eZ systems

Welcome!

Share your information

eZ systems

What is Time?

- Time quantifies or measures the interval between events, or the duration of events. (wikipedia)
- The standard unit for time is the SI second.
- A second is the duration of 9,192,631,770 periods of the radiation corresponding to the transition between the two hyperfine levels of the ground state of the caesium-133 atom.
- We use larger units: minute, hour, day, week, month, year, century.
- Those additional units make things hard.

eZ systems

Problems with Dates and Time

Ambiguities:

```
06/08/04
```

- August 6th, 2004
- June 8th, 2004
- August 4th, 2006

Unreadable:

```
20040425010541
```

- April 25th, 2005, 01:05:41

"Weird" formats:

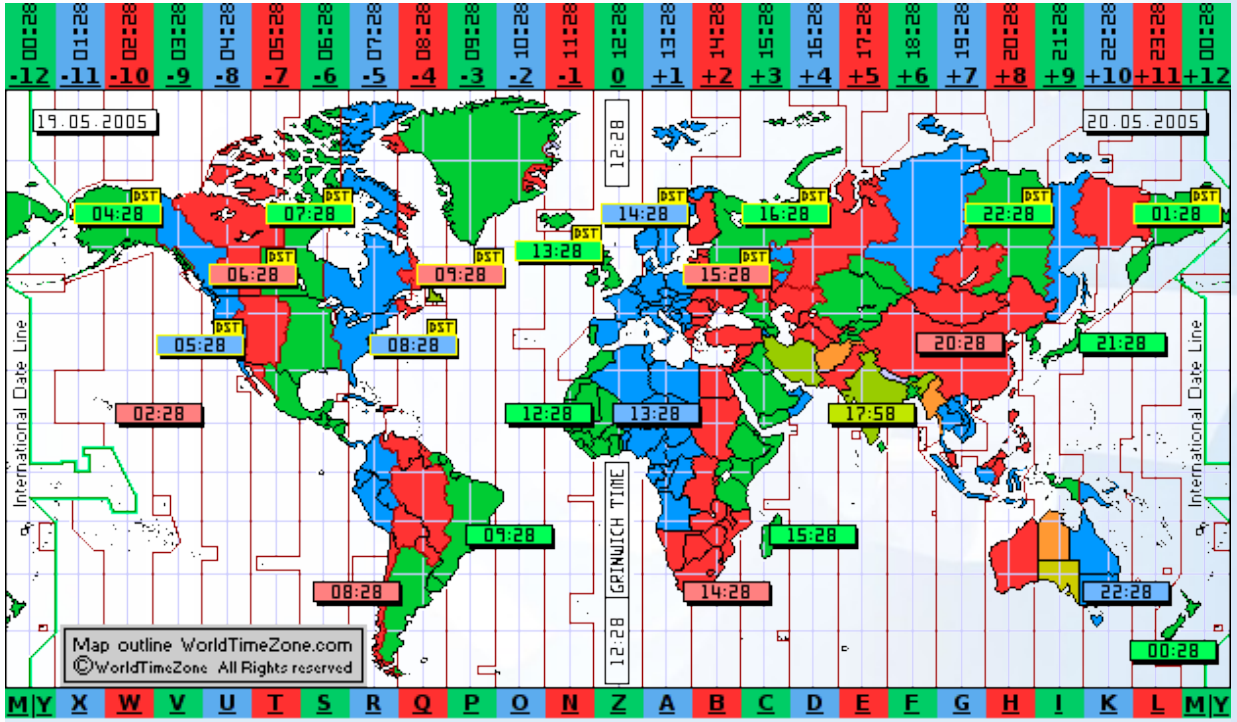
```
third saturday
2004-03-10 16:33:17.11403+01
2001-11-29T13:20:01.123-05:00
23:41:00.0Z
04:05:07.789 +0930
1999.238
```

Share your information

eZ systems

Share your information

Problems



- Most places have whole-hour timezone offsets
- Some places have half-hour or even quarter-hour timezone offsets
- Some places change timezones during the year

Problems

One identifier can mean different zones:

- PST: Pacific Standard Time, Pakistan Standard Time
- EST: Eastern Standard Time (USA), Eastern Standard Time (Australia) and Eastern Brazil Standard Time

