

С. В. Знаменский

## Компактные отметки времени для эклектичных компьютерных систем

Аннотация. Важной характеристикой сложности большой компьютерной системы является её эклектичность, то есть наличие независимых источников изменений структур и алгоритмов обработки данных. Эклектичными являются интернет в целом, крупномасштабные распределённые и суперкомпьютерные системы, высокая доступность и катастрофоустойчивость которых обеспечивается децентрализованное управление резервированием, локальными перезапусками и обновлением исполняемого кода.

Эклектичные системы нужны и для децентрализованных разработок решения сложных междисциплинарных задач.

Мониторинг и аудит эклектичной компьютерной системы нуждаются в сохранении всех изменений критичных данных с пометками временем изменения. При этом для различных данных точность задания времени различается и важна для корректной обработки.

Предлагается унифицированный формат компактного представления меток времени для данных различного назначения, включающий информацию о точности задания и оптимизированный по длине записи и времени обработки.

*Ключевые слова и фразы:* эклектичные компьютерные системы, управление версиями, информационные системы и сети, суперкомпьютеры, темпоральность, курирование данных.

### 1. Апология эклектичности

Популярная идея единой полной универсальной онтологии, на которой базируются стандарты и теория разработки информационных систем, проекты Semantic Web, Web 2.0 и Web 3.0, к сожалению принципиально не реализуема. Теорема Гёделя о неполноте подсказывает, что полная непротиворечивая онтология возможна лишь для примитивных систем.

Чтобы показать, что онтологии изменчивы в любой практически значимой области, рассмотрим символ астрономической точности — измерение времени.

В сутках 24 часа. Однако в системе, учитывающей переход на летнее поясное время, всё гораздо сложнее и количество часов в сутках иногда оказываться 23 или 25, что зависит от административного подчинения. С секундами в минуте сложнее. В последнюю минуту полугодия иногда (но почти каждый год) по итогам астрономических наблюдений вводится дополнительная високосная 61-я секунда. Никто не знает, будет ли последняя секунда следующего года високосной. Более того, Международный астрономический союз всерьёз обсуждает возможности отмены високосных секунд в частности с заменой их високосными часами.

Даже в математике общеизвестные факты уточняются с новых позиций. Пифагорейцы хранили в тайне иррациональность корня из двух поскольку это противоречило понятию числа как отношения. Ещё до рождения Христова было известно, что сумма углов в треугольнике равна развёрнутому, а квадрат любого числа положителен. Всё это неверно в геометриях Лобачевского и Пуанкаре и алгебре комплексных чисел. В учебниках по математическому анализу написано, что разрывные функции производных не имеют. Но теория обобщённых функций изучает их производные.

В предметных областях, отличных от математики и астрономии, уточнение онтологии ещё быстрее выводит на передний край науки и научно-технической политики и становится зыбким и неустойчивым. В частности, это верно и для сетевых технологий [1].

Безупречность онтологии оказывается жёстко ограниченной во времени и привязанной к конкретной сфере приложений.

Правильное понимание недолговечно. Но именно правильное понимание является основой стандартов разработки компьютерных систем. Вывод парадоксален и неутешителен: *императивные компьютерные системы в обозримой перспективе обречены продолжать разочаровывать пользователей* [2].

Выход подсказывает общепризнанные принципы разделения труда и разделения зон внимания (separation of concerns). Будущее за надёжными согласованно развивающимися взаимопроницаемыми системами, улучшаемыми независимыми группами, компетентными в своих областях [3–5]. Такие системы правильно называть эклектичными. Интернет в целом — это пример эклектичной системы.

Негативное отношение, закрепившееся за термином и системами, исходит от наивной мечты об абсолютном знании и отсутствия теории, технологий и примеров согласованных эклектичных систем.

Не исключено, что технологии эклектичных систем откроют путь к эффективной децентрализованной разработке на фоне постоянно доступного качественного сервиса.

Подобно тому, как объединив творческие силы мира Википедия превзошла по широте, полноте, доступности, актуальности и популярности все лучшие энциклопедии мира, совместное творчество независимых групп разработчиков [6] способно привести к результату, превосходящему лучшие ожидания [7]. Речь идёт о качественно превосходной точности, надёжности и дружелюбности взаимодействующих компьютерных систем, о самовосстанавливающихся, эволюционирующих и мультиагентных системах, о качественном скачке функциональности, надёжности и удобства пользовательских интерфейсов.

## 2. Задача унификации меток времени

Организация мониторинга и аудита эклектичных систем является важной нерешённой проблемой. Сложность сбора согласованной информации о происходящих в эклектичной системе изменениях показывают работы о сохранении истории ресурсов интернет [8].

