



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: 2011142764/08, 17.03.2010

(24) Дата начала отсчета срока действия патента:
17.03.2010

Приоритет(ы):

(30) Конвенционный приоритет:
23.03.2009 US 61/210,888;
07.08.2009 US 12/538,086

(43) Дата публикации заявки: 10.05.2013 Бюл. № 13

(45) Опубликовано: 10.08.2014 Бюл. № 22

(56) Список документов, цитированных в отчете о
поиске: US 2007/0183493A1, 09.08.2007. US
2003/0009535 A1, 09.01.2003. RU 2303288 C2,
20.07.2007. RU 2288504 C2, 27.11.2006

(85) Дата начала рассмотрения заявки РСТ на
национальной фазе: 24.10.2011

(86) Заявка РСТ:
US 2010/027720 (17.03.2010)

(87) Публикация заявки РСТ:
WO 2010/111098 (30.09.2010)

Адрес для переписки:

129090, Москва, ул. Б. Спасская, 25, строение 3,
ООО "Юридическая фирма Городисский и
Партнеры"

(72) Автор(ы):

ПЕРЛМАН Стефен Г. (US),
ВАН ДЕР ЛАН Роджер (US),
КОТТЕР Тимоти (US),
ФЕРМАН Скотт (US),
МАККУЛ Роберт (US),
БАКЛИ Ян (US)

(73) Патентообладатель(и):

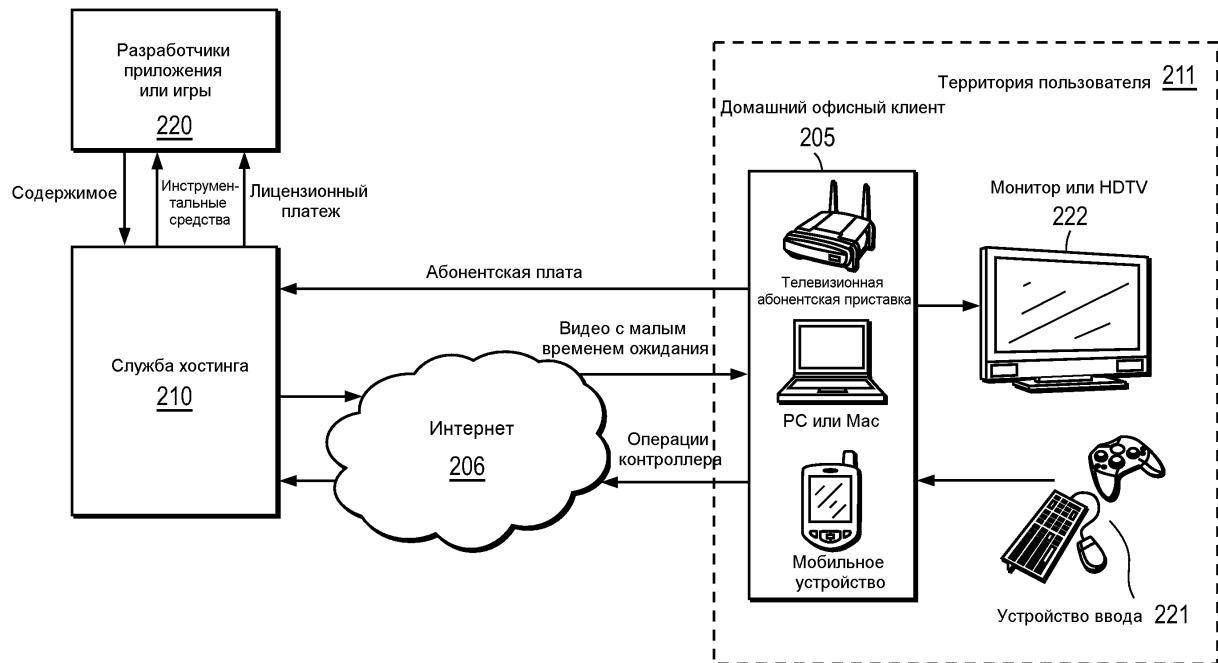
ОЛ2, ИНК. (US)

(54) СПОСОБ И УСТРОЙСТВО ВРЕМЕННОГО ДЕКОДЕРА

(57) Реферат:

Изобретение относится к средствам передачи потоком видео из сервера к клиенту. Техническим результатом является повышение быстродействия за счет стабилизации скорости передачи данных от сервера к клиенту. В способе принимают на сервере запрос клиента на видеосодержимое, определяют конфигурацию программного обеспечения и аппаратных средств клиента, формируют и/или выбирают временной декодер на основе определенной конфигурации средств

клиента, передают временной декодер клиенту, кодируют и передают поток запрошенного видеосодержимого из сервера клиенту, декодируют видеосодержимое временным декодером и визуализируют его на клиенте, определяют окончание сеанса клиента с сервером и осуществляют временное отключение и/или удаление временного декодера из клиента. 2 н. и 21 з.п. ф-лы, 55 ил.



Фиг.2а



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(12) ABSTRACT OF INVENTION

(21)(22) Application: 2011142764/08, 17.03.2010

(24) Effective date for property rights:
17.03.2010

Priority:

(30) Convention priority:
23.03.2009 US 61/210,888;
07.08.2009 US 12/538,086

(43) Application published: 10.05.2013 Bull. № 13

(45) Date of publication: 10.08.2014 Bull. № 22

(85) Commencement of national phase: 24.10.2011

(86) PCT application:
US 2010/027720 (17.03.2010)

(87) PCT publication:
WO 2010/111098 (30.09.2010)

Mail address:
129090, Moskva, ul. B. Spasskaja, 25, stroenie 3,
OOO "Juridicheskaja firma Gorodisskij i Partnery"

(72) Inventor(s):

**PERLMAN Stefen G. (US),
VAN DER LAN Rodzher (US),
KOTTER Timoti (US),
FERMAN Skott (US),
MAKKUL Robert (US),
BAKLI Jan (US)**

(73) Proprietor(s):

OL2, INK. (US)

(54) METHOD AND APPARATUS FOR TIME DECODER

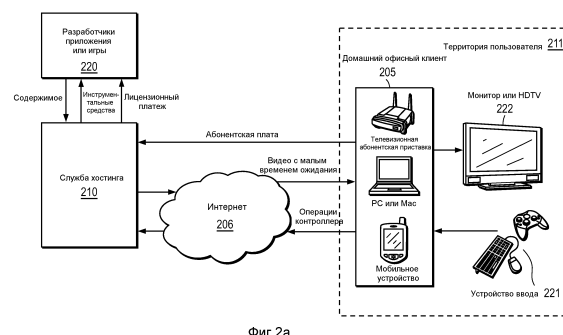
(57) Abstract:

FIELD: physics, video.

SUBSTANCE: invention relates to means of streaming video from a server to a client. The method includes receiving a client video content request at a server, determining client software and hardware configuration, creating and/or selecting a time decoder based on the determined client equipment configuration, transmitting the time decoder to the client, encoding and streaming the requested video content from the server to the client, decoding the video content with the time decoder and rendering said video content at the client, determining the end of the client/server session and performing temporary switch-off and/or deleting the time decoder from the client.

EFFECT: faster operation owing to stabilisation of rate of transmitting data from a server to a client.

23 cl, 55 dwg



Фиг.2а

Родственная заявка

Эта заявка является частичным продолжением (CIP) заявки с порядковым номером 10/315460, поданной 10 декабря 2002 г., озаглавленной "APPARATUS AND METHOD FOR WIRELESS VIDEO GAMING", права на которую принадлежат заявителю настоящей

заявки CIP.

Область техники, к которой относится изобретение

Настоящее раскрытие предмета изобретения в целом относится к области систем обработки данных, которые улучшают способность пользователей манипулировать аудио- и видеоносителями и подключаться к ним.

Уровень техники

Записываемые аудионосители и киноносители являются одной стороной общественной жизни со времен Томаса Эдисона. В начале 20-ого века были широко распространены записываемые аудионосители (валики и грампластинки) и киноносители (синематограф и фильмы), но обе технологии находились, однако, в своей ранней стадии развития. В конце 1920-ых годов кинофильмы были объединены со звуковым сопровождением на основе рынка товаров широкого потребления, вслед за этим появились цветные кинофильмы со звуковым сопровождением. Радиовещание постепенно приняло форму ширококвещательных средств распространения аудиоинформации рынка товаров широкого потребления, пользующихся поддержкой рекламы. Когда в середине 1940-ых годов был установлен стандарт телевизионного (TV) вещания, телевидение присоединилось к радио как форма ширококвещательных средств распространения рынка товаров широкого потребления, и ранее записанные кинофильмы или киноизображения в прямом эфире пришли в дом.

К середине 20-ого века в большинстве американских домов были граммофоны для воспроизведения записанных аудионосителей, радиоприемник для приема прямых аудиопередач и телевизор для воспроизведения аудио/видео (A/V) носителей прямой передачи. Очень часто эти 3 "медиаплеера" (проигрыватель, радиоприемник и телевизор) были объединены в одном корпусе с совместным использованием обычных динамиков, который стал "медиа-центром" для дома. Несмотря на то, что выбор носителей информации был ограничен для потребителя, "экосистема" носителей информации была довольно устойчивой. Большинство потребителей знали, как пользоваться "медиаплеерами" и могли в полной мере использовать их функциональные возможности. Вместе с тем, издатели мультимедиа (в значительной степени киностудии и телестудии и музыкальные компании) могли поставлять свои носители информации и в театры и для домашнего использования и не страдали от широко распространенного пиратства или "вторичных продаж", т.е. перепродаж используемого носителя информации. Как правило, издатели не получают доход от "вторичных продаж", и, по существу, это уменьшает доход, который издатели иначе могли бы получить от покупателя используемого носителя информации вследствие новых продаж. Несмотря на то, что в середине 20-ого века, конечно, использовались проданные грампластинки, такие продажи не оказывали большого влияния на издателей грампластинок, потому что, в отличие от кино или видеопрограммы - которые взрослый человек, как правило, смотрит один раз или только несколько раз - музыкальная фонограмма может прослушиваться сотни или даже тысячи раз. Поэтому носитель музыкальной информации является гораздо менее "преходящим" (т.е. он имеет непреходящую ценность для взрослого потребителя), чем кино/видео носители. После покупки грампластинки, если потребителю понравилась музыка, то он, вероятно, будет хранить ее в течение длительного времени.

С середины 20-ого века до настоящего времени экосистема носителей информации подверглась ряду радикальных перемен, как с выгодой для потребителей и издателей, так и в ущерб им. При широком распространении введения магнитофонов, особенно кассетных магнитных лент с высококачественным стереозвуком, конечно, потребителям стало намного удобнее. Но это также отметило начало того, что является теперь широко распространенной практикой по отношению к потребительским носителям информации: пиратство. Конечно, многие потребители использовали кассетные магнитные ленты для записи на нее своих собственных грампластинок просто для удобства, но все возрастающее количество потребителей (например, студенты в студенческом общежитии при свободном доступе к коллекциям пластинок друг друга) могли делать пиратские копии. Кроме того, потребители могли записывать на магнитную ленту музыку, передаваемую по радио, вместо покупки грампластинки или магнитной ленты у издателя.

Появление потребительского VCR (кассетного видеоманитфона) привело к еще большему удобству потребителей, так как теперь кассетный видеоманитфон мог быть установлен для записи телепрограммы, которую можно было посмотреть в более позднее время, и это также привело к созданию видеопроката, где доступ к фильмам, а также телепрограммам, мог предоставляться "по требованию". Быстрое развитие домашних устройств хранения данных рынка товаров широкого потребления с середины 1980-ых годов привело к беспрецедентному уровню выбора и удобства для потребителя, а также привело к быстрому расширению издательского рынка мультимедиа.

В настоящее время, потребители сталкиваются с большим выбором мультимедиа, а также с множеством устройств хранения данных, многие из которых привязаны к конкретным видам мультимедиа или конкретным издателям. Страстный потребитель мультимедиа может иметь стек устройств, подключенных к телевизорам и компьютерам, находящимся в различных помещениях дома, что в результате приводит к паутине кабелей, ведущих к одному или нескольким телевизорам и/или персональным компьютерам (PC), а также к группе пультов дистанционного управления. (В контексте настоящей заявки термин "персональный компьютер" или "PC" относится к любому виду компьютера, пригодного для дома или офиса, включающего в себя настольный компьютер, Macintosh® или другие компьютеры, не требующие использования операционной системы Windows, совместимые с Windows устройствами, разновидности UNIX, ноутбуки и т.д.) Эти устройства могут включать в себя видеоигровую консоль, кассетный видеоманитфон, DVD-плеер, звуковой процессор/усилитель объемного звука, спутниковую телевизионную абонентскую приставку, телевизионную абонентскую приставку кабельного телевидения и т.д. И, для страстного потребителя, может существовать множество устройств с аналогичными функциями из-за вопросов совместимости. Например, потребителю могут принадлежать как HD-DVD, так и Blu-ray DVD плеер, или как Microsoft Xbox®, так и Sony Playstation® игровая видеосистема. На самом деле, из-за несовместимости некоторых игр по всем версиям игровых консолей, потребителю могут принадлежать как Xbox, так и более поздняя версия, например, Xbox 360®. Часто, потребителей сбивает с толку то, какой видеовход и какой пульт дистанционного управления использовать. Даже после того, как диск вставлен в соответствующий плеер (например, DVD, HD-DVD, Blu-ray, Xbox или Playstation), выбраны видеовход и аудиовход для этого устройства, и найден соответствующий пульт дистанционного управления, потребитель, тем не менее, сталкивается с техническими проблемами. Например, в случае широкоэкранный DVD, пользователю может потребоваться сначала определить и затем установить соответствующий формат изображения на своем экране монитора или телевизора (например, 4:3, Full, Zoom, Wide

Zoom, Cinema Wide и т.д.). Аналогично, пользователю может потребоваться сначала определить и затем установить соответствующий аудиоформат системы объемного звучания (например, AC3, Dolby Digital, DTS и т.д.). Часто потребитель не осознает, что он, возможно, не использует мультимедийное содержимое в полном объеме функциональных возможностей своей телевизионной или аудио системы (например, при просмотре фильма в сжатом виде при неправильном формате изображения или при прослушивании аудио со стереофоническим звуком, а не с объемным звуком).

Все больше и больше устройств хранения данных, основанных на Internet-технологиях, добавляются к стеку устройств. Звуковые устройства, подобные системе Sonos® Digital Music, передают потоком аудио непосредственно из Internet. Аналогично, устройства, подобные телевизионной приставке Slingbox™, записывают видео и передают его потоком через домашнюю сеть или через Internet, где его можно смотреть удаленно на PC. И службы IP-телевидения (IPTV) предлагают услуги, подобные кабельному телевидению, через цифровую абонентскую линию (DSL) или другие соединения с Internet в доме. Недавно также предпринимались попытки для интеграции множества мультимедиа-функций в одном устройстве, например, Moxi® Media Center и PC под управлением операционной системы Windows XP Media Center Edition. Несмотря на то, что каждое из этих устройств предлагает элемент механизма для функций, которые оно выполняет, в каждом отсутствует повсеместно распространенный и простой доступ к большинству носителей информации. Кроме того, производство таких устройств часто стоит сотни долларов, часто из-за необходимости дорогостоящей обработки и/или локального запоминающего устройства. Кроме того, эти современные потребительские электронные устройства, как правило, потребляют много мощности, даже в режиме ожидания, что означает, что они требуют больших затрат со временем и нерационально используют энергоресурсы. Например, устройство может продолжать работать, если потребитель забыл выключить его или переключился на другой видеовход. И, так как ни одно из устройств не является законченным решением, то оно должно быть интегрировано с другим стеком устройств в доме, что по-прежнему оставляет пользователя с паутиной проводов и большим количеством пультов дистанционного управления.

Кроме того, когда многие новейшие устройства, основанные на Internet-технологиях, действительно работают должным образом, они, как правило, предлагают мультимедиа в более общей форме, чем оно могло иначе быть доступным. Например, устройства, которые передают потоком видео через Internet часто передают потоком только видеоматериал, а не интерактивные "дополнительные материалы", которые часто сопровождают DVD, подобные "мнению о" видео, играм или комментариям кинорежиссера. Это происходит вследствие того, что часто интерактивный материал выводится в конкретном формате, предназначенном для конкретного устройства, которое обрабатывает интерактивность локально. Например, каждый из дисков DVD, HD-DVD и Blu-ray имеет свой собственный конкретный интерактивный формат. Любое домашнее устройство хранения данных или локальный компьютер, который мог быть разработан для поддержки всех популярных форматов, могут потребовать некоторого уровня усложнения и гибкости, что, вероятно, сделало бы их функционирование недопустимо дорогим и сложным для потребителя.

Кроме указанной проблемы, если позже будет введен новый формат, то в локальном устройстве, может не существовать аппаратного обеспечения для поддержки этого нового формата, что будет означать то, что потребитель должен будет купить модернизированное локальное устройство хранения данных. Например, если позже

введено видео с высоким разрешением или стереоскопическое видео (например, один видеопоток для каждого глаза), то локальное устройство может не иметь вычислительной мощности для декодирования видео, или оно может не иметь аппаратного обеспечения для вывода видео в новом формате (например, предположим, что стереоскопическое восприятие достигается посредством видео с частотой 120 кадр/с, синхронизированного с затворными очками, причем кадры подаются на каждый глаз с частотой 60 кадр/с, если видеоаппаратура потребителя может поддерживать только видео с частотой 60 кадр/с, то эта опция не будет доступной без покупки модернизированного аппаратного обеспечения).

Вопрос сложности и морального износа устройства хранения данных является серьезной проблемой, когда дело доходит до усложненного интерактивного мультимедиа, особенно видеоигр.

Современные видеоигровые приложения в основном разделены на четыре главные машинозависимые аппаратные платформы: Sony PlayStation® 1, 2 и 3 (PS1, PS2 и PS3), Microsoft Xbox® и Xbox 360® и Nintendo Gamecube® и Wii™, а также основанные на PC игры. Каждая из этих платформ отличается от других так, что игры, написанные для исполнения на одной платформе обычно не исполняются на другой платформе. Могут также существовать проблемы совместимости устройства от поколения к поколению. Несмотря на то, что большинство разработчиков игровых программ создают игровые программы, независимые от конкретной платформы, для исполнения конкретной игры на конкретной платформе, причем требуется собственный слой программного обеспечения (часто называемый "механизмом разработки игровых программ") для адаптации игры для использования на конкретной платформе. Каждая платформа продается потребителю как "консоль" (т.е., автономный блок, подсоединяемый к телевизору или монитору/динамикам), или она сама является PC. Как правило, видеоигры продаются на оптическом носителе информации, например, DVD Blu-ray, DVD-ROM или CD-ROM, который содержит видеоигру, воплощенную как усложненное приложение реального времени. Так как скорости домашнего широкополосного доступа увеличились, видеоигры становятся все более и более доступными для загрузки.

Требования к специфике для достижения совместимости платформы с программным обеспечением видеоигры являются чрезвычайно высокими из-за характера режима реального времени и высоких вычислительных требований усовершенствованных видеоигр. Например, можно было бы ожидать полную совместимость видеоигр от поколения к поколению (например, от Xbox к Xbox 360 или от Playstation 2 ("PS2") к Playstation 3 ("PS3")), аналогично тому, как существует общая совместимость рабочих приложений (например, Microsoft Word) от одного PC к другому с более быстрым процессором или ядром. Однако с видеоиграми это не так. Так как производители видеоигр при выпуске поколения видеоигр, как правило, пытаются добиться максимально возможной производительности для данной ценовой точки, то часто существенно изменяют архитектуру системы так, что многие игры, написанные для системы предшествующего поколения, не исполняются на системе более позднего поколения. Например, Xbox выполнен на основе семейства процессоров x86, тогда как Xbox 360 выполнен на основе семейства PowerPC.

Могут быть использованы способы эмуляции предшествующей архитектуры, но с учетом того, что видеоигры являются приложениями реального времени, часто невозможно достичь идентичного поведения при эмуляции. Это наносит ущерб потребителю, производителю видеоигровой консоли и издателю программного обеспечения видеоигры. Для потребителя это означает необходимость наличия

видеоигровых консолей, как старого, так и нового поколения, подключенных к телевизору для возможности ведения всех игр. Для производителя консоли это означает затраты, связанные с эмуляцией, и более медленное внедрение новых консолей. И для издателя это означает необходимость выпуска множества версий новых игр для охвата

5 всех возможных потребителей - не только выпуск версии для каждого бренда видеоигры (например, XBox, Playstation), но часто версии для каждой версии данного бренда (например, PS2 и PS3). Например, отдельная версия "Madden NFL 08" Electronic Arts была разработана для XBox, XBox 360, PS2, PS3, Gamecube, Wii и PC, наряду с другими платформами.

10 Портативные устройства, например, сотовые ("cell") телефоны и портативные медиаплееры также представляют проблемы для разработчиков игр. Все большее количество таких устройств подключаются к беспроводным сетям передачи данных и могут загружать видеоигры. Но на рынке существует широкий выбор сотовых телефонов и устройств хранения данных с широким диапазоном разных разрешающих

15 способностей дисплея и вычислительных возможностей. Кроме того, так как такие устройства, как правило, имеют ограничения по потреблению мощности, стоимости и весу, у них, как правило, отсутствуют усовершенствованное аппаратное обеспечение ускорения выполнения графических операций, подобные графическому процессору ("GPU"), например, устройства, изготавливаемые корпорацией NVIDIA, Santa Clara, CA

20 (Санта-Клара, Калифорния). Следовательно, разработчики игровых программ, как правило, разрабатывают данную компьютерную игру одновременно для множества разных типов портативных устройств. Пользователь может обнаружить, что данная компьютерная игра не доступна конкретно для его сотового телефона или портативного медиаплеера.

25 В случае домашних игровых консолей производители аппаратных платформ, как правило, назначают лицензионный платеж для разработчиков игровых программ для возможности издания игры на своей платформе. Операторы сотовой связи также, как правило, назначают лицензионный платеж для издателя игр для загрузки игры в сотовый телефон. В случае компьютерных игр не существует лицензионного платежа,

30 выплачиваемого для издания игр, но разработчики игр, как правило, сталкиваются с высокими затратами из-за более высокого уровня косвенных затрат на обслуживание клиента для поддержки широкого диапазона конфигураций PC и из-за вопросов инсталляции, которые могут возникнуть. Кроме того, PC, как правило, представляют меньшие препятствия для пиратства игровых программ, так как опытный в техническом

35 отношении пользователь может легко их перепрограммировать, и игры можно легче нелегально переиздавать и легче распространять (например, через Internet). Соответственно, для разработчика игровых программ существуют затраты и неблагоприятные условия при публикации на игровых консолях, сотовых телефонах и PC.

40 Для издателей игровой консольной и компьютерной программы затраты не ограничиваются этим. Для распространения игр через розничные каналы, издатели назначают оптовую цену ниже продажной цены для розничных торговцев для получения чистой прибыли. Издатель также, как правило, должен оплачивать затраты на производство и распространение физических носителей информации, содержащих игру.

45 Розничный торговец также часто возлагает на издателя расход по "оплате ценовой защиты" для покрытия возможных непредвиденных расходов, например, когда игра не раскупается, или если цена игры снижается, или если розничный торговец должен возместить часть или всю оптовую цену и/или принять обратно игру от покупателя.

Кроме того, розничные торговцы также, как правило, возлагают на издателей расходы по оплате поддержки сбыта игр в рекламных листовках. Кроме того, розничные торговцы все чаще выкупают игры у пользователей, которые перестали играть в них, и затем продают их как подержанные игры, как правило, не разделяя дохода от их

5 продажи с издателем игр. Дополнением к бремени расходов, возложенных на издателей игр, является тот факт, что игры часто нелегально переиздают и распространяют через Internet для загрузки пользователями и изготовления ими бесплатных копий.

Так как скорости широкополосного доступа в Internet увеличиваются, и широкополосные соединения стали более широко распространенными в США и во

10 всем мире, в частности к дому и к Internet-"кафе", где арендуются PC, подключенные к Internet, игры все чаще распространяются посредством загрузки в PC или консоли. Кроме того, широкополосные соединения все чаще используют для ведения многопользовательских онлайн-игр и массовых многопользовательских онлайн-игр (и те и другие в настоящем раскрытии предмета изобретения обозначены

15 аббревиатурой "ММОГ"). Эти изменения уменьшают некоторые затраты и вопросы, связанные с распространением в розницу. Загрузка онлайн-игр направлена на преодоление некоторых неблагоприятных условий для издателей игр за счет того, что затраты на распространение, как правило, уменьшаются, и затраты из-за непроданного носителя информации являются небольшими или отсутствуют. Но загруженные игры,

20 по-прежнему, подвержены пиратству, и из-за своего размера (часто размером в много гигабайтов) они могут загружаться очень долго. Кроме того, множество игр могут заполнить дисководы небольшой (емкости), как те, которые продаются с портативными компьютерами или с видеоигровыми консолями. Однако для игр большого размера или ММОГ требуется, чтобы можно было вести эту игру по онлайн-соединению,

25 проблема пиратства уменьшается, так как обычно требуется, чтобы пользователь имел действительную учетную запись пользователя. В отличие от линейных мультимедиа (например, видео и музыка), которые можно скопировать камерой при съемке видео с экрана дисплея или микрофоном при записи аудио с динамиков, каждый опыт использования видеоигры уникален и не может быть скопирован с использованием

30 простой видео/аудио записи. Соответственно, даже в регионах, где не обеспечивается строгое выполнение законов об авторском праве, и пиратство сильно распространено, ММОГ могут быть ограждены от пиратства, и, следовательно, может поддерживаться предпринимательская деятельность. Например, ММОГ "World of Warcraft" медиаконгломерата Vivendi SA с успехом введен во всем мире и не страдает от пиратства.

35 И многие онлайн-игры или игры ММОГ, например, ММОГ "Second Life" ("Вторая Жизнь") компании Linden Lab формируют доход для операторов игр через экономические модели, встроенные в игры, где можно покупать, продавать и даже создавать ресурсы с использованием онлайн-инструментальных средств. Соответственно, для оплаты использования онлайн-игр могут использоваться механизмы в дополнение к

40 обычным покупкам игровых программ или абонентским обслуживаниям.

Несмотря на то, что пиратство во многих случаях может быть уменьшено из-за характера онлайн-игр (игр) или ММОГ, оператор интерактивных игр, тем не менее, сталкивается с остальными проблемами. Многие игры для исполнения их должным образом требуют значительных локальных (т.е., домашних) ресурсов обработки для

45 онлайн-игр (игр) или ММОГ. Если производительность локального компьютера пользователя является низкой (например, такой компьютер без GPU, как ноутбук младшей модели), то он, возможно, не сможет вести игру. Кроме того, так как игровые консоли устаревают, то они также отстают от современного уровня развития и,

возможно, не могут обрабатывать усовершенствованные игры. Даже, если предположить, что локальный РС пользователя может удовлетворять вычислительным требованиям игры, то часто существует сложность инсталляции. Может существовать несовместимость драйверов (например, если новая игра загружена, то она может

5 инсталлировать новую версию графического драйвера, который визуализирует ранее инсталлированную игру, зависящую от старой версии графического драйвера, которая не будет функционировать). Консоль может исчерпать пространство на локальном диске, когда загружается большее количество игр. Сложные игры, как правило,

10 принимают загружаемые заплатки от разработчика игр с течением времени, так как обнаружены и исправлены ошибки, или если в игре выполнены модификации (например, если разработчик игр считает, что уровень игры является слишком сложным или слишком простым). Эти заплатки требуется снова загружать. Но иногда не все пользователи выполняют загрузку всех заплат. В других случаях, загруженные заплатки вводят другие вопросы совместимости или расхода пространства на диске.

