
ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

УДК 621.397.132.19

ПРИМЕНЕНИЕ АЛГОРИТМА СОВМЕСТНОГО КЭШИРОВАНИЯ В РЕАЛИЗАЦИИ IPTV

Бородинский Алексей Андреевич, директор, Астраханский филиал открытого акционерного общества «Ростелеком», 414000, Россия, г. Астрахань, Театральный переулок, 7/8, e-mail: asicom@ast.south.rt.ru.

Статья описывает применение алгоритма совместного кэширования для повышения эффективности реализации услуги «Телевидение, сдвинутое по времени», входящее в комплекс услуг IPTV. Применение данного алгоритма позволяет снизить нагрузку на сеть передачи данных и передавать вещаемый трафик наиболее рациональным способом. Услуга «Телевидение, сдвинутое по времени» позволяет конечному пользователю смотреть телепрограммы с некоторой разницей по времени относительно прямого вещания, так что пользователь может начать просмотр телевизионной программы с самого начала, хотя вещание этой программы уже началось или даже уже закончилось. Исследования методов прокси-кэширования и распределенного размещения репликаций для сетей распределения контента показали, что каскадные алгоритмы, учитывающие расстояния и степень популярности контента, работают лучше, чем более простые эвристические алгоритмы. Особый интерес для решения данной задачи представляет метод кэширования со скользящим интервалом, где кэшируемая часть потока сначала – растущее значение, а затем переходит в динамически обновляемый скользящий интервал. Таким образом, последовательно поступающие запросы могут быть обслужены от начала до конца внутри данного окна. Более прогрессивным является использование совместно работающих прокси для осуществления кэширования, это позволяет достичь лучшую производительность и масштабируемость системы, чем в случае использования независимых прокси за счет возможностей балансировки нагрузки.

Ключевые слова: телевидение поверх протокола IP; телевидение, сдвинутое по времени; алгоритмы реального времени; прозрачный прокси; протокол вещания реального времени; сеть доступа; взаимодействующие кэши; кэширующий алгоритм; широкополосная сеть; скользящее окно.

USING OF CACHING ALGORITHM FOR THE IMPLEMENTATION OF IPTV SERVICES

Borodinsky Alexey A., director, Astrakhan branch of the Open Joint Stock Company “Ros-telecom”, 414000, Russia, Astrakhan, 7/8, Teatralny lane, e-mail: asicom@ast.south.rt.ru.

The article describes the application of the algorithm co-caching to improve the effectiveness of the service “Television, shifted in time” included in-service IPTV. Applying this algorithm to reduce the load on the network data traffic and transmit broadcasted the most rational way. The service “Television, shifted in time” allows the end user to watch TV with some difference in time

relative to the live broadcast, so the user can start watching a TV program from the start, although broadcast of this program has already begun, or has already ended. Research methods of proxy caching and replication for distributed deployment of content distribution networks showed that the cascade algorithms that take into account the distance and the popularity of content work better than a simple heuristic algorithms. Of particular interest to solve this problem is the method cache with a sliding interval, where the cached portion of the stream at first – the growing importance of, and then goes into a dynamically updated rolling interval. Thus, the sequence of the query can be served from start to finish within a given window. More progressive is the use of joint-working proxy for caching, it allows to achieve better performance and scalability of the system than in the case of independent proxy through load-balancing capabilities.

Key words: television over IP, television, shifted over time, algorithms for real-time, transparent proxy, the protocol broadcast of real-time network access, interact cache, the caching algorithm, broadband, a sliding window.

«Телевидение, сдвинутое по времени» позволяет конечному пользователю смотреть телепрограммы с некоторой разницей по времени относительно прямого вещания. То есть конечный пользователь может начать просмотр телевизионной программы с самого начала, хотя вещание этой программы уже началось или даже уже закончилось [1].

