

Время: звёзды, атомы и тараканы

Дмитрий Лапшин

ЛКШ 2016.Июль

9 июля

Что такое время?

- Кто-нибудь знает, что такое время?

Что такое время?

- Кто-нибудь знает, что такое время?
- На самом деле, всё, что мы можем сказать — некоторые события происходят *раньше* или *позже* других.

Что такое время?

- Кто-нибудь знает, что такое время?
- На самом деле, всё, что мы можем сказать — некоторые события происходят *раньше* или *позже* других.
- Какие-то из них происходят периодически (то есть: повторяются через примерно равные *промежутки* времени).

Что такое время?

- Кто-нибудь знает, что такое время?
- На самом деле, всё, что мы можем сказать — некоторые события происходят *раньше* или *позже* других.
- Какие-то из них происходят периодически (то есть: повторяются через примерно равные *промежутки* времени).
- И нам интересно, когда они повторятся.

Какие периодические события вы знаете?

Какие периодические события вы знаете?

На Земле, больше всего для нас важны:

День: Оборот Земли вокруг своей оси.
Его части: час, минута, секунда.

Год: Оборот Земли вокруг Солнца.
Его части: месяц, неделя.

Какие периодические события вы знаете?

На Земле, больше всего для нас важны:

День: Оборот Земли вокруг своей оси.
Его части: час, минута, секунда.

Год: Оборот Земли вокруг Солнца.
Его части: месяц, неделя.

На других планетах свои понятия дня и года. Например:

Сол: Сутки на Марсе, чуть длиннее земных.

Лунный месяц: Полный цикл фаз Луны, соответствует обороту Луны вокруг Земли относительно Солнца (почти лунный год).

День

- Днём мы называем среднюю продолжительность солнечных суток.

День

- Днём мы называем среднюю продолжительность солнечных суток.
- Один день содержит 24 часа, час — 60 минут, минута — 60 секунд.

День

- Днём мы называем среднюю продолжительность солнечных суток.
- Один день содержит 24 часа, час — 60 минут, минута — 60 секунд. Меньше используют кратные приставки (милли-, микро-, нано-).

День

- Днём мы называем среднюю продолжительность солнечных суток.
- Один день содержит 24 часа, час — 60 минут, минута — 60 секунд. Меньше используют кратные приставки (милли-, микро-, нано-).
- Давным-давно в Вавилоне любили 60-ричные системы счисления, оттуда и пришло.

Год, Юлианский календарь

- Ещё в древнем Египте считали, что год состоит из 365 дней, которые делились на 36 декад по 10 дней, и ещё 5 дней.

Год, Юлианский календарь

- Ещё в древнем Египте считали, что год состоит из 365 дней, которые делились на 36 декад по 10 дней, и ещё 5 дней.
- Но с чего бы году делиться нацело на день? И не делится!

Год, Юлианский календарь

- Ещё в древнем Египте считали, что год состоит из 365 дней, которые делились на 36 декад по 10 дней, и ещё 5 дней.
- Но с чего бы году делиться нацело на день? И не делится!
- Юлианский календарь (введён в Римской империи с 45 года до н.э.) уже содержит привычные нам месяцы и недели, а также *високосные* года (был каждый четвёртый).

Год, Юлианский календарь

- Ещё в древнем Египте считали, что год состоит из 365 дней, которые делились на 36 декад по 10 дней, и ещё 5 дней.
- Но с чего бы году делиться нацело на день? И не делится!
- Юлианский календарь (введён в Римской империи с 45 года до н.э.) уже содержит привычные нам месяцы и недели, а также *високосные* года (был каждый четвёртый). В високосный год в февраль добавлялся один день. В среднем, это 365.25 дней в году.

Год, Юлианский календарь

- Ещё в древнем Египте считали, что год состоит из 365 дней, которые делились на 36 декад по 10 дней, и ещё 5 дней.
- Но с чего бы году делиться нацело на день? И не делится!
- Юлианский календарь (введён в Римской империи с 45 года до н.э.) уже содержит привычные нам месяцы и недели, а также *високосные* года (был каждый четвёртый). В високосный год в февраль добавлялся один день. В среднем, это 365.25 дней в году.
- Как вы думаете, это точно?

Григорианский календарь

- Как выяснилось, нет. За многие годы астрономы измерили движение небесных тел гораздо точнее, в том числе — движение Солнца и Земли.

Григорианский календарь

- Как выяснилось, нет. За многие годы астрономы измерили движение небесных тел гораздо точнее, в том числе — движение Солнца и Земли.
- Длина солнечного 2000 года — 365.2421897 дней.

Григорианский календарь

- Как выяснилось, нет. За многие годы астрономы измерили движение небесных тел гораздо точнее, в том числе — движение Солнца и Земли.
- Длина солнечного 2000 года — 365.2421897 дней.
- В 1582 году в католических странах был введён Григорианский календарь (в остальных — позже ;). Високосным в нём считался год, номер которого делится на 400, а также те, что делятся на 4, но не делятся на 100.

