

August 2012
Geoff Huston

Leaping Seconds

The tabloid press are never lost for a good headline, but this one in particular caught my eye: "Global Chaos as moment in time kills the Interwebs". I'm pretty sure that "global chaos" is somewhat over the top, but there was a problem happening on the 1st of July this year, and yes, it impacted the Internet in various ways, as well as many other enterprises who rely on IT systems. And yes, the problem had a lot to do with time and how we measure it. This month I'd like to look at the cause of this problem in a little more detail.



<http://www.heraldsun.com.au/news/leap-second-crashes-qantas-and-leaves-passengers-stranded/story-e6frf7jo-1226413961235>

What is a Second?

I'd like to start a rather innocent question: What exactly is a second? Obviously it's a unit of time, but what defines a second? Well there are 60 of these seconds in a minute, 60 minutes in an hour and 24 hours in a day. That would infer that a "second" is $1/86400$ of a day, or $1/86400$ of the length of time it takes for the Earth to rotate about its own axis. Yes?

Almost, but this is still a little imprecise. What's the frame of reference that defines a unit of rotation of the Earth?

As was established in the work a century ago in attempting to establish a frame of reference for the measurement of the speed of light, these frame of reference questions can be quite tricky!

What is the frame of reference to calibrate the Earth's rotation about its own axis? A set of distant stars? The Sun? These days we use the Sun, which seems like a logical choice in the first instance. But cosmology is far from perfect, and far from being a stable measurement, this use of the length of time it takes for the Earth to rotate once about its axis relative to the Sun varies month by month by up to some 30 seconds from its mean value. This variation in the Earth's rotational period is an outcome of

both the Earth's elliptical orbit around the Sun, and the Earth's axial tilt. These variations mean that by the time of the March equinox the "solar day" is some 18 seconds shorter than the mean, at the time of the June solstice its some 13 seconds longer, at the September equinox its some 21 seconds shorter and in December its some 29 seconds longer. This variation in the rotational period of the Earth is unhelpful if you are looking for a stable way to measure time. To keep this unit at a constant value a second is based on an ideal version of the Earth's rotational period, and we have chosen to base the unit of measurement of time on "mean solar time." This "mean solar time" is the average time for the Earth to rotate about its own axis, relative to the Sun. This is a relatively constant value, as the variations in solar time work to cancel out each other in the course of a full year. So a second is defined as 1/86400 of mean solar time, or in other words 1/86400 of the average time it takes for the Earth to rotate on its axis. And how do we measure this "mean solar time"? Well that's derived from baseline interferometry from a number of distant radio sources.

So now we have a second as a unit of the measurement of time, based on the Earth's rotation about its own axis, and from this we can construct a uniform time system to measure not only intervals of time, but to allow us all to agree on a uniform value of absolute time. From this we can not only make calendars that are "stable, in that the calendar does not drift forward or backward in time from year to year, but accurate in that we can agree on absolute time down to units of minute fractions of a second. Well so one would've thought, but the imperfections of cosmology intrude once again.

The Earth has the Moon, and the Earth generates a tidal acceleration of the Moon, and, in turn the Moon decelerates the Earth's rotational speed. As well as this long term factor arising from the gravitational interaction between the Earth and the Moon, the Earth's rotational period is affected by climatic and geological events that occur on and within the Earth (Figure 1). This means that it's possible for the Earth's rotation to both slow down and speed up at times. So the two requirements of a second, namely that it is a constant unit of time and it is defined as 1/86400 of the mean time taken for the Earth to rotate on its axis cannot be maintained. Either one or the other has to go.

