**Migrating DICOM Dates and Times**

This document discusses a new approach to DICOMs Date(DA), Time(TM), and Date-Time(DT).

Issues:

1. Does DICOM have a gps coordinate element? It doesn't seem to.
2. How to handle Dates and Date-Times without time zones in a Dataset without a TimezoneOffsetFromUTC?
3. Which is better TimePoint, Timepoint, Timestamp…
4. Synonym for duration

## Problems with DICOM Dates and Times

* Dates, Times, and DateTimes don't support separator characters, and can't be formatted as internet Dates and Times.
* Times can be underspecified requiring only the hours digits.
* DateTimes can be underspecified requiring only the year digits.

# Current Handling in Open DICOMweb

## Time Zone

All Date(DA) and Date-Time(DT) elements are specified in the local time zone, if known; otherwise the Date-Time is assumed to be in UTC.

If the Dataset has a TimezoneOffsetFromUTC(0008,0201), then:

1. All Date(DA) elements in the Dataset have that time zone.
2. Date-Time(DT) elements that do not have a time zone component have that time zone.
3. Date-Time(DT) elements that have a time zone component are in the time zone specified by the component.

It is a ***Best Practice*** for all Datasets to contain a TimezoneOffsetFromUTC element.

## Date(DA)

The Date Element is currently a String; however, it has two Getters called:

* local
* utc

that return a Dart Date-Time object, which is encoded as the number of microseconds since the Unix Epoch (January, 1, 1970).

## Time(TM)

The Time(TM) Element is currently a String; however, it has a Getters called:

* asDuration

that returns a Dart Duration object.

## Date-Time(DT)

The Date-Time(DT) Element is a String; however, it has two Getters called:

* local
* utc

Each returns a Dart Date-Time object, which is encoded as the number of microseconds since the Unix Epoch (January, 1, 1970) in local or UTC time.

## Combining Date(DA) and Time(TM)

A Date and Time may be combined to form a Date-Time(DT) using the add method.

DateTime dt = date.add(time)

The resulting Date-Time(DT) will have the same time zone as the Date(DA).

# Proposed Enhancement –Time Point(TP) and Duration(DU)

It is proposed to add two new Value Representations(VR) to DICOM:

TODO: talk about support for Unix Epoch relative times and durations in all system today.

## Time Point(TP)

A Timepoint is a binary encoding of a point in time. For example, the point in time when a procedure started. This is like Date-Time, but is encoded as a 64-bit integer that is relative to the Unix Epoch (1970-01-01). It has the advantage that it requires no parsing, and is supported by all current computer systems[ref].

## Duration

A Duration is a binary encoding of an amount of time. For example, the time it took to perform a Study. This is like Time(TM), but is encoded as a 64-bit integer in microseconds. It has the advantage that it requires no parsing, and is supported by all current computer systems[ref].

TODO: talk about support for Unix Epoch relative times in all system today.

## Using Time Points and Durations

Time Points support the add method:

Suppose we have a starting Time Point start and a Duration d, then

TimePoint endTime = start + d

…

# Converting Dates and Times to Time Points

It is proposed to add new Time Point elements that correspond to all Date and Time element pairs. For example:

InstanceCreationDate() and InstanceCreationTime()

-> InstanceCreationTimePoint

And

StucyDate(0008, 00) and StudyTime(0008,00) will have a corresponding StudyTimePoint(0008,eeee)

Elements with start and end times

Dates and Times

* (0008,0201) TimezoneOffsetFromUTC
* DA -> TP at midnight on the date in the TimezoneOffsetFromUTC
* TM -> Duration
* DT -> TP
* New VR
  + TP – Binary Time – 8-bytes UTC
  + DU – Binary Duration 8-bytes

1. Converting DA to TP
2. Converting TM to DU
3. Converting DA + TDates, Times are in the Time Zone of the Element kTimezoneOffsetFromUTC
4. Date-Times if they have a Time Zone are in that time zone, if they don't have a time zone