One zone can have multiple identifiers

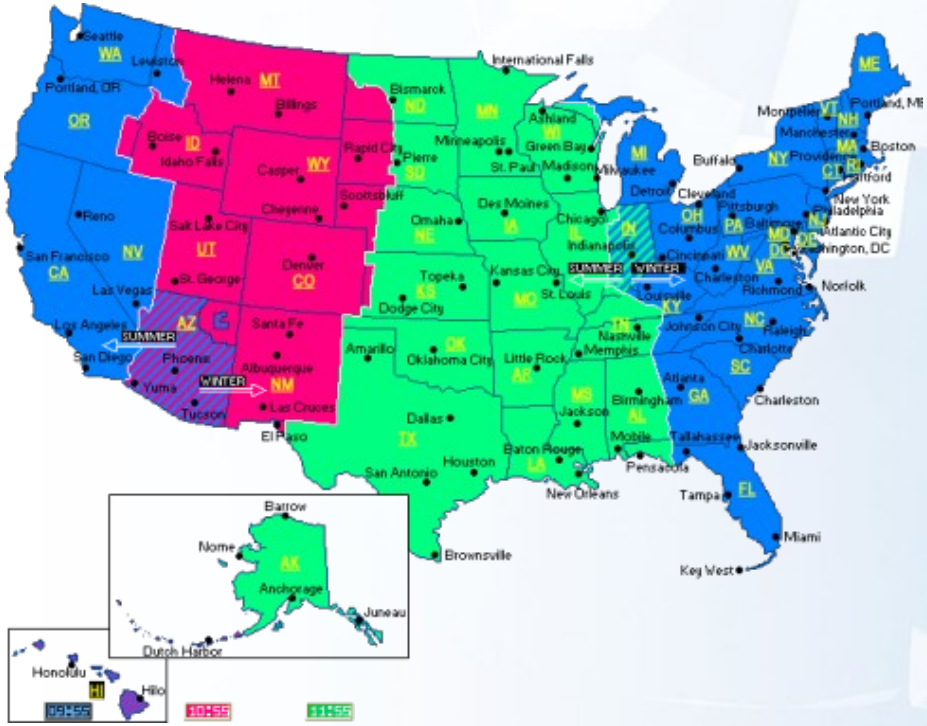
- Central Europe Summer Time: CEST or CETDST

Names can be different on Operating Systems

eZ systems

Problems

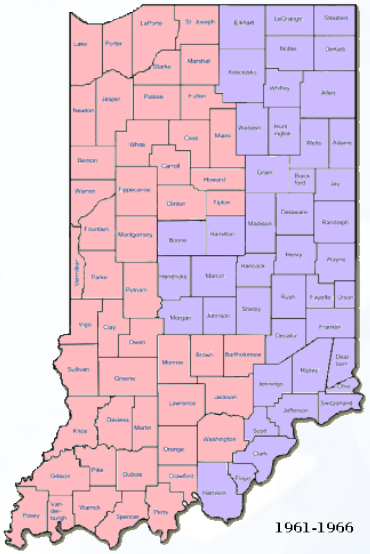
- Artificial time offset to save "daylight"
- Not all countires/areas use it
- Switches are not done at the same time for all areas
- There are plenty of exceptions



Share your information

eZ systems

Problems



(Currently) Three different rules:

- Most counties: Eastern, no DST
- Counties close to Chicago and Evansvile: Central, DST
- Counties close to Cincinnatti and Lousville: Eastern, DST

Share your information

eZ systems

Share your information

Problems

- Australia has vertical timezones
- Brazil changes change-over dates every year
- Lord Howe Island (Australia) moves only half an hour between DST and non-DST
- Nepal uses an quarter of an hour offsets
- And this continues for a while...

eZ systems

Date/Time Functions in PHP 4 and PHP 5.0

- checkdate() -- Validate a Gregorian date
- date() / gmdate() -- Format a local/GMT time/date
- getdate() -- Get date/time information (from a timestamp)
- gettimeofday() -- Get current time (as an array)
- localtime() -- Get the local time (as an array)
- mktime() / gmmktime() -- Get Unix timestamp for a date
- strftime() / gmstrftime() -- Format a local/GMT time/date according to locale settings
- strtotime() -- Parse about any English textual datetime description into a Unix timestamp

eZ systems

Date/Time Functions in PHP 4 and PHP 5.0

- Uses Unix timestamp as base unit (seconds since 1970-01-01, 00:00 GMT)
- Only 32 bit integers for timestamps (1902 to 2038)
- Limited to only positive numbers on some Operating Systems (1970 to 2038)
- strtotime() is buggy and very complex
- No way of dealing correctly with timezones
- Some functions are Operating System dependent