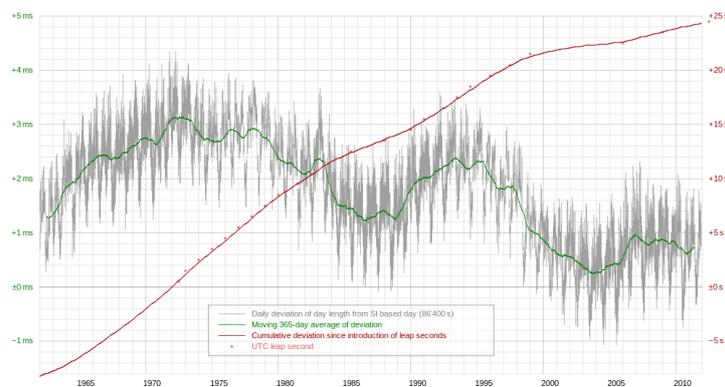


Figure 1 - The deviation of the mean solar day from the SI-based day 1962 - 2010

In 1955 we went down the route of a standard definition of a second, which was defined by the International Astronomical Union as 1/31,556,925.9747 of the 1900.0 "mean tropical year". This definition was also adopted in 1956 by the International Committee for Weights and Measures and in 1960 by the General Conference on Weights and Measures, becoming a part of the International System of Units (SI). This definition addressed the problem of the drift in the value of the mean solar year by specifying a particular year as the baseline for the definition.

However, by the mid 1960's this definition too was found to be inadequate for precise time measurements, so in 1967 the SI second was again redefined, this time in experimental terms as a repeatable measurement. The new definition of a second was 9,192,631,770 periods of the radiation emitted by a caesium-133 atom in the transition between the two hyperfine levels of its ground state.

Leaping Seconds

So we have the concept of a second as a fixed unit of time, but how does this relate to the astronomical measurement of time? For the past several centuries the length of the mean solar day has been increasing by an average of some 1.7ms per century. Given that the solar day was fixed on the mean solar day of 1900, then by 1961 the mean solar day was around a millisecond longer than 86400 SI seconds. Therefore, absolute time standards that change the date after precisely 86400 SI seconds, such as the International Atomic Time (TAI), get increasingly ahead of the time standards that are rigorously tied to the mean solar day, such as Greenwich Mean Time (GMT).

When the Coordinated Universal Time (UTC) standard was instituted in 1961, based on atomic clocks, it was felt necessary that this time standard maintain agreement with the Greenwich Mean Time (GMT) time of day, which until then had been the reference for broadcast time services. Thus, from 1961 to 1971, the rate of broadcast time from the UTC atomic clock source had to be constantly slowed to remain synchronised with GMT. During that period, therefore, the "seconds" of broadcast services were actually slightly longer than the SI second and closer to the GMT seconds.

In 1972 the "leap second" system was introduced, so that the broadcast UTC seconds could be made exactly equal to the standard SI second, while still maintaining the UTC time of day and changes of UTC date synchronised with those of UT1 (the solar time standard that superseded GMT). Reassuringly, a second is now a SI second in both the UTC and TAI standards, and the precise time when time transitions from one second to the next is synchronised in both these reference frameworks. But this fixing of the two time standards to a common unit of exactly one second means that to track the time of day it necessary to periodically add or remove entire seconds from the UTC time of day clock. Hence the use of so-called "leap seconds". By 1972 the UTC clock was already 10 seconds behind TAI, which had been synchronized with UT1 in 1958 but had been counting true SI seconds since then. After 1972, both clocks have been ticking in SI seconds, so the difference between their readouts at any time is 10 seconds plus the total number of leap seconds that have been applied to UTC.

Since 1 January 1988 the role of coordinating the insertion of these "leap second" corrections to the UTC time of day has been the responsibility of the International Earth Rotation and Reference Systems Service (IERS). IERS usually decides to apply a leap second whenever the difference between UTC and UT1 approaches 0.6s, in order to keep the absolute difference between UTC and the mean solar UT1 broadcast time from exceeding 0.9s.

The UTC standard allows leap seconds to be applied at the end of any UTC month, but since 1972 all of these leap seconds have been inserted either at the end of June 30 or December 31, making the final minute of the month in UTC, either one second longer or one second shorter when the leap second is applied. IERS publishes announcements every six months, whether leap seconds are to occur or not, in its "Bulletin C". Such announcements are typically published well in advance of each possible leap second date — usually in early January for a June 30 scheduled leap second and in early July for a December 31 leap second. Greater levels of advance notice are not possible because of the degree of uncertainty in predicting the precise value of the cumulative effect of fluctuations of the deviation of the Earth's rotational period from the value of the mean solar day.

