Time zone

From Wikipedia, the free encyclopedia https://en.wikipedia.org/wiki/Time_zone

A **time zone** is a region of the globe that observes a uniform standard time for legal, commercial, and social purposes. Time zones tend to follow the boundaries of countries and their subdivisions because it is convenient for areas in close commercial or other communication to keep the same time.

Most of the time zones on land are offset from Coordinated Universal Time (UTC) by a whole number of hours (UTC−12 to UTC+14), but a few zones are offset by 30 or 45 minutes (e.g. Newfoundland Standard Time is UTC−03:30, Nepal Standard Time is UTC+05:45, and Indian Standard Time is UTC+05:30).

Some higher latitude and temperate zone countries use daylight saving time for part of the year, typically by adjusting local clock time by an hour. Many land time zones are skewed toward the west of the corresponding nautical time zones. This also creates a permanent daylight saving time effect.

Contents

History

Early timekeeping

Before clocks were first invented, it was common practice to mark the time of day with apparent solar time (also called "true" solar time) – for example, the time on a sundial – which was typically different for every location and dependent on longitude.

When well-regulated mechanical clocks became widespread in the early 19th century,[1] each city began to use some local mean solar time. Apparent and mean solar time can differ by up to around 15 minutes (as described by the equation of time) because of the elliptical shape of the Earth's orbit around the Sun (eccentricity) and the tilt of the Earth's axis (obliquity). Mean solar time has days of equal length, and the difference between the two sums to zero after a year.

Greenwich Mean Time (GMT) was established in 1675, when the Royal Observatory was built, as an aid to mariners to determine longitude at sea, providing a standard reference time while each city in England kept a different local time.

Railway time

Plaque commemorating the Railway General Time Convention of 1883 in North America

Local solar time became increasingly inconvenient as rail transport and telecommunications improved, because clocks differed between places by amounts corresponding to the differences in their geographical longitudes, which varied by four minutes of time for every degree of longitude. For example, Bristol is about 2.5 degrees west of Greenwich (East London), so when it is solar noon in Bristol, it is about 10 minutes past solar noon in London.[2] The use of time zones accumulates these differences into longer units, usually hours, so that nearby places can share a common standard for timekeeping.

The first adoption of a standard time was on December 1, 1847, in Great Britain by railway companies using GMT kept by portable chronometers. The first of these companies to adopt standard time was the Great Western Railway(GWR) in November 1840. This quickly became known as Railway Time. About August 23, 1852, time signals were first transmitted by telegraph from the Royal Observatory, Greenwich. Even though 98% of Great Britain's public clocks were using GMT by 1855, it was not made Britain's legal time until August 2, 1880. Some British clocks from this period have two minute hands—one for the local time, one for GMT.[3]

Improvements in worldwide communication further increased the need for interacting parties to communicate mutually comprehensible time references to one another. The problem of differing local times could be solved across larger areas by synchronizing clocks worldwide, but in many places that adopted time would then differ markedly from the solar time to which people were accustomed.

On November 2, 1868, the then British colony of New Zealand officially adopted a standard time to be observed throughout the colony, and was perhaps the first country to do so. It was based on the longitude 172°30′ East of Greenwich, that is 11 hours 30 minutes ahead of GMT. This standard was known as New Zealand Mean Time.[4]

Timekeeping on the American railroads in the mid-19th century was somewhat confused. Each railroad used its own standard time, usually based on the local time of its headquarters or most important terminus, and the railroad's train schedules were published using its own time. Some junctions served by several railroads had a clock for each railroad, each showing a different time.[5]
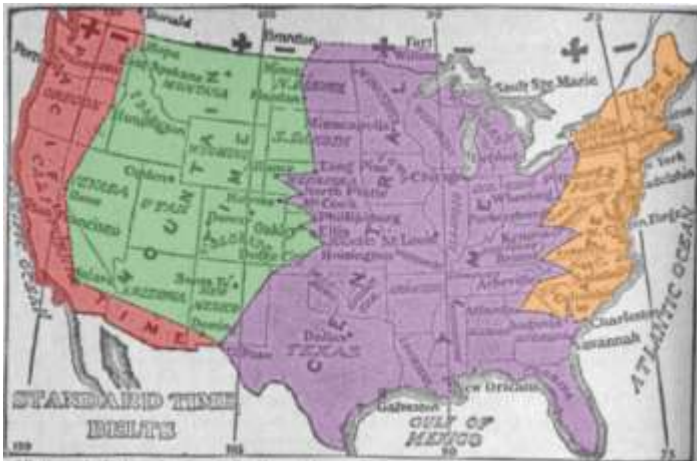
1913 time zone map of the United States, showing boundaries very different from today

Charles F. Dowd proposed a system of one-hour standard time zones for American railroads about 1863, although he published nothing on the matter at that time and did not consult railroad officials until 1869. In 1870 he proposed four ideal time zones (having north–south borders), the first centered on Washington, D.C., but by 1872 the first was centered on the meridian 75° W of Greenwich, with geographic borders (for example, sections of the Appalachian Mountains). Dowd's system was never accepted by American railroads. Instead, U.S. and Canadian railroads implemented a version proposed by William F. Allen, the editor of the *Traveler's Official Railway Guide*.[6] The borders of its time zones ran through railroad stations, often in major cities. For example, the border between its Eastern and Central time zones ran through Detroit, Buffalo, Pittsburgh, Atlanta, and Charleston. It was inaugurated on Sunday, November 18, 1883, also called "The Day of Two Noons",[7] when each railroad station clock was reset as standard-time noon was reached within each time zone. The zones were named Intercolonial, Eastern, Central, Mountain, and Pacific. Within a year 85% of all cities with populations over 10,000, about 200 cities, were using standard time.[8] A notable exception was Detroit (which is about halfway between the meridians of eastern time and central time) which kept local time until 1900, then tried Central Standard Time, local mean time, and Eastern Standard Time before a May 1915 ordinance settled on EST and was ratified by popular vote in August 1916. The confusion of times came to an end when Standard zone time was formally adopted by the U.S. Congress in the Standard Time Actof March 19, 1918.