eZ systems

Date/Time Functions in PHP 5.1 and up

- "Using a (signed) 64-bit value introduces a new wraparound date in about 290 billion years, on Sunday, December 4, 292,277,026,596. However, this problem is not widely regarded as a pressing issue."
- strtotime() has been rewritten
- Nothing is Operating System dependent
- Full support for timezones, DST, date modifications
- New format modifiers: e for timezone identifier and o for ISO Year
- Advanced date handling functions

eZ systems

Date/Time Functions in PHP 5.1 and up

"American" formats:

```
9/11                    4:08 pm
12/22/78                8:51:00 am
```

"Combined" formats:

```
Sat, 24 Apr 2004 21:48:40 +0200
2001-11-29T13:20:01.123-05:00
```

Descriptive formats:

```
tomorrow                last saturday
four months ago         +40 days 2 hours
```

Textual formats:

```
December 22. 1978     22-december-78
```

All ISO 8601 formats:

```
1978/12/22              1978-12-22              70-4-25
13:03:12.45678          15:57:41.0 pdt          13:03 CEST
13:03:12.45678 +0100     13:03:12.45678 CEST      04:05 -0930
15:57-8                  231431 CEST              23:41F
```

Database specific formats:

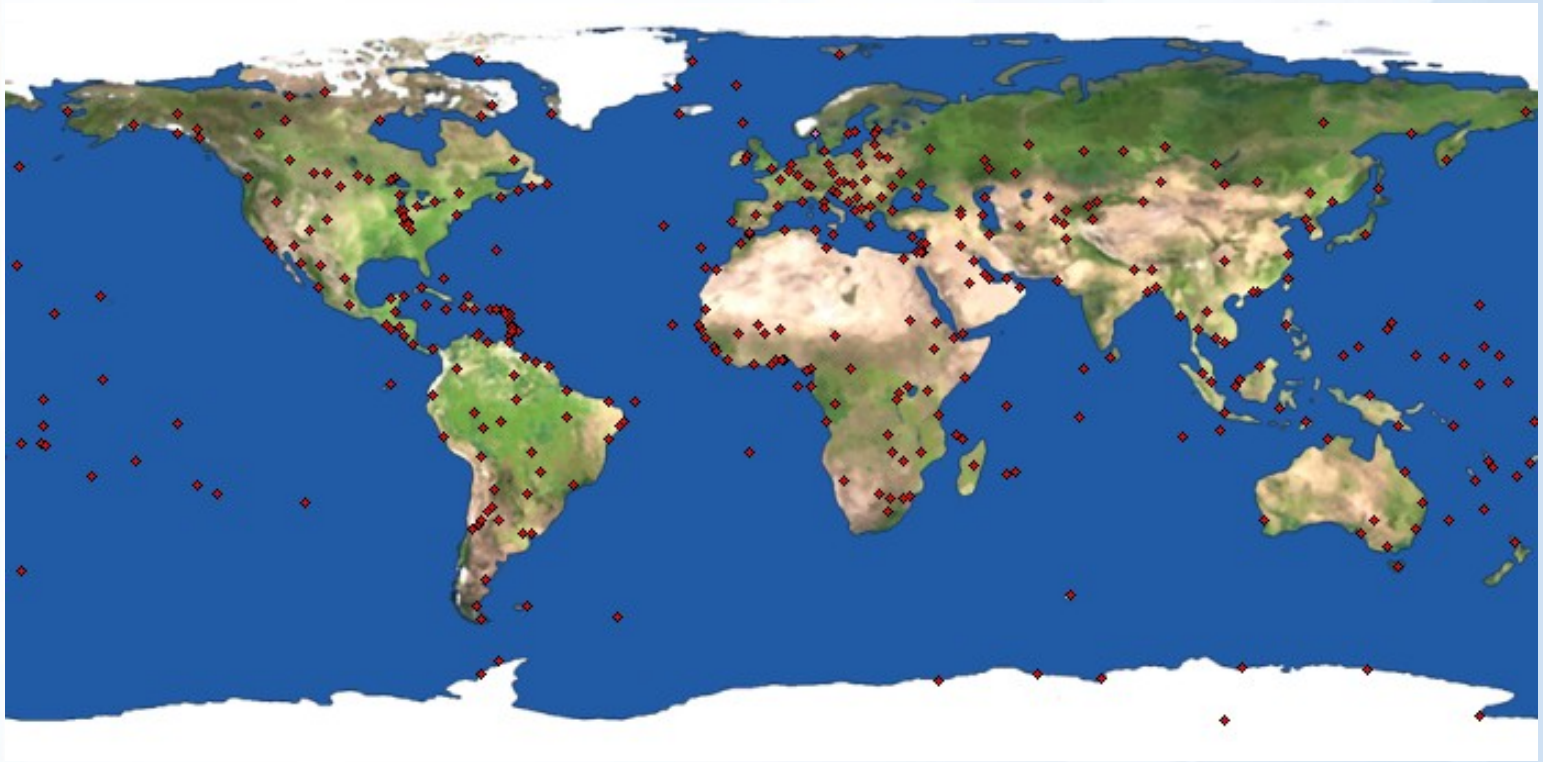
```
1999-Jan-08
1999.238
```

Share your information

eZ systems

Date/Time Functions in PHP 5.1 and up

- Bundled timezone database with 546 zones
- Not dependent on timezone abbreviations
- Timezones have the format: Continent/Location or Continent/Location/Sublocation - Like: Europe/Amsterdam, America/Indiana/Knox



Share your information

eZ systems

Timezone Support

- Each zone is identified by the city with the largest population in a specific area.
- Zones are divided into 10 major groups: Africa, America, Antarctica, Artic, Asia, Atlantic, Europe, Indian, Pacific.
- Examples are: America/Toronto, Europe/Berlin, America/New_York, Europe/Oslo, Asia/Singapore.
- There is also a group "Others" which contains outdated and backwards compatible names, such as Canada/Eastern, Etc/GMT-1 and US/Central. Those should not be used.
- Pick the zone that is closest to your location.
- http://php.net/timezones