Between 1972 and 2012 some 25 leap seconds have been added to UTC. On average this implies that a leap second has been inserted about every 19 months. However, the spacing of these leap seconds is quite irregular: there were no leap seconds in the seven-year interval between January 1, 1999 and December 31, 2005, but there were 9 leap seconds in the 8 years 1972–1979 (Figure 2). Since December 31 1998 there have been only 3 leap seconds, on December 31 2005, December 31 2008 and June 30 2012, each of which have added one second to that final minute of the month, at the UTC time of day.

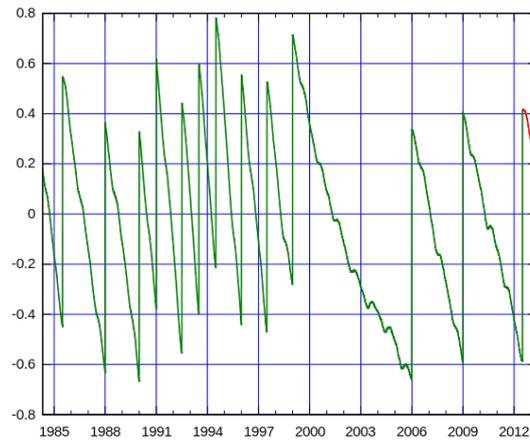


Figure 2 – The difference between UT1 and UTC
1984 – 2012

Leaping Seconds and Computer Systems

The June 30 2012 leap second did not exactly pass without a hitch, as reported by the tabloid press.

The side effect of this particular leap second appeared to include computer system outages and crashes – an outcome that was unexpected and surprising. This leap second managed to crash some servers used in the Amadeus airline management system, throwing the Qantas airline into a flurry of confusion on Sunday morning on the 1st of July in Australia. But not just the airlines were affected, as LinkedIn, Foursquare, Yelp, Opera were among a number online service operators who had their servers stumble in some fashion. This managed to also affect some internet service providers and data centre operators. One Australian service provider has reported that a large number of their Ethernet switches seize up over a two hour period following the leap second.

It appears that one common element here was the use of the Linux operating system.

But Linux is not exactly a new operating system, and the use of the Leap Second option in the Network Time Protocol (NTP) is not exactly novel either. Why didn't we see the same problems in early 2009, following the leap second that occurred on the 31st December 2008?

Ah, but there were problems then, but perhaps it was blotted out in the post new year celebratory hangover! Some folk noticed something wrong with their servers on the 1st of January 2009. Problems with the leap second were recorded with Red Hat Linux following the December 2008 leap second, where kernel versions of the system prior to 2.6.9 could encounter a deadlock condition in the kernel while processing the leap second. [https://bugzilla.redhat.com/show_bug.cgi?id=479765]

"[...] the leap second code is called from the timer interrupt handler, which holds xtime_lock. The leap second code does a printk to notify about the leap second. The printk code tries to wake up klogd (I assume to prioritize kernel messages), and (under some conditions), the scheduler attempts to get the current time, which tries to get xtime_lock => deadlock."
[<http://lkml.org/lkml/2009/1/2/373>]

The advice in January 2009 to sysadmins was to upgrade their systems to 2.6.9 or later, which contained a patch that avoided this kernel-level deadlock.

This time around it's a different problem, where the server's CPU encountered a 100% utilisation:

"The problem is caused by a bug in the kernel code for high resolution timers (hrtimers). Since they are configured using the CONFIG_HIGH_RES_TIMERS option and most systems

manufactured in recent years include the High Precision Event Timers (HPET) supported by this code, these timers are active in the kernels in many recent distributions.

"The kernel bug means that the hrtimer code fails to set the system time when the leap second is added. The result is that the hrtimer representation of the time taken from the kernel is a second ahead of the system time. If an application then calls a kernel function with a timeout of less than a second, the kernel assumes that the timeout has elapsed immediately after setting the timer, and so returns to the program code immediately. In the event of a timeout, many programs simply repeat the requested operation and immediately set a new timer. This results in an endless loop, leading to 100% CPU utilisation."