Worldwide time zones

The first known person to conceive of a worldwide system of time zones was the Italian mathematician Quirico Filopanti. He introduced the idea in his book *Miranda!* published in 1858. He proposed 24 hourly time zones, which he called "longitudinal days", the first centred on the meridian of Rome. He also proposed a universal time to be used in astronomy and telegraphy. But his book attracted no attention until long after his death.[9][10]

Scottish-born Canadian Sir Sandford Fleming proposed a worldwide system of time zones in 1879. He advocated his system at several international conferences, and is credited with "the initial effort that led to the adoption of the present time meridians".[11] In 1876, his first proposal was for a global 24-hour clock, conceptually located at the centre of the Earth and not linked to any surface meridian. In 1879 he specified that his universal day would begin at the anti-meridian of Greenwich (180th meridian), while conceding that hourly time zones might have some limited local use. He also proposed his system at the International Meridian Conference in October 1884, but it did not adopt his time zones because they were not within its purview. The conference did adopt a universal day of 24 hours beginning at Greenwich midnight, but specified that it "shall not interfere with the use of local or standard time where desirable".[12]

By about 1900, almost all time on Earth was in the form of standard time zones, only some of which used an hourly offset from GMT. Many applied the time at a local astronomical observatory to an entire country, without any reference to GMT. It took many decades before all time on Earth was in the form of time zones referred to some "standard offset" from GMT/UTC. By 1929, most major countries had adopted hourly time zones. Nepal was the last country to adopt a standard offset, shifting slightly to UTC+5:45 in 1956.[13]

Today, all nations use standard time zones for secular purposes, but they do not all apply the concept as originally conceived. Newfoundland, India, Iran, Afghanistan, Burma, Sri Lanka, the Marquesas, as well as parts of Australia use half-hour deviations from standard time, and some nations, such as Nepal, and some provinces, such as the Chatham Islands of New Zealand, use quarter-hour deviations. Some countries, such as China and India, use a single time zone even though the extent of their territory far exceeds 15° of longitude.[14]

Notation of time

ISO 8601

Main article: ISO 8601

ISO 8601 is an international standard that defines methods of representing dates and times in textual form, including specifications for representing time zones.[15]

UTC

If a time is in Coordinated Universal Time (UTC), a "Z" is added directly after the time without a separating space. "Z" is the zone designator for the zero UTC offset. "09:30 UTC" is therefore represented as "09:30Z" or "0930Z". Likewise, "14:45:15 UTC" is written as "14:45:15Z" or "144515Z".[16]
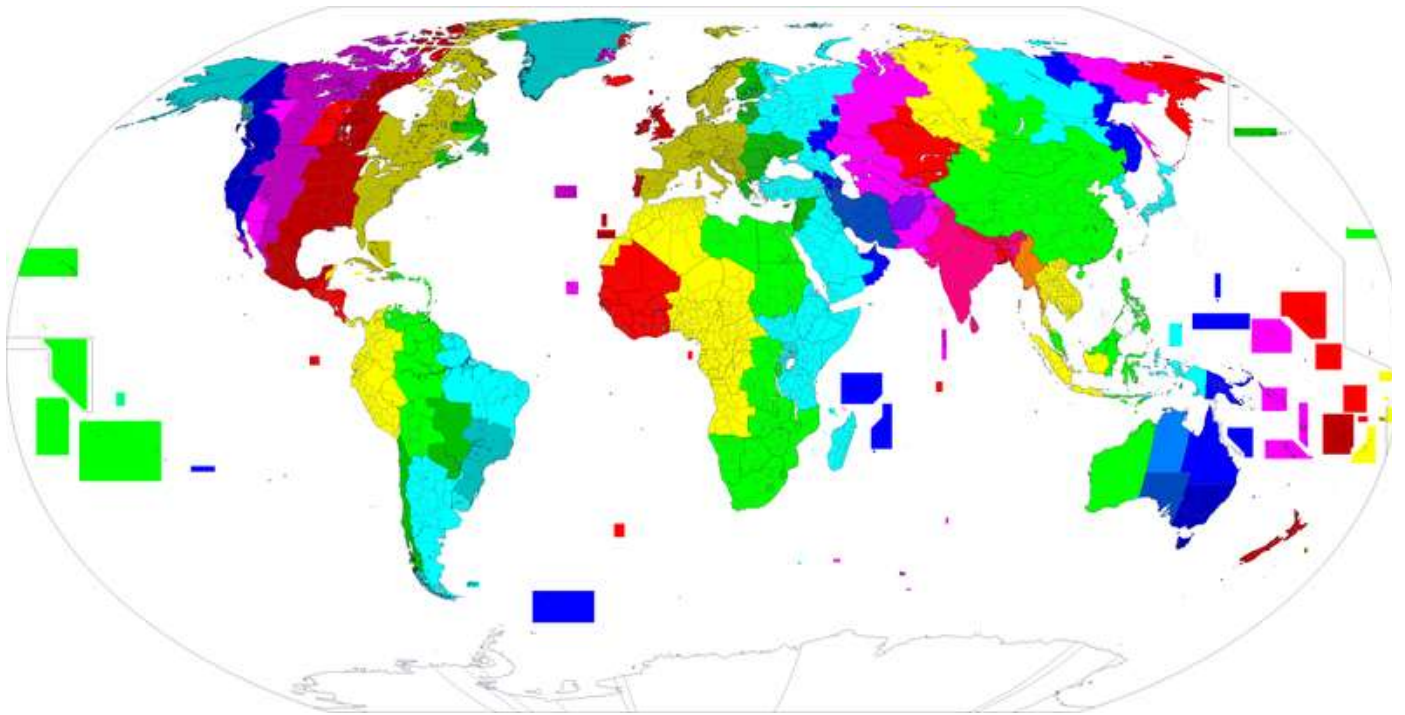
UTC time is also known as "Zulu" time, since "Zulu" is a phonetic alphabet code word for the letter "Z".[17]