Угроза рассогласования становится явной если обработка может соединить данные, не актуальные на общий момент времени. Поэтому проблема состоит в определении того, какие данные считать одновременно актуальными, то есть относящимися к одному состоянию системы.

Классические исследования показали запредельную техническую сложность синхронизации всех изменений в распределённой компьютерной системе. Система рассматривалась при этом как набор конечных автоматов, обменивающихся сообщениями о событиях. Время в ней трактовалось как вспомогательное средство для выстраивания всех событий в единый линейный порядок.

В интересующей нас эклектичной модели время это показания локальных часов, то есть физическая величина, измеренная с некоторой точностью.

Выделить актуальные в общий момент времени данные далеко не просто по многим причинам.

Во-первых, окончание актуальности данного  $X$  может быть настолько близко к началу актуальности данного  $Y$ , что момент их одновременной актуальности существует лишь с некоторой вероятностью.

Во-вторых, погрешность указания времени  $\Delta$  может варьироваться от наносекунд для данных быстро протекающих процессов до тысячелетий для археологических данных.

В-третьих, конкретная разница во времени событий может свидетельствовать об одновременности или о неодновременности событий в зависимости от постановки задачи.

В-четвёртых, различаются актуальность в некоторый момент промежутка и актуальность в течение всего промежутка.

В-пятых, проблематично корректно совместить данные, относящиеся к приближенно известной общей границе промежутков шкалы времени с одним из этих промежутков.

В-шестых, случается, что актуальная достоверная информация недоступна и пользователям порой нужен доступ к новейшим неполным или непроверенным данным. В таких ситуациях разумно предоставлять особый интерфейс к неполной сводной информации.

История изменений информационного объекта предстаёт упорядоченным по актуальности списком версий. Упорядоченность обеспечивает быстрый доступ к актуальной на любой заданный момент версии. Версии с недостаточно высокими значениями при этом быстро пропускаются.

Для прозрачно безупречного автоматического выделения одновременно актуальные версии данных

- запись момента времени должна отражать значение и погрешность в широких диапазонах строкой переменной длины,
- лексикографический порядок записей должен точно соответствовать порядку событий.

### **3. Способ кодирования времени**

Шкалой  $S$  назовём разбиение числовой прямой на полусегменты равной длины, называемой шагом дискретизации. Шаг дискретизации определяется прикладными задачами и должен быть практически несущественным. Например, для полученных из веб-формы данных не имеет практического значения, нажал ли пользователь кнопку отправки десятой долей секунды раньше или позже.

Правый конец в полусегмент не входит, этим из предыстории исключаются события, на обработку которых не было времени. Базовый пример даёт разбиение на целочисленные промежутки  $[n, n + 1), n \in \mathbb{Z}$ .

Мы будем использовать глобальные отметки времени (Новый год, полночь и т.д.) как разделители. Поскольку потребуется отчёт за год, то каждое событие должно определённо относиться либо к прошлому году, либо к будущему. Сложность с событием, произошедшим в момент такого разделителя: допустимая погрешность часов позволяет отнести его в одних подсистемах к предшествующему разделителю полусегменту, а в других к последующему за разделителем.

В таких случаях логически неправомерно производить обработку данных на момент конца года и нужно выбирать момент, на который череда изменений приостановилась. Пользователя однако может интересовать состояние на конец года. По-видимому, лучшее, что может сделать система в этой ситуации, это выдать корректный ответ на близкий момент времени. В ситуации продолжительной высокой интенсивности конфликтующих изменений входных данных, в которой пользователя интересует не только (вероятно отдалённый) момент истины, но заведомо логически небезупречная оперативная оценка текущего состояния дел.

Для получения такой оценки можно игнорировать несущественные расхождения во времени, но при записи результата указать оценку дефекта (времени рассогласованности)  $\tau$ . Эта оценка должна быть использована при выдаче пользователю предупреждения о размере возможной некорректности данных.

Одновременное использование многих шкал нуждается в их согласованности, означающей что из двух пересекающихся полусегментов один обязательно содержит другой. Обозначим  $\mathbb{B} = \{k \cdot 2^l | k, l \in \mathbb{Z}\}$  множество всех двоичных дробей и назовём *бинарным семейством* такое семейство шкал, все сегменты которого имеют вид  $[t - \delta, t) = [k \cdot 2^l, (k + 1) \cdot 2^l)$ , где  $k, l \in \mathbb{Z}$ .