15 Кроме того, во время ведения игры может потребоваться загрузка большого количества данных для обеспечения графической или поведенческой информации в локальный РС или консоль. Например, если пользователь входит в комнату в MMOG и сталкивается со сценой или персонажем, составленными из графических данных, или с линиями поведения, которых нет на локальной машине пользователя, тогда эти данные

20 персонажа или сцены должны быть загружены. Это может в результате привести к значительной задержке во время ведения игры, если подключение к Internet не является достаточно быстрым. И если персонаж или сцена, с которыми сталкиваются, требуют объем памяти или вычислительную мощность, превышающие возможности локального РС или консоли, то может создаться ситуация, когда пользователь не может продолжать

25 игру, или должен продолжать с графикой худшего качества. Соответственно, онлайн-игры или игры MMOG часто ограничивают свои требования к вычислительной сложности и/или по памяти. Кроме того, они часто ограничивают количество передач данных во время игры. И Онлайн-игры или игры MMOG также могут сузить рынок пользователей, которые могут вести эти игры.

30 Кроме того, опытные в техническом отношении пользователи все чаще декомпилируют локальные копии игр и модифицируют эти игры так, чтобы они могли обманывать. Обманы могут быть такими простыми, как то, чтобы сделать повторное нажатие клавиши более быстрым, чем это может человек (например, чтобы стрелять из огнестрельного оружия очень быстро). В играх, которые поддерживают

35 внутриигровые транзакции ресурсов, обман может достигать уровня фальсификации, которая в результате приводит к мошенническим транзакциям, включающим в себя ресурсы фактической экономической ценности. Когда экономическая модель онлайн-игр (игр) или игр MMOG основана на таких транзакциях ресурсов, это может в результате привести к последствиям, причиняющим значительный ущерб операторам

40 игр.

Стоимость разработки новой игры выросла, когда РС и консоли стали в состоянии выводить все более усложненные игры (например, с более реалистичной графикой, например, трассировка лучей в реальном времени, и более реалистичными линиями поведения, например, моделирование физики в реальном времени). На заре видеоигровой

45 промышленности разработка видеоигр была процессом очень похожим на разработку прикладного программного обеспечения, то есть, большую часть стоимости разработки составляла разработка программного обеспечения, в отличие от разработки таких графических, звуковых и поведенческих элементов или "ресурсов", как те, которые

могут быть разработаны для кинофильма с пространственными спецэффектами. В настоящее время многие программы работ по разработке усложненных видеоигр больше напоминают разработку кинофильмов со спецэффектами, чем разработку программного обеспечения. Например, многие видеоигры обеспечивают имитации 3D (трехмерных) миров, и формируют все более фотореалистичные (т.е., компьютерную графику, которая кажется такой реалистичной, как кадр с изображением игры актеров, фотографически) персонажи, реквизит и окружающую обстановку. Одним из самых перспективных аспектов разработки фотореалистичных игр является создание сформированного компьютером человеческого лица, которое нельзя отличить от человеческого лица при игре актеров. Технологии захвата лица, как Contour™ Reality Capture, разработанный Mova, San Francisco, CA (Сан-Франциско, Калифорния) захватывает и отслеживает точную геометрию лица актера с высоким разрешением, когда оно находится в движении. Эта технология обеспечивает возможность визуализации лица 3D на PC или игровой консоли, которое практически нельзя отличить от захваченного лица при игре актеров. Точный захват и визуализация "фотореального" человеческого лица полезны в нескольких отношениях. Во-первых, в видеоиграх часто используются очень легко узнаваемые знаменитости или спортсмены (часто высокооплачиваемые), и недостатки могут быть очевидными для пользователя, что отвлекает во время игры или делает просмотр неприятным. Часто требуется высокая степень детализации для достижения высокой степени фотореалистичности, с требованием визуализации большого количества многоугольников и текстур с высоким разрешением, возможно, с многоугольниками и/или текстурами, изменяющимися на кадровой основе во время движения лица.

Когда сцены, насчитывающие большое количество многоугольников, с детализированными текстурами быстро сменяются, PC или игровая консоль, поддерживающая игру, может не иметь достаточный объем памяти RAM для хранения данных достаточного количества текстур и многоугольников для необходимого количества кадров анимации, формируемых в сегменте игры. Кроме того, на PC или игровой консоли, как правило, доступен один накопитель на оптических дисках или один накопитель на магнитных дисках, которые обычно намного медленнее, чем RAM, и, как правило, не могут не отставать от максимальной скорости передачи данных, которую GPU может принимать при визуализации многоугольников и текстур. Современные игры, как правило, загружают большинство многоугольников и текстур в RAM, что означает, что данная сцена в значительной степени ограничена по сложности и продолжительности из-за объема RAM. В случае анимации лица, например, это может ограничивать PC или игровую консоль или до лица с низким разрешением, которое не является фотореальным, или до фотореального лица, которое может быть анимировано только для ограниченного количества кадров, до того, как игра приостановится и загрузит многоугольники и текстуры (и другие данные) для дополнительных кадров.

Наблюдение за медленным перемещением индикатора выполнения по экрану при выводе на экран PC или консоли сообщения наподобие "Loading..." ("Загрузка...") принимается современными пользователями как неотъемлемый недостаток сложных видеоигр. Задержка при загрузке следующей сцены с диска ("диск" в этом описании, если не оговорено иное, относится к энергонезависимым оптическим или магнитным носителям информации, а также к недисковым носителям информации, например, к полупроводниковой "Флэш"-памяти) может занимать несколько секунд или даже несколько минут. Это является пустой тратой времени и может совершенно разочаровать игрока. Как обсуждалось ранее, большая часть или вся задержка может происходить

из-за времени загрузки многоугольника, текстуры или других данных с диска, но также может случиться так, что часть времени загрузки тратится на подготовку процессором и/или GPU в PC или консоли данных для сцены. Например, видеоигра в футбол может обеспечивать возможность игрокам отбирать из большого количества игроков, команд, стадионов и метеорологических условий. Так, в зависимости от того, какая конкретная комбинация выбрана, для сцены могут потребоваться разные многоугольники, текстуры и другие данные (обобщенно "объекты") (например, разные команды имеют разные цвета и рисунки на своей форме). Может оказаться возможным перечислить многие или все различные переменные и предварительно вычислить многие или все объекты заранее, и сохранить эти объекты на диске, используемом для хранения игры. Но, если количество переменных является большим, то объем памяти, требуемый для всех объектов, может быть слишком большим и не соответствовать диску (или слишком нереальным для загрузки). Соответственно, существующие PC и консольные системы, как правило, являются ограниченными как по сложности, так и по продолжительности воспроизведения данных сцен, и страдают от долгого времени загрузки для сложных сцен.

Другим существенным ограничением систем прикладного программного обеспечения и игровых видеосистем на предшествующем уровне техники является то, что они все чаще используют большие базы данных, например, таких 3D объектов, как многоугольники и текстуры, которые требуется загружать в PC или игровую консоль для обработки. Как обсуждалось выше, загрузка таких баз данных может занимать много времени при хранении на локальном диске. Время загрузки, однако, обычно намного больше, если база данных хранится удаленно, и доступ к ней осуществляется через Internet. В таком случае могут потребоваться минуты, часы или даже дни для загрузки большой базы данных. Кроме того, создание таких баз данных часто связано с большими расходами (например, 3D модель детализированного парусного судна, снабженного высокой мачтой, для использования в игре, фильме или историческом документальном фильме), и они предназначены для продажи локальному конечному пользователю. Однако база данных подвергается риску несанкционированного использования после ее загрузки локальным пользователем. Во многих случаях пользователю требуется загрузка базы данных просто для ее оценки, чтобы проверить, удовлетворяет ли она его потребностям (например, имеет ли 3D костюм для персонажа игры удовлетворительный вид, когда пользователь выполняет конкретное перемещение). Длительное время загрузки может быть сдерживающим фактором для пользователя, оценивающего 3D базу данных до того, как принять решение о покупке.

Аналогичные вопросы имеют место в MMOG, в частности, как например, в играх, которые обеспечивают возможность пользователям использовать все более настраиваемых самостоятельно персонажей. Чтобы PC или игровая консоль выводили на экран персонажа, у них должен быть доступ к базе данных 3D геометрии (многоугольники, текстуры и т.д.), а также линий поведения (например, имеет ли персонаж щит, достаточно ли прочный этот щит, чтобы отклонить копьё или нет) для этого персонажа. Как правило, когда пользователь впервые ведет MMOG, большое количество баз данных для персонажа уже поставляется с исходной копией игры, которая доступна локально на оптическом диске игры или загружается на диск. Но, по мере продвижения игры, если пользователь сталкивается с персонажем или объектом, база данных которого не доступна локально (например, если другой пользователь создал персонажа, настраиваемого самостоятельно), то до того, как этот персонаж или объект могут быть выведены на экран, их база данных должна быть загружена. Это в

результате может привести к значительной задержке игры.

С учетом уровня усложненности и сложности видеоигр, другой проблемой для разработчиков видеоигр и издателей, связанной с видеоигровыми консолями предшествующего уровня техники, является то, что разработка видеоигры часто
 5 занимает от двух до трех лет при стоимости десятки миллионов долларов. С учетом того, что новые платформы видеоигровой консоли вводят со скоростью примерно по одной через каждые пять лет, разработчики игр должны начинать разработку этих игр за годы до выпуска новой игровой консоли, чтобы видеоигры появились в продаже одновременно с выпуском новой платформы. Иногда выпускают несколько консолей
 10 от конкурирующих производителей примерно в одно время (например, с интервалом один или два года), но еще неизвестна популярность каждой консоли, например, какая консоль сделает самые большие объемы продаж программного обеспечения видеоигры. Например, в недавнем цикле консолей, было запланировано ввести Microsoft Xbox 360, Sony Playstation 3 и Nintendo Wii примерно в один общий период времени. Но за годы
 15 до упомянутых введений разработчики игр по существу должны были "делать свои ставки" на то, какие платформы консоли будут более успешными, чем другие, и отдавать свои ресурсы разработки соответственно. Кинокомпании также должны соразмерно распределять свои ограниченные производственные ресурсы исходя из того, что, как они оценивают, будет вероятным успехом фильма, задолго до выпуска этого фильма.
 20 С учетом роста уровня инвестиций, требуемых для видеоигр, производство игр все больше становится похожим на производство кинофильмов, и компании по производству игр обычно отдают свои производственные ресурсы исходя из своей оценки будущего успеха конкретной видеоигры. Но в отличие от (они) кинокомпаний, эта ставка основана не просто на успехе самого производства, скорее она основана на успехе игровой
 25 консоли, на которой эта игра должна исполняться. Выпуск игры на многих консолях одновременно может уменьшить риск, но из-за этой дополнительной программы работ возрастает стоимость, и часто происходит задержка фактического выпуска игры.

Условия работы пользователя и условия разработки прикладного программного обеспечения на PC требуют все большего объема вычислений, становятся динамичными
 30 и интерактивными, не только, чтобы сделать их более привлекательными визуально для пользователей, но также и сделать их более полезными и интуитивно-понятными. Например, как новая операционная система Windows Vista™, так и последующие версии операционной системы Macintosh® включают в себя визуальные анимационные эффекты. Усовершенствованные графические средства, например, Maya™ от Autodesk, Inc.,
 35 обеспечивают способность к воспроизведению динамических изображений и самой усложненной 3D визуализации, которые расширяют границы современных CPU и GPU. Однако вычислительные требования этих новых инструментальных средств вызывают несколько практических вопросов у пользователей и разработчиков программного обеспечения таких продуктов.

40 Так как визуальное отображение операционной системы (OS) должно работать на широком диапазоне классов компьютеров - в том числе на компьютерах предшествующего поколения, которые больше не продаются, но на которых, тем не менее, можно заменить операционную систему (OS) на новую - графические требования OS ограничены в значительной степени "наименьшим общим знаменателем"
 45 компьютеров, для которых предназначена эта OS, который, как правило, включает в себя компьютеры, которые не включают в себя GPU. Это очень ограничивает функциональные возможности графических средств OS. Кроме того, портативные компьютеры с батарейным питанием (например, ноутбуки) ограничивают возможности

визуального отображения, так как высокий уровень вычислительной активности CPU или GPU, как правило, в результате приводит к более высокому уровню потребляемой мощности и меньшему времени работы аккумулятора. Портативные компьютеры, как правило, включают в себя программное обеспечение, которое автоматически понижает активность процессора для уменьшения потребляемой мощности, когда процессор не используется. В некоторых моделях компьютера пользователь может понизить активность процессора вручную. Например, ноутбук VGN-SZ280P корпорации Sony содержит переключатель, обозначенный "Stamina", с одной стороны (для низкого уровня производительности, большего времени работы аккумулятора) и "Speed" с другой (для высокого уровня производительности, меньшего времени работы аккумулятора). OS, под управлением которой работает портативный компьютер, должна быть пригодной к использованию даже в случае, если компьютер работает с производительностью, равной доле от своей максимальной производительности. Соответственно, производительность графической подсистемы OS часто остается гораздо ниже доступной на современном уровне техники вычислительной мощности.

Приложения с широкими функциональными возможностями, требующие большого объема вычислений, подобные Maya, часто продают с ожиданием того, что они будут использоваться на высокопроизводительных PC. Этим, как правило, определяется требование к "наименьшему общему знаменателю" более дорогому и менее портативному с гораздо большей производительностью. Как следствие, у таких приложений гораздо более ограниченная целевая аудитория, чем у универсальной OS (или универсального рабочего приложения, подобного Microsoft Office), и, как правило, (их) продают в гораздо меньшем объеме, чем программное обеспечение универсальной OS или универсальное прикладное программное обеспечение. Потенциальная аудитория также ограничена, потому что потенциальному пользователю часто трудно испытать такие приложения, требующие большого объема вычислений, заранее. Например, предположим, что студенту требуется узнать, как пользоваться Maya, или потенциальному покупателю, которому уже хорошо известны такие приложения, требуется испытать Maya до того, как вкладывать деньги в покупку (что также может включать в себя покупку машины старшей модели, которая может исполнять Maya). Несмотря на то, что студент или потенциальный покупатель могут загрузить демонстрационную версию Maya или получить физическую копию носителя информации с демонстрационной версией Maya, если у них нет компьютера, который может исполнять Maya со всеми его потенциальными возможностями (например, с обработкой сложной 3D сцены), то они не смогут полностью оценить этот продукт. Это существенно ограничивает аудиторию для таких приложений с широкими функциональными возможностями. Это также способствует увеличению продажной цены, так как стоимость разработки обычно погашается гораздо меньшим количеством покупок, чем в случае универсального приложения.

Дорогостоящие приложения также побуждают людей и компании к использованию "пиратских" копий прикладного программного обеспечения. В результате, высокопроизводительное прикладное программное обеспечение страдает от сильно распространенного пиратства, несмотря на существенные усилия издателей такого программного обеспечения уменьшить такое пиратство различными способами. Однако, даже при использовании "пиратских" приложений с широкими функциональными возможностями пользователи не могут избежать необходимости вкладывать деньги в дорогостоящие современные PC для исполнения пиратских копий. Следовательно, несмотря на то, что пользователи пиратского программного обеспечения могут

использовать приложения за цену, равную доле от его фактической розничной цены, они, тем не менее, должны покупать или приобретать дорогостоящий РС для полного использования этого приложения.

Это относится и к пользователям пиратских видеоигр с высокими характеристиками.

5 Несмотря на то, что пираты могут получить игры за цену, равную доле от их фактической цены, они, тем не менее, должны покупать дорогостоящую вычислительную технику (например, РС с усовершенствованным GPU или высокопроизводительную видеоигровую консоль, подобную XBox 360), необходимую для ведения игры должным образом. С учетом того, что видеоигры, как правило, являются времяпрепровождением
10 для потребителей, дополнительные затраты на высокопроизводительную игровую видеосистему могут являться препятствием. Эта ситуация ухудшается в странах (например, в Китае), где среднегодовой доход рабочих в настоящее время является довольно низким относительно среднегодового дохода рабочих в США. В результате гораздо меньший процент людей владеет высокопроизводительной игровой
15 видеосистемой или высокопроизводительным РС. В таких странах весьма распространены "Internet кафе", в которых пользователи оплачивают использование компьютера, подключенного к Internet. Часто такие Internet кафе имеют старую модель или низкопроизводительный РС без высокопроизводительных признаков, например, GPU, которые могли бы иначе обеспечить возможность игрокам вести видеоигры,
20 требующие большого объема вычислений. Это является решающим фактором успеха игр, которые исполняются на низкопроизводительных РС, например, "World of Warcraft" медиаконгломерата Vivendi, которая является очень успешной в Китае, и там в нее обычно играют в Internet кафе. Напротив, в игру, требующую большого объема вычислений, подобную "Second Life", с гораздо меньшей вероятностью можно играть
25 на РС, установленном в китайском Internet кафе. Такие игры фактически недоступны пользователям, которые имеют доступ только к низкопроизводительным РС в Internet кафе.

Препятствия также существуют для пользователей, которые рассматривают покупку видеоигры и хотели бы сначала испытать демонстрационную версию игры с загрузкой
30 ее через Internet в свой домашний компьютер. Демонстрационная версия видеоигры часто является полнофункциональной версией игры с некоторыми недоступными признаками или с ограничениями, наложенными на количество проведений игры. Может подразумеваться длительный процесс (возможно часы) загрузки гигабайтов данных до того, как игра будет инсталлирована и исполняться на РС или на консоли. В случае
35 РС может также подразумеваться выяснение того, какие специальные драйверы необходимы (например, драйверы OpenGL или DirectX) для этой игры, загрузка соответствующей версии, инсталляция их, и затем определение того, можно ли на РС вести эту игру. Этот последний этап может включать в себя определение того, существуют ли у РС достаточные возможности для обработки (CPU и GPU), достаточная
40 RAM и совместимая OS (например, некоторые игры исполняются на Windows XP, но не исполняются на Vista). Соответственно, после длительного процесса при попытке запуска демонстрационной версии видеоигры, пользователь может обнаружить, что демонстрационная версия видеоигры, возможно, не может быть воспроизведена, с учетом конфигурации РС пользователя. Хуже, если после загрузки новых драйверов
45 пользователем для испытания демонстрационной версии, эти версии драйверов могут оказаться несовместимыми с другими играми или приложениями, которые пользователь постоянно использует на РС, соответственно, инсталляция демонстрационной версии может привести ранее работающие игры или приложения в неработающее состояние.

Эти препятствия не только разочаровывают пользователя, но они создают препятствия издателям программного обеспечения видеоигр и разработчикам видеоигр для сбыта своих игр.

Другая проблема, которая в результате приводит к экономической неэффективности, имеет отношение к тому, что данные PC или игровая консоль обычно предназначены для обеспечения определенного уровня требования к производительности для приложений и/или игр. Например, некоторые PC имеют большую или меньшую RAM, более медленные или более быстрые CPU и более медленные или более быстрые GPU, если они вообще имеют GPU. Некоторые игры или приложения используют преимущества всей вычислительной мощности данного PC или консоли, в то время как многие игры или приложения не используют их. Если игре или приложению, которую выбирает пользователь, требуется производительность меньше максимальной производительности локального PC или консоли, то пользователь может зря потратить деньги на неиспользуемые признаки PC или консоли. В случае консоли, производитель консоли может заплатить больше, чем было необходимо для финансирования затрат на консоль.

Другая проблема, которая существует в сбыте и использовании видеоигр, включает в себя учет того, что пользователь просматривает другие игры до того, как он совершает покупку этой игры. На предшествующем уровне техники существует несколько подходов для записи видеоигр для повторного воспроизведения в дальнейшем. Например, в патентной заявке США № 5558339 сообщается о записи информации о состоянии игры, включающей в себя операции игрового контроллера, в течение "геймплей" (процесс игры) в компьютере клиента видеоигры (принадлежащего идентичному или другому пользователю). Эта информация о состоянии может быть использована в дальнейшем для повторного воспроизведения некоторых или всех операций игры на компьютере клиента видеоигры (например, PC или консоли). Существенным недостатком этого подхода является то, что для того, чтобы пользователю просмотреть записанную игру, он должен иметь компьютер клиента видеоигры, который может воспроизводить игру, и должен иметь видеоигровое приложение, исполняющееся на этом компьютере так, что геймплей является идентичным при повторном воспроизведении состояния записанной игры. Кроме того, видеоигровое приложение должно быть написано таким способом, что не существует возможности различия исполнения между записанной игрой и повторно воспроизводимой игрой.

Например, графическое представление игры обычно вычисляется на покадровой основе. Для многих игр, игровая логика иногда может занимать меньше или больше времени, чем один период кадра, для вычисления графического представления, выводимого на экран для следующего кадра, в зависимости от того, является ли сцена особенно сложной, или существуют ли другие задержки, которые замедляют исполнение (например, на PC, может исполняться другой процесс, который забирает циклы CPU у игровых приложений). В такой игре может со временем встретиться "пороговый" кадр, который вычисляется за несколько меньшее время, чем один период кадра (скажем, на несколько тактов CPU меньше). Когда снова вычисляется идентичная сцена с использованием идентичной информации о состоянии игры, она может легко занять на несколько тактов CPU больше, чем один период кадра (например, если внутренняя шина CPU немного не совпадает по фазе с внешней шиной DRAM, и это вводит несколько единиц времени цикла CPU задержки, даже если не существует большой задержки из-за другого процесса, забирающего миллисекунды процессорного времени у обработки игры). Следовательно, когда игра повторно воспроизводится, кадр

вычисляется в течение двух периодов кадра, а не в течение одного периода кадра.

Некоторые линии поведения основаны на том, как часто в игре вычисляется новый кадр (например, когда игра производит выборку входных данных из игровых контроллеров). Когда ведется игра, это несоответствие в привязке ко времени для

5 разных линий поведения не влияет на ведение игры, но это может в результате привести к тому, что при повторном воспроизведении игры выведется другой результат.

Например, если баллистика баскетбола вычисляется с устойчивой скоростью 60 кадр/с, но выборка входных данных игрового контроллера производится на основе скорости вычисленных кадров, то скорость вычисленных кадров могла быть 53 кадр/с, когда

10 игра записывалась, но 52 кадр/с, когда игра повторно воспроизводится, что может привести к несовпадению между тем, вошел ли баскетбольный мяч в корзину или нет, что в результате приведет к другому исходу. Соответственно, использование состояния игры для записи видеоигр требует очень осторожного проектирования игровой программы для обеспечения того, что повторное воспроизведение с использованием

15 идентичной информации о состоянии игры выведет идентичный исход.

Другим подходом предшествующего уровня техники для записи видеоигры является простая запись видеовыхода PC или игровой видеосистемы (например, на VCR, устройство записи на диски DVD или на плату захвата видео на PC). Видео после этого может быть перемотано и повторно воспроизведено, или в качестве альтернативы,

20 записанное видео может быть выгружено в Internet, как правило, после сжатия.

Недостатком этого подхода является то, что, когда последовательность кадров 3D игры повторно воспроизводится, пользователь ограничен просмотром этой последовательности кадров только с точки обзора, с которой эта последовательность кадров была записана. Другими словами, пользователь не может изменить точку обзора

25 сцены.

Кроме того, когда сжатое видео записанной последовательности кадров игры, воспроизводимое на домашнем PC или игровой консоли, делают доступным другим пользователям через Internet, даже если это видео сжато в реальном времени, может оказаться невозможной выгрузка этого сжатого видео в реальном времени в Internet.

30 Причиной, по которой это происходит, является то, что много домов в мире, которые подключены к Internet, имеют в высокой степени асимметричные широкополосные соединения (например, DSL и кабельный модем, как правило, имеют гораздо большую полосу пропускания нисходящего потока данных, чем полосу пропускания восходящего потока данных). Сжатые видеопоследовательности с высоким разрешением часто

35 имеют большие полосы пропускания, чем пропускная способность восходящего потока данных сети, что делает невозможной выгрузку их в реальном времени. Соответственно, может иметь место существенная задержка после воспроизведения последовательности кадров игры (возможно, минуты или даже часы) до того, как другой пользователь в Internet сможет просмотреть игру. Несмотря на то, что эта задержка допустима в

40 определенных ситуациях (например, для просмотра выполнений игрока, которые происходили раньше), она делает невозможным просмотр игры в прямом эфире (например, соревнование по баскетболу с участием победителей) или с возможностью "повтора", когда игра воспроизводится в прямом эфире.

Другой подход предшествующего уровня техники обеспечивает возможность зрителю, имеющему телевизор, просматривать видеоигры в прямом эфире, но только под

45 управлением телевизионной производственной группы. Некоторые телевизионные каналы, как в США, так и в других странах, обеспечивают каналы для просмотра видеоигр, по которым телевизионная аудитория может смотреть определенных

пользователей видеоигры (например, игроков с самым высоким рейтингом, участвующих в соревнованиях). Это выполняется при наличии видеовыхода игровых видеосистем (PC и/или консолей), подаваемого в оборудование для обработки и распространения видео для телевизионного канала. Это мало чем отличается от трансляции по телевизионному каналу баскетбольной игры в прямом эфире, в которой несколько камер обеспечивают линии передачи в прямом эфире под разными углами вокруг баскетбольной площадки. После этого телевизионный канал может использовать свое оборудование для спецэффектов и обработки видео/аудио для манипуляции выходом из различных игровых видеосистем. Например, телевизионный канал может накладывать текст поверх видео из видеоигры, в которой указывается статус разных игроков (аналогично тому, как можно накладывать текст во время баскетбольной игры в прямом эфире), и телевизионный канал может накладывать дополнительное аудио от комментатора, который может обсуждать действия, происходящие во время игр. Кроме того, видеоигровой выход может быть комбинирован с камерами, записывающими видео фактических игроков игр (например, с показом их эмоциональной реакции на игру).

Одной проблемой, связанной с этим подходом, является то, что такие линии передачи видео в прямом эфире должны быть доступны для оборудования телевизионного канала для обработки и распространения видео в реальном времени для того, чтобы у него было возбуждение прямой передачи. Однако, как обсуждалось ранее, это часто бывает невозможно (осуществить), когда игровой видеосистемой управляют из дома, особенно, если часть трансляции включает в себя видео в прямом эфире из камеры, которая захватывает реальное видео игрока. Кроме того, во время соревнования, существует беспокойство, что игрок, находящийся дома, может модифицировать игру и обманывать, как описано выше. По этим причинам такие трансляции видеоигры по телевизионным каналам часто komponуют так, что игроки и игровые видеосистемы сосредоточены в общественном месте (например, в телевизионной студии или на арене), где телевизионное производственное оборудование может принимать линии передачи видео из множества игровых видеосистем и, возможно, камер для прямой передачи.