Offsets from UTC

Offsets from UTC are written in the format ±[hh]:[mm], ±[hh] [mm], or ±[hh] (either hours ahead or behind UTC). For example, if the time being described is one hour ahead of UTC (such as the time in Berlin during the winter), the zone designator would be "+01:00", "+0100", or simply "+01". This numeric representation of time zones is appended to local times in the same way that alphabetic time zone abbreviations (or "Z", as above) are appended. The offset from UTC changes with daylight saving time, e.g. a time offset in Chicago, which is in the North American Central Time Zone, is "−06:00" for the winter (Central Standard Time) and "−05:00" for the summer (Central Daylight Time).[18]

Abbreviations

Main article: List of time zone abbreviations

Time zones are often represented by alphabetic abbreviations such as "EST", "WST", and "CST", but these are not part of the international time and date standard ISO 8601 and their use as sole designator for a time zone is discouraged. Such designations can be ambiguous; for example, "ECT" could be interpreted as "Eastern Caribbean Time" (UTC−4h), "Ecuador Time" (UTC−5h), or "European Central Time" (UTC+1h).[19]

UTC offsets worldwide

Main article: List of UTC time offsets

A great part of the world has a gap between the official time and the solar time

UTC−12:00 ... UTC−07:00		UTC−06:00 ... UTC−01:00		UTC±00:00 ... UTC+05:45		UTC+06:00 ... UTC+11:30		UTC+12:00 ... UTC+14:00	
Oceania / North America / Antarctica		North and South America / Antarctica		Europe / Africa / Asia / Antarctica		Asia / Antarctica		Asia / Oceania / Antarctica	
No DST in summer	DST in summer	No DST in summer	DST in summer	No DST in summer	DST in summer	No DST in summer	DST in summer	No DST in summer	DST in summer
−12:00	−12:00 /−11:00 N: US-	−06:00	−06:00 /−05:00 N: US-, MX-	±00:00 IS	±00:00 /+01:00 N: GB, IE, PT	+06:00 RU-, KZ--	+06:00 /+07:00	+12:00 KI-, RU-	+12:00 /+13:00 S: NZ-
						+06:30 MM			
								+12:45	+12:45 /+13:45 S: NZ

−11:00 US-	−11:00/−10:00	−05:00 BO, CO, PA, PE	−05:00/−04:00 N: CA-, CU, US-	+01:00 TN, CG, CD-, DZ, NE, NG	+01:00/+02:00 N: AT, BA, BE, CH, CZ, DE, DK, ES-, FR, HR, HU, IT, LI, LU, MK, NL, NO, PL, SE, SI, SK S: NA	+07:00 RU-, VN, LA, TH, KH, ID-	+07:00/+08:00 N: MN-	+13:00 KI-
−10:00 US-	−10:00/−09:00 US-	−04:00	−04:00/−03:00 S: CL-	+02:00 Africa: BI, BW, CD-, EG, LY, MW, MZ, RW, ZA, ZM, ZW	+02:00/+03:00 N: FI, EE, LV, LT, UA, BG, GR, MD, RO	+08:00 AU-, CN, HK, ID, MY, RU-, PH, SG, TW,	+08:00/+09:00 N: MN-	+14:00 KI-
		−03:30	−03:30/−02:30 S: CA-					
−09:00	−09:00/−08:00 N: US-	−03:00 S: ARCL-	−03:00/−02:00 S: BR-	+03:00 Europe: BY, RU-, TR, Africa: KE, SD, SO, SS, ER, Asia: IQ, SA	+03:00/+04:00	+09:00 RU-, JP, KR, ID-	+09:00/+10:00	
				+03:30	+03:30/+04:30 IR	+09:30	+09:30/+10:30 AU-	
−08:00	−08:00/−07:00 N: CA-, US-, MX-	−02:00 BR-	−02:00/−01:00	+04:00 RU-, GE	+04:00/+05:00	+10:00 RU-	+10:00/+11:00	

−07:00 US-, MX-	−07:00/−06:00 N: CA-, US-, MX- S: CL-	−01:00	−01:00/±00:00	+04:30 AF			
				+05:00 KZ-, PK	+05:00/+06:00	+11:00 RU-	+11:00/+12:00
				+05:30 IN		+11:30 NF	
				+05:45 NP			

XX = ISO 3166-1 alpha-2 country code, XX- = parts of the country, N = North, S = South, UTC = Universal Coordinated Time, DST = Daylight Saving Time]]
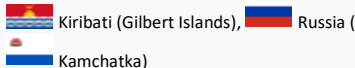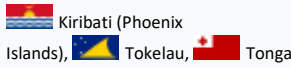
List of UTC offsets