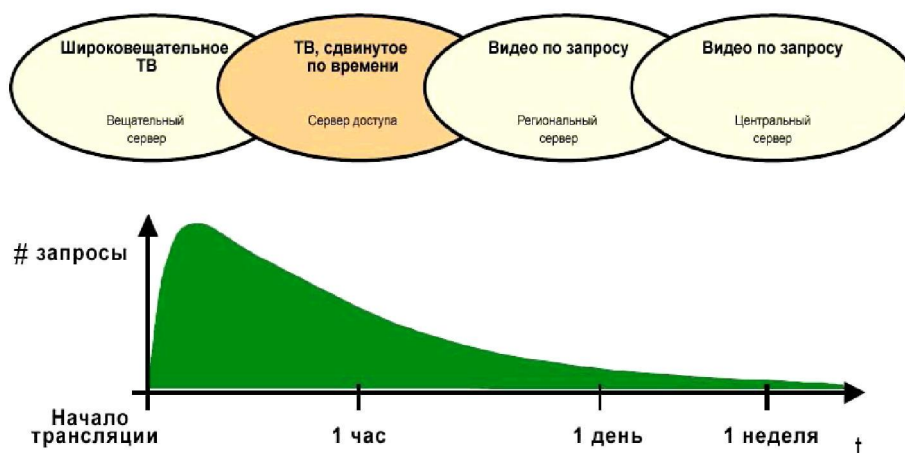


Рис. 1. Зависимость количества запросов от продолжительности трансляции

Как показывают статистические данные (рис. 1), популярность телевизионных трансляций обычно достигает своего пика в течение нескольких минут после начала вещания и затем экспоненциально уменьшается. Это означает, что кэширование сегмента со скользящим окном в несколько минут каждой транслируемой программы может удовлетворить значительную часть запросов со стороны потребителей, что делает целесообразным использование распределенных накопителей с ограниченной емкостью.

ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

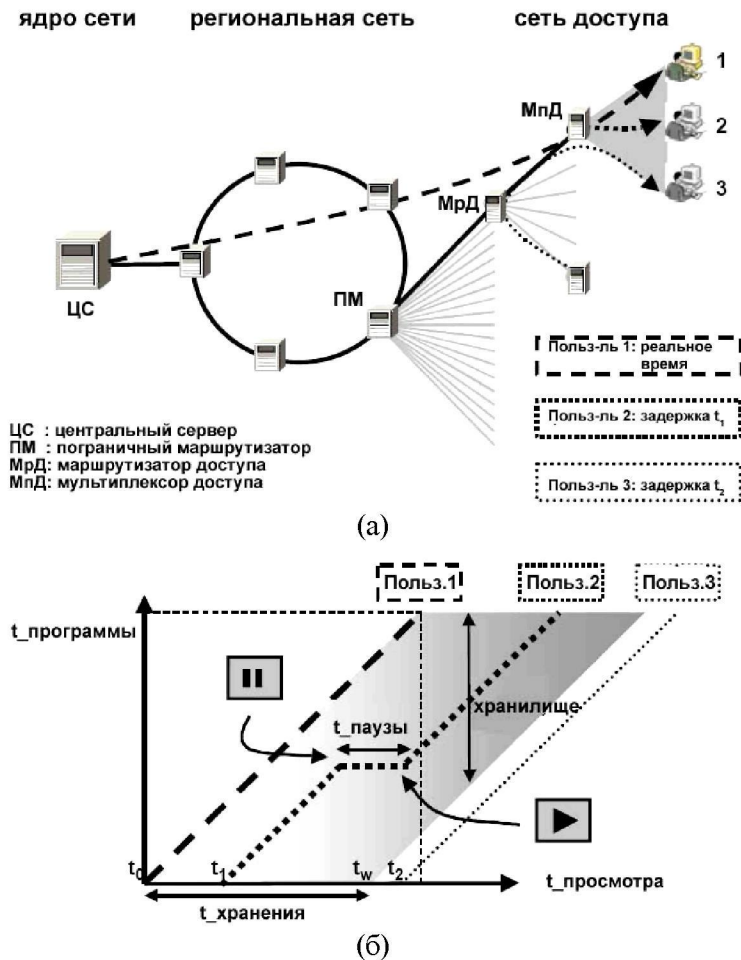


Рис. 2. Телевидение, сдвинутое по времени: (а) топология сети доступа, (б) диаграмма потока услуги