Григорианский календарь

- Как выяснилось, нет. За многие годы астрономы измерили движение небесных тел гораздо точнее, в том числе — движение Солнца и Земли.
- Длина солнечного 2000 года — 365.2421897 дней.
- В 1582 году в католических странах был введён Григорианский календарь (в остальных — позже ;). Високосным в нём считался год, номер которого делится на 400, а также те, что делятся на 4, но не делятся на 100.
- Это дало среднюю длину года в 365.2425 дней. Чтобы разойтись на сутки с Солнцем, этому календарю нужно порядка 10 000 лет (против 128 у Юлианского).

Григорианский календарь

- Как выяснилось, нет. За многие годы астрономы измерили движение небесных тел гораздо точнее, в том числе — движение Солнца и Земли.
- Длина солнечного 2000 года — 365.2421897 дней.
- В 1582 году в католических странах был введён Григорианский календарь (в остальных — позже ;). Високосным в нём считался год, номер которого делится на 400, а также те, что делятся на 4, но не делятся на 100.
- Это дало среднюю длину года в 365.2425 дней. Чтобы разойтись на сутки с Солнцем, этому календарю нужно порядка 10 000 лет (против 128 у Юлианского).
- *Пока* это всех устраивает...

Часовые пояса

- Все хотят, чтобы в начале дня солнце вставало, а вечером садилось. Но в разных местах земли это происходит в разные моменты времени...

Часовые пояса

- Все хотят, чтобы в начале дня солнце вставало, а вечером садилось. Но в разных местах земли это происходит в разные моменты времени...
- Решение — часовые пояса. В каждой местности своё время, чтобы в полдень солнце было в зените.

Часовые пояса

- Все хотят, чтобы в начале дня солнце вставало, а вечером садилось. Но в разных местах земли это происходит в разные моменты времени...
- Решение — часовые пояса. В каждой местности своё время, чтобы в полдень солнце было в зените.
- Понятно, что на каждом меридиане своё такое время, но это неудобно.

GMT и друзья

- Часовые пояса традиционно вводятся как разница с временем обсерватории в Гринвиче, Лондон.

GMT и друзья

- Часовые пояса традиционно вводятся как разница с временем обсерватории в Гринвиче, Лондон.
- Время там называется **Greenwich Mean Time (GMT)**.

GMT и друзья

- Часовые пояса традиционно вводятся как разница с временем обсерватории в Гринвиче, Лондон.
- Время там называется **Greenwich Mean Time (GMT)**.
- ЛКШ находится (в момент создания презентации) в часовом поясе **GMT +03:00**.

GMT и друзья

- Часовые пояса традиционно вводятся как разница с временем обсерватории в Гринвиче, Лондон.
- Время там называется **Greenwich Mean Time (GMT)**.
- ЛКШ находится (в момент создания презентации) в часовом поясе **GMT +03:00**.
- Ещё популярные обозначения поясов:
 - PST: Pacific Standart Time**, равно **GMT -08:00**, любят в Америке.
 - MSK: Московское время**.

Линия перемены дат

- Как несложно заметить, на карте была граница часовых поясов **GMT +12:00** и **GMT -12:00**

Линия перемены дат

- Как несложно заметить, на карте была граница часовых поясов **GMT +12:00** и **GMT -12:00**
- При её пересечении надо не забыть перевести часы на *сутки* (плюс-минус разности часов). Помните «Вокруг света за 80 дней»?

Летнее время

- Летом Солнце встаёт раньше, а время такое же. Хотелось бы, чтобы утром, пока все спят, Солнце не вставало, а наоборот вечером было дольше. Ещё иногда что-то говорят про экономию электричества на освещении...

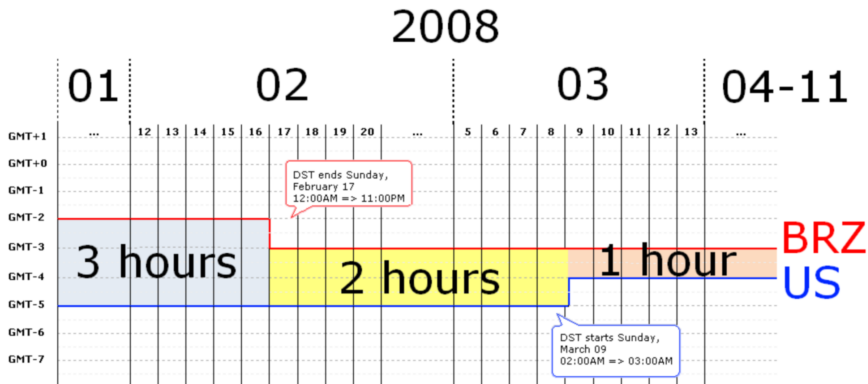
Летнее время

- Летом Солнце встаёт раньше, а время такое же. Хотелось бы, чтобы утром, пока все спят, Солнце не вставало, а наоборот вечером было дольше. Ещё иногда что-то говорят про экономию электричества на освещении...
- Идея: переводить часы летом на некоторое время вперёд, чтобы с точки зрения часов Солнце вставало позже, а события происходили раньше. Часовой пояс на время такого перехода помечается как **Daylight saving time (DST)**.