These examples give the local time at various locations around the world when daylight saving time is not in effect:

Time offset	Example time (ISO 8601 notation)	Example locations that do not use DST	Example locations that in summer use DST
UTC−12:00	2019-01-07T21:51:29-12:00	Baker Island, Howland Island (both uninhabited)	
UTC−11:00	2019-01-07T22:51:29-11:00	American Samoa, Niue	
UTC−10:00	2019-01-07T23:51:29-10:00	French Polynesia (most), United States (Hawaii) (Cook Islands)	United States (Aleutian Islands)
UTC−09:30	2019-01-08T00:21:29-09:30	French Polynesia (Marquesas Islands)	
UTC−09:00	2019-01-08T00:51:29-09:00	French Polynesia (Gambier Islands)	United States (most of Alaska)
UTC−08:00	2019-01-08T01:51:29-08:00	Pitcairn Islands	Canada (most of British Columbia), Mexico (Baja California), United States (California, most of Nevada, most of Oregon, Washington)
UTC−07:00	2019-01-08T02:51:30-07:00	Canada (northeastern British Columbia), Mexico (Sonora), United States (most of Arizona)	Canada (Alberta), Mexico (Chihuahua), United States (Colorado)
UTC−06:00	2019-01-08T03:51:30-06:00	Belize, Canada (most of Saskatchewan), Costa Rica, Ecuador (Galápagos Islands), El	Canada (Manitoba), United States (Illinois, most of Texas), Mexico (most), Chile (Easter Island)

		Salvador, Guatemala, Honduras, Nicaragua	
UTC−05:00	2019-01-08T04:51:30-05:00	Brazil (Acre), Colombia, Ecuador (continental), Haiti, Jamaica, Mexico (most of Quintana Roo), Panama, Peru	Bahamas, Canada (most of Ontario, most of Quebec), Cuba, United States (most of Florida, Georgia, New York, Ohio, Pennsylvania)
UTC−04:00	2019-01-08T05:51:30-04:00	Barbados, Bolivia, Brazil (most of Amazonas, Rondônia, Roraima), Dominican Republic, Puerto Rico, Trinidad and Tobago, Venezuela	Brazil (most of Mato Grosso, Mato Grosso do Sul), Canada (Nova Scotia, New Brunswick and most of Labrador, Prince Edward Island), Chile (continental), Paraguay
UTC−03:30	2019-01-08T06:21:30-03:30		Canada (island of Newfoundland, southeastern Labrador)
UTC−03:00	2019-01-08T06:51:30-03:00	Argentina, Brazil (Bahia, Ceará, Maranhão, Pará, Pernambuco), Chile (Magallanes Region), Falkland Islands, Saint Pierre and Miquelon, Uruguay	Brazil (Federal District, Minas Gerais, Paraná, Rio Grande do Sul, Rio de Janeiro, São Paulo), Greenland (most)
UTC−02:00	2019-01-08T07:51:30-02:00	Brazil (Fernando de Noronha), South Georgia and the South Sandwich Islands	
UTC−01:00	2019-01-08T08:51:30-01:00	Cape Verde	Portugal (Azores)
UTC±00:00	2019-01-08T09:51:30+00:00	Ivory Coast, Ghana, Iceland, Saint Helena, Senegal, Mali	Faroe Islands, Ireland, Portugal (continental, Madeira), Spain (Canary Islands), United Kingdom
UTC+01:00	2019-01-08T10:51:30+01:00	Algeria, Angola, Benin, Cameroon, Democratic Republic of the Congo (west), Gabon, Morocco, Niger, Nigeria, Tunisia, Western Sahara	Albania, Andorra, Austria, Belgium, Bosnia and Herzegovina, Croatia, Czech Republic, Denmark, France (metropolitan), Germany, Hungary, Italy, Liechtenstein, Luxembourg, Monaco, Malta, Macedonia, Netherlands (European), Norway, Poland, Serbia, Slovakia, Slovenia, Spain (continental), Sweden, Switzerland, Vatican City
UTC+02:00	2019-01-08T11:51:30+02:00	Burundi, Egypt, Malawi, Mozambique, Namibia, Russia (Kaliningrad), Rwanda, South Africa, Sudan, Swaziland, Zambia, Zimbabwe	Bulgaria, Cyprus, Estonia, Finland, Greece, Israel, Jordan, Latvia, Lebanon, Lithuania, Moldova, Palestine, Romania, Syria, Ukraine
UTC+03:00	2019-01-08T12:51:30+03:00	Belarus, Djibouti, Eritrea, Ethiopia, Iraq, Kenya, Kuwait, Madagascar, Russia (most of European part), Saudi Arabia, Qatar, Somalia, South Sudan, Tanzania, Turkey, Uganda, Yemen,	