Несмотря на то, что такие видеоигровые телевизионные каналы предшествующего уровня техники могут обеспечить очень захватывающее представление для телевизионной аудитории, которое является практическим опытом близким к реальному спортивному событию, например, с видеоигроками, представленными как "спортсмены", и в смысле их действий в мире видеоигры, и в смысле их действий в реальном мире, эти игровые видеосистемы часто ограничены положениями, когда игроки находятся в непосредственной физической близости друг с другом. И, так как телевизионные каналы транслируют, то каждый вещательный канал может показывать только один видеопоток, который выбирается производственной группой телевизионного канала. Из-за этих ограничений и высокой стоимости эфирного времени, производственного оборудования и производственных групп, по таким телевизионным каналам, как правило, показывают только игроков с самым высоким рейтингом, участвующих в самых важных соревнованиях.

Кроме того, данный телевизионный канал, транслирующий полноэкранное изображение видеоигры всей телевизионной аудитории, показывает только одну видеоигру в данный момент времени. Это очень ограничивает возможности выбора телезрителя. Например, телезритель, возможно, не интересуется игрой(ами), показываемой(ыми) в данное время. Другой зритель может интересоваться только просмотром игры конкретного игрока, которую не показывают по телевизионному

каналу в данное время. В других случаях зритель может интересоваться только просмотром того, как опытный игрок управляет конкретным уровнем в игре. Однако другие зрители могут захотеть управлять точкой обзора, с которой смотрят видеоигру, которая отличается от точки обзора, выбранной производственной группой и т.д.

5 Одним словом, телезритель может иметь множество предпочтений при просмотре видеоигр, которые не обеспечиваются конкретной трансляцией по телевизионной сети, даже если доступны несколько разных телевизионных каналов. По всем вышеупомянутым причинам видеоигровые телевизионные каналы предшествующего уровня техники имеют существенные ограничения в представлении видеоигр
10 телезрителям.

Другой недостаток игровых видеосистем предшествующего уровня техники и систем прикладного программного обеспечения состоит в том, что они являются сложными, и обычно страдают от ошибок, аварийных отказов и/или непреднамеренных и
15 нежелательных линий поведения (обобщенно, "ошибки"). Несмотря на то, что игры и приложения, как правило, проходят через процесс настройки и отладки (часто называемый "гарантии качества программного обеспечения" или SQA) перед выпуском, почти неизменно, после выпуска игры или приложения для широкой аудитории, в условиях эксплуатации неожиданно возникают ошибки. К сожалению, разработчику программного обеспечения трудно идентифицировать и обнаружить многие из ошибок
20 после выпуска. Разработчикам программного обеспечения может быть трудно узнать об ошибках. Даже когда они узнают об ошибке, у них может иметься в распоряжении только ограниченный объем информации для идентификации причины этой ошибки. Например, пользователь может позвонить по телефону в службу поддержки клиентов разработчика игры и оставить сообщение о том, что при ведении игры, экран начал
25 мерцать, потом стал темно-синим, и PC завис. Это дает группе SQA очень мало полезной информации для обнаружения ошибки. Некоторые игры или приложения, с которыми устанавливают связь в режиме онлайн, могут иногда давать больше информации в определенных случаях. Например, иногда может использоваться "сторожевой" процесс для осуществления текущего контроля игры или приложения на предмет "аварийного
30 отказа". Сторожевой процесс может собирать статистику о статусе процесса приложений или игры (например, о статусе стека, использования памяти, того, как далеко игра или приложения продвинулись и т.д.), когда происходит аварийный отказ, и затем передавать эту информацию в группу SQA через Internet. Но в сложной игре или приложении расшифровка такой информации может занять очень много времени для точного
35 определения того, что пользователь делал во время аварийного отказа. Даже после этого, может оказаться невозможным определить последовательность событий, которая привела к аварийному отказу.

Еще одной проблемой, связанной с PC и игровыми консолями, является то, что они подлежат обслуживанию, что причиняет большое неудобство потребителю. Вопросы
40 обслуживания также влияют на производителя PC или игровой консоли, так как они, как правило, обязаны отправлять специальный ящик для безопасной перевозки сломанных PC или консоли, и затем нести расходы по ремонту, если PC или консоль находятся на гарантии. На издателя прикладного программного обеспечения или игры может также влиять потеря продаж (или использование онлайн-обслуживания)
45 вследствие того, что PC и/или консоли находятся в состоянии ремонта.

На фиг.1 изображена игровая видеосистема предшествующего уровня техники, например, Sony Playstation® 3, Microsoft Xbox 360®, Nintendo Wii™, персональный компьютер на базе Windows или Macintosh корпорации Apple. Each of these systems includes

a central processing unit (CPU) for executing program code, typically a graphical processing unit (GPU) for performing advanced graphical operations, and multiple forms of input/output (I/O) for communicating with external devices and users. Каждая из этих систем включает в себя центральный процессор (CPU) для исполнения управляющей программы, как правило, графический процессор (GPU) для выполнения усовершенствованных графических операций и разнообразные виды ввода/вывода (I/O) для обмена информацией с внешними устройствами и пользователями. Для простоты эти компоненты изображены объединенными вместе как один блок 100. Также изображено, что игровая видеосистема предшествующего уровня техники по фиг.1 включает в себя накопитель 104 на оптических носителях информации (например, накопитель высокой емкости для дисков), накопитель 103 на жестких дисках для хранения данных и управляющей программы видеоигры, сетевое соединение 105 для ведения игр с несколькими участниками, для загрузки игр, заплат, демонстрационных версий или другого мультимедиа, оперативное запоминающее устройство (RAM) 101 для хранения управляющей программы, в настоящее время исполняемой CPU/GPU 100, игровой контроллер 106 для приема входных команд от пользователя во время геймплея и дисплей 102 (например, SDTV/HDTV или компьютерный монитор).

Система предшествующего уровня техники, изображенная на фиг.1, имеет несколько ограничений. Во-первых, накопители 104 на оптических дисках и накопители 103 на жестких дисках, как правило, имеют гораздо более медленные скорости доступа по сравнению со скоростью доступа RAM 101. При работе непосредственно через RAM 101, CPU/GPU 100 может, на практике, обрабатывать гораздо больше многоугольников в секунду, чем это возможно, когда управляющая программа и данные считываются непосредственно с накопителя 103 на жестких дисках или накопителя 104 на оптических дисках вследствие того, что RAM 101 обычно имеет гораздо большую полосу пропускания и не страдает от задержек дисковых механизмов относительно долгого поиска. Но в этих системах предшествующего уровня техники обеспечен только ограниченный объем RAM (например, 256-512 МБайт). Следовательно, часто требуется последовательность кадров "Загрузка...", во время которой RAM 101 периодически заполняется данными для следующей сцены видеоигры.

В некоторых системах делается попытка совмещать загрузку управляющей программы одновременно с геймплея, но это можно сделать только, когда известна последовательность событий (например, если автомобиль едет по дороге, то может быть загружена геометрия для приближающиеся зданий, находящихся на краю дороги, в то время как автомобиль едет). Для сложной и/или быстрой смены сцен, этот тип совмещения обычно не работает. Например, в случае, когда пользователь находится в середине сражения, и RAM 101 полностью заполнена данными, представляющими объекты, находящиеся в поле зрения в этот момент, если пользователь быстро перемещает изображение влево для просмотра объектов, которые в настоящее время не загружены в RAM 101, то в результате произойдет нарушение непрерывности в действии, так как не будет достаточно времени, чтобы загрузить новые объекты из накопителя 103 на жестких дисках или оптического носителя 104 информации в RAM 101.

Другая проблема, связанная с системой по фиг.1, возникает из-за ограничений емкости памяти накопителей 103 на жестких дисках и оптического носителя информации 104. Несмотря на то, что дисковые запоминающие устройства могут быть изготовлены с относительно большой емкостью памяти (например, 50 гигабайтов или больше), они, тем не менее, не обеспечивают достаточное количество емкости памяти для

определенных сценариев, встречающихся в современных видеоиграх. Например, как упоминалось ранее, в видеоигре в футбол пользователю может обеспечиваться возможность отбирать из десятков команд, игроков и стадионов по всему миру. Для каждой команды, каждого игрока и каждого стадиона необходимо большое количество карт отображения текстуры и карт окружения для описания 3D поверхностей в мире (например, каждая команда имеет уникальную футболку, причем каждая требует уникальной карты отображения текстуры).

Один способ, используемый для решения этой последней проблемы, заключается в том, чтобы в игре предварительно вычислять карты отображения текстуры и карты окружения после того, как они выбраны пользователем. Это может включать в себя несколько процессов, требующих большого объема вычислений, в том числе восстановление сжатых изображений, 3D отображение, затенение, организация структур данных и т.д. В результате для пользователя может иметь место задержка, в то время как в видеоигре выполняются эти вычисления. Одним способом уменьшения этой задержки, в принципе, является выполнение всех этих вычислений - включающих в себя каждую переменную команды, состава команды и стадиона - когда игра первоначально разрабатывалась. Выпускаемая версия игры тогда может включать в себя все эти предварительно обработанные данные, хранящиеся на оптическом носителе 104 информации, или на одном или нескольких серверах в Internet с только переключаемыми предварительно обработанными данными для данных команды, состава команды, выбора стадиона, загружаемыми через Internet в накопитель 103 на жестких дисках, когда пользователь делает выбор. Как практический вопрос, однако, такие предварительно загруженные данные каждой переменной, возможной при ведении игры, могут легко составлять терабайты данных, что намного превышает емкость современных оптических устройств хранения данных. Кроме того, данные для данных команды, состава команды, выбора стадиона могут легко составлять сотни мегабайтов данных или больше. В отношении домашнего сетевого подключения, скажем, 10 Мбит/с, загрузка этих данных через сетевое соединение 105 может занимать больше времени, чем вычисление упомянутых данных локально.

Соответственно, при архитектуре игры предшествующего уровня техники, изображенной на фиг.1, у пользователя происходят существенные задержки между основными монтажными переходами сложных игр.

Другая проблема, связанная с такими подходами предшествующего уровня техники, как тот, который изображен на фиг.1, состоит в том, что со временем видеоигры становятся более усовершенствованными и требуют большей вычислительной мощности CPU/GPU. Соответственно, даже при предположении неограниченного объема RAM, требования к аппаратному обеспечению видеоигры превышают максимальный уровень вычислительной мощности, имеющейся в этих системах. В результате пользователи должны модернизировать игровую аппаратуру через каждые несколько лет, чтобы не отставать (или вести более новые игры при более низких уровнях качества). Одним последствием тенденции еще большего усовершенствования видеоигр является то, что домашние видеоигровые вычислительные машины, как правило, являются экономически неэффективными, потому что их стоимость обычно определяется в соответствии с требованиями игры самой высокой производительности, которую они могут поддерживать. Например, Xbox 360 может использоваться для ведения игры, подобной "Gears of War", которая требует высокоэффективных CPU, GPU и сотни мегабайтов RAM, или Xbox 360 может использоваться для игры Pac Man, игры из 1970-ых, которая требует только килобайты RAM и CPU очень низкой производительности. На самом

деле, Xbox 360 обладает достаточной вычислительной мощностью для размещения многих параллельных игр Pac Man одновременно.

Видеоигровые вычислительные машины, как правило, выключены в течение большего количества времени в неделю. Согласно исследованию Nielsen Entertainment в июле 2006 г. активных игроков в возрасте 13 лет и старше, в среднем, активные игроки проводят четырнадцать часов в неделю за игрой в консольные видеоигры, или только 12% всего времени в неделю. Это означает, что средняя видеоигровая консоль не используется 88% времени, что является неэффективным использованием дорогостоящего ресурса. Это особенно важно с учетом того, что видеоигровые консоли часто финансируются производителем для снижения покупной цены (с ожиданием, что эта дотация будет возвращена вследствие лицензионных платежей от будущих покупок программного обеспечения видеоигры).

Видеоигровые консоли также несут расходы, связанные с почти любым потребительским электронным устройством. Например, электроника и механизмы систем должны быть смонтированы в корпусе. Производитель должен предоставлять гарантию на сервисное обслуживание. Розничный торговец, который продает систему, должен получать прибыль от любой продажи системы и/или от продажи программного обеспечения видеоигры. Все эти факторы добавляют к стоимости видеоигровой консоли, которая должна финансироваться или производителем, с передачей потребителю, или обоими.

Кроме того, основной проблемой видеоигровой промышленности является пиратство. Механизмы безопасности, используемые фактически на каждой основной игровой видеосистеме, "взламывают" со временем, что в результате приводит к несанкционированному копированию видеоигр. Например, система безопасности Xbox 360 была взломана в июле 2006 г., и пользователи теперь могут загружать нелегальные копии в режиме онлайн. Игры, которые можно загружать, (например, игры для PC или Mac) являются особенно уязвимыми для пиратства. В определенных регионах мира, где пиратство слабо контролируется, по существу, не существует жизнеспособного рынка для автономной программного обеспечения видеоигры, потому что пользователи могут покупать пиратские копии так же легко, как и легальные копии за цену, равную очень незначительной доле от цены. Кроме того, во многих частях мира цена игровой консоли составляет такой высокий процент от дохода, что, даже если бы пиратство контролировали, то мало людей могли бы позволить себе современную игровую систему.

Кроме того, рынок поддержанных игр уменьшает доход видеоигровой промышленности. Когда пользователь теряет интерес к игре, он может продать игру розничному торговцу, который перепродает эту игру другим пользователям. Это несанкционированная но обычная практика значительно уменьшает доходы издателей игр. Аналогично, обычно происходит сокращение продаж приблизительно на 50% при переходе на другую платформу каждые несколько лет. Это происходит потому, что пользователи перестают покупать игры для старых платформ, когда они узнают, что скоро должна быть выпущена новая версия платформы (например, когда собираются выпустить Playstation 3, пользователи перестают покупать игры для Playstation 2). Вместе взятые, уменьшение продаж и увеличение затрат на разработку, связанные с новыми платформами, могут оказать очень существенное неблагоприятное влияние на рентабельность разработчиков игр.

Новые игровые консоли также являются очень дорогими. Xbox 360, Nintendo Wii и Sony Playstation 3 - все в розницу продаются за сотни долларов. Мощные игровые системы персональных компьютеров могут стоить до 8000 долларов. Это представляет

для пользователей существенное вложение денег, особенно с учетом того, что аппаратное обеспечение устаревает после нескольких лет, и того, что многие системы покупают для детей.

Одним подходом к решению вышеизложенных проблем являются онлайн-игры, в которых программа, управляющая игрой, и данные размещены на сервере и доставляются в клиентские машины по требованию как сжатое видео и аудио, которые передаются потоком по широкополосной цифровой сети. Некоторые компании, например, G-Cluster в Финляндии (теперь дочерняя компания японской корпорации SOFTBANK Broadmedia) в настоящее время оказывают эти услуги в режиме онлайн. Аналогичные игровые услуги стали доступны в локальных сетях, например, в локальных сетях гостиниц, и предлагаются провайдерами DSL и кабельного телевидения. Основным недостатком этих систем является проблема времени ожидания, т.е. время, которое требуется сигналу для прохождения до и от игрового сервера, который обычно расположен в "центральной станции" оператора. Динамичные видеоигры-боевики (также известные как видеоигры "twitch" ("подергивания")) требуют очень малого времени ожидания между временем, когда пользователь выполняет операцию с игровым контроллером, и временем, когда экран дисплея обновляется и показывает результат операции пользователя. Малое время ожидания требуется для того, чтобы у пользователя было ощущение, что игра отвечает "немедленно". Пользователи могут быть удовлетворены разными интервалами времени ожидания, в зависимости от типа игры и уровня мастерства пользователя. Например, время ожидания 100 мс может быть допустимой для медленной бессистемной игры (подобной бакгаммон) или медленной ролевой игры, но в динамичной игре-боевике время ожидания, превышающая 70 или 80 мс, может ухудшать игру пользователя и, соответственно, является неприемлемым. Например, в игре, которая требует быстрого времени реакции, резко ухудшается точность по мере увеличения времени ожидания от 50 до 100 мс.

Когда игровой сервер или сервер приложений установлены в близлежащей, управляемой сетевой среде, или там, где сетевой тракт до пользователя является предсказуемым, и/или он может выдерживать пики полосы пропускания, намного проще управлять временем ожидания, и с точки зрения максимального времени ожидания, и с точки зрения постоянства времени ожидания (например, так, что пользователь наблюдает устойчивое движение от цифрового видео, передаваемого потоком через сеть). Такого уровня управления можно достичь между распределительным устройством сети кабельного телевидения до дома абонента кабельного телевидения, или от центральной АТС DSL до дома абонента DSL, или в среде коммерческой учрежденческой локальной сети (LAN) от сервера или пользователя. Также можно получить специально разделенные частные соединения точка-точка между компаниями, которые имеют гарантированную полосу пропускания и время ожидания. Но в игровой или прикладной системе, которая размещает игры в серверном центре, подключенном к общему Internet, и затем передает потоком сжатое видео пользователю через широкополосное соединение, многие факторы воздействуют на время ожидания, что в результате приводит к серьезным ограничениям в применении систем предшествующего уровня техники.

В типичном доме, подключенном к широкополосному каналу, пользователь может иметь DSL или кабельный модем для услуг широкополосного доступа. Для таких услуг широкополосного доступа время передачи сигнала туда и обратно между домом пользователя и общим Internet обычно составляет 25 мс (а иногда и больше). Кроме того, существуют величины времени ожидания передачи сигнала туда и обратно,

которые являются следствием маршрутизации данных через Internet в серверный центр. Время ожидания передачи через Internet меняется в зависимости от маршрута, который задается данным, и задержек, которые происходят вследствие маршрутизации по нему. В дополнение к задержкам маршрутизации, время передачи сигнала туда и обратно также является следствием скорости света, проходящего через светопровод, который соединяет большую часть Internet. Например, на каждые 1000 миль (1600 км), время передачи сигнала туда и обратно приблизительно равно 22 мс вследствие скорости света через светопровод и других потерь.

Дополнительное время ожидания может произойти из-за скорости передачи данных, передаваемых потоком через Internet. Например, если пользователю предоставляется услуга DSL, которую продают как "услугу DSL 6 мегабит в секунду", то на практике, пользователь, вероятно, получит меньше 5 мегабит в секунду пропускной способности нисходящего потока данных в лучшем случае и, вероятно, будет наблюдать, что соединение периодически ухудшается из-за различных факторов, например, перегрузки во время периодов пиковой нагрузки в мультиплексоре доступа к цифровой абонентской линии (DSLAM). Аналогичный вопрос может возникнуть с уменьшением скорости передачи данных кабельного модема, (используемого) для соединения, проданного как "услуга кабельной модемной связи 6 мегабит в секунду", до скорости, гораздо меньшей, чем указанная, если произойдет перегрузка в локальном общем коаксиальном кабеле, образующем линию связи через окрестности, или в другом месте в сети системы кабельных модемов. Если пакеты данных с устойчивой скоростью 4 Мбит/с будут передаваться потоком в одном направлении в формате протокола передачи дейтаграмм пользователя (UDP) от серверного центра через такие соединения, если все будет работать исправно, то пакеты данных будут передаваться без дополнительного времени ожидания, но в случае перегрузки (или других препятствий), и если потоковым данным к пользователю доступно только 3,5 Мбит/с, то в типичной ситуации или пакеты будут сброшены, что в результате приведет к потере данных, или пакеты будут стоять в очереди в точке перегрузки, пока они не будут отправлены, следовательно, с введением дополнительного времени ожидания. В разных точках перегрузки существует разная вместимость очереди для хранения задерживаемых пакетов, соответственно, в некоторых случаях пакеты, которые не могут пройти через точку перегрузки, немедленно сбрасываются. В других случаях, в очереди стоят несколько мегабитов данных и в итоге отправляются. Но, почти во всех случаях, очереди в точках перегрузки ограничены по вместимости, и после превышения этих ограничений очереди переполняются, и пакеты сбрасываются. Соответственно, для устранения дополнительного времени ожидания (или, что еще хуже, потери пакетов), необходимо устранение превышения возможностей скорости передачи данных из игрового сервера или сервера приложений пользователю.

Время ожидания также является следствием времени, требуемого для сжатия видео в сервере и восстановления сжатого видео в клиентском устройстве. Время ожидания также имеет место, когда в видеоигре, исполняющейся на сервере, вычисляется следующий кадр, который должен быть выведен на экран. Имеющиеся в настоящее время доступные алгоритмы сжатия видео страдают или из-за высоких скоростей передачи данных или из-за большого времени ожидания. Например, алгоритм сжатия движущихся изображений MJPEG является только-внутрикадровым алгоритмом сжатия с потерями, который отличается малым временем ожидания. Каждый кадр видео сжимается независимо от каждого другого кадра видео. Когда клиентское устройство принимает кадр видео, сжатого по алгоритму сжатия движущихся изображений MJPEG, то оно может немедленно восстановить сжатый кадр и вывести его на экран с

получающимся в результате очень малым временем ожидания. Но из-за того, что каждый кадр сжимается отдельно, упомянутый алгоритм не может использовать сходные черты между последовательными кадрами, и в результате только-внутрикадровые алгоритмы сжатия видео страдают от очень высоких скоростей передачи данных. Например, видео 640×480, 60 кадр/с (кадров в секунду), сжатое по алгоритму сжатия движущихся изображений MJPEG, может потребоваться 40 Мбит/с (мегабит в секунду) или большая (скорость передачи) данных. Такие высокие скорости передачи данных для видеоокна с таким низким разрешением требуют чрезмерно больших затрат во многих широкополосных приложениях (и конечно для большинства потребительских приложений, основанных на Internet-технологиях). Кроме того, из-за того, что каждый кадр сжимается независимо, артефакты в кадрах, которые могут получиться в результате сжатия с потерями, вероятно, будут появляться в разных местах в последовательных кадрах. Это может в результате привести к тому, что для зрителя выглядит как движущиеся визуальные артефакты, когда восстанавливается сжатое видео.

Другие алгоритмы сжатия, например, MPEG2, H.264 или VC9 от корпорации "Майкрософт", как они используются в конфигурациях предшествующего уровня техники, могут достигать высоких коэффициентов сжатия, но за счет большого времени ожидания. Такие алгоритмы используют межкадровое, а также внутрикадровое сжатие. Периодически такие алгоритмы выполняют только-внутрикадровое сжатие кадра. Такой кадр известен как ключевой кадр (как правило, называемый I-кадр). После этого эти алгоритмы, как правило, сравнивают I-кадр с предшествующими кадрами и с последующими кадрами. Вместо того, чтобы сжимать предшествующие кадры и последующие кадры независимо, упомянутый алгоритм определяет то, что изменилось в изображении предшествующих и последующих кадров по сравнению с I-кадром, и после этого сохраняет эти изменения как так называемые В-кадры для изменений, предшествующих I-кадру, и Р-кадры, для изменений, следующих за I-кадром. Это в результате приводит к гораздо меньшим скоростям передачи данных, чем только-внутрикадровое сжатие. Но это, как правило, достигается за счет большего времени ожидания. I-кадр обычно намного больше, чем В-кадр или Р-кадр (часто в 10 раз больше), и в результате его передача занимает пропорционально больше времени при данной скорости передачи данных.

Рассмотрим, например, ситуацию, когда I-кадры в 10 раз больше В-кадров и Р-кадров, и существует 29 В-кадров+30 Р-кадров=59 промежуточных кадров для каждого отдельного I-интракадра, или всего 60 кадров для каждой "Группы Кадров" (GOP). Соответственно, при 60 кадр/с, каждую секунду существует 1 60-кадровая GOP. Предположим, что максимальная скорость передачи данных на канале передачи равна 2 Мбит/с. Для получения видео высшего качества в этом канале, алгоритм сжатия может выводить поток данных 2 Мбит/с, и, с учетом вышеупомянутых коэффициентов, в результате этого получается $2 \text{ мегабита (Мбит)}/(59+10)=30 \text{ 394 бита}$ в каждом интракадре и 303 935 битов в каждом I-кадре. Когда алгоритм восстановления сжатых данных принимает сжатый видеопоток, для устойчивого воспроизведения видео, требуется восстанавливать каждый сжатый кадр и выводить их на экран с постоянным интервалом (например, 60 кадр/с). Для получения этого результата, если для какого-либо кадра существует время ожидания передачи, то все кадры должны быть задержаны, по меньшей мере, на время этого времени ожидания, соответственно, самая большая время ожидания кадра определяет время ожидания для каждого видеокadra. I-кадры вводят самые большие величины времени ожидания передачи, так как они являются

наибольшими, и весь I-кадр должен быть принят до того, как может быть восстановлен сжатый I-кадр и выведен на экран (или любой промежуточный кадр, зависящий от I-кадра). С учетом того, что скорость передачи данных по каналу равна 2 Мбит/с, передача I-кадра займет $303\,935/2$ Мбит=145 мс.

- 5 В системе межкадрового сжатия видео, как описано выше, с использованием большого процента полосы пропускания канала передачи существуют большие величины времени ожидания из-за большого размера I-кадра относительно среднего размера кадра. Или, другими словами, несмотря на то, что при алгоритмах межкадрового сжатия предшествующего уровня техники достигается меньшая средняя
- 10 покадровая скорость передачи данных, чем при только-внутрикадровых алгоритмах сжатия (например, 2 Мбит/с по сравнению с 40 Мбит/с), они, тем не менее, страдают от высокой пиковой покадровой скорости передачи данных (например, $303\,935 \times 60 = 18,2$ Мбит/с) из-за больших I-кадров. Однако следует принимать во внимание то, что вышеупомянутый анализ предполагает, что все Р-кадры и В-кадры намного меньше I-кадров.
- 15 Несмотря на то, что это в общем справедливо, это не справедливо для кадров с высоким уровнем сложности изображения, некоррелированных с предшествующим кадром, с большим перемещением или со сменами сцен. В таких ситуациях Р-кадры или В-кадры могут становиться такими большими, как I-кадры (если Р-кадр или В-кадр становится больше I-кадра, то усложненный алгоритм сжатия, как правило,
- 20 "обеспечивает" I-кадр и заменяет Р-кадр или В-кадр I-кадром). Соответственно, в цифровом видеопотоке в любой момент могут иметь место пики скорости передачи данных, имеющие размер I-кадра. Соответственно, в случае со сжатым видео, когда средняя скорость передачи данных видео приближается к пропускной способности скорости передачи данных каналов передачи (как это часто происходит, с учетом
- 25 требований высокой скорости передачи данных для видео), высокие пиковые скорости передачи данных из-за I-кадров или больших Р-кадров или В-кадров в результате приводят к большому времени ожидания кадра.