Летнее время

- Летом Солнце встаёт раньше, а время такое же. Хотелось бы, чтобы утром, пока все спят, Солнце не вставало, а наоборот вечером было дольше. Ещё иногда что-то говорят про экономию электричества на освещении...
- Идея: переводить часы летом на некоторое время вперёд, чтобы с точки зрения часов Солнце вставало позже, а события происходили раньше. Часовой пояс на время такого перехода помечается как **Daylight saving time (DST)**.
- Обычно переводят на час, но кто, когда, на сколько, каждый решает сам.

«Веселье» с летним временем



See this graph and others on www.daylight-savings-time.info

Ещё «веселья»

- Все эти часовые пояса не так сложно учитывать.

Ещё «веселья»

- Все эти часовые пояса не так сложно учитывать... Если бы они не менялись!

Ещё «веселья»

- Все эти часовые пояса не так сложно учитывать... Если бы они не менялись!
- В России часовые пояса так или иначе модифицировались в 1992, 2012 и 2014, например. В том числе отменили летнее время.

Ещё «веселья»

- Все эти часовые пояса не так сложно учитывать... Если бы они не менялись!
- В России часовые пояса так или иначе модифицировались в 1992, 2012 и 2014, например. В том числе отменили летнее время.
- По миру, часовые пояса меняются много раз за год. Зачем? Не знаю, им хочется.

Ещё «веселья»

- Все эти часовые пояса не так сложно учитывать... Если бы они не менялись!
- В России часовые пояса так или иначе модифицировались в 1992, 2012 и 2014, например. В том числе отменили летнее время.
- По миру, часовые пояса меняются много раз за год. Зачем? Не знаю, им хочется.
- Если у вас билет на самолёт, который летит из или в территорию с изменяющимся часовым поясом — будьте внимательны, лететь час меньше или больше самолёт не сможет.

Точное время

- Последние два века появилась потребность в ещё более точном времени для малых интервалов времени. Это также связано с изучением микромира и ядерной физики.

Точное время

- Последние два века появилась потребность в ещё более точном времени для малых интервалов времени. Это также связано с изучением микромира и ядерной физики.
- Были созданы сверхточные атомные часы, измеряющие колебания определённых атомов.

Точное время

- Последние два века появилась потребность в ещё более точном времени для малых интервалов времени. Это также связано с изучением микромира и ядерной физики.
- Были созданы сверхточные атомные часы, измеряющие колебания определённых атомов.
- Раньше секунда определялась как некоторая часть какого-то периода обращения небесных тел (суток, года, лунного месяца...) или периода колебания маятника.
- С 1967 в системе СИ секунда определяется как ровно **9 192 631 770** периодов колебания атома цезия-133 (^{133}Cs).

А как же $1/86400$ часть суток?

- Когда создавали СИ, секунда соответствовала нужной части суток.

А как же $1/86400$ часть суток?

- Когда создавали СИ, секунда соответствовала нужной части суток.
- А сейчас нет. Приливы замедляют вращение Земли.

А как же $1/86400$ часть суток?

- Когда создавали СИ, секунда соответствовала нужной части суток.
- А сейчас нет. Приливы замедляют вращение Земли.
- А ещё мы научились замечать, как длина дня «колеблется» с периодом в несколько лет. Полюса нашей планеты немного двигаются...
- А ещё мы знаем, что на поверхности Земли время искажается Землёй, следуя теориям относительности Эйнштейна...

TAI и UTC

- В середине прошлого века были введены новые мировые стандарты времени, на замену GMT.

TAI и UTC

- В середине прошлого века были введены новые мировые стандарты времени, на замену GMT.
- Координированное атомное время, измеряемое на Земле в секундах СИ, называется **International Atomic Time (TAI)**. Оно используется в сверхточных замерах времени, например в современной астрономии и навигации.

TAI и UTC

- В середине прошлого века были введены новые мировые стандарты времени, на замену GMT.
- Координированное атомное время, измеряемое на Земле в секундах СИ, называется **International Atomic Time (TAI)**. Оно используется в сверхточных замерах времени, например в современной астрономии и навигации.
- Для использования людьми на замену GMT пришло **Universal Coordinated Time (UTC)**, которое тоже измеряется в секундах СИ.

TAI и UTC

- В середине прошлого века были введены новые мировые стандарты времени, на замену GMT.
- Координированное атомное время, измеряемое на Земле в секундах СИ, называется **International Atomic Time (TAI)**. Оно используется в сверхточных замерах времени, например в современной астрономии и навигации.
- Для использования людьми на замену GMT пришло **Universal Coordinated Time (UTC)**, которое тоже измеряется в секундах СИ.
- Оно не соответствует солнечному времени **UT1**. Разность между ними постоянно измеряют и обозначают **DUT1**.

