NAME
        Image::ExifTool::Shift.pl - ExifTool time shifting routines

DESCRIPTION
        This module contains routines used by ExifTool to shift date and time
        values.

METHODS
    ShiftTime
        Shift date/time value

            use Image::ExifTool;
            $err = Image::ExifTool::ShiftTime($dateTime, $shift);

        Inputs:
            0) Date/time string in EXIF format (eg. "2016:01:30 11:45:00").

            1) Shift string (see below) with optional leading sign for shift
            direction.

            2) [optional] Direction of shift (-1 or +1), or 0 or undef to use
            the sign from the shift string.

            3) [optional] Reference to time-shift hash -- filled in by first
            call to ShiftTime, and used in subsequent calls to shift date/time
            values by the same relative amount (see "TRICKY" section below).

            or

            0) Shift string (and $_ contains the input date/time string).

        Return value:
            Error string, or undef on success and the input date/time string is
            shifted by the specified amount.

SHIFT STRING
        Time shifts are applied to standard EXIF-formatted date/time values
        (eg. "2005:03:14 18:55:00"). Date-only and time-only values may also
        be shifted, and an optional timezone (eg. "-05:00") is also supported.
        Here are some general rules and examples to explain how shift strings
        are interpreted:

        Date-only values are shifted using the following formats:

            'Y:M:D'   - shift date by 'Y' years, 'M' months and 'D' days
            'M:D'   - shift months and days only
            'D'   - shift specified number of days

        Time-only values are shifted using the following formats:

            'h:m:s'   - shift time by 'h' hours, 'm' minutes and 's' seconds
            'h:m'   - shift hours and minutes only
            'h'   - shift specified number of hours

        Timezone shifts are specified in the following formats:

            '+h:m'   - shift timezone by 'h' hours and 'm' minutes
            '-h:m'   - negative shift of timezone hours and minutes
            '+h'   - shift timezone hours only
            '-h'   - negative shift of timezone hours only

        A valid shift value consists of one or two arguments, separated by a
        space. If only one is provided, it is assumed to be a time shift when
        applied to a time-only or a date/time value, or a date shift when
        applied to a date-only value. For example:

            '1'   - shift by 1 hour if applied to a time or date/time

```
                        value, or by one day if applied to a date value
    '2:0'          - shift 2 hours (time, date/time), or 2 months (date)
    '5:0:0'        - shift 5 hours (time, date/time), or 5 years (date)
    '0:0:1'        - shift 1 s (time, date/time), or 1 day (date)
```

If two arguments are given, the date shift is first, followed by the time shift:

```
    '3:0:0 0'          - shift date by 3 years
    '0 15:30'          - shift time by 15 hours and 30 minutes
    '1:0:0 0:0:0+5:0' - shift date by 1 year and timezone by 5 hours
```

A date shift is simply ignored if applied to a time value or visa versa.

Numbers specified in shift fields may contain a decimal point:

```
    '1.5'          - 1 hour 30 minutes (time, date/time), or 1 day (date)
    '2.5 0'        - 2 days 12 hours (date/time), 12 hours (time) or
                     2 days (date)
```

And to save typing, a zero is assumed for any missing numbers:

```
    '1::'          - shift by 1 hour (time, date/time) or 1 year (date)
    '26:: 0'       - shift date by 26 years
    '+:30'         - shift timezone by 30 minutes
```

Below are some specific examples applied to real date and/or time values ('Dir' is the applied shift direction: '+' is positive, '-' is negative):

```
    Original Value            Shift    Dir    Shifted Value
    --------------------      -------  ---    --------------------
    '20:30:00'                '5'       +     '01:30:00'
    '2005:01:27'              '5'       +     '2005:02:01'
    '2005:01:27 20:30:00'     '5'       +     '2005:01:28 01:30:00'
    '11:54:00'                '2.5 0'   -     '23:54:00'
    '2005:11:02'              '2.5 0'   -     '2005:10:31'
    '2005:11:02 11:54:00'     '2.5 0'   -     '2005:10:30 23:54:00'
    '2004:02:28 08:00:00'     '1 1.3'   +     '2004:02:29 09:18:00'
    '07:00:00'                '-5'      +     '07:00:00'
    '07:00:00+01:00'          '-5'      +     '07:00:00-04:00'
    '07:00:00Z'               '+2:30'   -     '07:00:00-02:30'
    '1970:01:01'              '35::'    +     '2005:01:01'
    '2005:01:01'              '400'     +     '2006:02:05'
    '10:00:00.00'             '::1.33'  -     '09:59:58.67'
```

NOTES
     The format of the original date/time value is not changed when the time
     shift is applied. This means that the length of the date/time string
     will not change, and only the numbers in the string will be modified.
     The only exception to this rule is that a 'Z' timezone is changed to
     '+00:00' notation if a timezone shift is applied. A timezone will not
     be added to the date/time string.

TRICKY
     This module is perhaps more complicated than it needs to be because it
     is designed to be very flexible in the way time shifts are specified
     and applied...

     The ability to shift dates by Y years, M months, etc, conflicts with
     the design goal of maintaining a constant shift for all time values
     when applying a batch shift. This is because shifting by 1 month can
     be equivalent to anything from 28 to 31 days, and 1 year can be 365 or
     366 days, depending on the starting date.

     The inconsistency is handled by shifting the first tag found with the
     actual specified shift, then calculating the equivalent time difference
     in seconds for this shift and applying this difference to subsequent
     tags in a batch conversion. So if it works as designed, the behaviour
     should be both intuitive and mathematically correct, and the user

shouldn't have to worry about details such as this (in keeping with
Perl's "do the right thing" philosophy).

BUGS
       Due to the use of the standard time library functions, dates are
       typically limited to the range 1970 to 2038 on 32-bit systems.

AUTHOR
       Copyright 2003-2020, Phil Harvey (philharvey66 at gmail.com)

       This library is free software; you can redistribute it and/or modify it
       under the same terms as Perl itself.

SEE ALSO
       Image::ExifTool(3pm)

perl v5.18.2   2020-01-02   SHIFT(1)