(Preprint) AAS 13-507

VOCABULARY FOR TIME-SCALES

Russell O. Redman*

Much of the discussion regarding the future of Coordinated Universal Time (UTC) has focused on a "Yes / No" question on whether we should continue to insert leap seconds into UTC to keep it synchronized with the rotation of the Earth. Closer analysis of the problem reveals several other options, which can be explained using some proposed new vocabulary to dissect the concept of a time-scale into its component parts.

INTRODUCTION

The proposal to suppress the introduction of new leap seconds in UTC has been stalled for over a decade. It is clear that some applications would benefit from using a leap-second-free time-scale to ensure predictable behavior in an increasingly networked world, while other applications depend upon the close approximation of UT1 that UTC currently provides. Beyond the purely technical issues, there are also political, cultural and even religious issues that need to be addressed. To understand how these non-technical issues have been dealt with (or not), it may be useful to review the regulatory environment in which the proposed change is being considered.

The definition of UTC was formulated and adopted by the International Telecommunication Union, Radiocommunication Sector (ITU-R) in Recommendation TF.460†, because of the role played by radio signals in the coordination and transmission of national time scales. The definition of UTC remains a responsibility of the ITU-R. There are 193 administrations that are members of ITU-R, of whom 55 maintain time services that participate in the coordination of UTC. Changes in the definition of UTC require the approval of the member administrations at a World Radiocommunication Conference (WRC).

ITU-R is a voluntary organization and individual administrations remain sovereign within their own borders, so controversial changes in fundamental systems must normally reach a high degree of consensus before being accepted. Although it rarely happens, it is possible for administrations to ignore ITU-R recommendations if they are unhappy with the result of a vote on a difficult issue. In addition, for those administrations that include ITU-R TF.460 in their law codes by reference, a change in the definition of UTC would take effect immediately upon acceptance by a WRC, making it important to understand the consequences of the proposed changes before requesting their consent.

ITU-R delegates the discussion of proposed changes to study groups, and within each study group to a working party of technical experts. The proposal to suppress leap seconds is assigned

* Herzberg Astronomy and Astrophysics Programs, National Research Council of Canada, 5071 West Saanich Road, Victoria, BC, Canada V9E 2E7.

† Available from http://www.itu.int/rec/R-REC-TF.460/

to Study Group 7 (SG7), Working Party 7A (WP7A). These committees normally consider only technical issues; nontechnical issues are dealt with during the WRC after the technical committees have either achieved a consensus or demonstrated that a decision about the proposed changes is needed urgently.

Despite over ten years of discussion of the proposal to suppress leap seconds within UTC, neither WP7A nor SG7 could achieve the required consensus prior to the last WRC in 2012, and discussions at the Radiocommunication Assembly that preceded the WRC showed a level of disagreement amongst the participating administrations too high to predict the outcome of a vote, with powerful and technically sophisticated administrations advocating both sides of the question. The issue was referred back to SG7 and WP7A in Resolution 563 of WRC-12, but with a modified mandate that now includes the clause, "to consider the feasibility of achieving a continuous reference time-scale, whether by the modification of UTC **or some other method**" (emphasis added)[1*]. In preparation for the next WRC in 2015, SG7 and WP7A are now tasked with providing a set of technically valid options for the future handling of leap seconds, with proper estimates of associated costs and benefits.

The International Organization for Standardization (ISO), through its Technical Committee 37 (ISO/TC 37, whose scope is "Standardization of principles, methods and applications relating to terminology and other language and content resources in the contexts of multilingual communication and cultural diversity") has submitted a contribution to WP7A. ISO/TC 37's contribution objects to the proposal as it stands, in particular to the proposal to change the physical significance of the term UTC without changing its name. This would result in the term "UTC" becoming polysemic, i.e. a term with more than one meaning within a specific technical field. ISO forbids the use of such terms in all its standards. While the rules used internationally by ISO may not be binding on ITU-R, their advice on such issues should carry considerable weight and it would clearly complicate the further development of the field if ISO and those subscribing to its international standards were prevented from using ITU-R terminology.