TAI и UTC

- В середине прошлого века были введены новые мировые стандарты времени, на замену GMT.
- Координированное атомное время, измеряемое на Земле в секундах СИ, называется **International Atomic Time (TAI)**. Оно используется в сверхточных замерах времени, например в современной астрономии и навигации.
- Для использования людьми на замену GMT пришло **Universal Coordinated Time (UTC)**, которое тоже измеряется в секундах СИ.
- Оно не соответствует солнечному времени **UT1**. Разность между ними постоянно измеряют и обозначают **DUT1**.
- Когда DUT1 по модулю больше или равна 0.9 секунд, в UTC вводится **секунда координации** (или високосная секунда).

Секунда координации

- Изначально UTC было скорректировано на 10 секунд от TAI. Далее, секунду координации (leap second) вводит руками International Earth Rotation and Reference Systems Service (IERS).

Секунда координации

- Изначально UTC было скорректировано на 10 секунд от TAI. Далее, секунду координации (leap second) вводит руками International Earth Rotation and Reference Systems Service (IERS).
- Секунда координации может вводиться в конце 30 июня и 31 декабря часового пояса Гринвича.

Секунда координации

- Изначально UTC было скорректировано на 10 секунд от TAI. Далее, секунду координации (leap second) вводит руками International Earth Rotation and Reference Systems Service (IERS).
- Секунда координации может вводиться в конце 30 июня и 31 декабря часового пояса Гринвича.
- Если секунду вводят для компенсации замедления вращения земли, эти сутки расширяются на 1 секунду:

Секунда координации

- Изначально UTC было скорректировано на 10 секунд от TAI. Далее, секунду координации (leap second) вводит руками International Earth Rotation and Reference Systems Service (IERS).
- Секунда координации может вводиться в конце 30 июня и 31 декабря часового пояса Гринвича.
- Если секунду вводят для компенсации замедления вращения земли, эти сутки расширяются на 1 секунду:

23:59:58 30 Jun 2015 UTC

Секунда координации

- Изначально UTC было скорректировано на 10 секунд от TAI. Далее, секунду координации (leap second) вводит руками International Earth Rotation and Reference Systems Service (IERS).
- Секунда координации может вводиться в конце 30 июня и 31 декабря часового пояса Гринвича.
- Если секунду вводят для компенсации замедления вращения земли, эти сутки расширяются на 1 секунду:

23:59:58 30 Jun 2015 UTC

23:59:59 30 Jun 2015 UTC

Секунда координации

- Изначально UTC было скорректировано на 10 секунд от TAI. Далее, секунду координации (leap second) вводит руками International Earth Rotation and Reference Systems Service (IERS).
- Секунда координации может вводиться в конце 30 июня и 31 декабря часового пояса Гринвича.
- Если секунду вводят для компенсации замедления вращения земли, эти сутки расширяются на 1 секунду:

23:59:58 30 Jun 2015 UTC

23:59:59 30 Jun 2015 UTC

23:59:60 30 Jun 2015 UTC

Секунда координации

- Изначально UTC было скорректировано на 10 секунд от TAI. Далее, секунду координации (leap second) вводит руками International Earth Rotation and Reference Systems Service (IERS).
- Секунда координации может вводиться в конце 30 июня и 31 декабря часового пояса Гринвича.
- Если секунду вводят для компенсации замедления вращения земли, эти сутки расширяются на 1 секунду:

23:59:58 30 Jun 2015 UTC

23:59:59 30 Jun 2015 UTC

23:59:60 30 Jun 2015 UTC

00:00:00 01 Jul 2015 UTC

Секунда координации

- Изначально UTC было скорректировано на 10 секунд от TAI. Далее, секунду координации (leap second) вводит руками International Earth Rotation and Reference Systems Service (IERS).
- Секунда координации может вводиться в конце 30 июня и 31 декабря часового пояса Гринвича.
- Если секунду вводят для компенсации замедления вращения земли, эти сутки расширяются на 1 секунду:

23:59:58 30 Jun 2015 UTC

23:59:59 30 Jun 2015 UTC

23:59:60 30 Jun 2015 UTC

00:00:00 01 Jul 2015 UTC

00:00:01 01 Jul 2015 UTC

Секунда координации

- Если же секунду введут для компенсации ускорения вращения Земли (чего ещё не случилось), эти сутки укорачиваются на 1 секунду:

Секунда координации

- Если же секунду введут для компенсации ускорения вращения Земли (чего ещё не случилось), эти сутки укорачиваются на 1 секунду:

23:59:57 30 Jun 2015 UTC

Секунда координации

- Если же секунду введут для компенсации ускорения вращения Земли (чего ещё не случилось), эти сутки укорачиваются на 1 секунду:

