December 07, 2016

Leap-smeared representation of time for high-accuracy applications

Michael Shields

Leap-smeared representation of time for high-accuracy applications

ABSTRACT

Civil time is based upon coordinated universal time (UTC), which is synchronized to the length of the day. However, the earth's rotational speed is slowing over time, i.e., the length of the day is increasing. In order for UTC to keep pace with the growing length of the day, UTC is delayed at pre-announced dates by one second. This delay is effected by adding a "leap second" at the end of the day of each pre-announced date. Although leap second correction achieves synchronization between UTC and length-of-day, computer systems that presume a smooth, always-forward movement of time may malfunction when such corrections are carried out.

"Leap smear" techniques are used to achieve synchronization between UTC and length-of-day while keeping smooth the flow of time for computers. Rather than add a second at a particular time of a day (e.g., at the end of the day), each second of the date (or another period) over which leap-second correction is carried out is dilated. For example, for the 86,401 atomic-clock seconds on the day of leap-second addition (with the leap second being excess over a normal day of 86,400 seconds), each of the 86,400 seconds of the day is adjusted to be of duration 1.00001157407 (86401÷86,400) atomic-clock seconds. Leap-smearing works well for computer applications other than those that require high sub-second accuracy. This disclosure describes techniques that allow use of leap smearing to applications that require sub-second accuracy.

KEYWORDS: Leap smear, Leap second, UTC, TAI

BACKGROUND

A network of globally distributed atomic clocks produces international atomic time (TAI), a highly precise measure of time. Civil time is based upon UTC, which is synchronized to the length of the day. Atomic clocks that have an accuracy of one second per millions of years showed that the earth's rotational speed has a slowing trend, that is, the length of day is increasing over time. In order for UTC to maintain synchronization with the growing length of the day, each day is continuously measured in atomic-clock seconds. Before the cumulative error between atomic-clock seconds and the length of the day reaches one second, a "leap second" is added, such that the cumulative error is kept under 900 milliseconds. For example, a 24-hour day is expected to have 86,400 atomic-clock seconds, but a TAI measurement may indicate that it is actually a few milliseconds longer (or shorter). After several months, cumulative error between the length of a day and TAI may exceed 900 milliseconds, at which time a leap second is added (or removed). Leap seconds are added on pre-announced dates. On the pre-announced day when a leap-second is added, the day spans 86,401 TAI seconds. The first leap second correction was made on June 30, 1972.

The rate of change of the earth's rotation is not fixed. Indeed, it is possible that for certain time intervals, the earth's rotational speed increases. Consequently, the date of the leap second addition (or subtraction) is fixed only upon actual measurement of the earth's rotational speed and comparison with TAI. Therefore, the exact date of leap second correction can be known only known a few months in advance. Further, the duration between leap second corrections is not fixed. Another unknown is whether the leap second is to be added or subtracted. Timekeeping authorities publish in advance a "leap table," which comprises a schedule of leap-second corrections over a few months into the future. Leap-second corrections, if any, are added on the last day of a calendar month.

Computer systems that expect or assume a smooth, always-forward flow of time may malfunction upon such adjustments, e.g., the addition of a leap second or subtraction of a leap second from a day. Moreover, many computer systems are unable to represent minutes that have sixty-one or fifty-nine seconds. In order to maintain a smooth flow of time, and to allow compatibility with computer systems that expect exactly sixty seconds in a minute, techniques known as “leap smear” are used. Under leap smear, a leap second is not added at a particular time (e.g., at the end of the pre-announced date) of leap-second correction. Rather, each second of the day (or another period) of leap-second correction is dilated by a fraction of a SI (Standard International) second, so that the cumulative effect of many dilated seconds amounts to the addition (or subtraction) of a leap second. For example, in a traditional approach, the pre-announced day of leap-second correction may have 86,401 SI seconds, while under a leap-smeared approach, that day will have 86,400 seconds like any other day, except each second would be of the duration 86401÷86,400=1.00001157407 SI seconds.