- Несомненно, в вышеизложенном обсуждении описывается только время ожидания алгоритма сжатия, создаваемое большими В-кадрами, Р-кадрами или I-кадрами в GOP.
- 30 Если используются В-кадры, то время ожидания будет еще больше. Причиной, по которой это происходит, является то, что до того, как В-кадр может быть выведен на экран, все В-кадры, следующие за В-кадром и I-кадром, должны быть приняты. Соответственно, в последовательности группы изображений (GOP), например, BBBB-VIPRRRBBBBIIPRRR, в которой существуют 5 В-кадров перед каждым I-кадром,
- 35 первый В-кадр не может быть выведен на экран устройством восстановления сжатого видео до тех пор, пока не будут приняты последующие В-кадры и I-кадр. Соответственно, если видео передают потоком со скоростью 60 кадр/с (т.е. 16,67 мс/кадр), то до того, как может быть восстановлен сжатый первый В-кадр, потребуется $16,67 \times 6 = 100$ мс для приема пяти В-кадров и I-кадра, независимо от скорости полосы
- 40 пропускания канала, и это только с 5 В-кадрами. Сжатые видеопоследовательности с 30 В-кадрами являются довольно распространенными. И, при небольшой полосе пропускания канала, как 2 Мбит/с, влияние времени ожидания, вызванного размером I-кадра, является большим дополнением к влиянию времени ожидания из-за ожидания поступления В-кадров. Соответственно, на канале 2 Мбит/с, с большим количеством
- 45 В-кадров довольно просто превысить время ожидания 500 мс или больше с использованием технологии сжатия видео предшествующего уровня техники. Если В-кадры не используются (за счет меньшего коэффициента сжатия для данного уровня качества), то В-кадр не вызывает время ожидания, но, по-прежнему, время ожидания

вызывают пиковые размеры кадра, описанные выше.

Проблема усугубляется самим характером многих видеоигр. Алгоритмы сжатия видео, использующие структуру GOP, описанную выше, в значительной степени оптимизированы для использования с видео в прямом эфире или материалом кинофильма, предназначенного для пассивного просмотра. Как правило, камера (либо реальная камера или виртуальная камера в случае создаваемой компьютером анимации) и сцена являются относительно устойчивыми, просто потому что, если камера или сцена перемещаются слишком отрывисто, то материал фильма или видео (а) как правило, не приятно смотреть, и (б) если его смотрят, то обычно зритель не внимательно следит за действием, когда камера внезапно резко поворачивается (например, если камера ударяется при съемке, когда ребенок задувает свечи на торте ко дню рождения, и внезапно резко поворачивается в сторону от торта и возвращается обратно, то зрители, как правило, сосредоточены на ребенке и торте, и не обращают внимания на краткое прерывание, когда камера внезапно перемещается). В случае видеointервью или видеотелеконференции, камера может удерживаться в фиксированном положении и совсем не двигаться, что в результате приводит к совсем очень маленькому количеству пиков данных. Но 3D видеоигра с высокой активностью отличается постоянным движением (например, рассмотрим 3D гонки, где весь кадр находится в быстром движении в течение гонки, или рассмотрим игры-боевики от первого лица, где виртуальная камера постоянно перемещается рывками). Такие видеоигры могут в результате привести к последовательностям кадров с большими и частыми пиками, где пользователю может быть необходимо ясно видеть то, что происходит в течение этих внезапных движений. Как таковой, артефакты сжатия гораздо менее приемлемы в 3D видеоиграх с высокой активностью. Соответственно, видеовыход многих видеоигр, вследствие их характера, выводит сжатый видеопоток с очень высокими и частыми пиками.

С учетом того, что пользователи динамичных видеоигр-боевиков не терпимы к большому времени ожидания, и с учетом всех вышеупомянутых причин времени ожидания, до настоящего времени существовали ограничения для размещенных на сервере видеоигр, которые передавали потоком видео в Internet. Кроме того, пользователи приложений, которые требуют высокой степени интерактивности, страдают от аналогичных ограничений, если эти приложения размещены в общем Internet и передают видео потоком. Такие услуги требуют конфигурации сети, в которой хостинговые серверы устанавливаются непосредственно в распределительном устройстве (в случае кабельного широкополосного канала) или в центральной АТС (в случае цифровых абонентских линий (DSL)), или в пределах LAN (или на специально разделенном частном соединении) в коммерческой обстановке так, что маршрутом и расстоянием от клиентского устройства до сервера управляют для минимизации времени ожидания, и пики могут быть адаптированы так, что они не будут вызывать время ожидания. Сети LAN (как правило, имеющие скорость в пределах 100 Мбит/с - 1 Гбит/с) и арендуемые линии связи с соответствующей полосой пропускания, как правило, могут поддерживать требования по пиковой полосе пропускания (например, пиковая полоса пропускания 18 Мбит/с равна незначительной доле от пропускной способности LAN 100 Мбит/с).

Требования по пиковой полосе пропускания могут также быть обеспечены инфраструктурой широкополосной сети, связанной с жилыми домами, если выполнены специальные адаптации. Например, в системе кабельного телевидения, цифровому видеотрафику может быть предоставлена выделенная полоса пропускания, которая

может обрабатывать пики, например, большие I-кадры. И, в системе DSL, может быть обеспечен модем DSL с более высокой скоростью, с учетом больших пиков, или может быть обеспечено специально разделенное соединение, которое может обрабатывать данные при более высоких скоростях передачи. Но общепринятый кабельный модем и инфраструктура DSL, подсоединенные к общему Internet, имеют гораздо меньший допуск для требований по пиковой полосе пропускания для сжатого видео.

Соответственно, онлайн-сервисы, которые размещают видеоролики или приложения в серверных центрах, находящихся на большом расстоянии от клиентских устройств, и затем передают потоком сжатый видеопоток по Internet через обычные широкополосные соединения, связанные с жилыми домами, страдают от существенного времени ожидания и ограничений по пиковой полосе пропускания - особенно в отношении игр и приложений, которые требуют очень малого времени ожидания, (например, игры-боевики от первого лица и другие многопользовательские, интерактивные игры-боевики или приложения, требующие малого времени отклика).

Краткое описание чертежей

Настоящее раскрытие предмета изобретения станет полностью понятно из нижеследующего подробного описания и из прилагаемых чертежей, которые, однако, не должны восприниматься как ограничивающие раскрываемый предмет изобретения конкретными изображенными вариантами осуществления, а должны восприниматься как предназначенные только для объяснения и понимания.

На фиг.1 изображена архитектура игровой видеосистемы предшествующего уровня техники.

На фиг.2а-фиг.2b изображена архитектура системы высокого уровня согласно одному варианту осуществления.

На фиг.3 изображены фактические, оцененные и требуемые скорости передачи данных для передачи информации между клиентом и сервером.

На фиг.4а изображена служба хостинга и клиент, применяемые согласно одному варианту осуществления.

На фиг.4b изображены иллюстративные величины времени ожидания, связанные с передачей информации между клиентом и службой хостинга.

На фиг.4с изображено клиентское устройство согласно одному варианту осуществления.

На фиг.4d изображено клиентское устройство согласно другому варианту осуществления.

На фиг.4е изображена иллюстративная блок-схема клиентского устройства по фиг.4с.

На фиг.4f изображена иллюстративная блок-схема клиентского устройства по фиг.4d.

На фиг.5 изображена иллюстративная форма сжатия видео, которая может быть применена согласно одному варианту осуществления.

На фиг.6а изображена иллюстративная форма сжатия видео, которая может быть применена в другом варианте осуществления.

На фиг.6b изображены пики скорости передачи данных, связанной с передачей видеопоследовательности с низкой активностью, низким уровнем сложности.

На фиг.6с изображены пики скорости передачи данных, связанной с передачей видеопоследовательности с высокой активностью, высоким уровнем сложности.

На фиг.7а-фиг.7b изображены иллюстративные способы сжатия видео, применяемые в одном варианте осуществления.

На фиг.8 изображены дополнительные иллюстративные способы сжатия видео, применяемые в одном варианте осуществления.

На фиг.9а-фиг.9с изображены иллюстративные способы, применяемые в одном варианте осуществления для уменьшения пиков скорости передачи данных.

На фиг.10а-фиг.10b изображен один вариант осуществления, который эффективно упаковывает фрагменты внутри пакетов.

5 На фиг.11а-фиг.11d изображены варианты осуществления, которые применяют способы прямого исправления ошибок.

На фиг.12 изображен один вариант осуществления, который использует многоядерные процессоры для сжатия.

10 На фиг.13а-фиг.13b изображено географическое положение и связь между службами хостинга согласно различным вариантам осуществления.

На фиг.14 изображены иллюстративные величины времени ожидания, связанные с передачей информации между клиентом и службой хостинга.

На фиг.15 изображена иллюстративная архитектура серверного центра службы хостинга.

15 На фиг.16 изображен иллюстративный моментальный снимок экрана одного варианта осуществления интерфейса пользователя, который включает в себя множество видеоокон в реальном масштабе времени.

На фиг.17 изображен интерфейс пользователя по фиг.16 после выбора конкретного видеоокна.

20 На фиг.18 изображен интерфейс пользователя по фиг.17 после распаивания упомянутого конкретного видеоокна во весь экран.

На фиг.19 изображены иллюстративные совместные пользовательские видеоданные, совмещенные на экране игры с несколькими участниками.

25 На фиг.20 изображена иллюстративная страница пользователя для игрока в службе хостинга.

На фиг.21 изображена иллюстративная 3D интерактивная реклама.

На фиг.22 изображена иллюстративная последовательность этапов для вывода фотореального изображения, имеющего текстурированную поверхность, исходя из захвата поверхности живого исполнения.

30 На фиг.23 изображена иллюстративная страница интерфейса пользователя, которая обеспечивает возможность выбора линейного мультимедийного содержимого.

Фиг.24 - график, на котором изображено время, которое пройдет до того, как web-страница станет активной, по сравнению со скоростью соединения.

35 На фиг.25а-фиг.25b изображены варианты осуществления изобретения, которые используют канал обратной связи из клиентского устройства в службу хостинга.

На фиг.26а-фиг.26b изображен вариант осуществления, в котором фрагменты/кадры кодируются на основе последнего фрагмента/кадра, о котором известно, что он успешно принят.

40 На фиг.27а-фиг.27b изображен вариант осуществления, в котором состояние игры или приложения переносится из первой службы хостинга или сервера во вторую службу хостинга или сервер.

На фиг.28 изображен один вариант осуществления, в котором состояние игры или приложения переносится с использованием разностных данных.

45 На фиг.29 изображен один вариант осуществления изобретения, в котором на клиентском устройстве используется временный декодер.

На фиг.30 изображено то, как "I-фрагмент" вставляется во все "R-кадры" согласно одному варианту осуществления изобретения.

На фиг.31а-фиг.31h изображены варианты осуществления изобретения, в которых

формируется поток в прямом эфире и/или один или несколько HQ-потоков.

Описание иллюстративных вариантов осуществления

В нижеследующем описании изложены конкретные детали, например, типы устройств, конфигурации систем, способы обмена информацией и т.д. для обеспечения полного понимания настоящего раскрытия предмета изобретения. Однако специалистам в соответствующих областях техники, к которым относится данное изобретение, будет понятно, что эти конкретные детали не являются обязательными для применения описанных вариантов осуществления.

На фиг.2а-фиг.2b обеспечена архитектура высокого уровня двух вариантов осуществления, в которых видеоигры и приложения размещаются службой 210 хостинга и к ним получают доступ клиентские устройства 205, находящиеся на территории 211 пользователя (отметим, что "территория пользователя" означает любое место, где находится пользователь, в том числе на открытом воздухе, если он использует мобильное устройство), через Internet 206 (или другую общедоступную или частную сеть) согласно абонентскому обслуживанию. Клиентские устройства 205 могут быть универсальными компьютерами, например, PC Windows Microsoft или PC Linux или компьютерами Macintosh корпорации Apple, Inc. с проводным или беспроводным соединением с Internet, с встроенным или внешним дисплеем 222, или они могут быть специализированными клиентскими устройствами, например, телевизионной абонентской приставкой (с проводным или беспроводным соединением с Internet), которая выводит видео и аудио на монитор или телевизор 222, или они могут быть мобильными устройствами, предположительно с беспроводным соединением с Internet.

Любое из этих устройств может иметь свои собственные устройства ввода пользователя (например, клавиатуры, кнопки, сенсорные экраны, трекпад или световое перо, камеры видеозахвата и/или камеры, отслеживающие движение, и т.д.), или они могут использовать внешние устройства 221 ввода (например, клавиатуры, мыши, игровые контроллеры, световое перо, камеры видеозахвата и/или камеры, отслеживающие движение, и т.д.), подключенные посредством проводов или беспроводным способом. Как описано более подробно ниже, служба 210 хостинга включает в себя серверы различных уровней производительности, в том числе серверы с мощными средствами обработки CPU/GPU. Во время ведения игры или использования приложения в службе 210 хостинга, домашнее или офисное клиентское устройство 205 принимает входные данные контроллера и/или клавиатуры от пользователя, и затем оно отправляет эти входные данные контроллера через Internet 206 в службу 210 хостинга, который в ответ исполняет программу, управляющую игрой, и формирует последовательные кадры видеовыхода (последовательность видеоизображений) для игры или прикладного программного обеспечения (например, если пользователь нажимает на кнопку, которая может перемещать персонажа на экране вправо, тогда игровая программа создает последовательность видеоизображений, на которых изображен персонаж, перемещающийся вправо). Эта последовательность видеоизображений после этого сжимается с использованием устройства сжатия видео с малым временем ожидания, и после этого служба 210 хостинга передает видеопоток с малым временем ожидания через Internet 206. Домашнее или офисное клиентское устройство после этого декодируют сжатый видеопоток и визуализируют восстановленные видеоизображения на мониторе или телевизоре. Следовательно, требования к вычислительной и графической технике клиентского устройства 205 значительно уменьшаются. Требуется, чтобы клиент 205 имел вычислительную мощность только для пересылки входных данных клавиатуры/контроллера в Internet

206 и декодирования и восстановления сжатого видеопотока, принятого из Internet 206, что в настоящее время фактически любой персональный компьютер может выполнять в программном обеспечении на своем центральном процессоре (например, двухъядерный процессор корпорации Intel (Intel Corporation Core Duo CPU)), работающий с тактовой частотой приблизительно 2 ГГц, может восстанавливать сжатое HDTV 1280×720, закодированное с использованием таких устройств сжатия, как H.264 и Windows Media VC9). И, в случае любых клиентских устройств, также специализированные интегральные схемы могут выполнять восстановление сжатого видео для таких стандартов в реальном времени с гораздо меньшими затратами и с гораздо меньшим потреблением мощности, чем универсальный CPU, например, который может потребоваться для современного РС. А именно, для выполнения функции пересылки входных данных контроллера и восстановления сжатого видео, в домашних клиентских устройствах 205 не требуются никакие специализированные графические процессоры (GPU), накопитель на оптических дисках или накопители на жестких дисках, например, игровая видеосистема предшествующего уровня техники, изображенная на фиг.1.

По мере того, как игры и прикладное программное обеспечение становится более сложным и более фотореалистичным, они требуют GPU, CPU более высокой производительности, большего количества RAM и более быстрых накопителей на магнитных дисках большей емкости, и вычислительная мощность службы 210 хостинга может непрерывно модернизироваться, но конечному пользователю не потребуется обновлять домашнюю или офисную клиентскую платформу 205, так как требования к ее обработке будут оставаться постоянными для разрешающей способности дисплея и частоты кадров при данном алгоритме восстановления сжатого видео. Соответственно, аппаратные ограничения и вопросы совместимости, имеющиеся в настоящее время, не существуют в системе, изображенной на фиг.2а-фиг.2b.

Кроме того, так как игра и прикладное программное обеспечение исполняются только в серверах в службе 210 хостинга, не существует копии игры или прикладного программного обеспечения (ни в виде оптического носителя информации, ни как загружаемого программного обеспечения) в офисе ("офис", как используется в этом описании, если не оговорено иное, включает в себя любое нежилое окружение, не связанное с постоянным проживанием, в том числе классные комнаты, например) или в доме пользователя. Это значительно уменьшает вероятность того, что игра или прикладное программное обеспечение будут незаконно скопированы (незаконно использоваться), а также уменьшает вероятность того, что ценная база данных, которая может быть (использована) игрой или прикладным программным обеспечением, будет незаконно использоваться. На самом деле, если для ведения игры требуются специализированные серверы (например, с требованием очень дорогого, большого или шумного оборудования) или прикладное программное обеспечение, которое на практике нельзя использовать дома или в офисе, то, даже если будет получена пиратская копия игры или прикладного программного обеспечения, то ее нельзя будет использовать дома или в офисе.

В одном варианте осуществления, служба 210 хостинга обеспечивает инструментальные средства разработки программного обеспечения разработчикам 220 прикладного программного обеспечения или игры (что, как правило, относится к компаниям по разработке программного обеспечения, игровым или телевизионным студиям или издателям прикладного программного обеспечения или игры), которые проектируют видеоигры, для того, чтобы они могли проектировать игры, которые можно будет исполнять в службе 210 хостинга. Такие инструментальные средства

обеспечивают возможность разработчикам использовать признаки службы хостинга, которые обычно не доступны в автономном РС или игровой консоли (например, быстрый доступ к очень большим базам данных сложной геометрии ("геометрия", если не оговорено иное, используется в этом описании для ссылки на многоугольники, текстуры, снаряжение, освещение, линии поведения и другие компоненты и параметры, которые определяют базы данных 3D)).

Разные бизнес-модели возможны по этой архитектуре. По одной модели, служба 210 хостинга получает абонентскую плату от конечного пользователя и платит лицензионный платеж разработчикам 220, как изображено на фиг.2а. В альтернативной реализации, изображенной на фиг.2b, разработчики 220 получают абонентскую плату непосредственно от пользователя и платят службе 210 хостинг за хостинг содержимого приложения или игры. Эти лежащие в основе принципы не ограничены какой-либо конкретной бизнес-моделью для обеспечения хостинга приложения или онлайн-игр.

Характеристики сжатого видео

Как обсуждалось ранее, одной существенной проблемой, связанной с оказанием услуг по видеоиграм или услуг по прикладному программному обеспечению в режиме онлайн, является проблема времени ожидания. Время ожидания 70-80 мс (от момента, когда устройство ввода приводится в действие пользователем, до момента, когда ответ выводится на экран дисплея) находится на верхней границе для игр и приложений, требующих малого времени отклика. Однако достичь такой задержки очень трудно в контексте архитектуры, изображенной на фиг.2а и фиг.2b, из-за нескольких практических и физических ограничений.

Как указано на фиг.3, когда пользователь подписывается на Internet-услугу, соединение обычно оценивается номинальной максимальной скоростью 301 передачи данных до офиса или дома пользователя. В зависимости от политики провайдеров и характеристик оборудования для маршрутизации, может быть более или менее обеспечено строгое выполнение этой максимальной скорости передачи данных, но, как правило, фактическая доступная скорость передачи данных является более низкой по одной или нескольким разным причинам. Например, в центральной АТС DSL или в локальной линии кабельной модемной связи может находиться слишком много сетевого трафика, или в кабельной сети может присутствовать шум, вызывающий сброс пакетов, или провайдер может установить максимальное количество битов в месяц для каждого пользователя. В настоящее время, максимальная скорость передачи нисходящего потока данных для услуг DSL и кабельной связи, как правило, колеблется от нескольких сотен килобит в секунду (Кбит/с) до 30 Мбит/с. Услуги сотовой связи обычно ограничены до сотен Кбит/с нисходящего потока данных. Однако скорость услуг широкополосного доступа и количество пользователей, которые подписываются на услуги широкополосного доступа, существенно увеличатся со временем. В настоящее время, некоторые аналитики оценивают, что 33% американских абонентов широкополосного доступа имеют скорость передачи нисходящего потока данных 2 Мбит/с или больше. Например, некоторые аналитики предсказывают, что к 2010 г. свыше 85% американских абонентов широкополосного доступа будут иметь скорость передачи данных 2 Мбит/с или больше.

Как указано на фиг.3, фактическая доступная максимальная скорость 302 передачи данных может колебаться с течением времени. Соответственно, в контексте прикладного программного обеспечения или онлайн-игр с малым временем ожидания иногда трудно предсказать фактическую доступную скорость передачи данных для конкретного

видеопотока. Если скорость 303 передачи данных, требуемая для поддержки данного уровня качества при данном количестве кадров в секунду (кадр/с) с данной разрешающей способностью (например, 640×480, 60 кадр/с) для определенной величины сложности сцены и перемещения, поднимается выше фактической доступной максимальной скорости 302 передачи данных (как указано посредством пика на фиг.3), то может возникнуть несколько проблем. Например, в некоторых Internet-услугах будут просто сбрасываться пакеты, что в результате приведет к потере данных и искаженным/потерянным изображениям на видеоэкране пользователя. В других услугах дополнительные пакеты будут временно помещены в буфер (т.е., поставлены в очередь), и эти пакеты будут обеспечены клиенту с имеющейся скоростью передачи данных, что в результате приведет к увеличению времени ожидания - неприемлемый результат для многих видеоигр и приложений. Наконец, некоторые провайдеры Internet-услуг будут рассматривать увеличение скорости передачи данных как злонамеренную атаку, например, как атаку типа отказ в обслуживании (известный способ, (используемый) хакерами для блокирования сетевых соединений), и прервут соединение пользователя с Internet на определенный период времени. Соответственно, в вариантах осуществления, описанных в этом документе, предпринимаются шаги для обеспечения того, что скорость передачи данных, требуемая для видеоигры, не будет превышать максимальную имеющуюся скорость передачи данных.

Архитектура службы хостинга

На фиг.4а изображена архитектура 210 службы хостинга согласно одному варианту осуществления. Служба 210 хостинга или может быть расположена в одном серверном центре, или может быть распределена по всему множеству серверных центров (для обеспечения соединений с пользователями с меньшим временем ожидания, которые имеют маршруты с меньшим временем ожидания до определенных серверных центров, чем другие, для обеспечения выравнивания нагрузки среди пользователей и обеспечения избыточности на случай отказа одного или нескольких серверных центров). Служба 210 хостинга может со временем включать в себя сотни тысяч или даже миллионов серверов 402, обслуживающих очень большое количество пользователей. Система 401 управления службы хостинга обеспечивает централизованное управление для службы 210 хостинга и управляет маршрутизаторами, серверами, системами сжатия видео, системами учета и формирования счетов и т.д. В одном варианте осуществления, система 401 управления службы хостинга реализована в распределенной системе обработки данных на основе операционной системы Linux, соединенной с дисковыми массивами типа RAID, используемыми для хранения баз данных для пользовательской информации, информации для сервера и статистических данных системы. В вышеизложенных описаниях, различные операции, реализованные службой 210 хостинга, если они не отнесены к другим специальным системам, иницируются и управляются системой 401 управления службы хостинга.

Служба 210 хостинга включает в себя несколько серверов 402, например, таких серверов, которые в настоящее время имеются в продаже, от Intel, IBM, Hewlett Packard, и других. В качестве альтернативы, серверы 402 могут быть собраны в заказной конфигурации из компонентов, или, в конце концов, они могут быть интегрированы так, что весь сервер будет реализован как однокристалльная интегральная схема. Несмотря на то, что на этой схеме для примера изображено небольшое количество серверов 402, при фактическом применении может быть только один сервер 402 или миллионы серверов 402 или больше. Все серверы 402 могут быть сконфигурированы одинаково (в качестве примера некоторых параметров конфигурации, с одинаковыми

производительностью и типом CPU, с или без GPU, и если с GPU, то с одинаковыми производительностью и типом GPU, с одинаковым количеством CPU и GPU, с одинаковыми объемом и типом/скоростью RAM и с одинаковой конфигурацией RAM), или различные подмножества серверов 402 могут иметь одинаковую конфигурацию (например, 25% серверов могут быть выполнены определенным способом, 50% другим способом и 25% еще одним способом), или все серверы 402 могут быть разными.

В одном варианте осуществления серверы 402 без диска, т.е. вместо того, чтобы иметь свое собственное локальное массовое запоминающее устройство (независимо от того, является ли оно оптическим или магнитным запоминающим устройством, или запоминающим устройством на полупроводниках, например флэш-памятью, или другим средством массовой памяти, выполняющим аналогичную функцию), каждый сервер имеет доступ к общей массовой памяти через быстродействующую магистраль или ускоренное сетевое соединение. В одном варианте осуществления этим высокоскоростным соединением является сеть 403 хранения данных (SAN), соединенная с набором матриц 405 независимых дисковых накопителей с избыточностью (RAID) с соединениями между устройствами, реализованными с использованием технологии Gigabit Ethernet. Как известно специалистам в данной области техники, SAN 403 можно использовать для объединения множества дисковых массивов 405 типа RAID вместе, что в результате приводит к чрезвычайно широкой полосе пропускания -

приближающейся или, возможно, превышающей полосу пропускания, доступную из RAM, используемой в современных игровых консолях и PC. И, в то время как дисковые массивы типа RAID на основе вращающихся носителей информации, например, магнитных носителей информации, часто имеют существенное время ожидания доступа во время поиска, дисковые массивы типа RAID на основе полупроводниковой памяти могут быть реализованы с гораздо меньшим временем ожидания доступа. В другой конфигурации, некоторые или все серверы 402 предоставляют некоторые или все свои собственные массовые запоминающие устройства локально. Например, на сервере 402 может храниться информация, к которой часто обращаются, например ее операционная система и копия видеоигры или приложения, на локальном запоминающем устройстве на основе флэш-памяти с малым временем ожидания, но он может использовать SAN для доступа к дисковым массивам 405 типа RAID, на основе вращающихся носителей информации с большим временем поиска, для получения доступа к большим базам данных геометрии или информации о состоянии игры с меньшей частотой.

Кроме того, в одном варианте осуществления, служба 210 хостинга применяет логику 404 сжатия видео с малым временем ожидания, описанную подробно ниже. Логика 404 сжатия видео может быть реализована в программном обеспечении, аппаратном обеспечении или их любой комбинации (определенные варианты осуществления которой описаны ниже). Логика 404 сжатия видео включает в себя логику для сжатия аудио и визуального материала.