To help find a solution acceptable to all parties, ISO/TC 37 suggests that, "it would be useful to develop a 'concept system' which would explicitly clarify the meaning and relationships between the various time scale protocols." Within the ITU-R system, the natural repository for such a "concept system" is the glossary defined in ITU-R TF.686-2[†].

To help break the impasse at ITU-R, to explore the options allowed by the expanded mandate for WP7A, and to clarify the terminology, it is important to analyze the technical issue in more detail. A contribution to WP7A has been prepared giving the result of one such effort as a set of proposed changes to the glossary, adding new terms and modifying existing terms in the glossary that are related to time keeping. This document will discuss some of the most important changes that are being proposed, dissecting the definition of a "time-scale" into its component parts and re-examining the distribution of different time-scales to help understand what is failing and how it might be fixed. Better language to describe the technical aspects of the subject might also allow us to address the non-technical issues more convincingly.

In the following text, the proposed definitions of new terms have been copied from the original contribution submitted to WP7A. In many cases the copied text is abbreviated, omitting parts that would be repetitive. In a few cases, the text has been modified to improve its clarity in this

[*] See also agenda item 1.14 and Resolution 653 (WRC 2012) in the preliminary agenda for WRC-15 available from http://www.itu.int/oth/R1201000001/
[†] Available from http://www.itu.int/rec/R-REC-TF.686/

context. It should be borne in mind that the proposed changes are still under discussion and are likely to be modified further before acceptance by ITU-R; a precise quote of the original proposal is not essential for the current purpose, which is to introduce the ideas for wider discussion. These blocks of text are indented in the text with clauses bulleted using filled circles, but are not enclosed in quotation marks.

CONTEXT FOR THE VOCABULARY

In the past, astronomy was intimately involved in the definition and determination of time. The development of atomic clocks that are far more stable than the rotation of the Earth, and of time signals distributed by radio whose propagation can be characterized with steadily improving precision, has rendered most astronomical approaches obsolete for precision time-keeping in the near-Earth environment. Correspondingly, the terminology appropriate to astronomical time keeping has been evolving to meet requirements that no longer directly influence developments in precision time keeping.

The context for the terms defined in the ITU-R glossary is the science, technology and regulation of radio communications. Time-related terms occur in all subfields of this subject and must be defined to be widely applicable, but the terms under discussion in this document are specifically related to the coordination of atomic clocks in the UTC system. The existing terminology has been motivated by the communication of time signals between real clocks, which can be characterized with considerable precision and verified experimentally. This approach keeps the vocabulary firmly grounded in the actual process of time keeping and in the sound practices of metrology.

ITU-R terminology is used for legal purposes by many administrations and needs to be free from the risk that changes made by non-governmental organizations for their own valid purposes will not inadvertently change the legal interpretation of terms defined by the ITU-R. Following these considerations, the new or modified definitions for terms discussed in this document will, to the extent possible, avoid references to concepts outside the framework of clocks, time-scales and time signals.

PHYSICAL TIME

Historically, the most fundamental unit of time in most people's lives has been the day, and the original time-scales were intended to divide the day unto useful parts. The astronomical basis of all time-scales based on the day is the orientation of the Earth in space. Today, this is most conveniently summarized through the time-scale UT1.

Following the long traditions of western time keeping, each day was divided into 24 hours, with 60 minutes per hour, and 60 seconds per minute, giving 86,400 seconds per day. As a physical unit, the length of the second proved to be more practical than the day, so that the fundamental unit of time in the International System of Units (SI) is the second, currently defined as the duration of 9 192 631 770 periods of the radiation corresponding to the transition between the two hyperfine levels of the ground state of the cesium 133 atom. This change from an astronomical to a physical definition of the second became critically important with the advent of atomic clocks in the 1950's that were far more precise than the rotation of the Earth. Variations in the length of the day over the course of a year are now hundreds of thousands of times larger than the precision with which the start of each second is known using state of the art atomic clocks.