23:59:57 30 Jun 2015 UTC

23:59:58 30 Jun 2015 UTC

Секунда координации

- Если же секунду введут для компенсации ускорения вращения Земли (чего ещё не случилось), эти сутки укорачиваются на 1 секунду:

23:59:57 30 Jun 2015 UTC

23:59:58 30 Jun 2015 UTC

~~23:59:59 30 Jun 2015 UTC~~

00:00:00 01 Jul 2015 UTC

Секунда координации

- Если же секунду введут для компенсации ускорения вращения Земли (чего ещё не случилось), эти сутки укорачиваются на 1 секунду:

23:59:57 30 Jun 2015 UTC

23:59:58 30 Jun 2015 UTC

~~23:59:59 30 Jun 2015 UTC~~

00:00:00 01 Jul 2015 UTC

00:00:01 01 Jul 2015 UTC

Секунда координации

- Если же секунду введут для компенсации ускорения вращения Земли (чего ещё не случилось), эти сутки укорачиваются на 1 секунду:

23:59:57 30 Jun 2015 UTC

23:59:58 30 Jun 2015 UTC

~~23:59:59 30 Jun 2015 UTC~~

00:00:00 01 Jul 2015 UTC

00:00:01 01 Jul 2015 UTC

- Все, кто считал, что в минуте ровно 60 секунд — можете порадоваться.

Секунда координации

- Если же секунду введут для компенсации ускорения вращения Земли (чего ещё не случилось), эти сутки укорачиваются на 1 секунду:

23:59:57 30 Jun 2015 UTC

23:59:58 30 Jun 2015 UTC

~~23:59:59 30 Jun 2015 UTC~~

00:00:00 01 Jul 2015 UTC

00:00:01 01 Jul 2015 UTC

- Все, кто считал, что в минуте ровно 60 секунд — можете порадоваться.
- Високосная секунда происходит везде и сразу, вне зависимости от часового пояса (то есть в ЛКШ это будет не в полночь).

Все секунды

На нынешний момент, секунду координации ввели в:

Июнь 1972	Декабрь 1972	Декабрь 1973	Декабрь 1974
Декабрь 1975	Декабрь 1976	Декабрь 1977	Декабрь 1978
Декабрь 1979	Июнь 1981	Июнь 1982	Июнь 1983
Июнь 1985	Декабрь 1987	Декабрь 1989	Декабрь 1990
Июнь 1992	Июнь 1993	Июнь 1994	Декабрь 1995
Июнь 1997	Декабрь 1998	Декабрь 2005	Декабрь 2008
	Июнь 2012	Июнь 2015	

Все секунды

На нынешний момент, секунду координации ввели в:

Июнь 1972	Декабрь 1972	Декабрь 1973	Декабрь 1974
Декабрь 1975	Декабрь 1976	Декабрь 1977	Декабрь 1978
Декабрь 1979	Июнь 1981	Июнь 1982	Июнь 1983
Июнь 1985	Декабрь 1987	Декабрь 1989	Декабрь 1990
Июнь 1992	Июнь 1993	Июнь 1994	Декабрь 1995
Июнь 1997	Декабрь 1998	Декабрь 2005	Декабрь 2008
	Июнь 2012	Июнь 2015	

Текущая разность **TAI** - **UTC** равна 36 секундам.

Использование UTC

- Вместо GMT в мире используется UTC, со всеми часовыми поясами и летними переводами.

Использование UTC

- Вместо GMT в мире используется UTC, со всеми часовыми поясами и летними переводами.
- UTC — не самое удобное время. Високосные секунды вводятся руками, а не по алгоритму. В том числе — невозможно точно вычислить, сколько времени будет до момента в будущем.

Использование UTC

- Вместо GMT в мире используется UTC, со всеми часовыми поясами и летними переводами.
- UTC — не самое удобное время. Високосные секунды вводятся руками, а не по алгоритму. В том числе — невозможно точно вычислить, сколько времени будет до момента в будущем.
- Если хочется смотреть в далёкое будущее — используют TAI.

Использование UTC

- Вместо GMT в мире используется UTC, со всеми часовыми поясами и летними переводами.
- UTC — не самое удобное время. Високосные секунды вводятся руками, а не по алгоритму. В том числе — невозможно точно вычислить, сколько времени будет до момента в будущем.
- Если хочется смотреть в далёкое будущее — используют TAI.
- В том числе — GPS использует время, равное **TAI - 19с** (это соответствовало UTC в момент запуска системы).

Использование UTC

- Вместо GMT в мире используется UTC, со всеми часовыми поясами и летними переводами.
- UTC — не самое удобное время. Високосные секунды вводятся руками, а не по алгоритму. В том числе — невозможно точно вычислить, сколько времени будет до момента в будущем.
- Если хочется смотреть в далёкое будущее — используют TAI.
- В том числе — GPS использует время, равное **TAI - 19с** (это соответствовало UTC в момент запуска системы).
- А ещё обычно у людей часы не настолько точные, чтобы високосная секунда всё испортила.

А что мы хотим?

