

Re: Question about ISO8601 vs. strftime()

Source: <http://coding.derkeiler.com/Archive/General/comp.programming/2006-12/msg00656.html>

- *From:* Ian Woods <newspub3@wuggydotorg>
 - *Date:* Sun, 24 Dec 2006 08:02:27 -0600
-

Randy Howard <randyhoward@xxxxxxxxxxxxxxxxxxxx> wrote in
<news:0001HW.C1B3555300346B9FF0203648@xxxxxxxxxxxxxxxxxxxx>:

On Sat, 23 Dec 2006 21:27:29 -0600, Ian Woods wrote
(in article <Xns98A32331FB75Cnewspub3atwuggydotor@xxxxxxxxxxxxxxxxxxxx>):

So, my question is, with respect to ISO 8601, are both forms valid? Or is the strftime() one actually correct, and a slew of online references to ISO8601 just get it wrong? I don't have official ISO 8601 documentation, just references to it via the web.

I don't have a copy of the real ISO 8601 either. AFAIK, both the forms with and without separators are allowed and equivalent. This certainly seems to be the case from a number of sources including:

<http://www.cl.cam.ac.uk/~mgk25/iso-time.html>

Unless I missed something, that link only shows the form with the UTC offset without the ':' separator. It shows regular time such as hh:mm, but not the offset as far as I can tell.

It does say:

"Time zone

Without any further additions, a date and time as written above is assumed to be in some local time zone. In order to indicate that a time is measured in Universal Time (UTC), you can append a capital letter Z to a time as in

23:59:59Z or 2359Z

Re: Question about ISO8601 vs. strftime()

Note: The Z stands for the "zero meridian", which goes through Greenwich in London, and it is also commonly used in radio communication where it is pronounced "Zulu" (the word for Z in the international radio alphabet). Universal Time (sometimes also called "Zulu Time") was called Greenwich Mean Time (GMT) before 1972, however this term should no longer be used. Since the introduction of an international atomic time scale, almost all existing civil time zones are now related to UTC, which is slightly different from the old and now unused GMT.]

The strings

+hh:mm, +hhmm, or +hh

can be added to the time to indicate that the used local time zone is hh hours and mm minutes ahead of UTC. For time zones west of the zero meridian, which are behind UTC, the notation

-hh:mm, -hhmm, or -hh"

But, there's no "full example" with the time zone in these formats that I can obviously see.

http://en.wikipedia.org/wiki/ISO_8601 (looking at the "formats" on the RHS)

Again, I don't see it used that way in the examples there for the UTC offset value. BTW, I tried editing in a comment about the inconsistencies between the wiki entry and the C99 standard description, and it was summarily expunged within a few minutes. Makes wiki suspect, when questions of technical accuracy are deleted, rather than clarified, especially when it happens so quickly, and nothing is said even on the discussion page.

It does say that:

"Other time zones are specified by their offset from UTC, in the format ±[hh]:[mm], ±[hh][mm] or ±[hh]."

But again, there's no "full example" with the time zone in these other formats. Unfortunately the one "combine representation" shown has all of the seemingly optional separators in.

This is why I showed a number of sources: any single source could be hilariously incorrect but it's less likely (but still possible) for a number of separate sources to all be wrong in the same way.

Re: Question about ISO8601 vs. strftime()

<http://hydracen.com/dx/iso8601.htm>

This link definitely shows several forms for the UTC offset, including the 'short form' I haven't seen before of hours only.

Since there are so many variations, it somewhat calls into question the value of using such a loose standard, as it doesn't exactly make it easier on the "consumer" of the data, which still has to know how to interpret all the variations. Oh well, I guess I'll stick with this specifier, which seems to match most platform utilities that attempt a date at least similar to ISO8601:

```
"%Y-%m-%dT%H:%M:%S%z"
```

That said, people seem to dislike the "T" notation for the time component, usually resulting in this variation being used instead:

```
"%Y-%m-%d %H:%M:%S%z"
```

Will the sign of the timezone produced by strftime() be correct for ISO8601?

Ian Woods

.