При операции во время ведения видеоигры или использования приложения на территории 211 пользователя посредством клавиатуры, мыши, игрового контроллера или другого устройства 421 ввода, логика 413 управляющего сигнала на клиенте 415 передает управляющие сигналы 406a-406b (как правило, в виде пакетов UDP), представляющие нажатия клавиши (и другие виды вводов пользователя), приводимые в действие пользователем, в службу 210 хостинга 210. Управляющие сигналы от данного пользователя маршрутизируются в соответствующий сервер 402 (или серверы, если на устройство ввода пользователя реагируют множество серверов). Как изображено на фиг.4а, управляющие сигналы 406a могут маршрутизироваться в серверы 402 через

SAN. В качестве альтернативы или в дополнение, управляющие сигналы 406b могут маршрутизироваться непосредственно в серверы 402 по сети службы хостинга (например, по локальной сети Ethernet). Независимо от того, как они были переданы, сервер или серверы исполняют игру или прикладное программное обеспечение в ответ на управляющие сигналы 406a-406b. Несмотря на то, что на фиг.4a не изображены различные сетевые компоненты, например, брандмауэр(ы) и/или шлюз(ы), они могут обрабатывать входящий и исходящий трафик на границе службы 210 хостинга (например, между службой 210 хостинга и Internet 410) и/или на границе территории 211 пользователя между Internet 410 и домашним или офисным клиентом 415. Графический выход и аудиовыход исполняемой игры или прикладного программного обеспечения - т.е. новые последовательности видеоизображений - обеспечиваются в логику 404 сжатия видео с малым временем ожидания, которая сжимает последовательности видеоизображений согласно способам сжатия видео с малым временем ожидания, например, согласно способам, описанным в этом документе, и передает сжатый видеопоток, обычно со сжатым или несжатым аудио, назад в клиента 415 через Internet 410 (или, как описано ниже, через оптимизированную высокоскоростную сетевую службу, которая предаёт в обход общего Internet). После этого логика 412 восстановления сжатого видео с малым временем ожидания в клиенте 415 восстанавливает сжатые видео- и аудиопотоки и визуализирует восстановленный видеопоток, и, как правило, воспроизводит восстановленный аудиопоток, на дисплее 422. В качестве альтернативы, аудио может быть воспроизведено в динамиках, отделенных от дисплея 422, или совсем не воспроизводиться. Отметим, что, несмотря на то, что устройство 421 ввода и дисплей 422 изображены на фиг.2a и фиг.2b как автономные устройства, они могут быть интегрированы внутри клиентских устройств, например, портативных компьютеров или мобильных устройств.

Домашний или офисный клиент 415 (описанный ранее на фиг.2a и фиг.2b как домашний или офисный клиент 205) могут быть очень экономичным устройством с малым энергопотреблением, с очень ограниченной вычислительной производительностью или производительностью графической подсистемы, и может иметь очень ограниченную локальную массовую память, или она совсем может отсутствовать. Напротив, каждый сервер 402, соединенный с SAN 403 и множеством RAID 405, может быть исключительно высокопроизводительной вычислительной системой, и, безусловно, если совместно используются множество серверов в конфигурации параллельной обработки, то почти не существует ограничений по вычислительной производительности системы и производительности графической подсистемы, которые могут быть использованы. И, из-за сжатия 404 видео с малым временем ожидания и сжатия 412 видео с малым временем ожидания, у пользователя создается впечатление, что упомянутая вычислительная мощность серверов 402 предоставляются пользователю. Когда пользователь нажимает клавишу на устройстве 421 ввода, изображение на дисплее 422 обновляется в ответ на нажатие этой клавиши без значимой задержки с точки зрения восприятия так, как в случае локального исполнения игры или прикладного программного обеспечения. Соответственно, с домашним или офисным клиентом 415, который является компьютером с очень низкой производительностью или только экономичной интегральной схемой, в которой реализовано восстановление сжатого видео с малым временем ожидания и логика 413 управляющего сигнала, пользователь обеспечен фактически произвольно выбираемой вычислительной мощностью из удаленного местоположения, причем создается впечатление, что они находятся локально. Это дает пользователям возможность

запускать наиболее усовершенствованные, требующие интенсивной работы процессора (обычно новые) видеоигры и самые высокопроизводительные приложения.

На фиг.4с изображено очень элементарное и экономичное домашнее или офисное клиентское устройство 465. Это устройство является вариантом осуществления домашнего или офисного клиента 415 по фиг.4а и фиг.4b. Длина его приблизительно равна 5 сантиметрам. У него есть гнездо 462 Ethernet, которое обеспечивает интерфейс с кабелем Ethernet с питанием через Ethernet (PoE), через которое оно получает питание и получает соединение с Internet. Оно может исполнять преобразование сетевых адресов (NAT) внутри сети, которая поддерживает NAT. В офисной обстановке многие новые коммутаторы Ethernet имеют PoE и доводят PoE непосредственно до гнезда Ethernet в офисе. (В) таком случае, все, что требуется, - это кабель Ethernet от настенного гнезда до клиента 465. Если имеющееся соединение Ethernet не переносит энергию (например, в доме с DSL или кабельным модемом, но без PoE), то в продаже имеются экономичные настенные "модули" (т.е. источники питания), которые принимают кабель Ethernet без питания и выводят Ethernet с PoE.

Клиент 465 содержит логику 413 управляющего сигнала (по фиг.4а), которая соединена с беспроводным интерфейсом Bluetooth, который обеспечивает интерфейс с устройствами 479 ввода Bluetooth, например, клавиатурой, мышью, игровым контроллером и/или микрофоном и/или наушниками. Кроме того, один вариант осуществления клиента 465 может выводить видео со скоростью 120 кадр/с, при соединении с дисплеем 468 может поддерживать видео 120 кадр/с и передавать сигналы (как правило, через инфракрасное излучение) в затворные очки 466 для попеременного закрытия то одного глаза, то другого с каждым последовательным кадром. Результатом, воспринимаемым пользователем, является стереоскопическое 3D изображение, которое "выскакивает" из экрана дисплея. Одним таким дисплеем 468, который поддерживает такую операцию, является Samsung HL-T5076S. Так как видеопоток для каждого глаза является отдельным, в одном варианте осуществления два независимых видеопотока сжимаются службой 210 хостинга, кадры чередуются во времени, и сжатые кадры восстанавливаются как два независимых процесса восстановления сжатых данных внутри клиента 465.

Клиент 465 также содержит логику 412 восстановления сжатого видео с малым временем ожидания, которая восстанавливает сжатое входящее видео и аудио и выводит через HDMI (мультимедийный интерфейс высокой четкости), соединительный кабель 463, который подключают к SDTV (телевидение стандартной четкости) или HDTV (телевидение высокой четкости) 468, в случае, если телевизор с видео и аудио, или к монитору 468, который поддерживает HDMI. Если монитор 468 пользователя не поддерживает HDMI, то может быть использован HDMI-DVI (цифровой видеоинтерфейс), но аудио будет утрачено. Согласно стандарту HDMI, характеристики 464 дисплея (например, поддерживаемые разрешающие способности, частоты кадров) передаются из дисплея 468, и эта информация затем возвращается через подключение 462 к Internet обратно в службу 210 хостинга, поэтому она может передавать потоком сжатое видео в формате, подходящем для дисплея.

На фиг.4d изображено домашнее или офисное клиентское устройство 475, которое является идентичным домашнему или офисному клиентскому устройству 465, изображенному на фиг.4с, за исключением того, что у него больше внешних интерфейсов. Кроме того, клиент 475 может принимать любой PoE для питания, или он может работать от внешнего адаптера источника питания (не изображен), который вставляется в контактное гнездо на стене. С использованием входа USB клиента 475, видеокамера

477 обеспечивает сжатое видео в клиент 475, которое выгружается клиентом 475 в службу 210 хостинга для использования, описанного ниже. Устройство сжатия с малым временем ожидания, использующее способы сжатия, описанные ниже, является встроенным в камеру 477.

5 В дополнение к разъему Ethernet для подключения к Internet клиент 475 также имеет беспроводной интерфейс 802.11g с Internet. Оба интерфейса могут использовать NAT внутри сети, которая поддерживает NAT.

Кроме того, в дополнение к разъему HDMI для вывода видео и аудио, клиент 475 также имеет разъем Dual Link DVI-I (двухканальный DVI-I), который включает в себя
10 аналоговый выход (и с эталонным кабелем адаптера обеспечивает выход VGA). Он также имеет аналоговые выходы для композитного видео и S-видео.

Для аудио, клиент 475 имеет левое/правое гнезда RCA аналогового стерео, и для вывода цифрового аудио он имеет выход TOSLINK (оптический выход).

В дополнение к беспроводному интерфейсу Bluetooth с устройствами 479 ввода, он
15 также имеет гнезда USB для обеспечения интерфейса с устройствами ввода.

На фиг.4е изображен один вариант осуществления внутренней архитектуры клиента 465. Или все или некоторые из устройств, изображенные на схеме, могут быть реализованы в программируемой пользователем логической матрице, заказной схеме ASIC или в нескольких отдельных устройствах, или заказных или имеющихся в готовом
20 виде.

Ethernet с PoE 497 подсоединяется к интерфейсу 481 Ethernet. Питание 499 подается из Ethernet с PoE 497 и соединяется с остальными устройствами в клиенте 465. Шина 480 является общей шиной для обмена информацией между устройствами.

Управляющий CPU 483 (почти любой небольшой CPU, например, CPU серии R4000
25 с быстродействием миллион команд в секунду (MIPS) и тактовой частотой 100 МГц со встроенной RAM отвечает требованиям), исполняющий небольшое клиентское управляющее приложение из флэш-памяти 476, обеспечивает выполнение стека протоколов для сети (т.е. интерфейс Ethernet), и также осуществляет связь со службой 210 хостинга, и конфигурирует все устройства в клиенте 465. Он также управляет
30 интерфейсами с устройствами 469 ввода и отправляет пакеты обратно в службу 210 хостинга с пользовательскими данными контроллера, защищенными посредством прямого исправления ошибок, в случае необходимости. Кроме того, управляющий CPU 483 осуществляет текущий контроль пакетного трафика (например, если пакеты утеряны или задержаны, а также делает отметку времени их поступления). Эта
35 информация отправляется обратно в службу 210 хостинга так, чтобы она могла постоянно осуществить текущий контроль сетевого соединения и корректировать то, что она отправляет, соответственно. Во флэш-память 476 первоначально во время изготовления загружают с управляющую программу для управляющего CPU 483, а также серийный номер, который является уникальным для конкретной клиентской 465
40 единицы оборудования. Этот серийный номер обеспечивает возможность службе 210 хостинга однозначно идентифицировать клиентскую 465 единицу оборудования.

Интерфейс 484 Bluetooth осуществляет связь с устройствами 469 ввода беспроводным способом через свою антенну, находящуюся внутри клиента 465.

Устройство 486 восстановления сжатого видео является устройством восстановления
45 сжатого видео с малым временем ожидания, сконфигурированным для реализации восстановления сжатого видео, описанного в этом документе. Существует большое количество устройств восстановления сжатого видео, или имеющихся в готовом виде, или в виде интеллектуальной собственности (IP) конструкции, которая может быть

интегрирована в FPGA или заказную схему ASIC. Одной компанией, предлагающей IP для декодера H.264, является Ocean Logic, Manly, NSW Australia. Преимуществом использования IP является то, что способы сжатия, используемые в этом описании, не соответствуют стандартам сжатия. Некоторые стандартные устройства восстановления сжатых данных являются достаточно гибкими для обеспечения способов сжатия, описанных в этом документе, но некоторые не могут (их обеспечивать). Но, посредством IP, существует полная гибкость для перепроектирования устройства восстановления сжатых данных так, как требуется.

Выход устройства восстановления сжатого видео соединен с подсистемой 487 вывода видео, которая соединяет видео с видеовыходом интерфейса 490 HDMI.

Подсистема 488 восстановления сжатого аудио реализуется или с использованием стандартного устройства восстановления сжатого аудио, которое имеется в продаже, или она может быть реализована как IP, или восстановление сжатого аудио может быть реализовано внутри управляющего процессора 483, который может, например, реализовывать устройство восстановления сжатого аудио Vorbis.

Устройство, которое осуществляет восстановление сжатого аудио, соединено с подсистемой 489 вывода аудио, которая соединяет аудио с аудиовыходом интерфейса 490 HDMI.

На фиг.4f изображен один вариант осуществления внутренней архитектуры клиента 475. Как можно видеть, эта архитектура является идентичной архитектуре клиента 465 за исключением дополнительных интерфейсов и необязательного внешнего питания DC от адаптера источника питания, который вставляют в контактное гнездо на стене, и если так используется, заменяет питание, которое может поступать из PoE 497 Ethernet. Функциональные возможности, которые присутствуют у клиента 465 не будут повторяться ниже, но дополнительные функциональные возможности описываются следующим образом.

CPU 483 осуществляет связь с дополнительными устройствами и конфигурирует их.

Подсистема 482 WiFi обеспечивает беспроводной доступ к Internet, в качестве альтернативы, к Ethernet 497 через свою антенну. В продаже имеются подсистемы WiFi от широкого круга производителей, включающих в себя Atheros Communications, Santa Clara, CA (Санта-Клара, Калифорния).

Подсистема 485 USB обеспечивает альтернативный вариант по отношению к связи Bluetooth для проводных устройств 479 ввода USB. Подсистемы USB являются довольно стандартными и общедоступными для матриц FPGA и схем ASIC, а также они часто являются встроенными в имеющиеся в готовом виде устройства, выполняющие другие функции, подобные восстановлению сжатого видео.

Подсистема 487 вывода видео выводит более широкий диапазон видеовыходов, чем внутри клиента 465. В дополнение к обеспечению видеовыхода HDMI 490, она обеспечивает DVI-I 491, S-видео 492 и композитное видео 493. Кроме того, когда для цифрового видео используется интерфейс DVI-I 491, характеристики 464 дисплея передаются обратно из дисплея в управляющий CPU 483 для того, чтобы он мог уведомлять службу 210 хостинга о характеристиках дисплея 478. Все интерфейсы, обеспечиваемые подсистемой 487 вывода видео, являются довольно стандартными интерфейсами и общедоступными во многих видах.

Подсистема 489 вывода аудио выводит аудио в цифровой форме через цифровой интерфейс 494 (S/PDIF и/или TOSLINK) и аудио в аналоговой форме через стереофонический аналоговый интерфейс 495.

Анализ времени передачи сигнала туда и обратно

Несомненно, для реализации преимуществ, изложенных в предыдущем разделе, время передачи сигнала туда и обратно между действием пользователя с использованием устройства 421 ввода и появлением результата этого действия на дисплее 420, не должно превышать 70-80 мс. Это время ожидания должно учитывать все факторы на пути от устройства 421 ввода на территории 211 пользователя до службы 210 хостинга и обратно до территории 211 пользователя, до дисплея 422. На фиг.4b изображены различные компоненты и сети, через которые должны проходить сигналы, и выше этих компонентов и сетей находится временная шкала, на которой выведены в упорядоченном виде иллюстративные величины времени ожидания, которые можно ожидать в практической реализации. Отметим, что фиг.4b является упрощенной, и изображена только маршрутизация по критическому пути. Другая маршрутизация данных, используемая для других признаков системы, описана ниже. Двухнаправленные стрелки (например, стрелка 453) указывают время передачи сигнала туда и обратно, и однонаправленные стрелки (например, стрелка 457) указывают время передачи сигнала в одну сторону, и "~" обозначают приблизительное измерение. Следует указать, что будут реальные ситуации, в которых нельзя будет получить перечисленные величины времени ожидания, но в большинстве случаев в США, с использованием соединений посредством кабельного модема и DSL с территорией 211 пользователя, эти величины времени ожидания могут быть получены в случаях, описанных в следующем разделе. Также отметим, что, несмотря на то, что сотовая беспроводная связь с Internet безусловно будет функционировать в изображенной системе, в большинстве современных американских сотовых систем передачи данных (например EVDO) существуют очень большие величины времени ожидания, и не могут быть получены величины времени ожидания, изображенные на фиг.4b. Однако эти лежащие в основе принципы могут быть реализованы в будущих сотовых технологиях, в которых может быть реализован этот уровень времени ожидания.

Начиная с устройства 421 ввода на территории 211 пользователя, после того, как пользователь приводит в действие устройство 421 ввода, управляющий сигнал пользователя отправляется в клиент 415 (который может быть автономным устройством, например, телевизионной абонентской приставкой, или он может быть программным обеспечением или аппаратным обеспечением, функционирующими в другом устройстве, например, PC или мобильном устройстве), и разбивается на пакеты (в формате UDP в одном варианте осуществления), и пакету дается адрес назначения для передачи в службу 210 хостинга. Пакет будет также содержать информацию для указания того, от какого пользователя поступают управляющие сигналы. После этого пакет(ы) управляющего сигнала пересылается(ются) через устройство 443 брандмауэра/маршрутизатора/NAT (преобразование сетевых адресов) в интерфейс 442 WAN. Интерфейс 442 WAN является интерфейсным устройством, обеспечиваемым для территории 211 пользователя провайдером ISP пользователя (провайдер Internet-услуг). Интерфейс 442 WAN может быть кабелем или модемом DSL, приемопередатчиком WiMax, приемопередатчиком для волоконно-оптической линии связи, интерфейсом сотовой системы передачи данных, интерфейсом передачи данных по протоколу IP через электросеть (Internet Protocol-over-powerline) или любым другим из множества интерфейсов с Internet. Кроме того, устройство 443 брандмауэра/маршрутизатора/NAT (и, возможно, интерфейс 442 WAN) может быть интегрировано в клиента 415. Примером этого может быть мобильный телефон, который включает в себя программное обеспечение для реализации функциональных возможностей домашнего или офисного клиента 415, а также средство для маршрутизации и подключения к Internet

беспроводным способом через некоторый стандарт (например, 802.11g).

После этого интерфейс 442 WAN маршрутизирует управляющие сигналы в то, что называется в этом описании "точка входа в Internet" 441 для провайдера Internet-услуг (ISP) пользователя, которая является средством, которое обеспечивает интерфейс между 5 средством передачи данных WAN, соединенным с территорией 211 пользователя, и общим Internet или частной сетью. Характеристики точки входа в Internet изменяются в зависимости от характера обеспечиваемой Internet-услуги. Для DSL это, как правило, будет центральная АТС телефонной компании, где расположен DSLAM. Для кабельных модемов это, как правило, будет центральная станция мультисистемного оператора 10 (MSO) кабельной линии. Для систем сотовой связи это, как правило, будет диспетчерская, связанная с антенной мачтой сотовой связи. Но какова бы ни была природа точки входа в Internet, она далее маршрутизирует пакет(ы) управляющего сигнала в общий Internet 410. Пакет(ы) управляющего сигнала далее маршрутизируются в интерфейс 441 WAN со службой 210 хостинга, через то, что наиболее вероятно будет интерфейсом 15 приемопередатчика для волоконно-оптической линии связи. WAN 441 далее маршрутизирует пакеты управляющего сигнала в логику 409 маршрутизации (которая может быть реализована многими различными способами, включающими в себя коммутатор Ethernet и серверы маршрутизации), которая определяет адрес пользователя и маршрутизирует управляющий(ие) сигнал(ы) в соответствующий сервер 402 для 20 данного пользователя.

После этого сервер 402 принимает управляющие сигналы как входные сигналы для игры или прикладного программного обеспечения, которые исполняются на сервере 402, и использует их для обработки следующего кадра игры или приложения. После формирования следующего кадра, видео и аудио выводятся из сервера 402 в устройство 25 404 сжатия видео. Видео и аудио могут быть выведены из сервера 402 в устройство 404 сжатия посредством различных средств. Во-первых, устройство 404 сжатия может быть встроено в сервер 402, соответственно, сжатие может быть реализовано локально внутри сервера 402. Или видео и/или аудио могут выводиться в виде пакетов через сетевое соединение, например соединение Ethernet, в сеть, которая является любой 30 частной сетью между сервером 402 и устройством 404 сжатия видео, или через совместно использованную сеть, например, SAN 403. Или видео может выводиться через разъем видеовыхода из сервера 402, например, разъем VGA или DVI, и затем захватываться устройством сжатия 404 видео. Кроме того, аудио может выводиться из сервера 402 или как цифровое аудио (например, через разъем S/PDIF или TOSLINK) или как 35 аналоговое аудио, который оцифрован и закодирован логикой сжатия аудио внутри устройства 404 сжатия видео.

После захвата устройством 404 сжатия видео видеокadra и аудио, формируемого в течение этого периода кадра, из сервера 402, устройство сжатия видео сжимает видео и аудио с использованием способов, описанных ниже. После сжатия видео и аудио, 40 формируются пакеты с адресом для отправки обратно в клиент 415 пользователя, и их маршрутизируют в интерфейс 441 WAN, который далее маршрутизирует видео и аудио пакеты через общий Internet 410, который далее маршрутизирует видео и аудио пакеты в точку 441 входа в Internet ISP пользователя, которая маршрутизирует видео и аудио пакеты в интерфейс 442 WAN на территории пользователя, который маршрутизирует 45 видео и аудио пакеты в устройство 443 брандмауэра/маршрутизатора/NAT, которое далее маршрутизирует видео и аудио пакеты в клиента 415.

Клиент 415 восстанавливает сжатые видео и аудио и после этого выводит видео на экран дисплея 422 (или на встроенный дисплей клиента) и отправляет аудио на дисплей

422, или в отдельные усилитель/динамики, или в усилитель/динамики, встроенные в клиента.

Для того, чтобы пользователь воспринимал весь только описанный процесс без запаздывания, двухсторонняя задержка должна быть меньше 70 или 80 мс. Некоторые из задержек времени ожидания в описанном пути туда и обратно контролируются службой 210 хостинга и/или пользователем, а другие не контролируются. Однако следующие измерения, основанные на анализе и тестировании большого количества реальных сценариев, являются приблизительными.

Время передачи в одном направлении для отправки управляющих сигналов 451 обычно меньше 1 мс, маршрутизация туда и обратно через территорию 452 пользователя обычно выполняется с использованием общедоступных коммутаторов потребительского уровня брандмауэра/маршрутизатора/NAT по локальной сети Ethernet приблизительно за 1 мс. I (Соединения) пользователь ISP значительно отличаются по их задержкам 453 передачи сигнала туда и обратно, но с провайдерами кабельной модемной связи и DSL, как правило, наблюдается (задержка) между 10 и 25 мс. Время передачи сигнала туда и обратно по общему Internet 410 может очень отличаться в зависимости от того, как маршрутизируется трафик, и существуют ли какие-нибудь отказы на маршруте (и эти вопросы обсуждаются ниже), но, как правило, общий Internet обеспечивает довольно оптимальные маршруты, и время ожидания в значительной степени определяется скоростью света через светопровод с учетом расстояния до пункта назначения. Как также обсуждается ниже, авторы установили службу 210 хостинга самое большее на расстоянии примерно 1000 миль (1600 км), на котором, как ожидается, она будет располагаться от территории 211 пользователя. Фактическое время передачи сигнала через Internet на (расстояние) 1000 миль (1600 км) (2000 миль (3200 км) туда и обратно) приблизительно равно 22 мс. Интерфейс 441 WAN со службой 210 хостинга является обычно интерфейсом волоконно-оптической линии связи с высокой скоростью коммерческого уровня с незначительным временем ожидания. Соответственно, время 454 ожидания общего Internet обычно находится между 1 и 10 мс. Может быть достигнуто время ожидания маршрутизации 455 в одном направлении через службу 210 хостинга меньше 1 мс. Сервер 402, как правило, вычисляет новый кадр для игры или приложения за время меньшее, чем один период кадра (который при 60 кадр/с равен 16,7 мс), соответственно, 16 мс является приемлемым для использования максимальным временем 456 передачи сигнала в одну сторону. В оптимизированной аппаратной реализации алгоритмов сжатия аудио и сжатия видео, описанных в этом документе, сжатие 457 может быть выполнено за 1 мс. В менее оптимизированных версиях, сжатие может занимать 6 мс (несомненно, еще менее оптимизированные версии могут занимать больше времени, но такие реализации могут влиять на общее время ожидания передачи туда и обратно, и может потребоваться, чтобы другие величины времени ожидания были меньше (например, может быть уменьшено допустимое расстояние через общий Internet) для поддержания целевого времени ожидания 70-80 мс). Величины времени передачи сигнала туда и обратно для Internet 454, (соединения) 453 пользователь ISP, и маршрутизации 452 по территории пользователя уже были рассмотрены, поэтому остается рассмотреть время ожидания восстановления 458 сжатого видео, которое, в зависимости от того, реализовано ли восстановление 458 сжатого видео в специализированном аппаратном обеспечении, или реализовано ли оно в программном обеспечении на клиентском устройстве 415 (например, PC или мобильном устройстве), может изменяться в зависимости от размера дисплея и производительности CPU для восстановления сжатых данных. Как правило, восстановление 458 сжатых данных

занимает между 1 и 8 мс.

Соответственно, посредством суммирования всех величин времени ожидания в самом плохом случае, наблюдаемом на практике, можно определить время передачи сигнала туда и обратно в самом плохом случае, которое, как можно ожидать, будет испытывать пользователь системы, изображенной на фиг.4а. Они равны: $1+1+25+22+1+16+6+8=80$ мс. И, безусловно, на практике (с оговорками, обсуждаемыми ниже), это примерно равно времени передачи сигнала туда и обратно, наблюдаемому при использовании версии пилотной модели, изображенной на фиг.4а с использованием имеющихся в готовом виде PC Windows в качестве клиентских устройств и домашних соединений кабельной модемной связи и DSL в пределах США. Несомненно, сценарии, которые лучше самого плохого случая, могут в результате привести к гораздо меньшим величинам времени ожидания, но на них нельзя основываться при разработке коммерческой услуги, которая широко используется.

Для получения величин времени ожидания, выведенных в упорядоченном виде на фиг.4b, (при передаче сигнала) через общий Internet, требуется, чтобы устройство 404 сжатия видео и устройство 412 восстановления сжатого видео по фиг.4а в клиенте 415 формировало поток пакетов (с) очень конкретными характеристиками, например, последовательность пакетов, формируемая на всем пути от службы 210 хостинга до дисплея 422, не подвергалась задержкам или чрезмерной потере пакетов и, в частности, постоянно удовлетворяла ограничениям полосы пропускания, доступной пользователю по подключению к Internet пользователя через интерфейс 442 WAN и брандмауэр/маршрутизатор/NAT 443. Кроме того, устройство сжатия видео должно создавать поток пакетов, который достаточно устойчив, чтобы он мог допускать неизбежную потерю пакетов и изменение порядка следования пакетов, которое происходит в обычном Internet и сетевых передачах.

Сжатие видео с малым временем ожидания

Для достижения вышеизложенных целей, в одном варианте осуществления используется новый подход к сжатию видео, посредством которого уменьшают время ожидания и требования по пиковой полосе пропускания для передачи видео. До описания этих вариантов осуществления будет обеспечен анализ современных способов сжатия видео согласно фиг.5 и фиг.6а-фиг.6b. Несомненно, эти способы могут быть применены согласно лежащим в основе принципам, если пользователь будет обеспечен полосой пропускания, достаточной для обработки данных со скоростью передачи, требуемой этими способами. Отметим, что в этом описании не рассматривается сжатие аудио, за исключением констатации того, что оно реализуется одновременно и синхронно со сжатием видео. Существуют способы сжатия аудио предшествующего уровня техники, которые удовлетворяют требованиям для этой системы.

На фиг.5 изображен один конкретный способ предшествующего уровня техники для сжатия видео, в котором каждый отдельный видеокادر 501-503 сжимается посредством логики 520 сжатия с использованием конкретного алгоритма сжатия для формирования последовательности сжатых кадров 511-513. Одним вариантом осуществления этого способа является "алгоритм сжатия движущихся изображений MJPEG", в котором каждый кадр сжимается согласно алгоритму сжатия неподвижного изображения, разработанный Объединенной группой экспертов по машинной обработке фотографических изображений (JPEG) на основе дискретного косинусного преобразования (DCT). Однако могут быть применены различные другие типы алгоритмов сжатия, хотя, по-прежнему, с выполнением этих лежащих в основе принципов (например, алгоритмы сжатия, основанные на вейвлете, например, JPEG-

2000).

