

Московский государственный университет имени М.В. Ломоносова
Факультет вычислительной математики и кибернетики

Реферат

Обзор методов синхронизации времени в распределенных системах

Остроушко Л. О. 321 группа.

Москва 2015 год

Оглавление

Алгоритм Кристиана.....	2
Алгоритм Беркли.....	4
Усредняющие алгоритмы.....	4
Алгоритм отметок Ламберта.....	5

Без синхронизации времени в распределенных системах хранения и обработки данных невозможно согласовать работу отдельных узлов системы. Необходимо добиться максимально точного совпадения текущего времени синхронизируемых узлов в момент начала как предыдущей синхронизации данных, так и текущей (должно соблюдаться единство данного временного интервала для точного определения синхронизируемого объема информации), а также в промежутке между этими событиями для определения очередности наступления событий. Например, в момент начала последней синхронизации системное время на одном из узлов было на секунду больше, чем на другом; если за начало интервала синхронизации взять системное время второго узла, то изменения, произошедшие на первом узле за эту секунду, будут утеряны.

Рассмотрим основные методы синхронизации времени в распределенных системах, сразу сформулировав два основных требования:

- в определенные моменты системное время узлов системы должно максимально совпадать;
 - невозможно уменьшение системного времени при синхронизации, недопустим перевод часов в обратную сторону (Это может привести к выходу за начало интервала синхронизации, либо сбоем при определении очередности событий).
- Мы можем только переводить время вперед или же замедлять его ход.

Алгоритм Кристиана.

В системе присутствует эталонный узел, время которого принимается за истинное. Каждый из узлов периодически посылает эталонному запрос его текущего времени. Эталонный узел должен максимально быстро отвечать на этот запрос. Однако между отправкой сообщения и его получением проходит некоторое время, оно зависит от текущей нагрузки сети и не является постоянным. Оценивать это время Кристиан предложил как половину времени

между отправкой запроса и получением ответа. Если известно примерное время работы сервера (ответа на запрос), то можно точнее определить время хождения по сети, вычитая из времени оборота время работы сервера. Хуже обстоят дела, если прохождение запроса и ответа занимает разное время, так как сигналы идут разными маршрутами. Для повышения точности Кристиан предложил делать серию измерений и использовать среднее время. Причем, если попадается измерение, которое выходит за рамки некоторого принятого ранее порога в сторону увеличения, то такое измерение отбрасывается, как недостоверное. Наоборот, наиболее быстрый ответ можно считать содержащим самое точное время.

Достоинства:

- простота реализации
- высокая эффективность в небольших сетях и сетях с малой загрузкой: поскольку в системе присутствует источник точного времени и синхронизация производится по нему, то время в системе в целом соответствует реальному.

Недостатки:

- наличие внешнего источника точного времени
- отсутствие встроенной защиты от перевода часов в обратную сторону
- некорректность работы в сетях с резкими скачками загрузки сети, а так же метод не позволяет корректно устанавливать время в случае, если запрос и ответ передаются по разным маршрутам (требуется разное время для передачи данных сообщений).

Остальные алгоритмы предполагают отсутствие такого источника времени

Алгоритм Беркли.

Данный алгоритм предназначен для случая, когда источник точного времени отсутствует. Сервер времени опрашивает все узлы для выяснения их текущего времени, усредняет это значение и рассылает команды на установку нового значения времени либо замедления часов.

Достоинства:

- простота реализации
- высокая эффективность для систем, некритичных к отклонениям по времени.

Недостатки:

- является необходимость выделенного узла – сервера времени.
- поскольку источник точного времени отсутствует, а за общесистемное время принимается усредненное время узлов, оно может иметь мало общего с реальным временем
- узлы с замедленным для коррекции временем (все еще некорректным) оказывают влияние на общесистемное время.

Два данных алгоритма являются централизованными

Усредняющие алгоритмы.

Семейство алгоритмов, которые имеют общий принцип работы. Каждый из узлов системы выполняет широковещательную рассылку текущего времени с заданной периодичностью. Затем ожидается ответ от остальных узлов сети с их текущим временем. После принятия всех сообщений вычисляется новое время узла по некоторому алгоритму. Например, усреднение полученных значений. Алгоритм может быть усовершенствован если отбросить некоторое количество самых больших и самых маленьких значений для защиты от заведомо неверных значений. Так же, зная топологию сети или каким-либо другим способом можно попытаться оценить время прохождения сообщения от источника для получения

более точных значений.

Достоинства:

- отсутствие выделенного узла – сервера времени
- высокая эффективность подстройки под конкретные задачи.