[<http://www.h-online.com/open/news/item/Leap-second-bug-in-Linux-wastes-electricity-1631462.html>]

Leap Smearing

Following a close monitoring of their systems in the earlier 2005 leap second Google engineers were aware of problems in their operating system when processing this leap second. They had noticed that some clustered systems stopped accepting work during the leap second of December 31 2005, and they wanted to ensure that this did not recur in 2008. Their approach was subtly different to that used by the Linux kernel maintainers.

Rather than attempt to hunt down bugs in the time management code streams in the system kernel, they noted that the intentional side effect of the Network Time Protocol was to continually perform slight time adjustments in the systems that are synchronising their time according to the NTP signal. If the quantum of an entire second in a single time update was a problem to their systems, then what about an approach that allowed the 1 second time adjustment to be smeared across a number of minutes or even a number of hours? That way the leap second would be represented as a larger number of very small time adjustments which, in NTP terms, was nothing exceptional. The result of these changes was that NTP itself would start slowing down the time of day clock on these systems some time in advance of the leap second by very slight amounts, so that at the time of the applied leap second, at 23:59:59 UTC, the adjusted NTP time would have already been wound back to 23:59:58. The leap second, which would normally be recorded as 23:59:60 was now a 'normal' time of 23:59:59, and whatever bugs that remained in the leap second time code of the system were not exercised.

[<http://googleblog.blogspot.de/2011/09/time-technology-and-leaping-seconds.html>]

More Leaping?

The topic of leap seconds remains a contentious one. There was a proposal from the United States to the ITU-R Study Group 7's Working Party 7-A back in 2005 to eliminate leap seconds. It's not entirely clear whether these leap seconds would be replaced by a less frequent "leap hour", or whether the entire concept of attempting to link UTC and the mean solar day would be allowed to drift, and over time we would see UTC time shifting away from UT1's concept of solar day time. This proposal was most recently considered by the ITU-R in January 2012, and there was evidently no clear consensus on this topic. France, Italy, Japan, Mexico and the US were reported to be in favor of abandoning leap seconds, while Canada, China, Germany and the UK were reportedly against these changes to UTC. At present a decision on this topic, or at the least a discussion on this topic, is scheduled for the 2015 World Radio Conference.

While these computing problems with processing leap seconds are annoying and for some folk extremely frustrating and sometimes expensive, I'm not sure this factor alone should drive the decision process about whether to drop leap seconds from the UTC time framework. With our increasing dependence on highly available systems, and the criticality of accurate time of day clocks as part of the

basic mechanisms of system security and integrity, it would be good to think that we have managed to debug this processing of leap seconds.

It's often the case in systems maintenance that the more a bug is exercised the more likely it is that the bug will be isolated and corrected. However with leap seconds this is a tough ask, as the occurrence of leap seconds is not an easily predicted occurrence. Whenever we next have to leap a second in time about the best we can do is hope that we are ready for it.

Further Reading

The story of calendars, time, time of day and time reference standards is a fascinating story. It includes ancient stellar observatories, the medieval quest to predict the date of Easter, the quest to construct an accurate clock that would allow the calculation of longitude, and the current constellations of time and location reference satellites, and these days much of this material can be found on the net.

A good starting point for the leap second can be found in Wikipedia under the topic of "Leap_second" [http://en.wikipedia.org/wiki/Leap_second]

Disclaimer

The views expressed are the author's and not those of APNIC, unless APNIC is specifically identified as the author of the communication. APNIC will not be legally responsible in contract, tort or otherwise for any statement made in this publication.

About the Author

Geoff Huston B.Sc., M.Sc., has been closely involved with the development of the Internet for many years, particularly within Australia, where he was responsible for the initial build of the Internet within the Australian academic and research sector. He is author of a number of Internet-related books, and has been active in the Internet Engineering Task Force for many years.

www.potaroo.net