With the extreme precision provided by atomic clocks, gravitational time dilation is an important correction term, so the passage of time must be considered in a general relativistic context. In principle, general relativity allows space-time to be described using arbitrary coordinate patches, but for practical time keeping in the vicinity of the Earth it is useful to construct a "coordinate time" that provides an approximation of Newtonian time for slowly moving observers by taking into account the timing consequences of both speed and accelerations (including gravity). Conceptually, within general relativity a coordinate time has an origin and will be represented by a (mathematical) real number.

There is one particular coordinate time that is used by all of the atomic clocks in the UTC system. This important concept has not previously had a name within the documents that define UTC, perhaps because no need was perceived to distinguish time-scales like UTC or International Atomic Time (TAI) from their underlying physical coordinate time. (This might be construed as polysemic use of the terms UTC and TAI.) To disambiguate the meanings of these terms, a new term is proposed:

- *REFERENCE COORDINATE TIME* (RCT): A general relativistic coordinate time defined as the proper time measured by clocks embedded on the rotating geoid and propagated to correct for the timing consequences of speeds and accelerations (including gravity) within a given approximation. The physical unit of RCT is the SI second on the rotating geoid.

Thus, an ideal clock that is stationary with respect to the rotating geoid will find that its proper time matches RCT if it is located on the geoid, but will run faster or slower than RCT if it is above or below the geoid, respectively. Actual clocks are rarely located precisely on the rotating geoid, so it is necessary to correct the clock rate to account for the different gravitational potential. As with all such coordinate times, RCT is conceptually a mathematical real number with a specifiable origin (zero point). It is useful to note that this definition does not specify any particular origin for RCT. Every clock will use its own origin internally, which can be changed by restarting the clock. For many, but not all, applications it will be important to specify the origin using an externally defined time-scale like UTC or TAI. In a similar vein, we can distinguish the general concept of "distance along a line from an origin measured in metres" from the more particular concept of "measurement using a particular meter stick with the origin at one end".

This definition of RCT allows us to recognize it as a common property of UTC, TAI and the proposed new time-scale International Time (TI), and helps us to distinguish these from older time-scales like UT1 that use a different basis for time.

REPRESENTATION OF TIME

Within ITU-R, the generic term for a representation of time is a time code:

- *TIME CODE*: A system of digital or analogue symbols used in a specified format to convey time information i.e. date, time of day or time interval.

Common examples of time codes include:

1. binary, *e.g*. as a floating point number, or a pair of integers giving a count of seconds and fractions of a second
2. character string, e.g. the formats defined by ISO 8601
3. date-time structure, dividing the time into a set of fields representing different units, such as year, month, day, hours minutes, seconds
4. rising or falling voltages in a timing circuit

The latter example is included to emphasize the generality of the concept of a time code, which can be almost any measureable quantity that changes with time. It is quite normal for time codes to have a limited range of validity defined by starting and ending times, such as the interval between when a timing circuit is triggered and the moment the voltage crosses a threshold that stops the circuit. Time codes do not have to be permanently unique; a North American wall clock typically provides a time code that is only unique within a twelve-hour period and cycles twice each day. The examples also emphasize that time codes often can be grouped into useful families suited for different purposes.

The time codes of greatest interest here normally divide time into some kind of date-time structure. This introduces the new term:

- *DATE-TIME CONVERSION ALGORITHM*: An algorithm that describes how to convert a coordinate time into a date-time structure, which often contains fields for years, months, days, hours, minutes, seconds and fractions thereof.

Ideally, a date-time conversion algorithm should be unambiguous, independent of the time at which it is invoked, and bi-directional, such that converting the date-time structure to coordinate time and back returns the same values that were present in the original date-time structure. A generic statement of the algorithm will normally treat the coordinate time as a mathematical real number. Note that binary time codes also have an associated date-time conversion algorithm, although they may not divide the time of day into hours, minutes and seconds.