На рис. 2а и 2б, например, пользователь 1 – первый, кто запросил трансляцию определенной телевизионной программы, получает ее с центрального сервера. После этого другие пользователи (например, пользователь 2) могут быть обслужены прокси-сервером, до тех пор, пока окно запрошенной программы растет. Через несколько минут окно перестает расти и начинает скольжение, так что пользователь 3 не может быть обслужен и будет перенаправлен на центральный или региональный сервер, а в случае совместного кэширования – на соседний прокси-сервер в соответствующем сегменте, при наличии таковых. Приостановка трансляции программы (параллельно горизонтальной оси на рис. 2б) также может быть реализована в рамках сегмента окна, а также быстрой перемотки вперед или назад (вдоль вертикальной оси).

Исследования методов прокси-кэширования и распределенного размещения реплик для сетей распределения контента показали, что каскадные алгоритмы, учитывающие расстояния и степень популярности контента, работают лучше, чем более простые эвристические алгоритмы, такие как LRU (Least Recently Used) или LFU (Least Frequently Used).

Методы кэширования, основанные на сегментировании, применяются для потоковой передачи мультимедиа из-за огромных размеров мультимедийных потоков по сравнению с традиционными сетевыми объектами. Особый интерес для решения данной задачи представляет метод кэширования со скользящим интервалом, где кэшируемая часть потока сна-

чала – растущее значение, а затем переходит в динамически обновляемый скользящий интервал. Таким образом, последовательно поступающие запросы могут быть обслужены от начала до конца внутри данного окна. Более прогрессивным является использование совместно-работающих прокси для осуществления кэширования, это позволяет достичь лучшую производительность и масштабируемость системы, чем в случае использования независимых прокси за счет возможностей балансировки нагрузки [1].

В результате проведенных ранее исследований в области применения алгоритмов кэширования, основанных на сегментировании на прокси-серверах с использованием протоколов RTP / RTCP / RTSP, были получены наилучшие результаты, когда совмещаются преимущества кэширования со скользящим интервалом и совместного кэширования и использования прокси на базе протокола RTSP, поддерживающего данные алгоритмы.

Рассмотрим алгоритм кэширования услуг «Телевидения, сдвинутого по времени». Так как мы считаем, что будут сохраняться только сегменты телепрограммы, объем кэш-памяти может быть ограничен до нескольких гигабайт или даже меньше в случае совместно-кэширования. Таким образом, появляется возможность устанавливать потоковые серверы меньшие по объему, что позволяет снизить стоимость развертывания услуги.

Будем считать, что кэш виртуально разделен на две части: небольшую часть *S* и основную часть *L*. Часть *S* будет использоваться для кэширования первых нескольких (например, 5) минут каждой вновь запрошенной пользователем (или вещаемой) программы, в основном для определения ее начальной популярности. Ее размер, как правило, меньше, чем 1 Гигабайт (что составляет примерно один час потокового видеоконтента).