UTC Offset	Timestamp	Countries (col 1)	Countries (col 2)
UTC+03:30	2019-01-08T13:21:31+03:30		Iran
UTC+04:00	2019-01-08T13:51:31+04:00	Armenia, Azerbaijan, Georgia, Mauritius, Oman, Russia (Samara), Seychelles, United Arab Emirates	
UTC+04:30	2019-01-08T14:21:31+04:30	Afghanistan	
UTC+05:00	2019-01-08T14:51:31+05:00	Kazakhstan (west), Maldives, Pakistan, Russia (Sverdlovsk, Chelyabinsk), Uzbekistan	
UTC+05:30	2019-01-08T15:21:31+05:30	India, Sri Lanka	
UTC+05:45	2019-01-08T15:36:31+05:45	Nepal	
UTC+06:00	2019-01-08T15:51:31+06:00	Bangladesh, Bhutan, British Indian Ocean Territory, Kazakhstan (most), Russia (Omsk)	
UTC+06:30	2019-01-08T16:21:31+06:30	Cocos Islands, Myanmar	
UTC+07:00	2019-01-08T16:51:31+07:00	Cambodia, Indonesia (west), Laos, Mongolia (west), Russia (Krasnoyarsk), Thailand, Vietnam	
UTC+08:00	2019-01-08T17:51:31+08:00	Australia (Western Australia), Brunei, People's Republic of China, Hong Kong, Indonesia (central), Macau, Malaysia, Mongolia (most), Philippines, Russia (Irkutsk), Singapore, Republic of China	
UTC+08:45	2019-01-08T18:36:31+08:45	Australia (Eucla unofficial)	
UTC+09:00	2019-01-08T18:51:31+09:00	Timor-Leste, Indonesia (east), Japan, North Korea[20], Russia (most of Sakha), South Korea	
UTC+09:30	2019-01-08T19:21:31+09:30	Australia (Northern Territory)	Australia (South Australia)
UTC+10:00	2019-01-08T19:51:31+10:00	Australia (Queensland), Papua New Guinea, Russia (Primorsky)	Australia (New South Wales, Tasmania, Victoria)
UTC+10:30	2019-01-08T20:21:31+10:30		Lord Howe Island

UTC+11:00	2019-01-08T20:51:31+11:00	New Caledonia, Russia (Magadan), Solomon Islands, Vanuatu	
UTC+12:00	2019-01-08T21:51:31+12:00	Kiribati (Gilbert Islands), Russia (Kamchatka)	Fiji, New Zealand (most)
UTC+12:45	2019-01-08T22:36:31+12:45		New Zealand (Chatham Islands)
UTC+13:00	2019-01-08T22:51:31+13:00	Kiribati (Phoenix Islands), Tokelau, Tonga	Samoa
UTC+14:00	2019-01-08T23:51:31+14:00	Kiribati (Line Islands)	

Where the adjustment for time zones results in a time at the other side of midnight from UTC, then the date at the location is one day later or earlier.

Some examples when UTC is 23:00 on Monday when or where daylight saving time is not in effect:

- Cairo, Egypt: UTC+02; 01:00 on Tuesday

- Wellington, New Zealand: UTC+12; 11:00 on Tuesday

Some examples when UTC is 02:00 on Tuesday when or where daylight saving time is not in effect:

- Honolulu, Hawaii, United States: UTC−10; 16:00 on Monday

- Toronto, Ontario, Canada: UTC−05; 21:00 on Monday

The time-zone adjustment for a specific location may vary because of daylight saving time. For example, New Zealand, which is usually UTC+12, observes a one-hour daylight saving time adjustment during the Southern Hemisphere summer, resulting in a local time of UTC+13.

Time zone conversions

show**Time of day by zone**

Conversion between time zones obeys the relationship

"time in zone A" − "UTC offset for zone A" = "time in zone B" − "UTC offset for zone B",

in which each side of the equation is equivalent to UTC. (The more familiar term "UTC offset" is used here rather than the term "zone designator" used by the standard.)

The conversion equation can be rearranged to

"time in zone B" = "time in zone A" − "UTC offset for zone A" + "UTC offset for zone B".

For example, the New York Stock Exchange opens at 09:30 (EST, UTC offset=−05:00). In Los Angeles (PST, UTC offset= −08:00) and Delhi (IST, UTC offset= +05:30), the New York Stock Exchange opens at

time in Los Angeles = 09:30 − (−05:00) + (−08:00) = 06:30.

time in Delhi = 09:30 − (−05:00) + (+05:30) = 20:00.

These calculations become more complicated near a daylight saving boundary (because the UTC offset for zone X is a function of the UTC time).

The table "Time of day by zone" gives an overview on the time relations between different zones.

Nautical time zones

Main article: Nautical time

Since the 1920s a nautical standard time system has been in operation for ships on the high seas. Nautical time zones are an ideal form of the terrestrial time zone system. Under the system, a time change of one hour is required for each change of longitude by 15°. The 15° gore that is offset from GMT or UT1 (not UTC) by twelve hours is bisected by the nautical date line into two 7.5° gores that differ from GMT by ±12 hours. A nautical date line is implied but not explicitly drawn on time zone maps. It follows the 180th meridian except where it is interrupted by territorial waters adjacent to land, forming gaps: it is a pole-to-pole dashed line.[21][22][23]

A ship within the territorial waters of any nation would use that nation's standard time, but would revert to nautical standard time upon leaving its territorial waters. The captain is permitted to change the ship's clocks at a time of the captain's choice following the ship's entry into another time zone. The captain often chooses midnight. Ships going in shuttle traffic over a time zone border often keep the same time zone all the time, to avoid confusion about work, meal, and shop opening hours. Still the time table for port calls must follow the land time zone.