The date-time conversion algorithm is an important property of a time-scale. TAI and UTC differ primarily in their choice of date-time conversion algorithm. Before discussing the algorithms appropriate to these two time-scales, it is important to understand in some detail what a leap second is and how it is implemented. Much of the conventional terminology about leap seconds is misleading and in need of clarification. The following modification of the definition is proposed:

- *LEAP SECOND*: An intentional change in the number of seconds per minute, to extend a designated minute by one extra second (a positive leap second) or to finish the minute early by one second (a negative leap second). The leap second is used to adjust coordinated universal time (UTC) to ensure approximate agreement with UT1 when both are expressed in days/hours/minutes/seconds.

Figure 1 shows the progression of seconds across the end of a minute for a normal minute without a leap second, with a positive leap second and with a negative leap second, as described by ITU-R TF.460-6. The last second in each minute is highlighted with a grey background. A decimal count of RCT seconds is shown (with an arbitrary starting epoch) in the left hand column, to emphasize that RCT increments by one second for each line.

RCT	Normal	Positive	Negative
123.0	:57.0	:57.0	:57.0
124.0	:58.0	:58.0	:58.0
125.0	:59.0	:59.0	:00.0
126.0	:00.0	:60.0	:01.0
127.0	:01.0	:00.0	:02.0

Figure 1. Positive and Negative Leap Seconds.

Although it can be misleading, it is conventional to describe a positive leap second as “inserting an extra second”, and a negative leap second as “omitting a second” from the affected minute. (This language is found explicitly in ITU-R TF.460-6.) These descriptions contrast the behavior of UTC as a time-scale from time-scales like TAI that uniformly have 60 seconds per minute. An essential point is that leap seconds affect only the labeling of time by a date-time conversion algorithm that breaks the time of day into hours, minutes and seconds. In a simple decimal or binary count of seconds that correctly represents RCT over decades, no second would ever be be omitted and no second would appear twice.

With this understanding of the operation of a leap second, it is possible to describe clearly the two most important date-time conversion algorithms in common usage.

- *UNIFORM DATE-TIME CONVERSION ALGORITHM*: An algorithm that converts a coordinate time to years, months, days, hours, minutes seconds and fractions thereof, assuming there are uniformly 60 seconds per minute, 60 minutes per hour and 24 hours per day, with the count of days being converted to years, months and days according to a standard (e.g. Gregorian) calendar.
- *UTC DATE-TIME CONVERSION ALGORITHM*: An algorithm that converts reference coordinate time into years, months, days, hours, minutes, seconds and fractions thereof, using 59, 60 or 61 seconds per minute as specified by ITU-R TF.460 and whenever required by IERS, but with 60 minutes per hour, 24 hours per day, and with years, months and days calculated using a standard (e.g. Gregorian) calendar.

The uniform date-time conversion algorithm is the correct algorithm to use for time-scales that uniformly have 60 seconds in every minute like TAI, or UT1 where its seconds are understood as UT1-seconds. The UTC date-time conversion algorithm should be used always and only for UTC.

A large body of software has been written that incorrectly uses the uniform date-time conversion algorithm to convert UTC to and from a binary "count of seconds" that nominally starts at some epoch. This trap is easy to fall into, and the author is aware of several times in his own career where he has written code that assumed every minute has exactly sixty seconds. It should be emphasized that historically this practice was well justified; in an era when most system administrators set the system clocks by hand, rebooted machines regularly, and networked only small groups of machines within a building, it was hardly necessary to include complex code to handle leap seconds that occurred once every year and a half on average. The urgency of the issue today reflects the rising demand for precision and synchronization by organizations that operate on a global scale and are automating ever more important parts of their operations.

Normal	Positive	Negative
:58.0 345.0	:58.0 345.0	:58.0 345.0 (346.0?)
:59.0 346.0	:59.0 346.0	:00.0 347.0 (346.0?)
:00.0 347.0	:60.0 omitted	:01.0 348.0
	:00.0 347.0	

Figure 2. UTC with Uniform Date-Time Conversion Algorithm.