Одна проблема, связанная с этим типом сжатия, состоит в том, что оно уменьшает скорость передачи данных каждого кадра, но оно не использует сходные черты между последовательными кадрами для уменьшения скорости передачи данных всего видеопотока. Например, как изображено на фиг.5, предположим, что частота кадров $640 \times 480 \times 24 \text{ бит/пиксел} = 640 \times 480 \times 24 / 8 / 1024 = 900 \text{ Кбайт/кадр}$ (Кбайт/кадр), для данного качества изображения, посредством алгоритма сжатия движущихся изображений MJPEG можно сжать поток с коэффициентом 10, что в результате приводит к потоку данных 90 Кбайт/кадр. При 60 кадров/сек потребовалась бы полоса пропускания канала 90 Кбайт $\times$ 8 бит $\times$ 60 кадров/сек=42,2 Мбит/с, которая была бы слишком широкой полосой пропускания почти для всех домашних подключений к Internet в США в настоящее время, и слишком широкой полосой пропускания для многих офисных подключений к Internet. Безусловно, с учетом того, что при этом потребуется постоянный поток данных с такой широкой полосой пропускания, и он будет обслуживать только одного пользователя, даже в офисной среде LAN, он будет потреблять большую часть полосы пропускания LAN Ethernet 100 Мбит/с и сильно нагружать коммутаторы Ethernet, поддерживающие LAN. Соответственно, сжатие для движущегося видео неэффективно при сравнении с другими способами сжатия (например, способами сжатия, описанными ниже). Кроме того, алгоритмы сжатия одного кадра, подобные JPEG и JPEG-2000, которые используют алгоритмы сжатия с потерями, выводят артефакты сжатия, которые могут быть не заметны на неподвижных изображениях (например, артефакт внутри густой листвы в сцене может не казаться артефактом, так как глаз не знает точно, как густая листва должна выглядеть). Но, если сцена движется, то артефакт может быть заметным, потому что глаз замечает, что артефакт (изменяется) от кадра к кадру, несмотря на то, что артефакт находится в области сцены, где он мог быть незаметным в неподвижном изображении. Это в результате приводит к восприятию "фоновых помех" в последовательности кадров, сходных по виду с помехами "снег", видимых во время граничного аналогового телевизионного приема. Несомненно, этот тип сжатия может, тем не менее, использоваться в определенных вариантах осуществления, описанных в этом документе, но, вообще говоря, для данного качества восприятия требуется устранение фоновых помех в сцене, высокая скорость передачи данных (т.е. низкий коэффициент сжатия).

Другие типы сжатия, например, H.264 или Windows Media VC9, MPEG2 и MPEG4, все являются более эффективными при сжатии видеопотока, потому что они используют сходные черты между последовательными кадрами. Все эти способы основаны на одинаковых общих способах сжатия видео. Соответственно, несмотря на то, что будет описан стандарт H.264, однако идентичные общие принципы относятся к различным другим алгоритмам сжатия. Большое количество устройств сжатия H.264 и устройство восстановления сжатых данных имеются в наличии, в том числе библиотека программного обеспечения с открытым исходным текстом x264 для сжатия H.264 и библиотека программного обеспечения с открытым исходным текстом FFmpeg для восстановления сжатых данных H.264.

На фиг.6a и фиг.6b изображен иллюстративный способ сжатия предшествующего уровня техники, в котором последовательность несжатых видеок кадров 501-503, 559-561 сжимают посредством логики 620 сжатия в последовательность "I-кадров" 611, 671, "P-кадров" 612-613 и "B-кадров" 670. Вертикальная ось на фиг.6a в общем обозначает получающийся в результате размер каждого из закодированных кадров (несмотря на то, что кадры не вычерчены в масштабе). Как описано выше, кодирование видео с

использованием I-кадров, В-кадров и Р-кадров понятно специалистам в данной области техники. В нескольких словах, I-кадр 611 является сжатием на основе DCT полного несжатого кадра 501 (аналогичным сжатою изображению JPEG, как описано выше). Р-кадры 612-613, в общем, значительно меньше по размеру, чем I-кадры 611, потому что они используют данные предыдущего I-кадра или Р-кадра, то есть они содержат данные, указывающие на изменения между предыдущим I-кадром или Р-кадром. В-кадры 670 являются аналогичными Р-кадрам за исключением того, что В-кадры используют кадр в следующем опорном кадре, а также, возможно, кадр в предыдущем опорном кадре.

В нижеследующем обсуждении будет предполагаться, что требуемая частота кадров равна 60 кадров/секунда, что каждый I-кадр равен приблизительно 160 Кбит, средний Р-кадр и В-кадр равен 16 Кбит, и что новый I-кадр формируется каждую секунду. С этим набором параметров средняя скорость передачи данных равна: $160 \text{ Кбит} + 16 \text{ Кбит} \times 59 = 1,1 \text{ Мбит/с}$. Эта скорость передачи данных вполне находится в пределах максимальной скорости передачи данных для многих современных высокоскоростных подключений к Internet в домах и офисах. Посредством этого способа также можно решать проблему фоновых помех при только-внутрикадровом кодировании, потому что Р- и В-кадры отслеживают различия между кадрами, поэтому артефакты сжатия не должны появляться и исчезать от кадра к кадру, следовательно, уменьшается проблема фоновых помех, описанная выше.

Одна проблема, связанная с вышеизложенными видами сжатия, состоит в том, что, несмотря на то, что средняя скорость передачи данных является относительно низкой (например, 1,1 Мбит/с), передача одного I-кадра может занимать несколько периодов кадра. Например, с использованием способов предшествующего уровня техники, сетевого соединения 2,2 Мбит/с (например, DSL или кабельный модем с пиком 2,2 Мбит/с максимальной доступной скорости передачи данных 302 по фиг.3а) обычно достаточно для передачи потоком видео со скоростью 1,1 Мбит/с с I-кадром 160 Кбит/с каждые 60 кадров. Это может быть выполнено, если устройство восстановления сжатых данных будет ставить в очередь 1 секунду видео до восстановления сжатого видео. За 1 секунду передается 1,1 Мбит данных, что может быть легко обеспечено при максимальной доступной скорости передачи данных 2,2 Мбит/с, даже с предположением, что доступная скорость передачи данных может периодически снижаться на 50%. К сожалению, этот подход предшествующего уровня техники в результате приводит ко времени ожидания в 1 секунду для видео из-за 1-секундного видеобуфера в приемнике. Такая задержка отвечает требованиям многих приложений предшествующего уровня техники (например, воспроизведение линейного видео), но является слишком долгим временем ожидания для динамичных видеоигр-боевиков, в которых не допускается время ожидания более 70-80 мс.

Если попытаться исключить 1-секундный видеобуфер, то это по-прежнему не приведет к сокращению времени ожидания, достаточному для динамичных видеоигр-боевиков. Например, использование В-кадров, как описано ранее, неизбежно влечет за собой прием всех В-кадров, предшествующих I-кадру, а также I-кадра. Если предположить, что 59 кадров не-I-кадров примерно разделены между Р- и В-кадрами, то принимается, по меньшей мере, 29 В-кадров и I-кадр до того, как любой В-кадр может быть выведен на экран. Соответственно, независимо от доступной полосы пропускания канала, это неизбежно влечет за собой задержку $29 + 1 = 30$ кадров длительностью $1/60$ секунды каждый, или 500 мс времени ожидания. Очевидно, что это слишком долго.

Соответственно, другой подход должен исключать В-кадры и использовать только

I- и P-кадры. (Одним следствием этого является то, что скорость передачи данных увеличивается для данного уровня качества, но для согласованности в этом примере, будем продолжать предполагать, что размер каждого I-кадра равен 160 Кбит, и среднего P-кадра равен 16 Кбит, и, соответственно, скорость передачи данных по-прежнему равна 1,1 Мбит/с.) Этот подход исключает неизбежное время ожидания, вводимое B-кадрами, так как декодирование каждого P-кадра зависит только от предыдущего принятого кадра. Проблема, которая остается при этом подходе, состоит в том, что I-кадр настолько больше среднего P-кадра, что на канале с небольшой полосой пропускания, как обычно бывает в большинстве домов и во многих офисах, передача I-кадра добавляет значительное время ожидания. Это изображено на фиг.6b. Скорость 624 передачи данных видеопотока ниже доступной максимальной скорости 621 передачи данных за исключением I-кадров, где пиковая скорость передачи данных, требуемая для I-кадров 623 намного превышает доступную максимальную скорость 622 передачи данных (и даже номинальную максимальную скорость 621 передачи данных). Скорость 15 передачи данных, требуемая P-кадрами, меньше доступной максимальной скорости передачи данных. Даже если пики в 2,2 Мбит/с доступной максимальной скоростью передачи данных постоянно остаются на своей пиковой скорости 2,2 Мбит/с, то передача I-кадра займет $160 \text{ Кбит} / 2,2 \text{ Мбит} = 71 \text{ мс}$, и если доступная максимальная скорость 622 передачи данных снизится на 50% (1,1 Мбит/с), то передача I-кадра займет 142 мс. 20 Соответственно, время ожидания при передаче I-кадра находится приблизительно между 71-142 мс. Это время ожидания является дополнением к величинам времени ожидания, идентифицированным на фиг.4b, которые в самом плохом случае составляют в сумме 70 мс, поэтому в результате общее время передачи сигнала туда и обратно, от момента, когда пользователь приводит в действие устройство 421 ввода до того, как 25 изображение появляется на дисплее 422, равно 141-222 мс, которое является слишком большим. И если доступная максимальная скорость передачи данных становится меньше 2,2 Мбит/с, то время ожидания увеличивается дополнительно.

Отметим также, что, в общем, существуют серьезные последствия "создания пробки" ISP посредством пиковой скорости 623 передачи данных, которая намного превышает доступную скорость 622 передачи данных. Оборудование у разных ISP ведет себя по-разному, но следующие линии поведения являются довольно распространенными среди ISP кабельной модемной связи и DSL при приеме пакетов с гораздо большей скоростью передачи данных, чем доступная скорость 622 передачи данных: (а) задержка пакетов посредством постановки их в очередь (введение времени ожидания), (б) сброс некоторых 35 или всех пакетов, (с) прерывание соединения на некоторый период времени (скорее всего, потому что ISP будет обеспокоен тем, что это является злонамеренной атакой, например, атакой "отказ в обслуживании"). Соответственно, передача потока пакетов с полной скоростью передачи данных с такими характеристиками, как характеристики, изображенные на фиг.6b, не является практически осуществимым вариантом. Пики 623 40 могут стоять в очереди в службе 210 хостинга и отправляться со скоростью передачи данных ниже доступной максимальной скорости передачи данных, что вводит неприемлемое время ожидания, описанное в предыдущем абзаце.

Кроме того, последовательность 624 скорости передачи данных видеопотока, изображенная на фиг.6b, является очень "нормальной" последовательностью скорости 45 передачи данных видеопотока и может быть видом последовательности скорости передачи данных, которая, как можно ожидать, получается в результате сжатия видео из видеопоследовательности, которая не изменяется очень сильно и обнаруживает очень мало движения (например, как обычно бывает при проведении

видеотелеконференции, где камеры находятся в фиксированном положении и их мало передвигают, и объекты в сцене, например, разговаривающие сидящие люди, мало двигаются).

Последовательность 634 скорости передачи данных видеопотока, изображенная на фиг.6с, является последовательностью, которую обычно можно ожидать наблюдать из видео с гораздо большим количеством действий, например, которая может формироваться в кинофильме или видеоигре, или в некотором прикладном программном обеспечении. Отметим, что наряду с пиками 633 I-кадра, существуют также пики Р-кадра, например 635 и 636, которые являются довольно большими и превышают доступную максимальную скорость передачи данных во многих случаях. Несмотря на то, что эти пики Р-кадра не совсем такие большие, как пики I-кадра, они, тем не менее, являются слишком большими для передачи их по каналу с полной скоростью передачи данных, и как и в случае с пиками I-кадра, они, пики Р-кадра, должны передаваться медленно (следовательно, с возрастающим временем ожидания).

На канале с широкой полосой пропускания (например, LAN 100 Мбит/с или частное соединение с широкой полосой пропускания 100 Мбит/с) в сети могут допускаться большие пики, например, пики 633 I-кадра или пики 636 Р-кадра, и, в принципе, может поддерживаться малое время ожидания. Но такие сети часто совместно используются многими пользователями (например, в офисной обстановке), и такие данные "с пиками" влияют на производительность LAN, в частности, если сетевой трафик маршрутизируется к частному, совместно используемому соединению (например, из удаленного центра обработки и хранения данных в офис). Прежде всего, следует учитывать, что в этом примере видеопоток с относительно низким разрешением 640×480 пикселей с 60 кадр/с. Потоки HDTV 1920×1080 с 60 кадр/с легко обрабатываются современными компьютерами и дисплеями, и все больше имеется в продаже дисплеев с разрешающей способностью 2560×1440 с 60 кадр/с (например, дисплей 30" корпорации Apple, Inc). Видеопоследовательности с высокой активностью с 1920×1080 с 60 кадр/с может потребоваться 4,5 Мбит/с с использованием сжатия H.264 для приемлемого уровня качества. Если предположить, что I-кадры достигают пика при номинальной скорости передачи данных 10X, то в результате получаются пики 45 Мбит/с, а также меньший, но, тем не менее, значительный, пик Р-кадра. Если несколько пользователей принимают видеопотоки по идентичной сети 100 Мбит/с (например, частное сетевое соединение между офисом и центром обработки и хранения данных), то легко увидеть, как пики видеопотока нескольких пользователей могут объединяться с переполнением полосы пропускания сети и с возможным переполнением полосы пропускания магистральной коммутаторов, поддерживающих пользователей в сети. Даже в случае сети по технологии Gigabit Ethernet, если достаточное количество пиков достаточного количества пользователей объединяются одновременно, то это может переполнить сеть или сетевые коммутаторы. И когда видео с разрешающей способностью 2560×1440 станет более обычным явлением, средняя скорость передачи данных видеопотока может быть равна 9,5 Мбит/с, что в результате, возможно, приведет к пиковой скорости передачи данных 95 Мбит/с. Не вызывает сомнения, что соединение 100 Мбит/с между центром обработки и хранения данных и офисом (которое в настоящее время является исключительно высокоскоростным соединением) будет полностью переполнено пиковым трафиком отдельного пользователя. Соответственно, несмотря на то, что LAN и частные сетевые соединения могут допускать потоковое видео с пиками, потоковое видео с высокими пиками не желательно, и может потребоваться специальное планирование и адаптация отделом ИТ офиса.

Несомненно, для стандартных приложений линейного видео, эти вопросы не являются проблемой, потому что скорость передачи данных "сглаживается" в момент передачи, и данные для каждого кадра ниже максимальной доступной скорости 622 передачи данных, и в буфере в клиенте сохраняется последовательность I-, P- и B-кадров до того, как они будут восстановлены. Соответственно, скорость передачи данных по сети остается близкой к средней скорости передачи данных видеопотока. К сожалению, это вводит время ожидания, даже если B-кадры не используются, что является неприемлемым для приложений с малым временем ожидания, например, видеоигры и приложения требуют малого времени отклика.

Одним решением предшествующего уровня техники для уменьшения видеопотоков, которые имеют высокие пики, является использование способа, часто называемого кодирование "Постоянный битрейт" (постоянная скорость передачи битов, CBR). Несмотря на то, что представляется, что термин CBR подразумевает, что все кадры сжимаются так, чтобы у них была одинаковая скорость передачи битов (т.е. размер), он обычно относится к схеме сжатия, в которой допускается максимальная скорость передачи битов по определенному количеству кадров (в нашем случае, 1 кадр). Например, в случае фиг.6с, если ограничение CBR применяется к кодированию, которое ограничивает скорость передачи битов, например, до 70% номинальной максимальной скорости 621 передачи данных, тогда алгоритм сжатия ограничивает сжатие каждого из кадров так, чтобы любой кадр, который обычно сжимается с использованием больше 70% номинальной максимальной скорости 621 передачи данных, сжимается посредством меньшего количества битов. Результатом этого является то, что кадры, которым обычно требуется больше битов для поддержания данного уровня качества, "испытывают недостаток" в битах, и качество изображения этих кадров хуже, чем качество изображения других кадров, которым не требуется большее количество битов, чем 70% (скорости) максимальной скорости 621 передачи данных. Этот подход может привести к приемлемым результатам для определенных типов сжатого видео, когда (а) ожидается мало движения или изменений сцены, и (б) пользователи могут считать приемлемым периодическое ухудшение качества. Хорошим примером применения, подходящего для CBR, является проведение видеотелеконференции, так как в ней мало пиков, и если качество ухудшается на короткий период (например, если камерой выполняют панорамирование, что в результате приводит к существенному движению сцены и большим пиками, то во время панорамирования может быть не достаточно битов для сжатия высококачественного изображения, что в результате может привести к ухудшенному качеству изображения), это является приемлемым для большинства пользователей. К сожалению, CBR плохо подходит для многих других применений, в которых существуют сцены высокого уровня сложности или с большим движением, и/или в которых требуется достаточно постоянный уровень качества.

Логика 404 сжатия с малым временем ожидания, применяемая в одном варианте осуществления, использует несколько разных способов для решения круга проблем, связанных с передачей потоком сжатого видео с малым временем ожидания, при поддержании высокого уровня качества. Во-первых, логика 404 сжатия с малым временем ожидания формирует только I-кадры и P-кадры, следовательно, уменьшается необходимость ожидания в течение нескольких периодов кадра для декодирования каждого B-кадра. Кроме того, как изображено на фиг.7а, в одном варианте осуществления, логика 404 сжатия с малым временем ожидания разбивает каждый несжатый кадр 701-760 в последовательность "фрагментов" ("tile", "тайл") и отдельно кодирует каждый фрагмент или как I-кадр или как P-кадр. Группа сжатых I-кадров и

Р-кадров называется в этом описании "R-кадры" 711-770. В конкретном примере, изображенном на фиг.7а, каждый несжатый кадр разбивается в матрицу 4×4 из 16 фрагментов. Однако эти лежащие в основе принципы не ограничены никакой конкретной схемой подразделения.

5 В одном варианте осуществления логика 404 сжатия с малым временем ожидания разделяет видеокадр на несколько фрагментов и кодирует (т.е. сжимает) один фрагмент из каждого кадра как I-кадр (т.е. этот фрагмент сжимается так, как если бы он был отдельным видеокадром размером 1/16 полного изображения, и сжатие, используемое для этого "мини"-кадра, является сжатием I-кадра), а остальные фрагменты изображения
10 как Р-кадры (т.е. сжатие, используемое для каждой "мини" 1/16 (части) кадра, является сжатием Р-кадра). Фрагменты, сжатые как I-кадры и как Р-кадры, должны называться "I- фрагментами" и "Р- фрагментами", соответственно. С каждым последующим видеокадром, фрагмент, который должен быть закодирован как I-фрагмент, меняется. Соответственно, в данный период кадра, только один фрагмент из фрагментов в
15 видеокадре является I-фрагментом, а остальные фрагменты являются Р-фрагментами. Например, на фиг.7а, фрагмент 0 несжатого кадра 701 закодирован как I-фрагмент I_0 , а остальные 1-15 фрагменты закодированы как Р-фрагменты с P_1 по P_{15} для вывода R-кадра 711. В следующем несжатом видеокадре 702, фрагмент 1 несжатого кадра 701 закодирован как I-фрагмент I_1 , а остальные фрагменты 0 и с 2 по 15 закодированы как
20 Р-фрагменты, P_0 и с P_2 по P_{15} , для вывода R-кадра 712. Соответственно, I-фрагменты и Р-фрагменты для фрагментов постепенно чередуются во времени по последовательным кадрам. Процесс продолжается до тех пор, пока не будет сформирован R-(кадр) 770 с последним фрагментом в матрице, закодированным как I-фрагмент (т.е. I_{15}). После
25 этого процесс повторяется с формированием еще одного R-кадра, например, кадра 711 (т.е. кодирование I-фрагмента для фрагмента 0) и т.д. Несмотря на то, что на фиг.7а не изображено, в одном варианте осуществления, первый R-кадр видеопоследовательности из R-кадров содержит только I-фрагменты (т.е. последующие (R)-кадры содержат данные опорного изображения, исходя из которого вычисляется движение). В качестве
30 альтернативы, в одном варианте осуществления, последовательность в начальный момент использует схему I-фрагмента, идентичную обычной, но не включает в себя Р-фрагменты для тех фрагментов, которые еще не закодированы посредством I-фрагмента. Другими словами, определенные фрагменты не кодируются ни какими данными до тех пор, пока не поступит первый I-фрагмент, следовательно, устраняются начальные пики
35 в скорости 934 передачи данных видеопотока по фиг.9а, что более подробно объясняется ниже. Кроме того, как описано ниже, различные другие размеры и формы могут использоваться для фрагментов, хотя, по-прежнему, с выполнением этих лежащих в основе принципов.

Логика 412 восстановления сжатого видео, исполняющаяся в клиенте 415,
40 восстанавливает каждый сжатый фрагмент, как если бы это была отдельная видеопоследовательность маленьких I- и Р-кадров, и затем передает каждый фрагмент в буфер кадров, управляющий дисплеем 422. Например, I_0 и P_0 из R-кадров 711-770 используются для восстановления сжатого фрагмента 0 видеоизображения и его визуализации. Точно так же I_1 и P_1 из R-кадров 711-770 используются для восстановления
45 фрагмента 1 и так далее. Как упоминалось выше, восстановление сжатых I-кадров и Р-кадров известно в данной области техники, а восстановление сжатых I-фрагментов и Р-фрагментов может выполняться посредством наличия множества устройств восстановления сжатого видео, функционирующих в клиенте 415. Несмотря на то, что

представляется, что увеличение количества процессов увеличивает затраты вычислительных ресурсов в клиенте 415, это фактически не так, потому что сам фрагмент пропорционально меньше относительно количества дополнительных процессов, поэтому количество выводимых на экран пикселей является идентичным, как если бы

5 существовал один процесс и использовались обычные I- и P-кадры полного размера.

Этот способ R-кадра значительно уменьшает пики полосы пропускания, обычно связанные с I-кадрами, изображенными на фиг.6b и фиг.6с, потому что любой данный кадр обычно составлен из P-кадров, которые обычно меньше I-кадров. Например, снова предположим, что обычный I-кадр равен 160 Кбит, тогда I-фрагменты каждого

10 из кадров, изображенных на фиг.7а, примерно равны $1/16$ этой величины или 10 Кбит. Аналогично, предположим, что обычный P-кадр равен 16 Кбит, тогда P-кадры для каждого из фрагментов, изображенные на фиг.7а, могут быть примерно равны 1 Кбит. Конечным результатом является R-кадр приблизительно $10 \text{ Кбит} + 15 \times 1 \text{ Кбит} = 25 \text{ Кбит}$. Соответственно, каждая последовательность из 60 кадров равна $25 \text{ Кбит} \times 60 = 1,5 \text{ Мбит/с}$. Соответственно, при 60 кадров/секунда потребуется канал с возможностью поддержки

15 полосы пропускания 1,5 Мбит/с, но с гораздо меньшими пиками, появляющимися из-за I-фрагментов, распределенных по всему интервалу из 60 кадров.

Отметим, что в предыдущих примерах с идентичным предположением о скоростях передачи данных для I-кадров и P-кадров, средняя скорость передачи данных была

20 равна 1,1 Мбит/с. Причиной этого является то, что в предыдущих примерах новый I-кадр вводится только один раз каждые 60 периодов кадра, тогда как в этом примере 16 фрагментов, которые составляют I-кадр, циклически повторяются через 16 периодов кадра, и, по существу, эквивалент I-кадра вводится каждые 16 периодов кадра, что в результате приводит к несколько более высокой средней скорости передачи данных.

На практике, тем не менее, более частое введение I-кадров не увеличивает скорость

25 передачи данных линейно. Это происходит вследствие того, что P-кадр (или P-фрагмент) первоначально кодирует отличие последующего кадра от предшествующего.

Соответственно, если предшествующий кадр является довольно похожим на следующий кадр, то P-кадр является очень маленьким, если предшествующий кадр весьма отличается

30 от следующего кадра, то P-кадр является очень большим. Но так как P-кадр в значительной степени получается из предыдущего кадра, а не из фактического кадра, то получающийся в результате закодированный кадр может содержать больше ошибок (например, визуальные артефакты), чем I-кадр с соответствующим количеством битов. И, когда один P-кадр следует за другим P-кадром, то может произойти накопление

35 ошибок, которое ухудшается при наличии длинной последовательности P-кадров.

Далее, усложненное устройство сжатия видео обнаруживает, что качество изображения ухудшается после последовательности P-кадров, и, в случае необходимости, оно выделяет больше битов последующим P-кадрам для повышения качества или, если это

40 будет наиболее эффективным образом действия, заменяет P-кадр I-кадром.

Соответственно, когда используются длинные последовательности P-кадров (например, 59 P-кадров, как в предыдущих примерах выше), в частности, когда сцена очень сложная и/или в ней много движения, обычно требуется больше битов для P-кадров по мере их

удаления от I-кадра.

Другими словами, если посмотреть на P-кадры с противоположной точки зрения,

45 P-кадры, которые непосредственно следуют за I-кадром, как правило, требуют меньшего количества битов, чем P-кадры, которые являются более удаленными от I-кадра. Так, в примере, изображенном на фиг.7а, все P-кадры удалены от предшествующего им I-кадра не более чем на 15 кадров, тогда как, например, в предыдущем примере, P-кадр

может находиться на расстоянии 59 кадров от I-кадра. Соответственно, при большей частоте I-кадров, Р-кадры меньше. Несомненно, точные относительные размеры меняются в зависимости от характера видеопотока, но в примере по фиг.7а, если I-фрагмент равен 10 Кбит, то размер Р-фрагментов в среднем может быть равен только 0,75 Кбит, что в результате приводит к $10 \text{ Кбит} + 15 \times 0,75 \text{ Кбит} = 21,25 \text{ Кбит}$, или при (скорости) 60 кадров в секунду, скорость передачи данных равна $21,25 \text{ Кбит} \times 60 = 1,3 \text{ Мбит/с}$, или приблизительно на 16% больше, чем скорость передачи данных потока с I-кадром, за которым следуют 59 Р-кадров, в 1,1 Мбит/с. И опять же, относительные результаты этих двух подходов к сжатию видео меняются в зависимости от видеопоследовательности, но, как правило, опыт показывает, что использование Р-кадров требует приблизительно на 20% больше битов для данного уровня качества, чем использование последовательностей I/Р-кадров. Но, несомненно, Р-кадры существенно уменьшают пики, что намного уменьшает время ожидания при использовании упомянутых видеопоследовательностей, по сравнению со временем ожидания для последовательности I/Р-кадров.