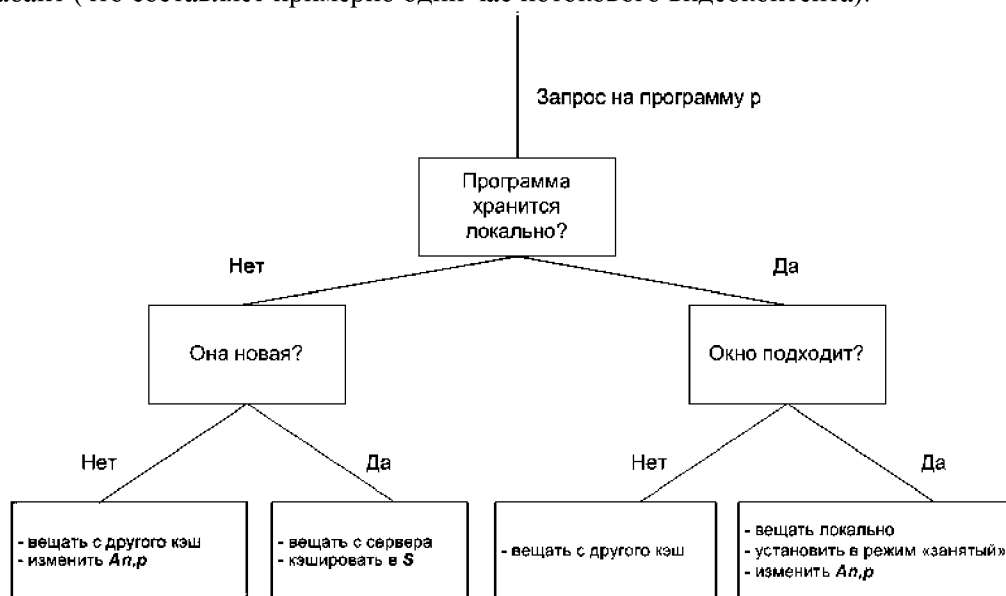


Рис. 3. Принципы алгоритма кэширования на каждом прокси

Часть *L* будет использоваться для хранения соответствующих сегментов (с растущими или скользящими окнами). В свою очередь, эта часть также виртуально разделена на два отдельных хранилища. Часть *L*₁ будет использоваться для хранения только уникальных сегментов. Она является общей для всех узлов, взаимодействующих на одном уровне доступа. Будем считать, что все кэши знают, какие уникальные сегменты хранятся на других кэшах, посредством протокола CSE (Cache State Exchange). Таким образом, все части *L*₁ на всех кэш-узлах представляют собой один большой кэш, главным образом, для разгрузки центрального сервера. Вторая часть *L*₂ используется для хранения сегментов, наиболее популяр-

ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

лярных локально. Основным назначением этой части является разгрузка сети доступа, при этом используется механизм совместного кэширования (запросы обслуживаются L_1 на соседнем кэше). Фактический размер каждого сегмента в части L_2 будет определен и, при необходимости, изменен после каждого интервала Δ (например, 5 минут). На рис. 3 показан основной принцип алгоритма кэширования. В течение каждого интервала Δ пользовательские запросы на просмотр программы поступают на различные прокси. Каждый раз параметр $A_{n,p}$ будет обновляться в прокси n для программы p . В общем, этот параметр пытается определить популярность программ, принимая во внимание показатели удаленности. Это означает, что популярная программа не может быть кэширована, поскольку более близкие прокси уже сохранили эту программу. Если $A_{n,p}$ для программы p на кэш-узле n недостаточен, то окно будет расти, в противном случае окно начнет скольжение или будет удалено в случае отсутствия запросов, обслуживаемых данным сегментом. $A_{n,p}$ рассчитывается следующим образом: каждый раз, когда запрос на программу p приходит на узел n , $A_{n,p}$ увеличивается на единицу (принимается во внимание только популярность программы) или на число сетевых сегментов между прокси n и обслуживаемым узлом (принимается во внимание популярность и удаленность).

После каждого интервала Δ все сегменты со статусом «заняты» (т.е. обслуживающие запросы) сохраняются в L_2 . После заполнения L_2 сегментами с увеличенными окнами для самых популярных программ (т.е. с наибольшими значениями $A_{n,p}$). Все другие сегменты удаляются, S очищается, и все значения p сбрасываются в 0.

Функциональность прокси разделена на две логические части. Связь с пользователями и центральным сервером включает сообщения, содержащие данные о том, какая программа или канал должен транслироваться или командах (подобных тем, которые используются в видеоманитофонах) паузы или остановки. Обычно для подобного взаимодействия используется протокол RTSP (Real-Time Streaming Protocol). Потоки инкапсулируются и доставляются посредством протокола вещания потокового видео RTP (Real-Time Protocol).