It is therefore important to consider the consequences of using the uniform date-time conversion algorithm for UTC time calculations. Figure 2 illustrates how the underlying "count of seconds" increments when it is calculated using the uniform date-time conversion algorithm. For purposes of illustration, the count of seconds is assumed to have the value 345.0 for second 58.0 within the selected minute. Using the uniform date-time conversion algorithm, second 0.0 of the following minute will *always* have the calculated value 347.0. During a normal minute, the count of seconds increments by one second per second, as expected. With a positive leap second, the uniform date-time conversion algorithm has no ability to represent second number 60.0 within the minute, so that second is omitted from the sequence; the count resumes with the value 347.0 after the leap second. For a negative leap second, the minute terminates one second early, so the calculated value jumps to 347.0 without ever encountering second 346.0. If the value 346.0 is converted to a UTC time, it is ambiguous whether the second following 345.0 is intended or the second preceding 347.0, and this ambiguity can have unpredictable consequences for time-critical code. The “count of seconds” with this scheme has gaps—omitting every positive leap second—and introduces a non-physical second at every negative leap second (fortunately, there have been none to date), and so does not represent RCT in any interval of time that includes a leap second.

It is often explained that the leap second causes the entire time-scale to shift by one second, but if the uniform date-time conversion algorithm is used consistently this is wrong. If a binary time stamp for second 345.0 is generated and stored using the uniform algorithm and the stored value converted back to UTC with the same uniform algorithm, the original UTC time will be recovered regardless of whether there has been an intervening leap second, so the time-scale does not shift at all. The “count of seconds” that results from the misapplication of the uniform date-time conversion algorithm to UTC may have a gap or non-physical second whenever a leap second occurs, but if used consistently for archival purposes the uniform date-time conversion algorithm is able to set, record and interpret its count of seconds (leap seconds excluded) as accurate time in the UTC system: except in the interval between the start of a leap second and the completion of the clock’s leap second correction it will be able to generate a unique, permanent and correct UTC-label.

A variety of ad hoc workarounds have been developed over the years to try to manage the problems caused by the use of an inappropriate date-time conversion algorithm. Inconsistency in the definition and management of these workarounds is a major source of complexity and operational problems that are incorrectly attributed to leap seconds.

Returning to the representation of time, it is now possible to provide a definition of the term time-scale that should allow further progress:

- *TIME-SCALE*: A family of time codes for a particular coordinate time that provide an unambiguous time ordering of events. A time-scale has five main properties:
  - the underlying **coordinate time**
  - a particular implementation of the underlying coordinate time by a **named clock or time service**, having a specifiable uncertainty
  - a **date-time conversion algorithm** (ideally bi-directional)
  - a **range of time** in which the date-time conversion algorithm is well-defined, with a mandatory starting epoch, and (optionally) an ending epoch after which the time-scale may be undefined or ambiguous
  - a **starting value** for the time-scale at the starting epoch

The definition of a time-scale as a family of time codes allows enormous generality; anything that can be used as a time code can define a time-scale. Recognizing that a time-scale is a family of time codes allows us to distinguish different families of time codes by their defining properties. The family of time codes for a system as fundamental as UTC is very large, but should not be extended arbitrarily. Always and necessarily, UTC divides the time of day into hours, minutes and seconds using the UTC date-time conversion algorithm, without which it would be impossible to define leap seconds.

Using this language we can now analyze UTC:

1. RCT is the coordinate time.
2. For precise work, the name of the time service (clock) providing the signal should be given in brackets, with the same codes used in BIPM (International Bureau of Weights and Measures) Circular T, e.g. UTC(NRC); for generic statements about the time-scale or low precision times it is acceptable to omit the name of the service.
3. The UTC date-time conversion algorithm as specified in ITU-R TF.460 must be used.
4. UTC date-times are well-defined in the interval starting 1972-01-01T00:00:00.0 and continuing up to the next opportunity to insert a leap second (nominally at the end of every month but with first preference at the end of June or December and second preference at the end of March or September) that IERS has not already scheduled to have or not have a leap second.
5. The starting value for UTC at the starting epoch is 1972-01-01T00:00:00.0.