Р-кадры могут быть сконфигурированы множеством различных способов, в зависимости от характера видеопоследовательности, надежности канала и доступной скорости передачи данных. В альтернативном варианте осуществления используется количество фрагментов, отличное от 16 в конфигурации 4×4. Например, может использоваться 2 фрагмента в конфигурации 2×1 или 1×2, может использоваться 4 фрагмента в конфигурации 2×2, 4×1 или 1×4, может использоваться 6 фрагментов в конфигурации 3×2, 2×3, 6×1 или 1×6, или может использоваться 8 фрагментов в конфигурации 4×2 (как изображено на фиг.7b), 2×4, 8×1 или 1×8. Отметим, что фрагменты не обязательно должны быть квадратными, также видеокادر не обязательно должен быть квадратным или даже прямоугольным. Фрагменты могут принимать любую форму, которая лучше всего подходит для используемого приложения или видеопотока.

В другом варианте осуществления циклическое повторение I- и Р-фрагментов не фиксировано количеством фрагментов. Например, в конфигурации 4×2 с 8 фрагментами может, однако, использоваться последовательность с периодическим повторением через 16 элементов, как изображено на фиг.7b. Последовательные несжатый кадры 721, 722, 723, каждый, разделены на 8 фрагментов, 0-7, и каждый фрагмент сжимают отдельно. В R-кадре 731, только фрагмент 0 сжат как I-фрагмент, а остальные фрагменты сжаты как Р-фрагменты. В следующем R-кадре 732 все 8 фрагментов сжаты как Р-фрагменты, и далее в следующем R-кадре 733, фрагмент 1 сжат как I-фрагмент, а остальные фрагменты все сжаты как Р-фрагменты. И такое установление последовательности продолжается для 16 кадров, при этом I-фрагмент формируется только через кадр, соответственно, последний I-фрагмент будет сформирован для фрагмента 7 в течение периода 15-ого кадра (на фиг.7b не изображен) и в течение периода 16-ого кадра сжимают R-кадр 780 с использованием всех Р-фрагментов. Далее последовательность снова начинается с фрагмента 0, сжатого как I-фрагмент, при этом другие фрагменты сжаты как Р-фрагменты. Как в предшествующем варианте осуществления, самый первый кадр всей видеопоследовательности обычно состоит из всех I-фрагментов для обеспечения опорных элементов (reference) для Р-фрагментов, начиная с этого момента. Циклическое повторение I-фрагментов и Р-фрагментов даже не должно быть четным, кратным числом количества фрагментов. Например, с 8 фрагментами, за каждым кадром с I-фрагментом может следовать 2 кадра со всеми Р-фрагментами, прежде, чем будет использован другой I-фрагмент. В еще одном варианте осуществления, для определенных фрагментов может устанавливаться

последовательность I-фрагментов чаще, чем для других фрагментов, если, например, известно, что в определенных областях экрана больше движения, что требует частых I-фрагментов, в то время как другие являются более статическими (например, с изображением счета в игре), что требует менее частых I-фрагментов. Кроме того, несмотря на то, что каждый кадр на фиг.7а-фиг.7b изображен с одним I-фрагментом, в одном кадре может быть закодировано множество I-фрагментов (в зависимости от полосы пропускания канала передачи). И наоборот, определенные кадры или последовательности кадров могут быть переданы без I-фрагментов (т.е. только P-фрагменты).

Причина, по которой подходы, описанные в предыдущем абзаце, работают хорошо, состоит в том, что, если не распределить I-фрагменты по каждому отдельному кадру, то, как представляется, это в результате приведет к большим пикам, поведение системы не является таким простым. Так как каждый фрагмент сжимают отдельно от других фрагментов, то, по мере того, как фрагменты становятся меньше, кодирование каждого фрагмента может стать менее эффективным, потому что устройство сжатия данного фрагмента не может использовать сходные признаки изображения и сходное движение из остальных фрагментов. Соответственно, деление экрана на 16 фрагментов в общем в результате приведет к менее эффективному кодированию, чем деление экрана на 8 фрагментов. Но, если экран разделен на 8 фрагментов, и это вызывает ввод данных полного I-кадра каждые 8 кадров, вместо каждых 16 кадров, то это в результате приводит к гораздо более высокой скорости передачи данных в целом. Соответственно, при введении полного I-кадра каждые 16 кадров, вместо того, чтобы вводить его каждые 8 кадров, общая скорость передачи данных уменьшается. Кроме того, при использовании 8 больших фрагментов, вместо 16 меньших фрагментов, уменьшается общая скорость передачи данных, что также уменьшает до некоторой степени пики данных, вызываемые большими фрагментами.

В другом варианте осуществления, логика 404 сжатия видео с малым временем ожидания по фиг.7а и фиг.7b управляет выделением битов различным фрагментам в R-кадрах или с предварительным конфигурированием посредством установок, на основе известных характеристик видеопоследовательности, которая должна быть сжата, или автоматически, на основе непрерывного анализа качества изображения в каждом фрагменте. Например, в некоторых видеоиграх-гонках, передняя часть автомобиля игрока (которая является относительно неподвижной в сцене) занимает большую часть нижней половины экрана, тогда как верхняя половина экрана полностью заполнена надвигающимся шоссе, зданиями и пейзажем, которые почти всегда находятся в движении. Если логика 404 сжатия выделяет равное количество битов каждому фрагменту, то фрагменты в нижней половине экрана (фрагменты 4-7) в несжатом кадре 721 по фиг.7b, будут, в общем, сжиматься с более высоким качеством, чем фрагменты (чем фрагменты) в верхней половине экрана (фрагменты 0-3) в несжатом кадре 721 по фиг.7b. Если известно, что эта конкретная игра или эта конкретная сцена игры имеют такие характеристики, то операторы 210 службы хостинга могут сконфигурировать логику 404 сжатия для выделения большего количества битов фрагментам, находящимся в верхней части экрана, чем фрагментам, находящимся в нижней части экрана. Или, логика 404 сжатия может оценивать качество сжатия фрагментов после сжатия кадров (с использованием одного или нескольких из множества показателей качества сжатия, например, пиковое отношение сигнал/шум (PSNR)) и если она определяет, что за определенное временное окно качество определенных фрагментов постоянно улучшается, тогда она постепенно выделяет большее количество битов фрагментам,

качество которых ухудшается до тех, пока уровень качества различных фрагментов не станет примерно одинаковым. В альтернативном варианте осуществления логика 404 устройства сжатия выделяет биты для получения более высокого качества в конкретном фрагменте или группе фрагментов. Например, с более высоким качеством в центре экрана, чем на краях, она может обеспечивать лучшее общее восприятие.

В одном варианте осуществления, для улучшения разрешающей способности определенных участков видеопотока, логика 404 сжатия видео использует меньшие фрагменты для кодирования областей видеопотока с относительно большей сложностью сцены и/или большим движением, чем для областей видеопотока с относительно меньшей сложностью сцены и/или меньшим движением. Например, как изображено на фиг.8, меньшие фрагменты применяются вокруг движущегося персонажа 805 в одной области одного R-кадра 811 (за которым, возможно, следует последовательность R-кадров с идентичными размерами фрагментов (не изображены)). Далее, когда персонаж 805 перемещается в новую область изображения, меньшие фрагменты используются вокруг этой новой области внутри другого R-кадра 812, как изображено. Как упоминалось выше, различные другие размеры и формы могут быть применены как "фрагменты", хотя, по-прежнему, с выполнением этих лежащих в основе принципов.

Несмотря на то, что циклические I/P-фрагменты, описанные выше, существенно уменьшают пики в скорости передачи данных видеопотока, они не устраняют пики полностью, в частности, в случае таких быстро изменяющихся или очень сложных видеоизображениях, которые встречаются в кинофильмах, видеоиграх и некотором прикладном программном обеспечении. Например, во время внезапного монтажного перехода, за сложным кадром может следовать другой сложный кадр, который полностью отличается от него. Несмотря на то, что несколько I-фрагментов могут находиться перед монтажным переходом только на расстоянии нескольких периодов кадра, они не помогут в этой ситуации, потому что материала нового кадра не имеет связи с предыдущими I-фрагментами. В такой ситуации (и в других ситуациях, когда, несмотря на то, что не все изменяется, большая часть изображения изменяется) устройство 404 сжатия видео определяет, что многие, если не все, R-фрагменты более эффективно кодировать как I-фрагменты, и в результате получается очень большой пик в скорости передачи данных для этого кадра.

Как обсуждалось ранее, это просто имеет место, что с большинством подключений к Internet потребительского уровня (и многими офисными соединениями), просто не выполнимо для данных, (создающих) "пробку", которые превышают доступную максимальную скорость передачи данных, изображенную позицией 622 на фиг.6с, вместе с номинальной максимальной скоростью 621 передачи данных. Отметим, что номинальная максимальная скорость 621 передачи данных (например, "DSL 6 Мбит/с") является, по существу, цифрой сбыта для пользователей, рассматривающих покупку подключения к Internet, но, в общем, она не гарантирует уровень производительности. Для целей этой заявки, это не имеет отношение к делу, так как единственной проблемой является доступная максимальная скорость 622 передачи данных во время передачи потоком видео через соединение. Следовательно, на фиг.9а и фиг.9с, так как мы описываем решение проблемы пиков, номинальная максимальная скорость передачи данных не изображена на графике, и изображена только доступная максимальная скорость 922 передачи данных. Скорость передачи данных видеопотока не должна превышать доступную максимальную скорость 922 передачи данных.

Для решения этой (проблемы), первым, что делает устройство 404 сжатия видео, является определение пиковой скорости 941 передачи данных, которая является

скоростью передачи данных, которую канал в состоянии обрабатывать устойчиво. Эта скорость может быть определена несколькими способами. Один такой способ состоит в постепенной отправки тестового потока с все более высокой скоростью передачи данных из службы по 210 размещению информации на сервере в клиента 415 по фиг.4a и фиг.4b, и обеспечении клиентом обратной связи в службу хостинга относительно уровня потери пакетов и времени ожидания. Когда потеря пакетов и/или время ожидания начинают резко увеличиваться, это указывает на приближение к доступной максимальной скорости 922 передачи данных. После этого служба 210 хостинга может постепенно уменьшать скорость передачи данных тестового потока до тех пор, пока клиент 415 не сообщит, что за достаточный период времени, тестовый поток был принят с приемлемым уровнем потери пакетов, и время ожидания является почти минимальным. Это устанавливает пиковую максимальную скорость 941 передачи данных, которая далее используется как пиковая скорость передачи данных для потокового видео. С течением времени пиковая скорость 941 передачи данных будет колебаться (например, если другой пользователь, живущий в этом доме, начнет интенсивно использовать подключение к Internet), и клиент 415 должен будет постоянно осуществлять текущий контроль над увеличением времени ожидания или потери пакетов с указанием того, что доступная максимальная скорость 922 передачи данных падает ниже ранее установленной пиковой скорости 941 передачи данных, и если это так, то пиковую скорость 941 передачи данных. Аналогично, если с течением времени клиент 415 находит, что потеря пакетов и время ожидания остаются на оптимальных уровнях, то он может отправлять запрос, чтобы устройство сжатия видео медленно увеличивало скорость передачи данных для проверки того, увеличилась ли доступная максимальная скорость передачи данных (например, прекратил ли другой пользователь, живущий в доме, интенсивно использовать подключение к Internet), и снова ожидать до тех пор, пока потеря пакетов и/или более высокое время ожидания не будут указывать на то, что доступная максимальная скорость 922 передачи данных была превышена, и снова может быть найден более низкий уровень для пиковой скорости 941 передачи данных, но тот, который может быть выше, чем уровень, который был до тестирования на увеличение скорости передачи данных. Соответственно, с использованием этого способа (и других способов, подобных этому) может быть найдена пиковая скорость 941 передачи данных и периодически корректироваться, по мере необходимости. Пиковая скорость 941 передачи данных устанавливает максимальную скорость передачи данных, которая может использоваться устройством 404 сжатия видео для передачи потоком видео пользователю. Логика для определения пиковой скорости передачи данных может быть реализована на территории 211 пользователя и/или в службе 210 хостинга. На территории 211 пользователя, клиентское устройство 415 выполняет вычисления для определения пиковой скорости передачи данных и передает эту информацию обратно в службу 210 хостинга, в службе 210 хостинга, сервер 402 в службе хостинга выполняет вычисления для определения пиковой скорости передачи данных на основе статистики, принятой из клиента 415 (например, потеря пакетов, время ожидания, максимальная скорость передачи данных и т.д.).

На фиг.9a изображена иллюстративная скорость 934 передачи данных видеопотока, который содержит значительную сложность сцены и/или значительное движение, сформированного с использованием способа сжатия циклического I/P-фрагмента, описанного ранее и изображенного на фиг.7a, фиг.7b и фиг.8. Устройство 404 сжатия видео было сконфигурировано для вывода сжатого видео со средней скоростью передачи данных, которая ниже пиковой скорости 941 передачи данных, и отметим, что, почти

все время, скорость передачи данных видеопотока остается ниже пиковой скорости 941 передачи данных. При сравнении скорости 934 передачи данных со скоростью 634 передачи данных видеопотока, изображенной на фиг.6с, создаваемой с использованием I/P/B- или I/P-кадров, видно, что на выходе сжатия циклического I/P-фрагмента

5 получается гораздо более гладкая скорость передачи данных. Однако в пике 952 2х кадра (который приближается к пиковой скорости 942 передачи данных 2х) и пике 954 4х кадра (который приближается к пиковой скорости 944 передачи данных 4х) скорость передачи данных превышает пиковую скорость 941 передачи данных, что является неприемлемым. На практике, даже с видео с высокой активностью в видеоиграх с

10 быстрым изменением, пики, превышающие пиковую скорость 941 передачи данных, встречаются меньше, чем в 2% кадров, пики, превышающие пиковую скорость 942 передачи данных 2х, встречаются редко, а пики, превышающие пиковую скорость 943 передачи данных 3х, не встречаются почти никогда. Но, когда они действительно встречаются (например, во время монтажного перехода), требующаяся им скорость

15 передачи данных является необходимой для вывода видеоизображения хорошего качества.

Один способ решения этой проблемы состоит в том, чтобы просто сконфигурировать устройство 404 сжатия видео так, что выход его максимальной скорости передачи данных является пиковой скоростью 941 передачи данных. К сожалению, получающееся

20 в результате качество видео на выходе во время пиковых кадров является плохим, так как алгоритм сжатия "испытывает недостаток" в битах. В результате появляются артефакты сжатия, когда осуществляются внезапные переходы или быстрое движение, и со временем, пользователь понимает, что артефакты всегда неожиданно возникают, когда происходят внезапные изменения или быстрое движение, и это может его весьма

25 раздражать.

Несмотря на то, что зрительная система человека довольно чувствительна к визуальным артефактам, которые появляются во время внезапных изменений или быстрого движения, она не является очень чувствительной, чтобы заметить уменьшение частоты кадров в таких ситуациях.

30 Фактически, когда происходят такие внезапные изменения, оказывается, что зрительная система человека занята отслеживанием изменений, и она не замечает, если частота кадров на короткое время падает с 60 кадр/с до 30 кадр/с и затем сразу возвращается к 60 кадр/с. И, в случае очень значительного перехода, подобного внезапному изменению сцены, зрительная система человека не замечает, если частота

35 кадров падает до 20 кадр/с или даже 15 кадр/с и затем сразу возвращается к 60 кадр/с. До тех пор, пока уменьшение частоты кадров происходит редко, человеку-наблюдателю, кажется, что видео непрерывно передается со скоростью 60 кадр/с.

Это свойство зрительной системы человека используется в способах, изображенных на фиг.9b. Сервер 402 (по фиг.4а и фиг.4b) выводит несжатый выходной видеопоток с

40 устойчивой частотой кадров (с 60 кадр/с в одном варианте осуществления). На временной шкале изображено, что каждый кадр 961-970 выводится каждую 1/60-ую долю секунды. Каждый несжатый видеокادر, начиная с кадра 961, выводится в устройство 404 сжатия видео с малым временем ожидания, которое сжимает кадр за время, меньшее, чем период кадра, с выводом для первого кадра сжатого кадра 981 1.

45 Данные, выводимые для сжатого кадра 981 1, могут быть большего размера или меньшего размера, в зависимости от многих факторов, как описано ранее. Если размер данных такой маленький, что они могут быть переданы клиенту 415 в течение периода кадра (1/60-ая доля секунды) или меньше с пиковой скоростью 941 передачи данных,

то они передаются в течение периода 991 передачи (xmit time) (длина стрелки указывает на продолжительность периода передачи). В течение следующего периода кадра сервер 402 выводит несжатый кадр 962 2, он сжимается в сжатый кадр 982 2, и он передается в клиент 415 в течение периода 992 передачи, который меньше, чем период кадра, с

5 пиковой скоростью 941 передачи данных.

Далее, в течение следующего периода кадра, сервер 402 выводит несжатый кадр 963 3. Когда он сжимается устройством 404 сжатия видео, получающийся в результате, сжатый кадр 983 3 является большим количеством данных, чем может быть передано с пиковой скоростью 941 передачи данных в течение одного периода кадра.

10 Соответственно, он передается в течение периода 993 передачи (пик 2х), который занимает весь период кадра и часть следующего периода кадра. Далее, в течение следующего периода кадра, сервер 402 выводит другой несжатый кадр 964 4 и выводит его в устройство 404 сжатия видео, но данные игнорируются, и они изображены посредством позиции 974. Это происходит потому, что устройство 404 сжатия видео

15 сконфигурировано для игнорирования последующих несжатых видеокадров, которые поступают, когда оно все еще передает предыдущий сжатый кадр. Несомненно, устройство восстановления сжатого видео клиента 415 не примет кадр 4, но оно просто будет продолжать выводить на экран дисплея 422 кадр 3 в течение двух периодов кадра (т.е. на короткое время уменьшит частоту кадров с 60 кадр/с до 30 кадр/с).

20 В течение следующего кадра 5, сервер 402 выводит несжатый кадр 965 5, который сжимается в сжатый кадр 985 5 и передается внутри 1 кадра в течение периода 995 передачи. Устройство восстановления сжатого видео клиента 415 восстанавливает сжатый кадр 5 и выводит его на экран дисплея 422. Далее, сервер 402 выводит несжатый кадр 966 6, устройство 404 сжатия видео сжимает его в сжатый кадр 986 6, но на этот

25 раз размер, получившихся в результате данных, является очень большим. Сжатый кадр передается в течение периода 996 передачи (пик 4х) с пиковой скоростью 941 передачи данных, но передача кадра занимает почти 4 периода кадра. В течение следующих 3 периодов кадра, устройство 404 сжатия видео игнорирует 3 кадра из сервера 402, и устройство восстановления сжатых данных клиента 415 постоянно выводит кадр 6 на

30 дисплей 422 в течение 4 периодов кадра (т.е. на короткое время уменьшает частоту кадров с 60 кадр/с до 15 кадр/с). И наконец, сервер 402 выводит кадр 970 10, устройство 404 сжатия видео сжимает его в сжатый кадр 987 10, и он передается в течение периода 997 передачи, и устройство восстановления сжатых данных клиента 415 восстанавливает сжатый кадр 10 и выводит его на экран дисплея 422, и снова вывод видео возобновляется

35 со скоростью 60 кадр/с.

Отметим, что, несмотря на то, что устройство 404 сжатия видео сбрасывает видеокадры видеопотока, формируемого сервером 402, оно не сбрасывает аудиоданные, независимо от того, в какой форме поступает аудио, и оно продолжает сжимать аудиоданные, когда видеокадры сбрасываются, и передает их в клиент 415, который

40 продолжает восстанавливать сжатые аудиоданные и обеспечивать аудио в устройство, которое использует пользователь для воспроизведения аудио, каково бы оно ни было. Соответственно, аудио продолжает воспроизводиться полностью в течение периодов, когда кадры сбрасываются. Сжатое аудио использует относительно маленький процент от полосы пропускания, по сравнению со сжатым видео, и в результате не оказывает

45 значительного влияния на общую скорость передачи данных. Несмотря на то, что это не изображено ни в одной из схем скорости передачи данных, всегда существует пропускная способность скорости передачи данных, зарезервированная для сжатого аудиопотока в пределах пиковой скорости 941 передачи данных.

Пример, только что описанный по фиг.9b, был выбран для иллюстрации того, как частота кадров падает во время пиков скорости передачи данных, но он не иллюстрирует то, как используются способы с циклическим I/P-фрагментом, описанные ранее, такие пики скорости передачи данных и последующие сброшенные кадры являются редкими, даже во время последовательностей с высокой активностью/с высокой сложностью сцены, например, последовательностей, которые встречаются в видеоиграх, кинофильмах и некотором прикладном программном обеспечении. Следовательно, уменьшенные частоты кадров являются нечастыми и кратковременными, и зрительная система человека не замечает их.

Если только что описанный механизм уменьшения частоты кадров применяется к скорости передачи данных видеопотока, изображенной на фиг.9a, то получающаяся в результате скорость передачи данных видеопотока изображена на фиг.9c. В этом примере пик 952 2х был уменьшен до сглаженного пика 953 2х, и пик 955 4х был уменьшен до сглаженного пика 955 4х, и вся скорость 934 передачи данных видеопотока остается на уровне пиковой скорости 941 передачи данных или ниже его.

Соответственно, с использованием способов, описанных выше, видеопоток с высокой активностью может быть передан с малым временем ожидания через общий Internet и через подключение к Internet потребительского уровня. Кроме того, в офисной обстановке в LAN (например, Ethernet 100 Мбит или беспроводная связь 802.11g) или в частной сети (например, соединение 100 Мбит/с между центром обработки и хранения данных и офисами) видеопоток с высокой активностью может передаваться без пиков так, что множество пользователей (например, передающих 1920×1080 с 60 кадр/с с 4,5 Мбит/с) могут использовать LAN или совместно используемое частное информационное соединение без накладывающихся пиков, переполняющих сеть или магистрали сетевого коммутатора.

Корректировка скорости передачи данных

В одном варианте осуществления служба 210 хостинга сначала оценивает доступную максимальную скорость 622 передачи данных и время ожидания канала для определения соответствующей скорости передачи данных для видеопотока и затем динамически корректирует скорость передачи данных в ответе. Для корректировки скорости передачи данных, служба 210 хостинга может, например, модифицировать разрешающую способность изображения и/или количество кадров/секунда видеопотока, который должен быть отправлен в клиент 415. Кроме того, служба хостинга может корректировать уровень качества сжатого видео. При изменении разрешающей способности видеопотока, например, с разрешающей способности 1280×720 на 640×360 логика 412 восстановления сжатого видео в клиенте 415 может приводить к масштабу изображение для поддержания идентичного размера изображения на экране дисплея.

В одном варианте осуществления, в ситуации, когда в канале пропадает сигнал, служба 210 хостинга приостанавливает игру. В случае игры с несколькими участниками служба хостинга сообщает другим пользователям, что упомянутый пользователь выбыл из игры и/или приостанавливает игру (с) другими пользователями.

Сброшенные или задержанные пакеты

В одном варианте осуществления, если данные потеряны из-за потери пакетов между устройством 404 сжатия видео и клиентом 415 по фиг.4a или фиг.4b, или из-за пакета, принимаемого не по порядку, который поступает слишком поздно, для восстановления сжатого кадра и удовлетворения требований ко времени ожидания восстановленного кадра, логика 412 восстановления сжатого видео может уменьшать визуальные артефакты. При реализации потокового I/P-кадра, если существует потерянный/

задержанный пакет, то это влияет на весь экран, возможно, вызывая полное зависание экрана на некоторый период времени или показ других визуальных артефактов по всему экрану. Например, если потерянный/задержанный пакет вызывает потерю I-кадра, то в устройстве восстановления сжатых данных будет отсутствовать опорный элемент для всех последующих R-кадров до тех пор, пока не будет принят новый I-кадр. Если R-кадр будет потерян, то это повлияет на R-кадры для всего экрана, которые за ним следуют. В зависимости от того, сколько времени пройдет до того, как появится I-кадр, это окажет более длительное или более короткое визуальное воздействие. С использованием чередующихся I/P-фрагментов, как изображено на фиг.7a и фиг.7b, гораздо менее вероятно, что потерянный/задержанный пакет повлияет на весь экран, так как он только повлияет на фрагменты, содержащиеся в затронутом пакете. Если данные каждого фрагмента отправляют в отдельном пакете, то, если пакет потерян, то это повлияет только на один фрагмент. Несомненно, продолжительность визуального артефакта будет зависеть от того, потерян ли пакет I-фрагмента, и, если R-фрагмент потерян, то, через какое количество кадров появится I-фрагмент. Но, с учетом того, что разные фрагменты на экране обновляются посредством I-кадров очень часто (возможно, каждый кадр), даже если затрагивается один фрагмент на экране, то другие фрагменты могут не затрагиваться. Кроме того, если некоторое событие вызывает потерю нескольких пакетов одновременно (например, всплеск напряжения электропитания рядом с линией DSL, который на короткое время прерывает поток данных), тогда некоторые из фрагментов будут затронуты больше, чем другие, но вследствие того, что некоторые фрагменты будут быстро обновлены посредством нового I-фрагмента, то их это затронет только на короткое время. Кроме того, с реализацией потокового I/P-кадра, I-кадры являются не только наиболее важным кадром, но и I-кадры являются чрезвычайно большими, поэтому, если существует событие, которое вызывает сброс/задержку пакета, то существует большая вероятность того, что будет затронут I-кадр (т.е., если какая-либо часть I-кадра потеряна, то маловероятно, что I-кадр может быть вообще восстановлен), чем намного меньший I-фрагмент. Вследствие всех этих причин, использование I/P-фрагментов в результате приводит к гораздо меньшим визуальным артефактам, когда пакеты сбрасываются/задерживаются, чем с I/P-кадрами.

Один вариант осуществления направлен на уменьшение влияния потерянных пакетов посредством рациональной упаковки сжатых фрагментов внутри пакетов TCP (протокол управления передачей) или пакетов UDP (протокол передачи дейтаграмм пользователя). Например, в одном варианте осуществления, фрагменты выравниваются по границам пакета всегда, когда это возможно. На фиг.10a изображено то, как фрагменты могут быть упакованы внутри последовательности пакетов 1001-1005 без реализации этого признака. А именно, на фиг.10a, фрагменты пересекают границы пакета и упакованы неэффективно так, что потеря одного пакета в результате приводит к потере множества кадров. Например, если потеряны пакеты 1003 или 1004, то потеряны три фрагмента, что в результате приводит к визуальным артефактам.

В отличие от этого, на фиг.10b изображена логика 1010 упаковки фрагмента для рациональной упаковки фрагментов внутри пакетов для уменьшения влияния потери пакета. Сначала логика 1010 упаковки фрагмента выравнивает фрагменты по границам пакета. Соответственно, фрагменты T1, T3, T4, T7 и T2 выровнены по границе пакетов 1001-1005, соответственно. Логика упаковки фрагмента также направлена на размещение фрагментов внутри пакетов наиболее эффективным возможным способом без пересечения границ пакета. На основе размера каждого из фрагментов, фрагменты T1

и Т6 объединены в один пакет 1001, Т3 и Т5 объединены в один пакет 1002, фрагменты Т4 и Т8 объединены в один пакет 1003, фрагмент Т(7) добавлен в пакет 1004 и фрагмент Т2 добавлен в пакет 1005. Соответственно, согласно этой схеме, потеря одного пакета в результате приведет к потере не больше, чем 2 фрагментов, (а не 3 фрагментов, как

5 изображено на фиг.10а).

