankündigung nur 59 Minuten vorher
- Ankündigung eventuell zu spät für NTP-Server
- Ankündigung geht verloren, wenn in dieser Stunde Empfang gestört ist

IRIG-Zeitcodes

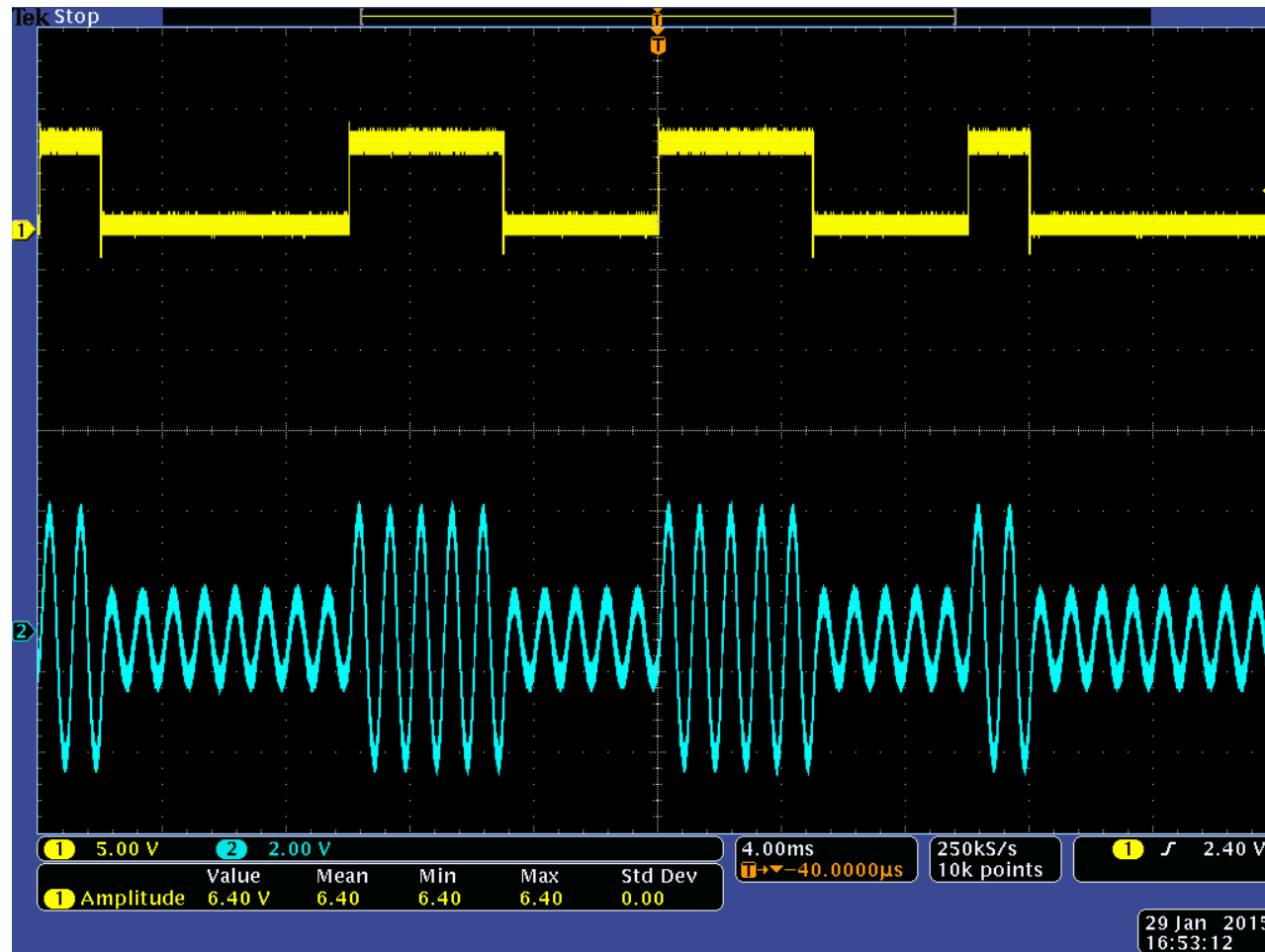
- Eingeführt ca. 1960, lange bevor Schaltsekunden „erfunden“ wurden
- Häufig verwendete IRIG-Formate enthalten keine Schaltsekunden-Ankündigung
- Selbst wenn Sekunde „60“ übertragen wird, ist das zu spät zur Verarbeitung