eZ systems

Changing Timezone Definitions

- An updated database is released about 20 times a year.
- Some of the changes are very sudden.
- PHP releases will therefore often have an outdated version.
- The PECL extension timezonedb provides a drop in replacement for the timezone database.
- pecl install timezonedb

eZ systems

Parsing Dates

Parsing strings for date time information with the strtotime() function:

```
<?php
    $ts = strtotime("2005-07-11 22:16:50 CEST");
?>
```

With initial timestamp:

```
<?php
    $date = strtotime("2005-07-11 22:16:50 CEST");
    $ts = strtotime("next week", $date);
?>
```

The timestamps returned are still 32 bit signed integers as this is all that PHP supports.

eZ systems

Parsing Dates - Take #2

Parsing strings for date time information with the date_create() function:

```
<?php
    $ts = date_create("2005-07-11 22:16:50");
?>
```

Alternatively you can just instantiate a DateTime object:

```
<?php
    $ts = new DateTime("2006-11-07 16:12:15");
?>
```

This function will not return the timestamp as an integer, but instead returns a DateTime object which is a wrapper around a 64 bit integer (with some additional functionality ofcourse).

eZ systems

Parsing Dates

Parsing strings with the date_parse() function:

```
<pre><?php
$date = "22apr2006 8:36:14.43 #^ Europe/Oslo CEST";
$t = date_parse( $date );

echo $t['year'], '-', $t['month'], '-', $t['day'], " ";
echo $t['hour'], ':', $t['minute'], ':', $t['second'] + $t['fraction'], " ";
echo $t['tz_id'], "\n";

if ( $t['warning_count'] )
    echo "Warnings:\n";
    foreach( $t['warnings'] as $pos => $message )
        echo "- $message @$pos\n";
if ( $t['error_count'] )
    echo "Errors:\n";
    foreach( $t['errors'] as $pos => $message )
        echo "- $message @$pos\n";
```

Share your information

eZ systems

Formatting Dates

Formatting using format specifiers:

```
<?php
    date_default_timezone_set("Europe/Oslo");
    $ts = date_create("1979-12-31 09:15");
    echo date_format($ts, "D Y-m-d H:i:s - \I\S\O \W/\Y: W/o"), "<br/>\n";
```

All format modifiers as supported by date() are supported too.