Skewing of zones

Difference between sun time and clock time during daylight saving time:

 1h ± 30 min behind

 0h ± 30m

 1h ± 30m ahead

 2h ± 30m ahead

 3h ± 30m ahead
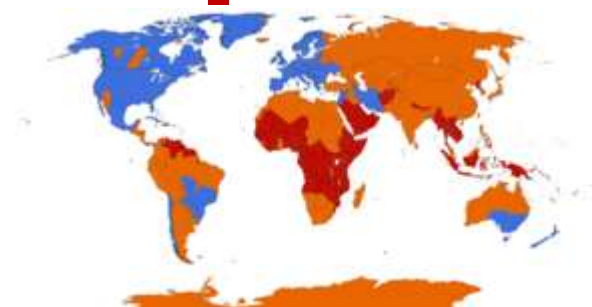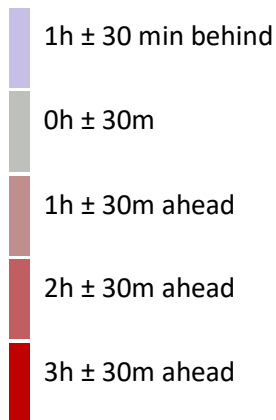
 DST observed

 DST formerly observed

 DST never observed

Ideal time zones, such as nautical time zones, are based on the mean solar time of a particular meridian located in the middle of that zone with boundaries located 7.5 degrees east and west of the meridian. In practice, zone boundaries are often drawn much farther to the west with often irregular boundaries, and some locations base their time on meridians located far to the east.

For example, even though the Prime Meridian (0°) passes through Spain and France, they use the mean solar time of 15 degrees east (Central European Time) rather than 0 degrees (Greenwich Mean Time). France previously used GMT, but was switched to CET (Central European Time) during the German occupation of the country during World War II and did not switch back after the war.[24] Similarly, prior to World War II, the Netherlands observed "Amsterdam Time", which was twenty minutes ahead of Greenwich Mean Time. They were obliged to follow German time during the war, and kept it thereafter. In the mid 1970s the Netherlands, as with other European states, began observing daylight saving (summer) time.

There is a tendency to draw time zone boundaries far to the west of their meridians. The main reason for this is that similar working day schedules around the world have led to people rising on average at 07:00 clock time and going to bed at 23:00 clock time[citation needed]. Another reason is that it can allow the more efficient use of sunlight.[25] This means that the middle of the period that people are awake ("*awake time noon*") occurs at 15:00 (= [7 + 23]/2) clock time, whereas - if using as clock time the time of the nautical time zone to which the location concerned geographically belongs - solar noon occurs at 12:00 (+/- 30 min) clock time. To make solar noon coincide more with *awake time noon* (i.e. make the sun reach its highest point closer to 15:00 clock time rather than 12:00 clock time), the time of one or even two nautical time zones to the east is chosen[citation needed]. Many of these locations also use DST, adding yet another nautical time zone to the east. As a result,[note 1] in summer, solar noon in the Spanish town of Muxía occurs at 14:37 clock time, indeed very close to *awake time noon* (15:00). This westernmost area of continental Spain never experiences sunset before 18:00 clock time, even in midwinter, despite its lying more than 40 degrees north of the equator. Near the summer solstice, Muxía has sunset times (after 22:00) similar to those of Stockholm, which is in the same time zone and 16 degrees further north. Stockholm has much earlier sunrises, though.

A more extreme example is Nome, Alaska, which is at 165°24′W longitude—just west of center of the idealized Samoa Time Zone (165°W). Nevertheless, Nome observes Alaska Time (135°W) with DST so it is slightly more than two hours ahead of the sun in winter and over three in summer.[26] Kotzebue, Alaska, also near the same meridian but north of the Arctic Circle, has an annual event on August 9 to celebrate *two* sunsets in the same 24-hour day, one shortly after midnight at the start of the day, and the other shortly before midnight at the end of the day.

Also, China extends as far west as 73°34′E, but all parts of it use UTC+08:00 (120°E), so solar "noon" can occur as late as 15:00 in western portions of China such as Xinjiang and Tibet.

Daylight saving time

Main article: Daylight saving time

Many countries, and sometimes just certain regions of countries, adopt daylight saving time (also known as "Summer Time") during part of the year. This typically involves advancing clocks by an hour near the start of spring and adjusting back in autumn ("spring forward", "fall back"). Modern DST was first proposed in 1907 and was in widespread use in 1916 as a wartime measure aimed at conserving coal. Despite controversy, many countries have used it off and on since then; details vary by location and change occasionally. Most countries around the equator do not observe daylight saving time, since the seasonal difference in sunlight is minimal.

Computer systems and the Internet

Computer operating systems include the necessary support for working with all (or almost all) possible local times based on the various time zones. Internally, operating systems typically use UTC as their basic time-keeping standard, while providing services for converting local times to and from UTC, and also the ability to automatically change local time conversions at the start and end of daylight saving time in the various time zones. (See the article on daylight saving time for more details on this aspect).

Web servers presenting web pages primarily for an audience in a single time zone or a limited range of time zones typically show times as a local time, perhaps with UTC time in brackets. More internationally oriented websites may show times in UTC only or using an arbitrary time zone. For example, the international English-language version of CNN includes GMT and Hong Kong Time,[27] whereas the US version shows Eastern Time.[28] US Eastern Time and Pacific Time are also used fairly commonly on many US-based English-language websites with global readership. The format is typically based in the W3C Note "datetime".

Email systems and other messaging systems (IRC chat, etc.)[29] time-stamp messages using UTC, or else include the sender's time zone as part of the message, allowing the receiving program to display the message's date and time of sending in the recipient's local time.

Database records that include a time stamp typically use UTC, especially when the database is part of a system that spans multiple time zones. The use of local time for time-stamping records is not recommended for time zones that implement daylight saving time because once a year there is a one-hour period when local times are ambiguous.