IEEE-Zeitcodes

- IEEE 1344 von 1995 spezifiziert, wie Datenfelder, die in IRIG-B122 reserviert sind, zur Übertragung von Zusatzdaten genutzt wird: 2-stellige Jahreszahl, Zeitoffset der Ortszeit, Sommerzeit-Status und Schaltsekunden-Ankündigung
- IEEE C37.118 von 2005 ist eine Überarbeitung von 1344, spezifiziert aber umgekehrtes Vorzeichen für Zeitoffset der Ortszeit
- Schaltsekunden-Ankündigung nur 59 Sekunden vorher, viel zu spät zur Verwendung als Zeitquelle für NTP-Server, aber ausreichend für Endgeräte, z.B. Anzeigen

Beispiel eines IIRIG-Zeitcode-Signals

Unmoduliertes Zeitcode-Signal (DC Level Shift, DCLS)



Moduliertes Zeitcode-Signal, 1 kHz Trägerfrequenz

Serielle Zeitlegramme

- Oft verwendet, um Zeit- und Statusinformationen von einer Funkuhr oder einem GPS-Empfänger an einen NTP-Zeitserver zu übertragen
- Nur wenige Telegrammformate enthalten ein Feld für Schaltsekunden-Ankündigung
- Viele oft verwendete Formate wie NMEA unterstützen dies nicht
- Einige GPS-Empfänger können die Schaltsekunden-Ankündigung über proprietäre Binärformate übertragen
- Die von Meinberg standardmäßig verwendeten Telegrammformate für GPS-Empfänger und andere Funkuhren können eine Schaltsekunden-Ankündigung übertragen
- Wenn das gewählte Telegrammformat dies nicht unterstützt, hat der GPS-Empfänger keine Möglichkeit, die Schaltsekunden-Ankündigung weiterzugeben

NIST Leap Second File

- Eine Textdatei mit einer Tabelle historischer und aktueller Schaltsekunden
- Kann von FTP-/HTTP-Servern heruntergeladen werden
- Enthält ein „Verfallsdatum“
- Kann mit NTP-Servern verwendet werden
- Neben USNO und NIST stellt jetzt auch der IERS selbst solch eine Datei zur Verfügung

Olson TZ Data Base

- Oft verwendet, um UTC in Ortszeit je nach Zeitzone umzurechnen
- Enthält ebenfalls ein Leap Second File
- Enthält historische Daten zu Zeitzonen und Schaltsekunden
- Schaltsekunden-Daten nur verwendet bei „right“-Zeitzone
- „right“-Zeitzone jedoch meist nicht verwendet
- NTP-Implementierung *chrony* verwendet dieses Leap Second File

Precision Time Protocol (PTP/IEEE 1588)

- Verwendet auf Protokoll-Ebene standardmäßig TAI-Zeitskala
- UTC-Offset wird durch Protokoll übertragen
- 2 unterschiedliche Flags für positive und negative Schaltsekunden
- Verwendung zweier Flags ist fehleranfällig bei der Implementierung
- Schaltsekunden-Ankündigung bis zu 12 Stunden vor UTC-Mitternacht
- Open Source-Implementierung (ptpd) hat ursprünglich die Schaltsekunden-Ankündigung gar nicht ausgewertet
- Problem erkannt und behoben ein paar Wochen vor der Schaltsekunde Ende Juni 2012
- Programm läuft unter Linux und FreeBSD
- Schaltsekunden-Ankündigung werden an den Kernel weitergegeben

