

**NAME**

Date::Parse – Parse date strings into time values

**SYNOPSIS**

```
use Date::Parse;

$time = str2time($date);

($ss, $mm, $hh, $day, $month, $year, $zone) = strptime($date);
```

**DESCRIPTION**

Date::Parse provides two routines for parsing date strings into time values.

str2time( [ DATE [, ZONE] ] )

str2time parses DATE and returns a unix time value, or undef upon failure. ZONE, if given, specifies the timezone to assume when parsing if the date string does not specify a timezone.

strptime( [ DATE [, ZONE] ] )

strptime takes the same arguments as str2time but returns an array of values (\$ss, \$mm, \$hh, \$day, \$month, \$year, \$zone, \$century). Elements are only defined if they could be extracted from the date string. The \$zone element is the timezone offset in seconds from GMT. An empty array is returned upon failure.

**MULTI-LANGUAGE SUPPORT**

Date::Parse is capable of parsing dates in several languages, these include English, French, German and Italian.

```
$lang = Date::Language->new('German');
$lang->str2time("25 Jun 1996 21:09:55 +0100");
```

**EXAMPLE DATES**

Below is a sample list of dates that are known to be parsable with Date::Parse

1995:01:24T09:08:17.1823213	ISO-8601
1995-01-24T09:08:17.1823213	
Wed, 16 Jun 94 07:29:35 CST	Comma and day name are optional
Thu, 13 Oct 94 10:13:13 -0700	
Wed, 9 Nov 1994 09:50:32 -0500 (EST)	Text in ()'s will be ignored.
21 dec 17:05	Will be parsed in the current time zone
21-dec 17:05	
21/dec 17:05	
21/dec/93 17:05	
1999 10:02:18 "GMT"	
16 Nov 94 22:28:20 PST	

**LIMITATION**

Date::Parse uses Time::Local internally, so is limited to only parsing dates which result in valid values for Time::Local::timelocal. This generally means dates between 1901-12-17 00:00:00 GMT and 2038-01-16 23:59:59 GMT

**BUGS**

When both the month and the date are specified in the date as numbers they are always parsed assuming that the month number comes before the date. This is the usual format used in American dates.

The reason why it is like this and not dynamic is that it must be deterministic. Several people have suggested using the current locale, but this will not work as the date being parsed may not be in the format of the current locale.

My plans to address this, which will be in a future release, is to allow the programmer to state what order they want these values parsed in.

**AUTHOR**

Graham Barr <gbarr@pobox.com>

**COPYRIGHT**

Copyright (c) 1995–2009 Graham Barr. This program is free software; you can redistribute it and/or modify it under the same terms as Perl itself.