Leap smearing maintains the property that the number of seconds in a minute is always known and is always sixty. However, leap smearing is a trade-off between accuracy and practicality, and may cause problems for computer applications that require high sub-second accuracy.

DESCRIPTION

This disclosure describes techniques that enable conversion between leap-smeared time and unsmeared, e.g., UTC or TAI, time. The techniques permit leap-smearing to be used even on computer systems that implement applications that require high sub-second accuracy.

Conversion between leap-smeared and UTC, or TAI, times is simply a matter of invoking the appropriate smeared-to-unsmeared convertor. In this manner, an application that requires high sub-second accuracy can access both leap-smeared as well as unsmeared times. A library that converts between smeared and unsmeared times is described. The library is based upon certain principles, including the following:

- A leap-smeared timestamp or time interval within the range of the leap table is converted to unsmeared time precisely.
- A leap-smeared timestamp or time interval outside the range of the current leap table results, upon conversion, in a range of possible unsmeared times.

Conversion from, for example, smeared to unsmeared time, using the library produces two result objects. If the conversion is precise, then the two objects are equal, and represent unsmeared time, e.g., UTC or TAI. If the conversion results in a range of possible unsmeared times, then the two objects are unequal, and represent respectively the lower and upper ends of the range. The library, or application programming interface, based upon the above principles can be called by any application. An application developer can determine how the application utilizes the two result objects provided by using the library functions. The library may be made available as a code package in a programming language, e.g., Go.

The following examples illustrate the application of the above principles in effecting conversions between smeared and unsmeared times. Examples below are based on a leap table that extends through 30-Dec-2016. The table shows that one and only one leap second was added between 1-Jan-2015 and 30-Dec-2016, and that the leap second was added on 30-Jun-2015. The leap table does not have information relating to leap-second corrections for 31-Dec-2016 and beyond. Time is denoted with the following notation: (year, month, day, hour, minute, second, nanoseconds). Thus, 9 PM on 2-Oct-2015 would be denoted as (2015, 10, 2, 21, 0, 0, 0).

Example 1: Determine the number of TAI seconds between two leap-smeared timestamps (2015, 8, 1, 10, 0, 0, 0) and (2015, 6, 1, 10, 0, 0, 0)

Both timestamps are within the range of the leap table. The given interval contains one positive leap second. The number of days between the two timestamps is 61. Therefore conversion will result in two equal objects, each comprising the following number of TAI seconds: $61 \times 24 \times 60 \times 60 + 1$.

Example 2: Determine the number of TAI seconds between two leap-smeared timestamps (2017, 2, 1, 10, 0, 0, 0) and (2016, 10, 1, 10, 0, 0, 0)

One of the timestamps is outside the range of the leap table. Between the two timestamps, there are four possible dates for leap-second correction - 31-Oct-2016, 30-Nov-2016, 31-Dec-2016, and 31-Jan-2017. Of these four possible dates, the leap table shows that 31-Oct-2016 and 30-Nov-2016 have no leap-second corrections. Corrections for 31-Dec-2016 and 31-Jan-2017 are at present unknown and may each be +1, 0, or −1 leap seconds. Thus, through the end of the interval, we may have leap-second corrections between −2 and +2 leap seconds. The number of days between the two timestamps is 123. Conversion to unsmeared (TAI) seconds will result in two unequal objects. The first object is at the lower end of the range of possible TAI seconds - $123 \times 24 \times 60 \times 60 - 2$, and the second object is at the higher end of the range of possible TAI seconds - $123 \times 24 \times 60 \times 60 + 2$.

Example 3: Determine the unsmeared UTC timestamp corresponding to the smeared timestamp (2016, 10, 1, 10, 0, 0, 0)

The timestamp occurs on a date that has no leap-second correction. Therefore the unsmeared UTC timestamp is identical to the smeared timestamp. Conversion to unsmeared timestamp will result in two equal objects, each comprising the timestamp (2016, 10, 1, 10, 0, 0, 0).