Network Time Protocol (NTP)

- Verwendet auf Protokoll-Ebene standardmäßig UTC-Zeit
- Das Protokoll unterstützt positive und negative Schaltsekunden
- Die Standard-Implementierung akzeptiert Schaltsekunden-Ankündigungen
 - ✓ von anderen NTP-Servern
 - ✓ von Hardware-Refclocks (Funkuhren, GPS-Empfänger)
 - ✓ NIST Leap Second File
- Die genaue Art der Verarbeitung hängt ab von der Software-Version
- Priorisierung der möglichen Quellen der Ankündigung
- Normalerweise wird die Schaltsekunden-Ankündigung an den Kernel durchgereicht, wenn der Kernel dies unterstützt (Linux, *BSD, Solaris) und Kernel Discipline nicht durch Verwendung von Parameter -x deaktiviert wurde

Unterschiede bei NTP-Versionen

- Bis v4.2.4 Verwendung des Leap Second File nur, wenn Autokey vollständig konfiguriert ist
- Ab v4.2.4 Schaltsekunden-Workaround für Windows enthalten
- Bis v4.2.4 Ankündigung von einzeltem Server akzeptiert
- Ab v4.2.6 Ankündigung nur von mehreren Servern akzeptiert
- Erst ab v4.2.6 Verarbeitung durch NTP, wenn kein Kernel Discipline (non-Windows)
- In v4.2.6 ist Leap Second File immer autoritativ, andere Quellen werden ignoriert
- In v4.2.6 ist **Leap Second File selbst autoritativ, wenn „Verfallsdatum“ überschritten**, d.h. keine Schaltsekunden Verarbeitung, selbst wenn andere Quellen ankündigen
- Ab v4.2.8 wird das „Verfallsdatum“ des Leap Second File beachtet
- Bis v4.2.8 Problem mit Workaround unter Windows 8 / Server 2012 aufgrund von Änderungen im Windows-Kernel
- Ab v4.2.8p1 modifizierter Workaround, der auch unter Windows 8 / Server 2012 korrekt funktioniert
- Seit v4.2.6 **Step der Zeit, selbst wenn -x angegeben** ([NTP Bug 2745](#)), Lösung ?

UNIX-Kernel

- Normalerweise Verarbeitung der Schaltsekunde selbständig, wenn rechtzeitig eine Ankündigung erfolgt
- Über ein Kernel-Interface können Programme wie ntpd oder ptpd die Schaltsekunden-Ankündigung an den Kernel weitergeben
- Über das gleiche Kernel-Interface können andere Programme den Schaltsekunden-Status des Kernels abfragen
- Bei den meisten aktuellen UNIX-Kernels springt die Zeit zu Beginn der Schaltsekunde um 1 s zurück

Windows

- Kennt gar keine Schaltsekunden
- Es gibt einen Workaround in NTP für Windows, bei dem die Zeit über 2 Sekunden nachgeführt wird

Linux Kernel Deadlock

- Während der Verarbeitung der Schaltsekunde auf einer ausgelasteten Maschine konnte der Kernel versuchen, ein Lock zu erhalten, was er aber bereits hatte
- Dadurch wurde ein Deadlock verursacht, sofortiger Stillstand der Maschine, Services nicht mehr verfügbar
- Problem in Kernels ca. 2.6.22 bis 2.6.26.6

Hohe CPU-Auslastung auf Linux-Systemen

- Passiert bei der Schaltsekunde Ende Juni 2012
- CPU-Auslastung konnte dauerhaft auf 100% gehen
- Dadurch signifikant erhöhter Stromverbrauch und Wärmeentwicklung in Data Centern
- Durch erhöhte Kühlung nochmals erhöhter Stromverbrauch
- Problem in Kernels bis ca. 2.6.32