A variety of terms have been used informally to describe time-scales that do not use leap seconds. They have sometimes been described as "purely atomic", although UTC is also generated by atomic clocks and the same servers can generate time signals for both UTC and any of the "purely atomic" time-scales. Another common term has been "continuous" although there are no discontinuities in UTC, which continuously and consistently labels every second. Both of these terms can be misleading, so this document urges the definition and use of the term:

- *UNIFORM TIME SCALE*: A time-scale that uses the uniform date-time conversion algorithm to convert the underlying coordinate time into a date-time structure comprising years, months, days, hours, minutes, seconds and fractions thereof. A uniform time-scale always has sixty seconds per minute.

Examples of uniform time-scales include TAI, UT1 and the proposed new time-scale TI. UTC is explicitly NOT a uniform time-scale.

It is useful also to be able to distinguish those time-scales that are based on RCT from those that are not, while avoiding the existing term "reference time-scale" that simply means a time-scale that is used as a reference for some purpose:

- *REFERENCE COORDINATE TIME SCALE*: A time-scale that represents the reference coordinate time (RCT), can be generated by a real clock, can be distributed using one or more time codes, and from which other time-scales can be derived.

Within the ITU-R glossary, the generic term for the value of a time-scale as measured at a particular instance is a "time-scale reading." It is proposed to add to the glossary the term "time stamp" as a particular kind of time-scale reading that is commonly used in computing:

- *TIME STAMP*: A time stamp is a digital representation of a time-scale reading that records, with a specifiable accuracy, the instant that an event occurred. Ideally, a time stamp should be unambiguous and permanently valid, i.e. always refers to the same instant in time. Time stamps can be stored for later reference or packaged for transmission.

DISTRIBUTION OF TIME

Part of the long running debate about UTC is the issue of whether the time-scale TAI is necessarily the time-scale that is generated by BIPM and distributed after the fact through Circular T, or whether it is a time-scale equivalent to UTC that can be generated by any clock that generates UTC and can be distributed as a standard time service.

ITU-R TF.460-6 explicitly states that the value of TAI differs from UTC by the integer number of leap seconds that have been introduced up to that date, so the algorithm is quite straightforward to convert a value of UTC to/from TAI, or UTC(*k*) to/from TAI(*k*). In ITU-R TF.460-6, ANNEX 1, Section E, the quantity DTAI = TAI − UTC is defined and clearly stated to be a quantity that can be "disseminated with time signals", making it available for use by anyone who receives the time signal to calculate TAI from the distributed value of UTC. A long-standing position of the CCTF (Consultative Committee on Time and Frequency), which reports to the CIPM (International Committee on Weights and Measures) that oversees BIPM, is that TAI—realized by time laboratory *k* as TAI(*k*)—may be made available and should be recommended for use in applications that need a uniform time-scale.

Synchronizing clocks over a global network would be such an application. We note in passing that the Precision Time Protocol (PTP), unlike the more widely used Network Time Protocol (NTP), explicitly uses a representation of TAI during transmission to avoid the ambiguities introduced by the current, flawed implementations of leap seconds in NTP. Recall also that, outside the hours-minutes-seconds format, any binary or decimal count of seconds in UTC that validly represents RCT is identical to the count of seconds in TAI.

The fact remains that relatively few applications use TAI in this way, and it may be that common usage is making the identity TAI ≡ TAI(*BIPM*) normative, so that in the near future any other use might be deprecated.

To bring some clarity to this discussion, it is useful to introduce the new term:

- *TIME SIGNAL*: A time signal is a process that creates and passes time stamps through some medium with a transit delay that can be characterized. Examples include the radio signals used to broadcast UTC, and the protocols used to distribute time over the internet.