- Хочется, чтобы компьютерная техника считала время правильно.

А что мы хотим?

- Хочется, чтобы компьютерная техника считала время правильно.
- А ещё и хорошо считала длину между двумя моментами времени.

А что мы хотим?

- Хочется, чтобы компьютерная техника считала время правильно.
- А ещё и хорошо считала длину между двумя моментами времени.
- И лучше бы программы понимали наши запросы.

А что мы хотим?

- Хочется, чтобы компьютерная техника считала время правильно.
- А ещё и хорошо считала длину между двумя моментами времени.
- И лучше бы программы понимали наши запросы.
- В обычные компьютеры встроен тактовый генератор, позволяющий считать время в секундах СИ. Его точность не самая идеальная, но обычно хватает.

А что мы хотим?

- Хочется, чтобы компьютерная техника считала время правильно.
- А ещё и хорошо считала длину между двумя моментами времени.
- И лучше бы программы понимали наши запросы.
- В обычные компьютеры встроен тактовый генератор, позволяющий считать время в секундах СИ. Его точность не самая идеальная, но обычно хватает.
- Для сохранения точности нужно синхронизировать часы. Желательно — автоматически и через Интернет.

А что мы хотим?

- Хочется, чтобы компьютерная техника считала время правильно.
- А ещё и хорошо считала длину между двумя моментами времени.
- И лучше бы программы понимали наши запросы.
- В обычные компьютеры встроен тактовый генератор, позволяющий считать время в секундах СИ. Его точность не самая идеальная, но обычно хватает.
- Для сохранения точности нужно синхронизировать часы. Желательно — автоматически и через Интернет.
- И надо что-то придумать про часовые пояса, их изменения, летние переходы и високосные секунды.

Наивный подход

- Давайте храним строчку, в которую запишем всё, что думаем про время:

```
16:07:23 06.07.2016 UTC +03:00 DST <GPS координаты>
```

Наивный подход

- Давайте храним строчку, в которую запишем всё, что думаем про время:
16:07:23 06.07.2016 UTC +03:00 DST <GPS координаты>
- Зачем время и дата — понятно.

Наивный подход

- Давайте храним строчку, в которую запишем всё, что думаем про время:
16:07:23 06.07.2016 UTC +03:00 DST <GPS координаты>
- Зачем время и дата — понятно.
- Зачем часовой пояс?

Наивный подход

- Давайте храним строчку, в которую запишем всё, что думаем про время:
16:07:23 06.07.2016 UTC +03:00 DST <GPS координаты>
- Зачем время и дата — понятно.
- Зачем часовой пояс? Когда нас спрашивают про разницу времён в каком-то другом месте, им интересен часовой пояс.

Наивный подход

- Давайте храним строчку, в которую запишем всё, что думаем про время:

16:07:23 06.07.2016 UTC +03:00 DST <GPS координаты>

- Зачем время и дата — понятно.
- Зачем часовой пояс? Когда нас спрашивают про разницу времён в каком-то другом месте, им интересен часовой пояс.
- Зачем место?

Наивный подход

- Давайте храним строчку, в которую запишем всё, что думаем про время:

16:07:23 06.07.2016 UTC +03:00 DST <GPS координаты>

- Зачем время и дата — понятно.
- Зачем часовой пояс? Когда нас спрашивают про разницу времён в каком-то другом месте, им интересен часовой пояс.
- Зачем место? Если часовой пояс поменяется, то нам, возможно, интереснее остаться не в том же часовом поясе, а в том, что в заданной местности.

Наивный подход

- Давайте храним строчку, в которую запишем всё, что думаем про время:

16:07:23 06.07.2016 UTC +03:00 DST <GPS координаты>

- Зачем время и дата — понятно.
- Зачем часовой пояс? Когда нас спрашивают про разницу времён в каком-то другом месте, им интересен часовой пояс.
- Зачем место? Если часовой пояс поменяется, то нам, возможно, интереснее остаться не в том же часовом поясе, а в том, что в заданной местности.
- Если храним момент времени, не равный текущему, Mail.ru предлагает хранить всё это и про момент создания, и про указанное время и место.

Наивный подход

- Давайте храним строчку, в которую запишем всё, что думаем про время:

```
16:07:23 06.07.2016 UTC +03:00 DST <GPS координаты>
```

- Зачем время и дата — понятно.
- Зачем часовой пояс? Когда нас спрашивают про разницу времён в каком-то другом месте, им интересен часовой пояс.
- Зачем место? Если часовой пояс поменяется, то нам, возможно, интереснее остаться не в том же часовом поясе, а в том, что в заданной местности.
- Если храним момент времени, не равный текущему, Mail.ru предлагает хранить всё это и про момент создания, и про указанное время и место.
- **Но нам всё ещё нужна полная история часовых поясов и секунд координации!**

Ошибка 2000

- Не надо экономить в строчке выше.

Ошибка 2000

- Не надо экономить в строчке выше.
- Вы когда-нибудь заполняли анкеты?