Standard-NTP-Software

- Bis v4.2.4 wurde Schaltsekunde nur verarbeitet, wenn Kernel Discipline verfügbar und verwendet
- In v4.2.4 wurde ein Workaround für Windows eingeführt
- In v4.2.6 Fix für andere Systeme ohne Kernel Discipline, Step zurück um 1 s zu Beginn der Schaltsekunde
- Bis 4.2.6 konnte es passieren, dass ein „Leap Second Loop“ unter vernetzten NTP-Servern entstand, wodurch auch am Ende jedes folgenden Monats eine Schaltsekunde eingefügt wurde, bis alle betroffenen ntpds neu gestartet wurden
- In 4.2.6 wurde Unterstützung negativer Schaltsekunden entfernt
- in 4.2.8 wurde die Schaltsekundenunterstützung in NTP überarbeitet
- Fix für -x (slew always vs. step leap second) eventuell in 4.2.8p3

Open Source ptpd

- Unterstützung für Schaltsekunden erst seit v2.2.2
- Im Grandmaster-Mode noch kein automatisches Update, ptpd Restart erforderlich
- Eventuell ein Fix für v2.3.1

Falsche Schaltsekunden-Ankündigung durch GPS-Empfänger von Fremdherstellern

- Erstmalig bekannt geworden nach Juli 2005, als GPS-Satelliten begannen, Schaltsekunden-Daten für **Dezember 2005** zu senden
- Falsche Ankündigung einer Schaltsekunde für **September 2005**
- Dadurch falsche Ankündigung einer Schaltsekunde auch durch NTP-Server
- Als Folge Modifikation in ntpd, als Client Schaltsekunden nur zu akzeptieren, wenn eine Mehrheit der konfigurierten NTP-Server die Ankündigung aussendet
- NTP-Server können trotzdem betroffen sein, denn Refclocks gelten als autoritativ, wenn kein (aktuelles) Leap Second File verfügbar ist
- Auf der NTP Pool-Mailingliste wurde berichtet, dass das Problem auch im Januar 2015 erneut auftrat, nachdem GPS-Satelliten begannen, die Schaltsekunde für Ende Juni 2015 anzukündigen
- Diese Fehler sind nicht in ntpd, sondern in der Firmware der GPS-Empfänger

Zeitpunkte der Einfügung

- 31. Dez. 1998 Do/Fr Neujahrstag
- 31. Dez. 2005 Sa/So Neujahrstag, Wochenende
- 31. Dez. 2008 Mi/Do Neujahrstag
- 30. Juni 2012 Sa/So Wochenende
- 30. Juni 2015 Di/Mi **normaler Arbeitstag**

Schaltsekunde 2015

- UTC: Mi 00:00 Uhr
- Deutschland: Mi 02:00 Uhr
- Asien / Japan: Mi 08:00 Uhr
- USA / Kalifornien: Di 16:00 Uhr

→ geringeres Risiko in Europa / Afrika, nachts

→ höheres Risiko in Asien / Amerika, zur Arbeitszeit

→ Problematische eigenständige Versuche, die Schaltsekunde anders zu verarbeiten

- Es ist wichtig, dass die Schaltsekunden-Ankündigung bis hinunter zu den Endgeräten rechtzeitig erfolgt
- Auf dem gesamten Weg gibt es verschiedene Stationen mit unterschiedlichen Verbindungsarten und Protokollen
- Auf jedem Weg und in jeder Station müssen „Worst Case“-Bedingungen eingehalten werden, damit z.B. bei einer Hierarchie von NTP-Clients/Servern mit 1024 s Polling Intervall die Information überall ankommt
- Zur Anbindung von GPS-Empfängern geeignetes Protokoll verwenden
- IRIG-ähnliche Zeitcodes und viele serielle Telegrammformate nicht geeignet
- Verwendung eines Leap Second Files kann hilfreich sein
- Leap Second File muss aktuell gehalten werden, auch wenn keine Schaltsekunde
- Bei abgelaufenem „Verfallsdatum“ keine Möglichkeit, zu unterscheiden, ob keine Schaltsekunde ansteht oder nur keine aktuelle Information zur Verfügung steht