Calendar systems nowadays usually tie their time stamps to UTC, and show them differently on computers that are in different time zones. That works when having telephone or internet meetings. It works less well when travelling, because the calendar events are assumed to take place in the time zone the computer or smartphone was on when creating the event. The event can be shown at the wrong time. For example, if a New Yorker plans to meet someone in Los Angeles at 9 AM, and makes a calendar entry at 9 AM (which the computer assumes is New York time), the calendar entry will be at 6 AM if taking the computer's time zone. There is also an option in newer versions of Microsoft Outlook to enter the time zone in which an event will happen, but often not in other calendar systems. Calendaring software must also deal with daylight saving time (DST). If, for political reasons, the begin and end dates of daylight saving time are changed, calendar entries should stay the same in local time, even though they may shift in UTC time. In Microsoft Outlook, time stamps are therefore stored and communicated without DST offsets.[30] Hence, an appointment in London at noon in the summer will be represented as 12:00 (UTC+00:00) even though the event will actually take place at 13:00 UTC. In Google Calendar, calendar events are stored in UTC (although shown in local time) and might be changed by a time-zone changes,[31] although normal daylight saving start and end are compensated for (similar to much other calendar software).

Operating systems

Unix

Main article: Unix time

Most Unix-like systems, including Linux and Mac OS X, keep system time in time_t format, representing the number of seconds that have elapsed since 00:00:00 Coordinated Universal Time (UTC) on Thursday, January 1, 1970,[32]. By default the external representation is as UTC (Coordinated Universal Time), though individual processes can specify time zones using the LC_TIME environment variable or less commonly the TZ environment variable. This allows users in multiple timezones to use the same computer, with their respective local times displayed correctly to each user. Time zone information most commonly comes from the IANA time zone database. In fact, many systems, including anything using the GNU C Library, can make use of this database.

Microsoft Windows

Windows-based computer systems prior to Windows 2000 used local time, but Windows 2000 and later can use UTC as the basic system time.[33] The system registry contains time zone information that includes the offset from UTC and rules that indicate the start and end dates for daylight saving in each zone. Interaction with the user normally uses local time, and application software is able to calculate the time in various zones. Terminal Servers allow remote computers to redirect their time zone settings to the Terminal Server so that users see the correct time for their time zone in their desktop/application sessions. Terminal Services uses the server base time on the Terminal Server and the client time zone information to calculate the time in the session.

Programming languages

Java

While most application software will use the underlying operating system for timezone information, the Java Platform, from version 1.3.1, has maintained its own timezone database. This database is updated whenever timezone rules change. Oracle provides an updater tool for this purpose.[34]

As an alternative to the timezone information bundled with the Java Platform, programmers may choose to use the Joda-Time library.[35] This library includes its own timezone data based on the IANA time zone database.[36]

As of Java 8 new DATE TIME API is there that can help converting timezones. Java 8 Date Time

JavaScript

There is very little in the way of timezone support for JavaScript. Essentially the programmer has to extract the UTC offset by instantiating a time object, getting a GMT time from it, and differencing the two. This does not provide a solution for daylight saving variations.

Perl

The DateTime object in Perl supports all time zones in the Olson DB and includes the ability to get, set and convert between time zones.[37]

PHP

The DateTime objects and related functions have been compiled into the PHP core since 5.2. This includes the ability to get and set the default script timezone, and DateTime is aware of its own timezone internally. PHP.net provides extensive documentation on this.[38] As noted there, the most current timezone database can be implemented via the PECL timezonedb.

Python

The standard module datetime included with Python stores and operates on the timezone information class tzinfo. The third party pytz module provides access to the full IANA time zone database.[39] Negated time zone offset in seconds is stored time.timezone and time.altzone attributes.

Smalltalk

Each Smalltalk dialect comes with its own built-in classes for dates, times and timestamps, only a few of which implement the DateAndTime and Duration classes as specified by the ANSI Smalltalk Standard. VisualWorks provides a TimeZone class that supports up to two annually recurring offset transitions, which are assumed to apply to all years (same behavior as Windows time zones). Squeak provides a Timezone class that does not support any offset transitions. Dolphin Smalltalk does not support time zones at all.

For full support of the tz database (zoneinfo) in a Smalltalk application (including support for any number of annually recurring offset transitions, and support for different intra-year offset transition rules in different years) the third-party, open-source, ANSI-Smalltalk-compliant Chronos Date/Time Library is available for use with any of the following Smalltalk dialects: VisualWorks, Squeak, Gemstone, or Dolphin.[40]

Time zones in outer space

This section **needs additional citations for verification**. Please help improve this article by adding citations to reliable sources. Unsourced material may be challenged and removed. *(September 2015)(Learn how and when to remove this template message)*

Orbiting spacecraft typically experience many sunrises and sunsets in a 24-hour period, or in the case of Apollo program astronauts travelling to the moon, none. Thus it is not possible to calibrate time zones with respect to the sun, and still respect a 24-hour sleep/wake cycle. A common practice for space exploration is to use the Earth-based time zone of the launch site or mission control. This keeps the sleeping cycles of the crew and controllers in sync. The International Space Station normally uses Greenwich Mean Time (GMT).[41][42]

Timekeeping on Mars can be more complex, since the planet has a solar day of approximately 24 hours and 39 minutes, known as a sol. Earth controllers for some Mars missions have synchronized their sleep/wake cycles with the Martian day,[43] because solar-powered rover activity on the surface was tied to periods of light and dark. The difference in day length caused the sleep/wake cycles to slowly drift with respect to the day/night cycles on Earth, repeating approximately once every 36 days.

See also

The control panel of the Time Zone Clock in front of Coventry Transport Museum.