Predefined formats:

```
<?php
    date_default_timezone_set("Europe/Oslo");
    $ts = date_create("December 22nd, 2005 15:41");
    echo date_format($ts, DATE_ISO8601), "<br/>\n";
    echo date_format($ts, DATE_RFC1036), "<br/>\n";
    echo date_format($ts, DATE_RSS), "<br/>\n";
```

Share your information

eZ systems

Updating Dates and Times

```
<?php
    $date = new DateTime( 'now' );
    echo $date->format(DateTime::ISO8601), "<br/>\n";

    $date->setTime( 15, 0, 7 );
    echo $date->format(DateTime::ISO8601), "<br/>\n";

    $date->setDate( 2006, 12, 22 );
    echo $date->format(DateTime::ISO8601), "<br/>\n";

    $date->setIsoDate( 2006, 45, 2 );
    echo $date->format(DateTime::ISO8601), "<br/>\n";
?>
```

Share your information

eZ systems

Modifying Dates

Modifying dates and times:

```
<?php
    date_default_timezone_set("Europe/Oslo");
    $ts = new DateTime("now");
    echo $ts->format(DATE_RFC2822), "<br/>\n";

    echo $ts->modify("+2 days");
    echo $ts->format(DATE_RFC2822), "<br/>\n";

    echo $ts->modify("fifth month");
    echo $ts->format(DATE_RFC2822), "<br/>\n";

    echo $ts->modify("Friday +3 weeks");
    echo $ts->format(DATE_RFC2822), "<br/>\n";

    echo $ts->modify("next friday");
    echo $ts->format(DATE_RFC2822), "<br/>\n";
?>
```

eZ systems

Using Timezones

Specifying timezone abbreviation while parsing:

```
<?php
    $ts = date_create("1978-12-22 09:15 CET");
?>
```

Using timezone abbreviations is deprecated, one should always use either a default timezone, or the full identifier.

Specifying timezone identifier while parsing:

```
<?php
    $ts = date_create("1978-12-22 09:15 Europe/Oslo");
?>
```

eZ systems

Default Timezones

Setting a default timezone:

```
<?php
    date_default_timezone_set("Europe/Oslo");
    $ts = date_create("1978-12-22 09:15");
    echo date_format($ts, "e");
?>
```

Getting a default timezone:

```
<?php
    $default_identifier = date_default_timezone_get();
    echo $default_identifier;
?>
```

Default timezone is 'guessed' in the following order:

- date_default_timezone_set() value
- TZ environment variable
- php.ini's date.timezone setting
- System's rendering of timezone abbreviation

Share your information

eZ systems

Using Timezones

Creating a timezone resource:

```
<?php
    $tz = timezone_open("Asia/Singapore");
?>
```

Using the timezone when parsing a string with a date representation:

```
<?php
    $tz = timezone_open("Pacific/Honolulu");
    $ts = date_create("1978-12-22 09:15", $tz);
?>
```

A passed timezone object does not override a parsed timezone:

```
<?php
    $tz = new DateTimeZone("Pacific/Honolulu");
    $ts1 = new DateTime("1978-12-22 09:15 CET", $tz);
    $ts2 = new DateTime("1978-12-22 09:15 Europe/Amsterdam", $tz);
    echo $ts2->format( DateTime::RFC2822 );
?>
```

Share your information

eZ systems

Using Timezones

Getting a timezone's name:

```
<?php
    $tz = timezone_open("Asia/Singapore");
    echo timezone_name_get($tz), ', ';

    $tz = timezone_open("CEST");
    echo timezone_name_get($tz);
?>
```

Getting the current offset to GMT with a timezone for a specific date:

```
<?php
    $tz = new DateTimeZone("Europe/Amsterdam");
    $d = new DateTime("2005-01-22 09:15");
    echo $tz->getOffset($d), ', ';
    $d->modify("+6 months");
    echo $tz->getOffset($d);
?>
```

Share your information

eZ systems

Using Timezones

Using the timezone when parsing a string with a date representation:

```
<?php
    $tz1 = timezone_open("Pacific/Honolulu");
    $tz2 = timezone_open("Europe/Amsterdam");
    $tz3 = timezone_open("Australia/Melbourne");

    $ts = date_create("1978-12-22 09:15", $tz1);
    echo $ts->getTimezone()->getName(), ': ',
        $ts->format(DATE_RFC822), "<br/>";

    $ts->setTimezone($tz2);
    echo $ts->getTimezone()->getName(), ': ',
        $ts->format(DATE_RFC822), "<br/>";

    date_timezone_set($ts, $tz3);
    echo timezone_name_get(date_timezone_get($ts)), ': ',
        date_format($ts, DATE_RFC822);
?>
```