- **Sorgfältiges Software-Design und sorgfältige Implementierung erforderlich**
- **Vergleich von Zeitstempeln von unterschiedlichen Systemen:**
 - ✓ Schaltsekunde kann unterschiedlich verarbeitet werden
 - ✓ Dadurch können um die Schaltsekunde herum ungewöhnlich große Zeitdifferenzen auftreten
- **Zeitstempel:**
 - ✓ Die Zeit kann zurückspringen
 - ✓ Dadurch können doppelte Zeitstempel auftreten
- **Zeitintervalle:**
 - ✓ Ein Tag kann 86400 ± 1 s lang sein
 - ✓ Auch wenn „historische Daten“ verarbeitet werden

- **Versuchen, Mehrdeutigkeit von Zeitstempeln zu vermeiden**
 - Wenn die Zeit aufgrund einer Schaltsekunde zurückspringt
 - Das Problem ist eigentlich ähnlich wie am Ende der Sommerzeit
 - Ein Schaltsekunden-Status wäre hilfreich, zu unterscheiden
- **GsoC 2013 Projekt der Network Time Foundation**
 - Untersuchung eines neuen Timestamp-Formats und einer entsprechenden API
 - Eventuell nicht geeignet, Zeitstempel sehr schnell und oft hintereinander zu lesen
 - Zusätzliche Berechnungen im Kernel verlängern die Ausführungszeit
 - Locking-Mechanismen erforderlich, um Konsistenz von Zeitstempel und zugehörigem Status zu garantieren

- **Vorschlag von Steve Allen: Systemzeit läuft mit TAI (linear)**
 - Verwendung der „right“-Zeitzone der TZ Database
 - PTP/IEEE 1588 verwendet sowieso TAI
 - UTC-Zeit kann eindeutig abgeleitet werden
 - TZ Leap Second File hat jetzt auch „Verfallsdatum“
- **Mögliche Verbesserungen:**
 - Eine API-Funktion, herauszufinden, ob Kernel auf UTC oder TAI läuft, so dass Programme wie ntpd oder ptpd das zu Laufzeit herausfinden können
 - Möglichkeiten, die Schaltsekunden-Information (Leap Second File) automatisch zu aktualisieren

- **Einschränkungen bei Verwendung der „right“-Zeitzone:**
 - Daten müssen regelmäßig aktualisiert werden
 - Leap Second File im NIST-Format muss heruntergeladen und konvertiert werden
 - Updates der TZ Database momentan nur als Software-Update verteilt
 - Embedded Systeme erhalten oft keine Software-Updates
- **Verbesserungen in Arbeit:**
 - Neues tzdist-Protokoll ermöglicht Updates der TZ-Datenbank und Schaltsekunden-Information via Netzwerk (HTTP/JSON)
 - Vorschlag von Poul-Henning Kamp, Schaltsekunden-Informationen per DNS zu verbreiten (DNSSEC, Pseudo-Einträge im DNS-Server)
 - Langwährende Diskussion, Schaltsekunden abzuschaffen

- Jedes Glied der Verarbeitungskette hat eigene Eigenschaften, Vorteile und Einschränkungen
- Es ist wichtig, diese Eigenschaften im Zusammenspiel der einzelnen Komponenten zu berücksichtigen
- Zukünftige Verbesserungen sollten versuchen, potentielle Fehlerquellen zu beseitigen
- Kompatibilität existierender Applikationen sollte weitestmöglich erhalten bleiben

