Subject: Re: hours since 1-1-1
Posted by Mark Hadfield on Wed, 27 Jul 2005 03:00:52 GMT
View Forum Message <> Reply to Message

qian wrote:
> Dear IDL users:
>
> Is there an IDL function/program that converts the unidata time unit to
> more readable format?
>
> For example: hours since 1-1-1 17338824 -> Jan 1 1979.

Oh no, not hours since 1-1-1 again! That must be the silliest ever
date-time origin ever invented for modern data! I mean, what calendar
were they using then and why should I have to care?

Anyway, IDL has the JULDAY function, which calculates time in Julian
days, ie time since 12:00 hours on 1 Jan 4713BC. (OK, I withdraw my
comment about 1 Jan 0001 being the silliest ever date-time origin.)

The confusing things you have to remember:

- JULDAY uses the weird month, day, year order for dates.

- The Julian date reaches an integral at 12:00 each day, not at
  00:00.

- JULDAY accepts optional hour, minute, second arguments. With them
  it returns the true Julian date-time as a double precision floating
  point number. Without them it returns the Julian date to be reached
  at 12:00 on the day in question as an integer. (Got that?)

Consider the following calculations

```
IDL> print, julday(1,1,1,0,0,0)
       1721423.5
IDL> print, julday(1,1,1979,0,0,0)-julday(1,1,1,0,0,0)
       722451.00
IDL> print, 24*(julday(1,1,1979,0,0,0)-julday(1,1,1,0,0,0))
       17338824.
```

It seems that JULDAY agrees with you and with Unidata, that
00:00 on 1 Jan 1979 was 722451 days == 17338824 hours after 00:00 on 1
Jan 0001.

IDL also offers the CALDAT function, to convert Julian date-times to
calendar date-times and the calendar format codes that let it express

real numbers as date-time strings (on the assumption they are Julian
date-times).

So to deal with "hours after 1-1-1" in IDl convert them to Julian
date-times immediately (divide by 24 then add 1721423.5D0) and then use
IDL's built-in facilities thereafter.

--
Mark Hadfield "Kei puwaha te tai nei, Hoea tahi tatou"
m.hadfield@niwa.co.nz
National Institute for Water and Atmospheric Research (NIWA)

Subject: Re: hours since 1-1-1
Posted by James Kuyper on Wed, 27 Jul 2005 04:40:53 GMT
View Forum Message <> Reply to Message

Mark Hadfield wrote:
...
> Anyway, IDL has the JULDAY function, which calculates time in Julian
> days, ie time since 12:00 hours on 1 Jan 4713BC. (OK, I withdraw my
> comment about 1 Jan 0001 being the silliest ever date-time origin.)

The epoch for the Julian Day system may seem silly, but it was
originally designed for use with ancient historical data. Its of
special interest to astronomers, who are interested in knowing exactly
how long ago an ancient astronomer observed a particular event. 4713 BC
is the year on which three different cycles associated with widely used
ancient calendar systems were all in sync. Of course, nobody was using
any of those calendar systems at that time, it's just a theoretical
connection based upon running those calendar systems backward until the
cycles are in sync. That synchronization makes it easier to calculate
the Julian day for each of those calendar systems, and therefore to
compare dates recorded in each of those systems with each other.

Subject: Re: hours since 1-1-1
Posted by Michael Wallace on Wed, 27 Jul 2005 21:32:34 GMT
View Forum Message <> Reply to Message

>> Anyway, IDL has the JULDAY function, which calculates time in Julian
>> days, ie time since 12:00 hours on 1 Jan 4713BC. (OK, I withdraw my
>> comment about 1 Jan 0001 being the silliest ever date-time origin.)
>
>
> The epoch for the Julian Day system may seem silly, but it was
> originally designed for use with ancient historical data.

Actually, I love Julian Day for one simple reason. No matter what data
I'm looking at, I never have to use negative numbers to represent the
date. You wouldn't believe what kind of hilarity ensues when you try to
work with dates as negative numbers.

I give the Silliest Epoch Award to Macintosh for their selection Jan 1,
1904 as their base epoch for MacOS. Apparently, they were originally
going to use Jan 1, 1900 which seems to be a logical choice, but then
someone remembered that 1900 was not a leap year. Every four years
after 1900 there would be a leap year until 2100. By not including
1900, they then didn't have to bother with coding those extra pesky leap
year rules that everyone always forgets. And to make things even
better, it was easiest for them to start their epoch on a leap year and
so 1904 became their epoch. Hopefully, no one will be running MacOS in
2100. Besides, the rest of us have to make it past 2038 first. :-)

-Mike

Subject: Re: hours since 1-1-1
Posted by R.Bauer on Thu, 04 Aug 2005 13:51:12 GMT
View Forum Message <> Reply to Message

kuyper@wizard.net wrote:

We prefer Julian Seconds which were initial defined by Ray Sterner at
JHUapl.

http://fermi.jhuapl.edu/s1r/idl/s1rlib/time/time.html

cheers

Reimar

> Mark Hadfield wrote:
> ...
>> Anyway, IDL has the JULDAY function, which calculates time in Julian
>> days, ie time since 12:00 hours on 1 Jan 4713BC. (OK, I withdraw my
>> comment about 1 Jan 0001 being the silliest ever date-time origin.)
>
> The epoch for the Julian Day system may seem silly, but it was
> originally designed for use with ancient historical data. Its of
> special interest to astronomers, who are interested in knowing exactly
> how long ago an ancient astronomer observed a particular event. 4713 BC
> is the year on which three different cycles associated with widely used

> ancient calendar systems were all in sync. Of course, nobody was using
> any of those calendar systems at that time, it's just a theoretical
> connection based upon running those calendar systems backward until the
> cycles are in sync. That synchronization makes it easier to calculate
> the Julian day for each of those calendar systems, and therefore to
> compare dates recorded in each of those systems with each other.