Share your information

eZ systems

Timezones Utilities

Creating a timezone resource:

```
<?php
    $tz = timezone_open("Europe/Amsterdam");
    $trs = timezone_transitions_get($tz);
    $trs = $tz->getTransitions();

    echo "<pre>\n";
    foreach ($trs as $tr) {
        printf("%20s %7d %d %s\n",
            $tr['time'], $tr['offset'],
            $tr['isdst'], $tr['abbr']);
    }
?>
```

Share your information

eZ systems

Timezones Utilities

All supported timezone identifiers:

```
<?php
    $ids = timezone_identifiers_list();
    echo "Number of identifiers: ", count($ids), "<br/>";
    echo implode(", ", array_slice($ids, 0, 5)), '...';
    echo implode(", ", array_slice($ids, -5));
?>
```

All supported timezone identifiers:

```
<?php
    $abbrs = timezone_abbreviations_list();
    echo "<pre>\n";
    foreach ($abbrs as $abbr => $ids) {
        foreach ($ids as $id) {
            printf("%-6s %6d %d %s\n", strtoupper($abbr),
                $id['offset'], $id['dst'], $id['timezone_id']);
        }
    }
?>
```

Share your information

eZ systems

Daylight Savings Time

```
<?php
$tz = new DateTimeZone(
    isset($_GET['zone']) ? $_GET['zone'] : "Europe/Amsterdam");

foreach (timezone_transitions_get($tz) as $tr)
    if ($tr['ts'] > time())
        break;

$d = new DateTime( "@{$tr['ts']}" );
printf("The timezone %s switches to %s on %s.<br/>The new GMT offset will be:
%d (%s)\n",
    $tz->getName(), $tr['isdst'] ? "DST" : "standard time",
    $d->format('d M Y @ H:i'), $tr['offset'], $tr['abbr']
);
?>
<form>
<select name='zone'>
<?php foreach(DateTimeZone::listIdentifiers() as $id) {
    echo "<option value='$id'>$id</option>\n"; } ?>
</select>
<input type="submit" value="go"/>
</form>
```

Share your information

eZ systems

Timezone and Date tips

- Store date/time in a timezone agnostic way: UTC
- Use a "Unix" timestamp, perhaps as string: date("U");

Share your information

eZ systems

Share your information

Sunset/Sunrise

```
<pre><font size="5"><?php
date_default_timezone_set('Europe/Oslo');
if (isset($_POST['country'])) {
mysql_connect('localhost', 'root'); mysql_select_db('geolocation');
$re = mysql_query( "SELECT * FROM city where country='{$_POST['country']}' AND
normalized_name='{$_POST['city']}'" );
$info = mysql_fetch_assoc( $re );
if ($info) {
    $lat = $info['lat']; $lon = $info['lon'];

    $info = date_sun_info(time(), $lon, $lat);
    echo "Sunrise is at ", date(DateTime::RSS, $info['sunrise']), "\n";
    echo "Sunset is at ", date(DateTime::RSS, $info['sunset']), "\n";
    echo "Day length is ",
    round(($info['sunset'] - $info['sunrise']) / 3600, 1), "hrs";
}
}
?></font></pre>
<form method="post">
Country: <input name="country"/> City: <input name="city"/><br/>
<input type="submit" value="Check"/>
</form>
```

eZ systems

User's Timezone

```
<?php
if (!isset($_GET['tzinfo'])) {
?>
<html>
<script type="text/javascript">
var d = new Date()
var tza = d.toLocaleString().split(" ").slice(-1)
var tzo = d.getTimezoneOffset()
window.location = window.location + '?tzinfo=' + tza + '|' + tzo
</script>
</html>
<?php
} else {
    list( $abbr, $offset ) = explode( '|', $_GET['tzinfo'] );
    echo timezone_name_from_abbr( $abbr, $offset * -60 );
}
?>
```

Share your information

eZ systems

When Is The Additional Functionality Available

In PHP 5.1 if you compile PHP with a special flag:

```
CFLAGS=-DEXPERIMENTAL_DATE_SUPPORT=1 ./configure
```

In PHP 5.2 and higher by default.

Share your information

eZ systems

Resources

This presentation: http://derickrethans.nl/talks.php
World timezones: http://www.worldtimezone.com/index24.php
Questions?: dr@ez.no

Share your information