Дата: __.__.20__

Ошибка 2000

- Не надо экономить в строчке выше.
- Вы когда-нибудь заполняли анкеты?

Дата: __.__.20__

- Много софта из прошлого века хранило от года 2 цифры.

Ошибка 2000

- Не надо экономить в строчке выше.
- Вы когда-нибудь заполняли анкеты?

Дата: __.__.20__

- Много софта из прошлого века хранило от года 2 цифры.
Догадайтесь, что бы с ним произошло 1 января 2000 года.

Ошибка 2000

- Не надо экономить в строчке выше.
- Вы когда-нибудь заполняли анкеты?

Дата: __.__.20__

- Много софта из прошлого века хранило от года 2 цифры. Догадайтесь, что бы с ним произошло 1 января 2000 года.
- Этого очень боялись, но обошлось.

UNIX-время

- С 1980-х в мире очень распространены UNIX-подобные операционные системы (доминирующая часть сетевого оборудования, GNU/Linux, Mac OS, Android), по части времени многие с ними согласны.

UNIX-время

- С 1980-х в мире очень распространены UNIX-подобные операционные системы (доминирующая часть сетевого оборудования, GNU/Linux, Mac OS, Android), по части времени многие с ними согласны.
- Там принято время хранить в виде **целого знакового числа секунд с момента 1 января 1970 года по UTC**. Дробная часть, если надо, хранится отдельно.

UNIX-время

- С 1980-х в мире очень распространены UNIX-подобные операционные системы (доминирующая часть сетевого оборудования, GNU/Linux, Mac OS, Android), по части времени многие с ними согласны.
- Там принято время хранить в виде **целого знакового числа секунд с момента 1 января 1970 года по UTC**. Дробная часть, если надо, хранится отдельно.
- 1 января 1970 года взято, потому что примерно в тех годах и был создан UNIX. Это время называют текущей эпохой (*epoch*).

UNIX-время

- С 1980-х в мире очень распространены UNIX-подобные операционные системы (доминирующая часть сетевого оборудования, GNU/Linux, Mac OS, Android), по части времени многие с ними согласны.
- Там принято время хранить в виде **целого знакового числа секунд с момента 1 января 1970 года по UTC**. Дробная часть, если надо, хранится отдельно.
- 1 января 1970 года взято, потому что примерно в тех годах и был создан UNIX. Это время называют текущей *эпохой* (*epoch*).
- UNIX-время не учитывает секунды координации. Оно просто монотонно растёт. В худшем случае оно их проходит дважды, откатываясь.

Ошибка 2038

- Берём 32-битный `int`. Храним в нём UNIX-время.

Ошибка 2038

- Берём 32-битный `int`. Храним в нём UNIX-время.
- Максимальное значение — 2 147 483 647. Потом переполнится.

Ошибка 2038

- Берём 32-битный `int`. Храним в нём UNIX-время.
- Максимальное значение — 2 147 483 647. Потом переполнится.
- На языке UNIX-времени — после очередной секунды утра вторника, 19 января 2038 года UNIX-время улетит в 1901 год. Ой.

Ошибка 2038

- Берём 32-битный `int`. Храним в нём UNIX-время.
- Максимальное значение — 2 147 483 647. Потом переполнится.
- На языке UNIX-времени — после очередной секунды утра вторника, 19 января 2038 года UNIX-время улетит в 1901 год. Ой.
- Решение: 64-битное время, уже почти везде так хранят. Его хватит на *гораздо* больше, чем возраст Вселенной.

tzdata

- Вся история часовых поясов предоставляется Internet Assigning Numbers Authority (IANA) в специальном формате, называемом **timezone database** или **zoneinfo database**, сокращённо **tz database** или **tzdata**.

tzdata

- Вся история часовых поясов предоставляется Internet Assigning Numbers Authority (IANA) в специальном формате, называемом **timezone database** или **zoneinfo database**, сокращённо **tz database** или **tzdata**.
- Также там предоставлены очень приятные имена часовых зон, например *America/New_York* или *Europe/Paris*. В таком формате очень удобно отслеживать изменения часовых поясов.

tzdata

- Вся история часовых поясов предоставляется Internet Assigning Numbers Authority (IANA) в специальном формате, называемом **timezone database** или **zoneinfo database**, сокращённо **tz database** или **tzdata**.
- Также там предоставлены очень приятные имена часовых зон, например *America/New_York* или *Europe/Paris*. В таком формате очень удобно отслеживать изменения часовых поясов.
- Обновления этой БД входят в обновления вашей ОС, рекомендуется ставить.

NTP

- Хотим синхронизировать время в Сети. Какие-то компьютеры подключены к точным часам, которые сами сообщают сверхточное время.

NTP

- Хотим синхронизировать время в Сети. Какие-то компьютеры подключены к точным часам, которые сами сообщают сверхточное время.
- Скажем, раз в день компьютер время подключится по Сети и выкачает это время.