This term, of course, merely makes formal a usage that has existed for some time, and makes explicit the importance of metrological considerations in the distribution of time. Elaborating this point, the most useful time signals are traceable, a term borrowed directly from metrology and widely used within ITU-R but in this proposed new term specialized to the requirements of the coordinated time services:

- *TRACEABLE TIME SIGNAL*: A traceable time signal has a specifiable uncertainty relative to a reference time-scale. Traceability can be extended along a chain, taking account of the uncertainties introduced during transmission.
- NOTE 1 – Although the time signal is generated in real time, the reference time-scale can be generated after the fact, e.g. UTC

The note raises an important point that may be unique to the field of time keeping. Even if a time signal is generated in real time, the reference time-scale can be generated after the fact; for a national time service *k*, UTC(*k*) is traceable to UTC*(BIPM),* but UTC*(BIPM)* is only known after the fact and is published through Circular T each month. The global acceptance of UTC owes much to the traceability of signals from the national time services back to UTC*(BIPM),* which ensures that they all broadcast the same time within known uncertainties.

These concepts allow us to specify precise conditions for a time-scale to be "distributable":

- *DISTRIBUTABLE TIME SCALE*: A distributable time-scale is a specific real-time implementation for creating time signals that are traceable to a source of reference coordinate time. UTC*(BIPM)* is not a distributable time-scale, but UTC(*k*) usually is.

A direct implication of this definition is that TAI*(BIPM)*, like UTC(*BIPM*), is not a distributable time-scale, but that TAI(*k*) from the time service *k* usually is because it meets exactly the same conditions as UTC(*k*).

BINARY TIME-SCALES AND TIME DISTRIBUTION

In practice, many of the protocols used to distribute time do not use distribute UTC directly. Rather, the server broadcasts a binary time stamp that typically encodes the current time as a small set of integers that count seconds and fractions of seconds, or something equivalent. Upon receiving a packet, clients will use the time stamp to adjust the system clock, which itself normally represents the time as a binary structure. It is only when the time needs to be displayed or used by an application program that we would expect the time to be converted into a date-time structure and/or character string.

As discussed above, binary representations of time that do not break the time of day into hours, minutes and seconds cannot be considered members of the UTC family of time codes. This raises an interesting question of what we mean when we speak of the distribution of UTC. In the terminology developed here, we are actually distributing a binary time-scale, which can be converted by client software into any other time-scale, including UTC but also including UT1, and any of the time zones used for civil time at the discretion of individual administrations.

The current set of proposed changes to the ITU-R glossary does not include any terminology specific to binary time-scales. This is primarily because there are only a few protocols that implement the binary representation of RCT correctly. PTP does and many web clocks transmit decimal counts of seconds, milliseconds or microseconds, but NTP does not when configured to distribute UTC, nor does Microsoft Windows, nor any Posix-compliant operating system. (NTP does represent RCT correctly when configured to distribute a uniform time-scale like TAI or the GPS time-scale.) In the same way that the IEEE Standard for Floating-Point Arithmetic (IEEE 754) allowed floating point arithmetic to be done consistently on different processors and allowed floating point data to be transmitted without corruption, it might be valuable to define a standard for the binary representation of RCT. The adoption and implementation of such a standard might remove much of the trouble that currently attends the distribution of time. Such a project lies outside the mandate of ITU-R, but a commitment to develop such a standard might be part of a comprehensive solution to the problem.

With the recognition that the distribution of time normally uses binary time-scales, the discussion about whether we should distribute UTC versus some uniform time-scale loses much of its force. So long as the distribution protocol transmits a faithful representation of RCT, client systems and application programs can convert the signal into any desired time-scale. For example, the Global Positioning Satellite (GPS) system can be used to set UTC clocks, but internally uses its own proprietary uniform and binary time-scales.

The recognition that client software is free to convert a time signal from any reference coordinate time-scale, including a binary time-scale, into any other desired time-scale on demand changes the nature of the discussion. It does not matter whether the packets created by a particular protocol are intended to encode UTC, TAI or a binary time-scale, provided only that they encode a valid representation of RCT. Client software can be, and currently is being, used to generate date-time structures appropriate for its own application domain. If we fix the operating system libraries and the network transmission protocols to properly represent RCT, and encourage clients to use time-scales appropriate to their applications, many of the problems in the current system will vanish.