Недостатки:

- сильная зависимость эффективности от загрузки сети
- низкая степень синхронизации
- отсутствие гарантии установки на всех узлах одинакового времени (часть сообщений на конкретном узле может не быть получена, не получена вовремя, отброшена как заведомо ложная).

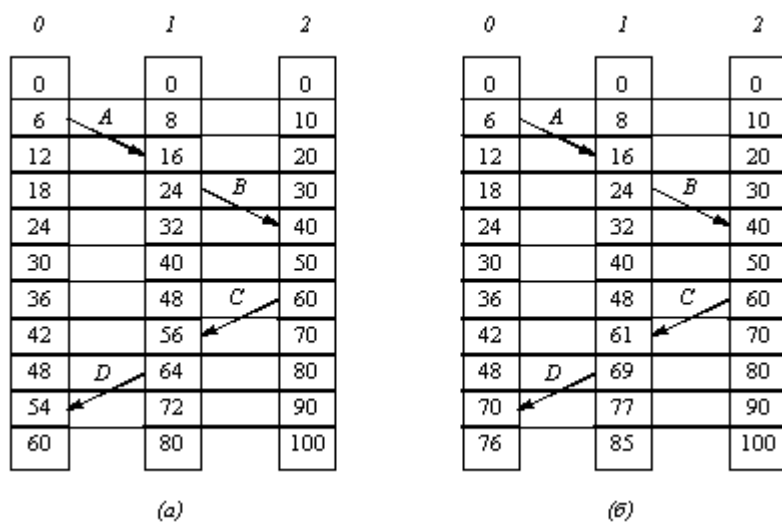
Отметки времени Лампорта.

Данным методом оперируют не физическими, а логическими часами. Лампорт предложил синхронизировать только связанные между собой процессы и ориентироваться не на точное время выполнения процессов, а на их порядок. При обмене данными между взаимодействующими узлами происходит также обмен сведениями об их локальном времени. Если отметка времени, указанная в сообщении, оказывается меньше, чем локальное время узла, такая ситуация считается штатной, так как сообщение было отправлено ранее, чем получено. Если же в сообщении указано более раннее время, чем локальное время узла, то есть сообщение было послано позже, чем получено, делается вывод о необходимости коррекции системного времени узла. Время выставляется в значение, равное отметке в сообщении плюс единица.

На рисунке показаны три процесса, выполняющихся на разных машинах, каждая из которых имеет свои часы, идущие со своей скоростью. Как видно из рисунка, когда часы процесса 0 показали время 6, в процессе 1 часы показывали 8, а в процессе 2 - 10. Предполагается, что все эти часы идут с постоянной для себя скоростью.

В момент времени 6 процесс 0 посылает сообщение А процессу 1. Это

сообщение приходит к процессу 1 в момент времени 16 по его часам. В логическом смысле это вполне возможно, так как $6 < 16$. Аналогично, сообщение В, посланное процессом 1 процессу 2 пришло к последнему в момент времени 40, то есть его передача заняла 16 единиц времени, что также является правдоподобным. Ну а далее начинаются весьма странные вещи. Сообщение С от процесса 2 к процессу 1 было отправлено в момент времени 64, а поступило в место назначения в момент времени 54. Очевидно, что это невозможно. Такие ситуации необходимо предотвращать. Решение Лампрта вытекает непосредственно из отношений "случилось до". Так как С было отправлено в момент 60, то оно должно прийти в момент 61 или позже. Следовательно, каждое сообщение должно нести с собой время своего отправления по часам машины-отправителя. Если в машине, получившей сообщение, часы показывают время, которое меньше времени отправления, то эти часы переводятся вперед, так, чтобы они показали время, большее времени отправления сообщения. На рисунке 'б' видно, что С поступило в момент 61, а сообщение D - в 70.



Достоинства:

- отсутствие выделенного узла – сервера времени
- высокая эффективность при малом числе узлов

- высокая степень синхронизации (время переводится только вперед, и нет необходимости ждать эффекта от замедления на спешащих узлах).

Недостатки:

- не принимается в расчет время передачи сообщений между узлами и поскольку имеет место постоянный перевод часов вперед – общесистемное время имеет мало общего с реальным.

Список литературы:

- Cristian F. Probabilistic Clock Synchronization Distributed Computing, vol. 3, pp. 146-158, 1989.
- Ю.А. Алдунин «Синхронизация времени в распределенных системах, в зависимости от выбранной модели непротиворечивости» ISSN 1995-4565. Вестник РГРТУ. Вып. 22. Рязань, 2007
- http://citforum.ru/operating_systems/sos/glava_13.shtml