NTP

- Хотим синхронизировать время в Сети. Какие-то компьютеры подключены к точным часам, которые сами сообщают сверхточное время.
- Скажем, раз в день компьютер время подключится по Сети и выкачает это время. Только сеть бывает очень медленная, а запросов много...

NTP

- Хотим синхронизировать время в Сети. Какие-то компьютеры подключены к точным часам, которые сами сообщают сверхточное время.
- Скажем, раз в день компьютер время подключится по Сети и выкачает это время. Только сеть бывает очень медленная, а запросов много...
- Есть специальный протокол **Network Time Protocol (NTP)**, позволяющий качественно синхронизировать время (с точностью до нескольких миллисекунд).

NTP

- Хотим синхронизировать время в Сети. Какие-то компьютеры подключены к точным часам, которые сами сообщают сверхточное время.
- Скажем, раз в день компьютер время подключится по Сети и выкачает это время. Только сеть бывает очень медленная, а запросов много...
- Есть специальный протокол **Network Time Protocol (NTP)**, позволяющий качественно синхронизировать время (с точностью до нескольких миллисекунд).
- Этот же протокол сообщает о предстоящих секундах координации.

- Пусть на сервере время отличается на δ . Считаем, что время, необходимое для того, чтобы сообщение дошло в обе стороны, одинаково. Задача — найти δ .

- Пусть на сервере время отличается на δ . Считаем, что время, необходимое для того, чтобы сообщение дошло в обе стороны, одинаково. Задача — найти δ .
- Клиент засекает своё время t_1 , отправляет запрос серверу.

- Пусть на сервере время отличается на δ . Считаем, что время, необходимое для того, чтобы сообщение дошло в обе стороны, одинаково. Задача — найти δ .
- Клиент засекает своё время t_1 , отправляет запрос серверу.
- Тот отвечает временем получения t_2 и временем отправки ответа t_3 .

- Пусть на сервере время отличается на δ . Считаем, что время, необходимое для того, чтобы сообщение дошло в обе стороны, одинаково. Задача — найти δ .
- Клиент засекает своё время t_1 , отправляет запрос серверу.
- Тот отвечает временем получения t_2 и временем отправки ответа t_3 .
- Клиент получил сообщение в t_4 . По его времени, сервер получил запрос и отправил ответ, соответственно, в $t'_2 = t_2 - \delta$ и $t'_3 = t_3 - \delta$.

- Пусть на сервере время отличается на δ . Считаем, что время, необходимое для того, чтобы сообщение дошло в обе стороны, одинаково. Задача — найти δ .
- Клиент засекает своё время t_1 , отправляет запрос серверу.
- Тот отвечает временем получения t_2 и временем отправки ответа t_3 .
- Клиент получил сообщение в t_4 . По его времени, сервер получил запрос и отправил ответ, соответственно, в $t'_2 = t_2 - \delta$ и $t'_3 = t_3 - \delta$.
- Так как мы считаем, что задержки в обе стороны были одинаковые, то

$$t_4 - t_3 + \delta = t_4 - t'_3 = t'_2 - t_1 = t_2 - \delta - t_1$$

откуда

$$\delta = \frac{-t_1 + t_2 + t_3 - t_4}{2}$$

Високосные секунды

- NTP позволяет сделать следующий трюк для элиминации високосных секунд.

Високосные секунды

- NTP позволяет сделать следующий трюк для элиминации високосных секунд.
- Точность в несколько миллисекунд нас устроит?

Високосные секунды

- NTP позволяет сделать следующий трюк для элиминации високосных секунд.
- Точность в несколько миллисекунд нас устроит?
- Тогда размажем високосную секунду по всем секундам дня перед ней. Успех!

Выводы

- Учёт времени — нетривиальная задача. В первую очередь потому, что каждый раз исправления должны вписываться в уже существующую практику (*legacy*).

Выводы

- Учёт времени — нетривиальная задача. В первую очередь потому, что каждый раз исправления должны вписываться в уже существующую практику (*legacy*).
- Не стоит пытаться учёт всего этого безобразия делать самому.

Выводы

- Учёт времени — нетривиальная задача. В первую очередь потому, что каждый раз исправления должны вписываться в уже существующую практику (*legacy*).
- Не стоит пытаться учёт всего этого безобразия делать самому.
- Не стоит напрягаться учётом точного времени, всё равно не нужно.

Выводы

- Учёт времени — нетривиальная задача. В первую очередь потому, что каждый раз исправления должны вписываться в уже существующую практику (*legacy*).
- Не стоит пытаться учёт всего этого безобразия делать самому.
- Не стоит напрягаться учётом точного времени, всё равно не нужно. А если нужно — вы занимаетесь чем-то крайне интересным.

Спасибо за внимание

Вопросы?

Спасибо за внимание

Вопросы?

Презентация будет на странице параллелей ejudge.lksh.ru,
а за пределами ЛКШ она будет на моём сайте:
ldvsoft.net/static/SIS/2016.July/time.pdf.