This is not a small chore, of course, and would probably require an investment comparable to the change in the Internet protocols from IPv4 to IPv6, but the latter transition has been recognized to be necessary and is in progress at the time of writing.

CONCLUSIONS

A set of new and modified terms has been proposed for the ITU-R glossary, which should clarify discussions about suppressing or retaining leap seconds in UTC. New terms have been introduced when necessary. Where an existing term has become ambiguous, care was taken to clarify the term so that the former usage would not be affected, introducing new terms when needed to make the necessary distinctions.

The most important new definition is “reference coordinate time”, which conceptually is the physical time that measured by clocks within the Coordinated Universal Time system. UTC provides a complete, consistent, continuous representation of RCT, as does TAI and the proposed new time-scale TI. UTC does not, however, provide minutes that uniformly contain 60 seconds, allowing additionally 61 seconds per minute (with a positive leap second) or 59 seconds (with a negative leap second).

The properties of a “time-scale” are analyzed, and include:

- the underlying coordinate time
- the name of the clock or time service that generates the time signal
- the date-time conversion algorithm
- the time interval within which the time-scale is valid and/or unambiguous, defined by the starting epoch and (optionally) the ending epoch
- the starting value for the time-scale at the starting epoch

The difference between UTC and TAI lies primarily in the choice of the “date-time conversion algorithm” and the limited time interval into the future for which UTC is unambiguous. In principle, UTC should always use the “UTC date-time conversion algorithm” as specified by ITU-R TF.460-6, and is only unambiguous before the next opportunity to insert a leap second for which IERS has not expressed an intention.

In practice, many important and widely used protocols in computer operating systems and network time protocols mistakenly use the “uniform date-time conversion algorithm” to convert UTC times between a binary representation and a date-time structure. This results in anomalous behavior that developers have attempted to work around with many complex routines (such as changing the system clock rate for arbitrary amounts of time before and/or after the leap second). Maintaining this system is complex and error prone, which is the proximal cause of many of the problems currently associated with the use of leap seconds.

The long debate on whether to continue to insert leap seconds in UTC or to suppress them indicates considerable unease about both the “yes” and ”no” answers to this question. The analysis that led to this paper suggests a different approach to fixing the existing system:

- New libraries of time handling software should be written that correctly implement the UTC date-time conversion algorithm. Time stamps generated by the new library should represent RCT, perhaps following a new standard for binary time-scales. The existing libraries that implement the uniform date-time conversion algorithm can be repurposed for TAI or TI with minor changes in the naming of structures and routines.

- Operating systems should be modified to allow system managers to configure them to use the new libraries.
- Network transmission protocols that do not already do so should be modified to use the same set of reference coordinate binary time-scales. It should be explicitly recognized that the modified protocols would not distribute either UTC or TAI directly, but that the signal can be converted by client software to either of these, or any other desired time-scale, on demand.
- Client software should be encouraged to use uniform time-scales like TAI or TI whenever long-term predictability and/or unambiguous interpretation of time is more important than a close connection to the orientation of the Earth.
- UTC with its current definition can continued to be used whenever it is appropriate to the client's purposes. Critical, safety-related equipment with built in receivers for UTC time signals would probably require these signals to continue to operate for many years. On a larger scale of concern, the daily lives of most citizens of the Earth depend upon the Sun, and political acceptance of the modified system will be much easier if the current definition of UTC is preserved so it can be used as a sun-synchronized reference time-scale from which civil time-scales can be offset using time zones. The global acceptance of UTC with its current definition indicates that it meets all of the political, cultural, and religious requirements of the world community, and such consensus should not be discarded lightly.
- UTC can continue to be the "referential time-scale", or this designation could be transferred to one of the uniform time-scales such as TAI or TI, with the understanding that all of them represent RCT and can be converted to any other reference coordinate time-scale on demand.

REFERENCES

[1] Final Acts WRC-12, ISBN: 9789261141417, (International Telecommunication Union Radiocommunication Sector, Geneva)