**Предложение 1.** *Полусегмент бинарного семейства однозначно определяется своей серединой. Множество середин таких полусегментов совпадает с  $\mathbb{B}$*

Простота идентификации полусегментов бинарного семейства делает их привлекательными для использования в качестве идентификаторов моментов событий с учётом погрешности определения времени. Например, момент события, предшествовавшего моменту 2 с точностью примерно 0.1%, идентифицируется как  $1.1111111111_{(2)}$ .

Для ускорения поиска нужной версии в бинарном дереве важно чтобы идентификаторы были лексикографически упорядочены по актуальности. Несмотря на простоту и естественность, порядок при таком представлении не согласуется с желаемым лексикографическим: правый конец полусегмента  $t$  существенно более значим, чем левый  $t - \delta$ .

К сожалению, количество дней в году и другие соотношения мер времени не являются степенями двойки (и, к тому же, не все являются заранее фиксированными числами). Моменты времени практически задаются конечными наборами  $t = (Y, M, D, h, m, s, \mu_0, \mu_1, \dots)$  из указаний года  $Y$ , месяца  $M$ , дня  $D$ , часа  $h$ , минут  $m$ , секунд  $s$ , миллисекунд  $\mu_0$ , микросекунд  $\mu_1$  и т.д.

**Предложение 2.** *Функция*

$$\begin{aligned} x(t) = & 2^{-3}Y + 2^{-7}(M - 1) + 2^{-12}(D - 1) \\ & + 2^{-18}h + 2^{-24}m + 2^{-30}s \\ & + \sum_{k \geq 0} 2^{-30-10k} \mu_k \end{aligned}$$

*монотонно и однозначно представляет моменты времени двоичными дробями.*

**Замечание 1.** *Функция теоремы 2 переводит привычные шкалы времени (годы, месяцы, дни, недели, часы и т. д) в шкалы бинарного семейства.*

Например, 13:35 30 июня 2014 года с секундной точностью идентифицируется двоичной дробью

$$\begin{aligned} x = & 2014 \cdot 2^{-3} + 29 \cdot 2^{-7} + 13 \cdot 2^{-12} \\ & + 35 \cdot 2^{-18} + 0.5 \cdot 2^{-24} + 2^{-31} \\ = & 11111011.110\ 0101\ 11101\ 01101\ 100010\ 111111\ 1_{(2)} \\ = & 2^{11} - 100.001\ 1010\ 00010\ 10010\ 011101\ 000000\ 1_{(2)}. \end{aligned}$$

Следующее утверждение описывает ускоряющий сравнение переход от чисел произвольной длины со знаком к битовым массивам (строкам произвольной длины).

**Предложение 3.** *Кусочно-линейная функция*

$$B(x) = \frac{1}{2} + \frac{\operatorname{sgn}(x)}{2} \left( 1 - \frac{3}{2^{k(x)+1}} - \frac{1 + |x|}{2^{2k(x)}} \right), \text{ где}$$

$$k(x) = \operatorname{ceil}(\log_2(1 + |x|)),$$

*непрерывна, монотонна и взаимно-однозначно отображает множество  $\mathbb{B}$  всех двоичных дробей на  $(0, 1) \cap \mathbb{B}$ .*

Для промежутка с 1984 по 2112 годы это выражение удобно масштабируется  $B(x(t) - 2^{11})$  и упрощается до  $B(x(t) - 2^{11}) = \frac{1}{2} + \frac{x}{t} 2$  потому что  $|x - 1| < 1$ . Для длительности  $\delta$  удобнее суточный масштаб времени  $2B(-2^1 2x(\delta)) - 1$ , а для дефекта  $\tau$  минутный  $2B(-2^2 4x(\delta)) - 1$ .

**Замечание 2.** *Обратная к  $y = B(x)$  функция вычисляется по формуле*

$$x = 1 + \operatorname{sgn}(2y - 1) (3\alpha - 2\alpha^2 |2y - 1| - 1), \text{ где}$$

$$\alpha(y) = 2^{k(y)-1},$$

$$\tilde{k}(y) = -\operatorname{floor}(\log_2(1 - |2y - 1|)).$$

**Замечание 3.** *Количество значащих цифр после запятой в двоичной записи  $B(x)$  равно сумме  $|k(x)|$  и количества цифр в записи двоичной дроби  $|x| + 1$ , взятого без заключительных нулей.*

В частности, образ  $B(x(t) - 2^{11})$  34-значной двоичной дроби из последнего примера длиннее записи самой дроби на два знака поскольку для неё  $k(x) = 2$ .

Для полусегментов  $[t - \delta, t)$  и дефектов  $\tau$  требуется монотонно закодировать лексикографически упорядоченные вектора из записи двоичных дробей  $B(x(t)), B(2^{12}x(-\delta))B(2^{24}x(-\tau)),$ .