- Mills, D.L., "A kernel model for precision timekeeping", Network Working Group Report RFC 1589, University of Delaware, March 1994
- Kuhn, Markus G., "UTC with Smoothed Leap Seconds (UTC-SLS)", 2005-12-14, <http://www.cl.cam.ac.uk/~mgk25/time/utc-sls/>
- Google Blog, "Time, technology and leaping seconds", 2011-09-15, <http://googleblog.blogspot.de/2011/09/time-technology-and-leaping-seconds.html>
- Linux Kernel Mailing List, "LKML Bug: Status/Summary of slashdot leap-second crash on new years 2008-2009", 2009-01-02, <https://lkml.org/lkml/2009/1/2/373>
- RedHat Bug 479765, "Leap second message can hang the kernel", 2009-01-12, https://bugzilla.redhat.com/show_bug.cgi?id=479765
- Linux Kernel Mailing List, "Potential fix for leapsecond caused futex related load spikes", 2012-07-01, <https://lkml.org/lkml/2012/7/1/27>

- RedHat Bug 836803, "RHEL6: Potential fix for leapsecond caused futex related load spikes", 2012-07-01, https://bugzilla.redhat.com/show_bug.cgi?id=836803
- "Leap second bug in Linux wastes electricity", 2012-07-03, <http://www.h-online.com/open/news/item/Leap-second-bug-in-Linux-wastes-electricity-1631462.html>
- NTP bug 508: "ntpd doesn't handle leap seconds gracefully on systems without kernel support for leap seconds", https://bugs.ntp.org/show_bug.cgi?id=508
- NTP Bug 2246: "sys_leap is sticky", https://bugs.ntp.org/show_bug.cgi?id=2246
- NTP Bug 2745: "ntpd -x steps clock on leap second", https://bugs.ntp.org/show_bug.cgi?id=2745

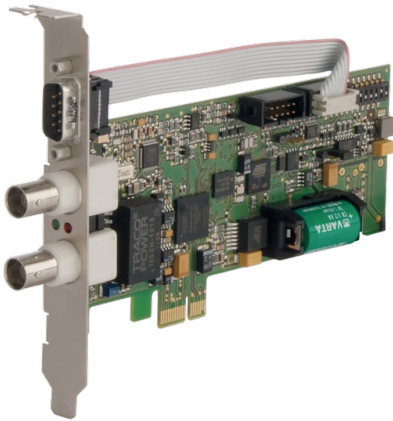
- Google Summer of Code 2013, "New Timestamp Format and API",
<http://www.google-melange.com/gsoc/project/details/google/gsoc2013/limifly/5727390428823552>
- Network Time Foundation, "New Timestamp and Library API", Google Summer of Code 2013,
<https://wiki.nwtime.org/GSoC/GSoC2013NewTimestampAndAPI>
- Steven L. Allen, "Timekeeping System Implementations: Options For The Pontifex Maximus",
October 2011 meeting "Decoupling Civil Timekeeping from Earth Rotation"



- Gegründet 1979 von Werner und Günter Meinberg
- Erste Produkte: DCF-77-Funkuhren für industrielle Anwendungen (1980)
- Eigene Entwicklung eines GPS-Zeitempfänger erstmalig 1993
- LANTIME NTP- und PTP-Timeserver, PCI-Karten
- Entwicklung, Design, Herstellung, Verkauf und Support aus einer Hand
- Produktion von 90% der mechanischen Komponenten sowie Herstellung der elektronischen Komponenten und Montage im eigenen Werk
- Etwa 90 Mitarbeiter, 20 davon in der Entwicklung tätig
- Firma in Bad Pyrmont, ca. 70 km südwestlich von Hannover

Produktspektrum umfaßt:

- GPS-Empfänger und Funkuhren für DCF-77, MSF, and WWVB
- NTP / PTP Timeserver und PTP Slave-Geräte mit großem Funktionsumfang
- IRIG / AFNOR / IEEE Zeitcode-Generatoren und -Empfänger
- Frequenz-Referenzsysteme für Telekommunikations-Lösungen



Danke



Danke für Ihre Aufmerksamkeit!