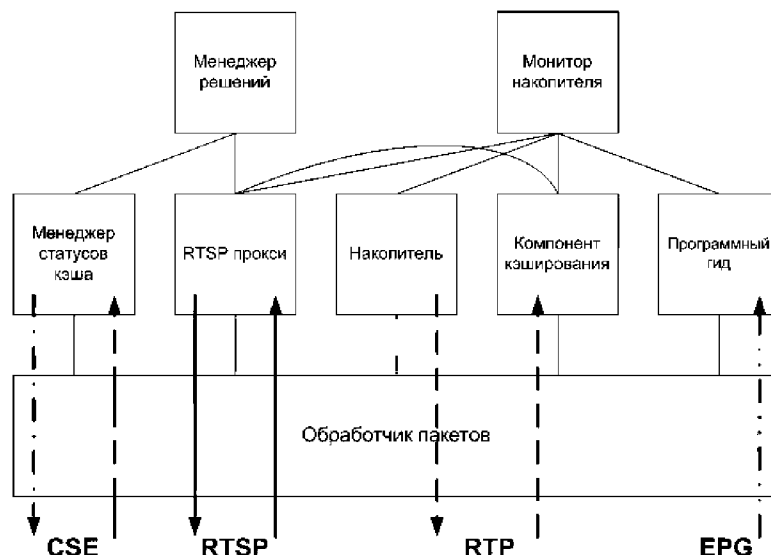


Рис. 4. Компоненты прокси

Первый функциональный компонент прокси – RTSP-прокси. Это компонент, который взаимодействует с клиентами услуги «Телевидение, сдвинутое по времени» и сервером, используя протокол RTSP. Он интерпретирует их сообщения и команды другим компонентам для того, чтобы выполнить эти запросы. Компонент RTSP-прокси передает решения кэ-

ширующего алгоритма другому компоненту – Менеджеру решений, который использует информацию Менеджера статусов кэша и обновляется посредством централизованного или распределенного протокола обмена статусами *Cache State Exchange*. Задача Компонента кэширования – хранение популярных потоков, переданных сервером или другим кэшем на прокси в скользящих окнах. Обработка потоков, посланных клиентам из этих окон – это функция компонента Накопитель. Прокси также отслеживает потоки, которые были посланы к нему (программы, каналы, время начала, прочее) посредством компонента Монитор накопителя, с помощью компонента Программный гид, который, в свою очередь, взаимодействует с сервером электронного программного гида (*Electronic Program Guide*). Обработчик пакетов выступает в качестве интерфейса, обеспечивающего взаимодействие с сетью.

Таким образом, для решения задачи повышения эффективности организации услуги «Телевидение, сдвинутое по времени», целесообразно использовать механизм кэширования популярного контента на прокси-серверах. Как показали результаты исследования, изложенные в данной статье, при использовании представленного алгоритма кэширования, использующего метрики популярности контента и расстояний, нагрузка на сеть может быть значительно снижена, особенно в случае совместного кэширования.

Список литературы

1. Liu J. Proxy caching for media streaming over the Internet / J. Liu, J. Xu // IEEE Communications Magazine. – Vol. 42, № 8. – August 2004. – Pp. 88–94.
2. O’Driscoll G. Next generation IPTV services and technologies / G. O’Driscoll // John Wiley & Sons, Inc., Hoboken. – New Jersey, 2008.

References

1. Liu J. Proxy caching for media streaming over the Internet / J. Liu, J. Xu // IEEE Communications Magazine. – Vol. 42, № 8. – August 2004. – Pp. 88–94.
2. O’Driscoll G. Next generation IPTV services and technologies / G. O’Driscoll // John Wiley & Sons, Inc., Hoboken. – New Jersey, 2008.

УДК 621.31 (075.8)

МУЛЬТИМЕДИЙНЫЕ ТЕХНОЛОГИИ В ИНЖЕНЕРНОМ ОБРАЗОВАНИИ

Панфилов Степан Александрович, доктор технических наук, Мордовский государственный университет, 430005, Россия, г. Саранск, ул. Большевикская, 68, e-mail: raphilovsa@freemail.mrsu.ru.

Некрасова Нинель Романовна, кандидат технических наук, Мордовский государственный университет, 430005, Россия, г. Саранск, ул. Большевикская, 68, e-mail: nekrasovanr@yandex.ru.

Современные государственные образовательные стандарты в вузе предусматривают существенное уменьшение числа аудиторных часов на изучение и освоение учебной дисциплины. Очевидно, что детально изложить весь объем учебного материала в отведенное расписанием время невозможно.

В данной статье авторами предлагается один из способов решения этой проблемы в процессе обучения студентов инженерных направлений на примере преподавания общепрофессиональной дисциплины «Теоретические основы электротехники». В статье обсуждается создание и применение обучающих видеофильмов в инженерном образовании, приво-