**Замечание 4.** *Можно монотонно закодировать лексикографически упорядоченные вектора из записи двоичных дробей  $(x_1, x_2, x_3) \in [0, 1]$  строками из  $\sum_{i=1}^3 \operatorname{ceil}\left(\frac{\operatorname{length}(x_i)}{6}\right) + 2$  байт, принимающих 65 различных значений.*

Для этого битовый массив разбивается на группы по 6 бит, последняя дополняется нулями, каждая группа бит кодируется байтом (Base64) и  $x_i$  представляется строкой (в нашем примере длины 6 байт). Пусть «!» - байт с меньшим кодом, чем использованные в Base64. Тогда конкатенация  $x_1.«!».x_2.«!».x_3$  даёт эффективную кодировку для замечания 4.

#### 4. Выводы

- (1) Конфликты версий неизбежны внутри сложных эклектичных систем.
- (2) Автоматическое версионирование на основе меток времени открывает путь к исключению таких конфликтов.
- (3) Описан способ маркировки моментов событий в эклектичных системах.

#### Список литературы

- [1] Kely C. *Conceiving Open Systems* // Wash. UJL & Pol'y, 2009. Vol. **30**, no.1, p. 139–177. ↑1
- [2] Merali Y., Papadopoulos T., Nadkarni T. *Information systems strategy: Past, present, future?* // J.Strateg. Inform. Syst., 2012, <http://dx.doi.org/10.1016/j.jsis.2012.04.002>. ↑1
- [3] Feiler P., Gabriel R. ., Goodenough J., Linger R., Longstaff T., Kazman R., Klein M., Northrop L., Schmidt D., Sullivan K., Wallnau K. *Ultra-Large-Scale Systems: The Software Challenge of the Future* Technical Report, Carnegie Mellon University Software Engineering Institute, 2006 ↑1
- [4] Northrop L. . *Ultra-Large-Scale Systems: Scale Changes Everything* // SMART Ultra-Large-Scale Systems Forum, March 6, 2008. ↑
- [5] Ncube C. *On the Engineering of Systems of Systems: key challenges for the requirements engineering community* // Requirements Engineering for Systems, Services and Systems-of-Systems (RESS), Aug.; 2011, p. 70–73. ↑1
- [6] Jeffery S., Sun L., DeLand M., Pendar N., Barber R., Galdi A. *Arnold: Declarative Crowd-Machine Data Integration* // 6th Biennial Conference on Innovative Data Systems Research (CIDR '13), January 6, p. 8. ↑1
- [7] Van Osch W., Avital M. *Collective Generativity: The Emergence of IT-Induced Mass Innovation* // Proceedings of JAIS Theory Development Workshop., <http://sprouts.aisnet.org/9-54>. ↑1
- [8] Xie Z., Van de Sompel H., Liu J., van Reenen J., Jordan R. *Archiving the Relaxed Consistency Web* // arXiv preprint arXiv:1308.2433, 2013. ↑2



Об авторе:



### Сергей Витальевич Знаменский

Автор критерия разрешимости уравнений свёртки в пространстве функций, голоморфных на множестве, понятий выпуклости в направлении и  $S$ -выпуклости, русификации Т<sub>Е</sub>X для журналов Отделения математики РАН, графического пакета `mfpic3d` и нового подхода к построению информационных систем.

e-mail:

[svz@latex.pereslavl.ru](mailto:svz@latex.pereslavl.ru)

Образец ссылки на эту публикацию:

С. В. Знаменский. Компактные отметки времени для эклектичных компьютерных систем // Программные системы: теория и приложения : электрон. научн. журн. 2013?. Т. 4?, № 4(17)?, с.??-??.

URL:

<http://psta.psiras.ru/read/>

S. V. Znamenskij. *Performance indicators for scheduling destruction.*

ABSTRACT. The eclectic nature of a large software system becomes important. It means the presence of independent sources of changes in the structures and data processing algorithms.

The internet in general is eclectic, also large-scale and distributed supercomputer systems mostly are: the high availability and disaster recovery are provided by decentralized management of local restarts and by locally managed executable code updates.

The development of decentralized solutions for complex interdisciplinary problems solving also need an eclectic system to combine updateable software from different vendors.

Monitoring and auditing eclectic computer system needs all changes of critical data to be saved with a change timestamp. The accuracy of time record usually varies and is important for proper treatment.

An unified format of a timestamp with various time accuracy is presented. The format provides the shortest records with lexicographical ordering of timestamps for shorter processing time. (*In Russian*)

*Key Words and Phrases:* backup, performance indicators, system restore, schedule, quality assessment.