Example 4: (an inversion of Example 3): Determine the smeared timestamp corresponding to the unsmeared UTC timestamp (2016, 10, 1, 10, 0, 0, 0)

The given timestamp occurs on a day that has no leap-second correction. Therefore the smeared timestamp will be identical to the unsmeared timestamp. Conversion to smeared timestamp will result in two equal objects, each comprising the timestamp (2016, 10, 1, 10, 0, 0, 0).

Example 5: Determine the unsmeared UTC timestamp corresponding to the smeared timestamp (2016, 12, 31, 21, 0, 0, 0)

The date is an end-of-month date outside the leap table. Hence it is unknown whether a leap-second addition, or subtraction, is applicable, if at all. Conversion to unsmeared time produces two objects - the first object indicates the start of the range of possible unsmeared times, and the second object indicates the end of the range of possible unsmeared times. At the moment of the timestamp (21:00:00), smearing, if applied starting at noon, would be 9÷24 = three-eighths complete. Thus the range of unsmeared times would extend above and below 21:00:00 hours by 0.375 seconds, that is, 375,000,000 nanoseconds. The two objects returned upon conversion would therefore comprise respectively the timestamps (2016, 12, 31, 20, 59, 59, 625000000) and (2016, 12, 31, 21, 0, 0, 375000000).

Example 6: Determine the error radius in TAI seconds for the smeared timestamp (2016, 12, 31, 21, 0, 0, 0)

The date is an end-of-month date outside the leap table. Hence it is unknown if leap-second addition, or subtraction, is applicable or not. At the moment of the timestamp (21:00:00), smearing, if applied starting at noon, would be 9÷24 = three-eighths completed. Therefore the range of unsmeared times would extend above and below 21:00:00 hours by 0.375 seconds. That is, the error radius is 0.375 TAI seconds.

Example 7: Determine the unsmeared UTC timestamp corresponding to the smeared timestamp (2017, 2, 9, 10, 0, 0, 0)

The date is outside the range of the leap table, but it is not an end-of-month date. Hence no leap-second correction is possible on the given date. Therefore, conversion to unsmeared timestamp results in two equal objects, each comprising a timestamp identical to the given timestamp (2017, 2, 9, 10, 0, 0, 0).

Example 8: Determine TAI time corresponding to smeared timestamp (2015, 8, 1, 20, 0, 0, 0)

As of the date of the given timestamp, the leap table indicates that 36 leap seconds have been added since the first one. No leap seconds have been added on the date of the given timestamp. Thus TAI time is exactly 36 seconds ahead of the given smeared timestamp. Conversion to TAI will result in two equal objects, each comprising the timestamp (2015, 8, 1, 20, 0, 36, 0).

Example 9: Determine the UTC time corresponding to TAI timestamp (2017, 8, 1, 20, 0, 0, 0)

The date in the timestamp is beyond the range of the leap table that indicates leap-second corrections only until 30-Dec-2016. Between the end-date of the leap table and the given timestamp, there are eight days - 31-Dec-2016, 31-Jan-2017, 28-Feb-2017, 31-Mar-2017, 30-Apr-2017, 31-May-2017, 30-Jun-2017, and 31-Jul-2017, on which leap second adjustments may be made. Thus the maximum number of leap second additions (or subtractions) until timestamp is 8. Until the end-date of the leap table, there have been 36 leap-second corrections. Therefore, until the date of the timestamp, there are between 36+8=44 and 36−8=28 leap seconds. Thus, conversion of the given TAI timestamp to UTC results in two objects that comprise timestamps that are respectively 28 and 44 seconds behind the given TAI timestamp. The two objects comprise the timestamps (2017, 8, 1, 19, 59, 32, 0) and (2017, 8, 1, 19, 59, 16, 0).

CONCLUSION

Techniques of this disclosure enable conversion between unsmeared, e.g., UTC or TAI, and leap-smeared times. A computer application that requires high sub-second accuracy can use a leap-smeared representation of time. Computer systems thus benefit from the advantages of leap-smeared representation, while retaining compatibility with applications that require sub-second time accuracy. A conversion library per this disclosure enables mapping leap-smeared time to UTC and TAI time.