- Daylight saving time

- ISO 8601

- Lists of time zones

- Time in Europe

- Metric time

- Time by country

- World clock

- Jet lag

Notes

1. ^ Spain may have chosen its time zone for other reasons, such as synchronising with trading partners, and adopting CET as a major member of the EU

References

1. ^ *The Mechanics of Mechanical Watches and Clocks | Ruxu Du | Springer.*

2. ^ Latitude and Longitude of World Cities http://www.infoplease.com/ipa/A0001769.html

3. ^ *"Bristol Time". Wwp.greenwichmeantime.com. Retrieved December 5, 2011.*

4. ^ *"Our Time. How we got it. New Zealand's Method. A Lead to the World". Papaerspast. Evening Post. p. 10. Retrieved October 2,2013.*

5. ^ *Alfred, Randy (November 18, 2010). "Nov. 18, 1883: Railroad Time Goes Coast to Coast". Wired. Retrieved July 30, 2018.*

6. ^ *"Economics of Time Zones" (PDF). Archived from the original (PDF) on May 14, 2012.* (1.89 MB)

7. ^ *"Historymatters.gmu.edu". Historymatters.gmu.edu. Retrieved December 5, 2011.*

8. ^ *"Resolution concerning new standard time by Chicago". Sos.state.il.us. Archived from the original on October 5, 2011. Retrieved December 5, 2011.*

9. ^ Quirico Filopanti from scienzagiovane, Bologna University, Italy. Archived January 17, 2013, at the Wayback Machine.

10. **^** Gianluigi Parmeggiani (Osservatorio Astronomico di Bologna),The origin of time zones Archived August 24, 2007, at the Wayback Machine.

11. **^** *"History & info - Standard time began with the railroads". www.webexhibits.org. Retrieved February 13, 2018.*

12. **^** *International conference held at Washington for the Purpose of Fixing a Prime Meridian and a Universal Day. October, 1884. Protocols of the proceedings., 1884, p. 201, retrieved July 23,2018*

13. **^** *"15 minutes of fame - Nepali Times". archive.nepalitimes.com. Retrieved August 22, 2018.*

14. **^** *Schiavenza, Matt (November 5, 2013). "China Only Has One Time Zone—and That's a Problem". The Atlantic. Retrieved August 22, 2018.*

15. **^** *"In Canada, You Can Just Write the Date Whichever Way You Want". Atlas Obscura. June 8, 2015. Retrieved August 22,2018.*

16. **^** *"Z – Zulu Time Zone (Time Zone Abbreviation)". www.timeanddate.com. Retrieved August 22, 2018.*

17. **^** *"Z – Zulu Time Zone (Time Zone Abbreviation)". www.timeanddate.com. Retrieved August 22, 2018.*

18. **^** *"What is UTC or GMT Time?". www.nhc.noaa.gov. Retrieved August 22, 2018.*

19. **^** *"Time zone ambiguities on Linux". www.pixelbeat.org. Retrieved August 22, 2018.*

20. **^** *Taylor, Adam (May 4, 2018). "The brief history of North Korea's time zone". Washington Post.*

21. **^** Bowditch, Nathaniel. *American Practical Navigator*. Washington: Government Printing Office, 1925, 1939, 1975.

22. **^** Hill, John C., Thomas F. Utegaard, Gerard Riordan. *Dutton's Navigation and Piloting*. Annapolis: United States Naval Institute, 1958.

23. **^** Howse, Derek. *Greenwich Time and the Discovery of the Longitude*. Oxford: Oxford University Press, 1980. ISBN 0-19-215948-8.

24. **^** *Poulle, Yvonne (1999). "La France à l'heure allemande"(PDF). Bibliothèque de l'école des chartes. 157 (2): 493–502. doi:10.3406/bec.1999.450989. Retrieved January 11, 2012.*

25. **^** *"法定时与北京时间". 人民教育出版社. Archived from the original on November 14, 2006.*

26. **^** *Doug O'Hara (March 11, 2007). "Alaska: daylight stealing time". Far North Science. Retrieved May 11, 2007.*

27. **^** *"International CNN". Edition.cnn.com. Retrieved December 5,2011.*

28. **^** *"United States CNN". Cnn.com. Retrieved December 5, 2011.*

29. **^** *"Guidelines for Ubuntu IRC Meetings". Canonical Ltd. August 6, 2008.*

30. **^** How time zone normalization works in Microsoft Outlook. Microsoft (2015).

31. **^** Use Google Calendar in different time zones. Google Calendar Help (as of Oct. 2015)

32. **^** *"The Open Group Base Specifications Issue 7, section 4.16 Seconds Since the Epoch". The Open Group. Retrieved January 22, 2017.*

33. **^** *"GetSystemTime function (Windows)". msdn.microsoft.com. Retrieved February 13, 2018.*

34. **^** *"Timezone Updater Tool". Java.sun.com. Retrieved December 5, 2011.*

35. **^** *"Joda-Time". Joda-time.sourceforge.net. Retrieved December 5, 2011.*

36. **^** *"tz database". Twinsun.com. December 26, 2007. Archived from the original on June 23, 2012. Retrieved December 5,2011.*

37. **^** *"DateTime". METACPAN. Retrieved April 14, 2014.*

38. **^** *"DateTime". Php.net. Retrieved December 5, 2011.*

39. **^** *"pytz module". Pytz.sourceforge.net. Retrieved December 5,2011.*

40. **^** Chronos Date/Time Library Archived April 5, 2014, at the Wayback Machine.

41. **^** *"Ask the Crew: STS-111".*

42. **^** *Ed Lu. "Day in the Life".*

43. **^** Megan Gannon, 2008, New Tricks Could Help Mars Rover Team Live on Mars Time, *space.com*