Одним дополнительным преимуществом в отношении варианта осуществления, изображенного на фиг.10b, является то, что фрагменты передаются в порядке, отличном от того, в котором они выводятся на экран внутри изображения. Соответственно, если соседние пакеты потеряны из-за идентичного события, препятствующего передаче, то

10 это затронет области, которые не находятся рядом друг с другом на экране, что создаст менее заметные артефакты на дисплее.

В одном варианте осуществления применяются способы прямого исправления ошибок (FEC) для защиты определенных частей видеопотока от канальных ошибок. Как известно в данной области техники, способы FEC, например Рида-Соломона и Витерби,

15 формируют и добавляют информацию данных для исправления ошибок к данным, передаваемым по каналу связи. Если происходит ошибка в основных данных (например, I-кадр), то можно использовать FEC для исправления этой ошибки.

Коды FEC увеличивают скорость передачи данных передачи, поэтому предпочтительно они используются только там, где они больше всего необходимы.

Если отправляются данные, которые не могут в результате привести к очень заметному визуальному артефакту, то может оказаться предпочтительным не использовать коды FEC для защиты данных. Например, потерянный Р-фрагмент, который непосредственно идет пред I-фрагментом, создаст на экране визуальный артефакт (т.е. на фрагменте на экране не будет обновлен) только на 1/60-ую долю секунды. Человеческий глаз не

25 замечает такой визуальный артефакт. Чем дальше Р-фрагменты отстоят от следующего I-фрагмента, тем все более заметной становится потеря Р-фрагмента. Например, если циклической схемой фрагмента является I-фрагмент, за которым следуют 15 Р-фрагментов до того, как I-фрагмент будет снова доступен, то, если потерян Р-фрагмент, непосредственно следующий за I-фрагментом, то это в результате приведет к тому, что

30 этот фрагмент будет показывать неправильное изображение в течение 15 периодов кадра (при 60 кадр/с, это равно 250 мс). Человеческий глаз легко замечает разрыв в потоке, продолжительностью 250 мс. Соответственно, чем дальше отстоит Р-фрагмент от следующего нового I-фрагмента (т.е. чем ближе Р-фрагменты расположены к предыдущему I-фрагменту), тем более заметным становится артефакт. Как обсуждалось ранее, несмотря на то, что, в общем, чем ближе Р-фрагмент расположен к предыдущему I-фрагменту, тем меньше данные для этого Р-фрагмента. Соответственно, Р-фрагменты, которые следуют за I-фрагментами не только более важно защищать от потери, но и они меньше по размеру. И, в общем, чем меньше данных, которые необходимо защищать, тем меньший код FEC нужен для их защиты.

Соответственно, как изображено на фиг.11а, в одном варианте осуществления, из-за важности I-фрагментов в видеопотоке, только I-фрагменты обеспечиваются кодами FEC. Соответственно, FEC 1101 содержит код исправления ошибок для I-фрагмента 1100, и FEC 1104 содержит код исправления ошибок для I-фрагмента 1103. В этом варианте осуществления FEC не формируется для Р-фрагментов.

В одном варианте осуществления, изображенном на фиг.11b, коды FEC также формируются для Р-фрагментов, которые, наиболее вероятно, вызовут визуальные артефакты, если будут потеряны. В этом варианте осуществления коды FEC 1105 обеспечивают коды исправления ошибок для первых 3 Р-фрагментов, но не для

последующих Р-фрагментов. В другом варианте осуществления коды FEC формируются для Р-фрагментов, размер данных которых является наименьшим (которые будут, как правило, самостоятельно выбирать Р-фрагменты, раньше всего встречающиеся после I-фрагмента, защита которых наиболее важна).

5 В другом варианте осуществления, вместо того, чтобы отправлять код FEC с фрагментом, фрагмент передается два раза, причем каждый раз в другом пакете. Если потерян/задержан один пакет, то используется другой пакет.

В одном варианте осуществления, изображенном на фиг.11с, коды 1111 и 1113 FEC формируются для аудио пакетов, 1110 и 1112, соответственно, передаются из службы хостинга одновременно с видео. Особенно важно поддерживать целостность аудио в видеопотоке, потому что искаженное аудио (например, щелканье или шипенье) в результате приводит к особенно неприятному практическому опыту пользователя. Коды FEC способствуют обеспечению представления содержимого аудио в компьютере 415 клиента без искажения.

15 В другом варианте осуществления, вместо того, чтобы отправлять код FEC с аудиоданными, аудиоданные передается два раза, причем каждый раз в другом пакете. Если потерян/задержан один пакет, то используется другой пакет.

Кроме того, в одном варианте осуществления, изображенном на фиг.11d, коды 1121 и 1123 FEC используются для команд 1120 и 1122 ввода пользователя, соответственно (например, нажатие клавиши), передаваемых в восходящем направлении из клиента 415 в службу 210 хостинга. Это важно, потому что потеря нажатия клавиши или движения мыши в видеоигре или приложении могут в результате привести к неприятному практическому опыту пользователя.

В другом варианте осуществления, вместо того, чтобы отправлять код FEC с данными команды ввода пользователя, данные команды ввода пользователя передаются два 25 раза, причем каждый раз в другом пакете. Если потерян/задержан один пакет, то используется другой пакет.

В одном варианте осуществления служба 210 хостинга оценивает качество канала связи с клиентом 415 для определения того, использовать ли FEC, и если да, то какие 30 части видео, аудио и команд пользователя с какими FEC следует применять. Оценка "качества" канала может включать в себя такие функции, как оценка потери пакетов, времени ожидания и т.д., как описано выше. Если канал является особенно ненадежным, то служба 210 хостинга может применять FEC ко всем I-фрагментам, Р-фрагментам, аудио и командам пользователя. Напротив, если канал является надежным, то служба 35 210 хостинга может применять FEC только к аудио и командам пользователя, или может не применять FEC ни к аудио ни к видео, или может совсем не использовать FEC. Могут применяться различные другие перестановки применения FEC, хотя, по-прежнему, с выполнением этих лежащих в основе принципов. В одном варианте осуществления, служба 210 хостинга непрерывно осуществляет текущий контроль условий на канале 40 и соответственно изменяет политику FEC.

В другом варианте осуществления, согласно фиг.4а и фиг.4b, когда пакет потерян/задержан, что в результате приводит к потере данных фрагмента, или если, возможно, из-за особенно большой потери пакетов, FEC не может исправить потерянные данные фрагмента, то клиент 415 оценивает, сколько кадров остается до того, как новый I-фрагмент будет принят, и сравнивает это количество со временем передачи сигнала 45 туда и обратно, от клиента 415 до службы 210 хостинга. Если время передачи сигнала туда и обратно меньше, чем количество кадров, оставшееся до ожидаемого поступления нового I-фрагмента, то клиент 415 отправляет сообщение в службу по размещению 210

информации на сервере с запросом нового I-фрагмента. Это сообщение маршрутизируется в устройство 404 сжатия видео, и вместо формирования Р-фрагмента для фрагмента, данные которого были потеряны, оно формирует I-фрагмент. С учетом того, что система, изображенная на фиг.4а и фиг.4b, предназначена для обеспечения времени передачи сигнала туда и обратно, которое, как правило, меньше 80 мс, в результате получим, что фрагмент исправляется в пределах 80 мс (при 60 кадр/с, продолжительность кадров равна 16,67 мс, соответственно, в полных периодах кадра, время ожидания 80 мс в результате приводит к исправлению фрагмента в пределах 83,33 мс, которые равны 5 периодам кадра - заметный разрыв, но намного менее заметный, чем, например, разрыв 250 мс для 15 кадров). Когда устройство 404 сжатия формирует такой I-фрагмент вне его обычного циклического порядка, если I-фрагмент вызывает то, что полоса пропускания этого кадра превышает доступную полосу пропускания, то устройство 404 сжатия задерживает циклы других фрагментов так, что другие фрагменты принимают Р-фрагменты в течение этого периода кадра (даже если один фрагмент обычно должен принимать I-фрагмент в течение этого кадра), и далее, начиная со следующего кадра, продолжается обычное циклическое повторение, и фрагмент, который обычно принимал I-фрагмент в предыдущем кадре, принимает I-фрагмент. Несмотря на то, что это действие на короткое время задерживает фазу циклического повторения R-кадра, визуально это обычно не заметно.

Реализация устройства сжатия/устройства восстановления сжатого видео и аудио

На фиг.12 изображен один конкретный вариант осуществления, в котором используется многоядерный процессор и/или мультипроцессор 1200 для сжатия 8 фрагментов параллельно. В одном варианте осуществления, используется компьютерная система с четырехъядерным CPU Xeon, сдвоенный процессор, функционирующая с тактовой частотой 2,66 ГГц или выше, причем каждое ядро реализует устройство сжатия H.264 ×264 с открытым исходным текстом как независимый процесс. Однако, могут использоваться различные другие конфигурации аппаратного/программного обеспечения, хотя, по-прежнему, с выполнением этих лежащих в основе принципов. Например, каждое из ядер CPU может быть заменено устройством сжатия H.264, реализованным в FPGA. В примере, изображенном на фиг.12, ядра 1201-1208 используются для одновременной обработки I-фрагментов и Р-фрагментов как восемь независимых потоков. Как известно в данной области техники, современные многоядерные и мультипроцессорные компьютерные системы по своей природе могут осуществлять многопоточность при интеграции с многопоточными операционными системами, например, Microsoft Windows XP Professional Edition (или 64-разрядная или 32-разрядная версия) и Linux.

В варианте осуществления, изображенном на фиг.12, так как каждое из этих 8 ядер отвечает только за один фрагмент, то оно функционирует в значительной степени независимо от других ядер, причем каждое исполняет отдельный экземпляр ×264. Карта захвата DVI, на основе PCI Express ×1, например, Sendero Video Imaging IP Development Board от Microtronix, Oosterhout, Нидерланды, используется для захвата несжатого видео с разрешающей способностью 640×480, 800×600 или 1280×720, и FPGA на карте использует прямой доступ к памяти (DMA) для передачи захваченного видео через DVI в системную RAM. Фрагменты компоуются в компоновку 1205 4×2 (несмотря на то, что они изображены как квадратные фрагменты, в этом варианте осуществления они имеют разрешающую способность 160×240). Каждый экземпляр ×264 сконфигурирован для сжатия одного из 8 фрагментов 160×240, и они синхронизированы так, что после начального сжатия I-фрагмента каждое ядро входит в цикл, причем каждый кадр не

совпадает по фазе с другим, для сжатия одного I-фрагмента, за которым следуют семь P-фрагментов, и изображенный на фиг.12.

Каждый период кадра, получающиеся в результате сжатые фрагменты объединяются в поток пакетов с использованием способов, описанных ранее, и после этого сжатые
5 фрагменты передаются в целевой клиент 415.

Несмотря на то, что не изображено на фиг.12, если скорость передачи данных объединенных 8 фрагментов превышает заданную пиковую скорость 941 передачи данных, то все 8 процессов x264 откладываются на такое количество периодов кадра, которое необходимо до тех пор, пока все данные упомянутых объединенных 8
10 фрагментов не смогут быть переданы.

В одном варианте осуществления клиент 415 реализован в виде программного обеспечения на PC, на котором исполняются 8 экземпляров FFmpeg. Принимающий процесс принимает 8 фрагментов, и каждый фрагмент маршрутизируется в экземпляр FFmpeg, который восстанавливает сжатый фрагмент и визуализирует его в
15 соответствующем месте расположения фрагмента на дисплее 422.

Клиент 415 принимает ввод клавиатуры, мыши или игрового контроллера из драйверов устройства ввода PC и передает его в сервер 402. Сервер 402 далее применяет принятые данные устройства ввода и применяет их к игре или приложению, исполняющемуся на сервере 402, который является PC, работающим по управлением
20 Windows с использованием двухъядерного (Core Duo) CPU Intel 2,16 ГГц. Сервер 402 далее выводит новый кадр и выводит его через свой выход DVI, или из расположенной на системной плате графической системы или через выход DVI платы PCI 8800GTX NVIDIA.

Одновременно сервер 402 выводит аудио, выводимое игрой или приложениями, через
25 свой цифровой аудиовыход (например, S/PDIF), который соединен с цифровым аудиовходом на sdвоенном четырехъядерном PC на основе Xeon, который реализует сжатие видео. Устройство сжатия аудио с открытым исходным текстом Vorbis используется для сжатия аудио одновременно с видео с использованием любого ядра, доступного для потока обработки. В одном варианте осуществления ядро, которое
30 выполняет сжатие своего фрагмента, сначала исполняет сжатие аудио. Сжатое аудио далее передается вместе со сжатым видео и восстанавливается в клиенте 415 с использованием устройства восстановления сжатого аудио Vorbis.

Распределение серверного центра службы хостинга

Свет через стекло, например светопровод, проходит со скоростью, равной некоторой
35 доле от скорости света в вакууме, и, соответственно, может быть определена точная скорость распространения света в светопроводе. Но, на практике, с учетом времени из-за задержек маршрутизации, неэффективности передач и других потерь, авторы наблюдали, что оптимальные величины времени ожидания в Internet отражают скорости передачи ближе к 50% от скорости света. Соответственно, оптимальное время передачи
40 сигнала туда и обратно на расстояние 1000 миль (1600 км) равно приблизительно 22 мс, и оптимальное время передачи сигнала туда и обратно на расстояние 3000 миль (4800 км) равно приблизительно 64 мс. Соответственно, один сервер, находящийся на одном побережье США, находится слишком далеко для обслуживания клиентов на другом побережье (которое может находиться на расстоянии 3000 миль (4800 км)) с
45 требуемым временем ожидания. Однако, как изображено на фиг.13а, если серверный центр 1300 службы 210 хостинга расположен в центре США (например, в Канзасе, в Небраске и т.д.) так, что расстояние до любой точки в континентальных США равно приблизительно 1500 миль (2400 км) или меньше, то время передачи сигнала туда и

обратно по Internet может быть равно только 32 мс. Согласно фиг.4b, отметим, что, несмотря на то что величины времени ожидания в самом плохом случае, допустимые для (соединения) 453 пользователь ISP, равны 25 мс, как правило, авторы наблюдали величины времени ожидания ближе к 10-15 мс с системами кабельной модемной связи и DSL. Кроме того, согласно фиг.4b предполагается, что максимальное расстояние от территории 211 пользователя до хостинг-центра 210 равно 1000 миль (1600км).

Соответственно, при типичном используемом времени передачи сигнала туда и обратно (по соединению) пользователь ISP 15 мс и максимальном расстоянии через Internet 1500 миль (2400 км) для времени передачи сигнала туда и обратно 32мс, общее время передачи сигнала туда и обратно от момента, когда пользователь приводит в действие устройство 421 ввода и наблюдает ответ на дисплее 422 равно $1+1+15+32+1+16+6+8=80$ мс.

Соответственно, (при передаче) по Internet на расстояние 1500 миль (2400 км), как правило, может получаться время ответа 80 мс. Это может обеспечить любую территорию пользователя достаточно коротким временем 453 ожидания (для соединения) пользователь ISP в континентальных США для доступа к одному серверному центру, который расположен в центре.

В другом варианте осуществления, изображенном на фиг.13b, серверные центры, HS1-HS6, службы 210 хостинга стратегически размещены по Соединенным Штатам (или другому географическому региону), причем определенные большие серверные центры службы хостинга расположены близко к центрам с высокой плотностью населения (например, HS2 и HS5). В одном варианте осуществления серверные центры HS1-HS6 обмениваются информацией через сеть 1301, которая может быть Internet, или частной сетью, или комбинацией обоих. В отношении множества серверных центров, услуги могут оказываться с меньшим временем ожидания для пользователей, у которых время 453 ожидания ISP пользователя является большим.

Несмотря на то, что расстояние по Internet определено является фактором, который увеличивает время передачи сигнала туда и обратно через Internet, иногда играют роль другие факторы, которые в основном не связаны со временем ожидания. Иногда поток пакетов маршрутизируется через Internet в удаленное местоположение и обратно, что в результате приводит ко времени ожидания из-за длинной линии связи. Иногда оборудование для маршрутизации, находящееся на пути (прохождения сигнала) не работает должным образом, что в результате приводит к задержке передачи. Иногда путь перегружен трафиком, что вводит задержку. И, иногда, имеет место отказ, который не дает возможность ISP пользователя осуществлять маршрутизацию в данный пункт назначения. Соответственно, несмотря на то, что общий Internet обычно обеспечивает соединения из одной точки в другую с довольно надежным и оптимальным маршрутом и временем ожидания, которое в значительной степени определяется расстоянием, (особенно в соединениях дальней связи, которые в результате приводят к маршрутизации за пределами локальной зоны пользователя), такие надежность и время ожидания ни в коем случае не гарантируется и часто не могут быть получены из территории пользователя в данный пункт назначения в общем Internet.

В одном варианте осуществления, когда клиент 415 пользователя вначале соединяется со службой 210 хостинга для воспроизведения видеоигры или использования приложения, клиент осуществляет связь с каждым из серверных центров HS1-HS6 службы хостинга, доступных после запуска (например, с использованием способов, описанных выше). Если время ожидания является достаточно малым для конкретного соединения, то используется это соединение. В одном варианте осуществления клиент осуществляет связь со всеми серверными центрами службы хостинга или их подмножеством, причем

выбирается серверный центр с соединением с самым малым временем ожидания. Клиент может выбирать центр обслуживания с соединением с самым малым временем ожидания, или центры обслуживания могут идентифицировать центр обслуживания с соединением с самым малым временем ожидания и обеспечивать эту информацию (например, в виде

5 адреса Internet) в клиент.

Если конкретный серверный центр службы хостинга перегружен и/или для приложения или игры пользователя время ожидания для соединения с другим, менее загруженным серверным центром службы хостинга является допустимым, то клиент 415 может быть перенаправлен в другой серверный центр службы хостинга. В такой ситуации игра или

10 приложение, исполняемые пользователем, приостанавливаются на сервере 402 в перегруженном серверном центре пользователя, и данные о состоянии приложения или игры передаются в сервер 402, находящийся в другом серверном центре службы хостинга. После этого игра или приложение возобновляются. В одном варианте осуществления служба 210 хостинга ожидает до тех пор, пока игра или приложение не достигнут

15 естественной точки приостановки (например, между уровнями в игре, или после того, как пользователь инициирует операцию "сохранение" в приложении), для упомянутой передачи. В еще одном варианте осуществления служба 210 хостинга ожидает до тех пор, пока активность пользователя не прекратится на заданный период времени (например, на 1 минуту), и тогда инициирует упомянутую передачу в это время.

20 Как описано выше, в одном варианте осуществления, служба 210 хостинга абонирует услугу 440 по обходу Internet по фиг.14 с целью обеспечения своим клиентам гарантируемого времени ожидания. Услуги по обходу Internet, как используется в этом описании, являются услугами, которые обеспечивают маршруты через частную сеть от одной точки до другой (точки) в Internet с гарантируемыми характеристиками

25 (например, время ожидания, скорость передачи данных и т.д.). Например, если служба 210 хостинга принимает большой объем трафика от пользователей, использующих услугу DSL AT&T, предлагаемую в Сан-Франциско, то вместо маршрутизации в находящиеся в Сан-Франциско центральные АТС AT&T служба 210 хостинга может арендовать частное информационное соединение с большой пропускной способностью

30 у провайдера услуг (возможно, у компании AT&T непосредственно или у другого провайдера) между находящимися в Сан-Франциско центральными АТС и одним или несколькими серверными центрами для службы 210 хостинга. Тогда, если маршруты от всех серверных центров HS1-HS6 службы хостинга через общий Internet до пользователя, находящегося в Сан-Франциско, с использованием DSL AT&T в результате

35 приводят к слишком большому времени ожидания, то вместо них может быть использовано частное информационное соединение. Несмотря на то, что частные информационные соединения, как правило, являются более дорогими, чем маршруты через общий Internet, до тех пор, пока они составляют небольшой процент от соединений службы 210 хостинга с пользователями, общее влияние на стоимость будет оставаться

40 на низком уровне, и пользователи будут испытывать более постоянный практический опыт обслуживания.

Серверные центры часто имеют два уровня резервного питания в случае отказа в системе питания. Первый уровень, как правило, является резервным питанием от батарей (или от альтернативного, находящегося в готовности к немедленному использованию,

45 источника энергии, такой маховик, который поддерживается в рабочем состоянии и подсоединен к генератору), которое немедленно обеспечивает питание, когда происходит отказ в питающей линии, и поддерживает в рабочем состоянии серверный центр. Если отказ в системе питания является непродолжительным, и происходит быстрый возврат

питающей линии (например, в течение одной минуты), то батареи - это все, что необходимо для поддержания серверного центра в рабочем состоянии. Но если отказ в системе питания происходит в течение более длительного периода времени, то, как правило, запускают генераторы (например, дизельные), которые принимают на себя функции батарей и могут работать, пока у них есть топливо. Такие генераторы являются чрезвычайно дорогими, так как они должны быть способны выводить такую мощность, которую обычно получает серверный центр из питающей линии.

В одном варианте осуществления каждая из служб HS1-HS5 хостинга совместно использует пользовательские данные так, что, если в одном серверном центре происходит отказ в системе питания, то он может приостановить игры и приложения, которые находятся в процессе (исполнения), и затем передает данные о состоянии приложения или игры из каждого сервера 402 в серверы 402, находящиеся в других серверных центрах, и затем уведомят клиента 415 каждого пользователя и предписывает, чтобы он передавал информацию в новый сервер 402. С учетом того, что такие ситуации происходят редко, может оказаться приемлемой передача пользователя серверному центру службы хостинга, который не может обеспечивать оптимальное время ожидания (т.е. пользователь просто должен будет мириться с более высоким временем ожидания в течение отказа в системе питания), что обеспечивает гораздо более широкие возможности для передачи пользователей. Например, с учетом разных часовых поясов в США, пользователи на Восточном побережье могут ложиться спать в 23:30, в то время как пользователи на Западном Побережье в 20:30 начинают максимально использовать видеоигры. Если в это время происходит отказ в системе питания в серверном центре службы хостинга, находящемся на Западном Побережье, то, возможно, нет достаточного количества серверов 402, находящихся на Западном Побережье, в других серверных центрах службы хостинга для управления всеми пользователями. В такой ситуации некоторые пользователи могут быть переданы в серверные центры службы хостинга, находящиеся на Восточном побережье, в которых существуют доступные серверы 402, и единственным последствием для пользователей будет большее время ожидания. После передачи пользователей из серверного центра, который остался без электричества, серверный центр может начать в должном порядке отключение своих серверов и оборудования так, чтобы все оборудование было отключено до разрядки батарей (или другого немедленно подключаемого резервного питания). Следовательно, серверный центр может избежать затрат на генератор.

В одном варианте осуществления во время периодов интенсивной нагрузки служб 210 хостинга (или вследствие пиковой пользовательской нагрузки, или вследствие отказа одного или нескольких серверных центров) пользователей передают в другие серверные центры на основе требований к времени ожидания игры или приложения, которые они используют. Соответственно, пользователям, использующим игры или приложения, которые требуют малого времени ожидания, отдается предпочтение (при предоставлении) доступных соединений с серверами с малым временем ожидания при ограниченном питании.

Признаки службы хостинга

На фиг.15 изображен вариант осуществления компонентов серверного центра для службы 210 хостинга, используемый в нижеследующем описании признаков. Как и в случае со службой 210 хостинга, изображенной на фиг.2а, компонентами этого серверного центра управляет и (их работу) координирует система 401 управления службы 210 хостинга, если не оговорено иное.

Входящий трафик 1501 в internet из клиентов 415 пользователя направляется во

входящую маршрутизацию 1502. Как правило, входящий трафик 1501 в internet входит в серверный центр через высокоскоростное оптоволоконное соединение с Internet, но будет достаточно любых средств сетевого соединения с отвечающими требованиям полосой пропускания, надежностью и малым временем ожидания. Входящая маршрутизация 1502 является системой сетевых (сеть может быть реализована как сеть Ethernet, сеть волоконно-оптических каналов или посредством любых других транспортных средств) коммутаторов и серверов маршрутизации, поддерживающих коммутаторы, которая принимает поступающие пакеты и маршрутизирует каждый пакет в соответствующий сервер 1521-1525 приложения/игры ("app/game"). В одном варианте осуществления пакет, который доставляется на конкретный сервер приложения/игры, представляет подмножество данных, принимаемых из клиента, и/или может быть преобразован/изменен другими сетевыми компонентами (например, такими сетевыми компонентами, как шлюзы и маршрутизаторы) в центре обработки и хранения данных. В некоторых случаях, пакеты маршрутизируются в несколько серверов 1521-1525 одновременно, например, если игра или приложение исполняются параллельно одновременно на множестве серверов. Дисковый массив 1511-1512 типа RAID соединен с сетью 1502 с входящей маршрутизацией так, что серверы 1521-1525 приложения/игры могут считывать с дисковых массивов 1511-1512 типа RAID и записывать на них. Кроме того, дисковый массив 1515 типа RAID (который может быть реализован как множество дисковых массивов типа RAID) также соединен с входящей маршрутизацией 1502, и данные из дискового массива 1515 типа RAID могут быть считаны из серверов 1521-1525 приложения/игры. Входящая маршрутизация 1502 может быть реализована в широком диапазоне архитектур сети предшествующего уровня техники, включающих в себя древовидную структуру коммутаторов, с входящим трафиком 1501 в internet в его корне, в ячеистой структуре, соединяющей все различные устройства, или как набор взаимосвязанных подсетей, с трафиком, сосредоточенным между (устройствами) внутренней связи, отделенным от трафика, сосредоточенного среди других устройств. Одним типом конфигурации сети является SAN, которая, несмотря на то, что она, как правило, используется для запоминающих устройств, она также может использоваться для обычной высокоскоростной передачи данных между устройствами. Кроме того, серверы 1521-1525 приложения/игра каждый может иметь множество сетевых соединений во входящую маршрутизацию 1502. Например, сервер 1521-1525 может иметь сетевое соединение с подсетью, подсоединенной к дисковым массивам 1511-1512 типа RAID, и другое сетевое соединение с подсетью, подсоединенной к другим устройствам.

Серверы 1521-1525 приложения/игры могут все быть сконфигурированы одинаково, некоторые - по-разному, или все - по-разному, как описано ранее в отношении серверов 402 в варианте осуществления, изображенном на фиг.4а. В одном варианте осуществления каждый пользователь при использовании службы хостинга обычно (использует), по меньшей мере, один сервер 1521-1525 приложения/игры. Для простоты объяснения предположим, что данный пользователь использует сервер 1521 приложения/игры, но множество серверов могут использоваться одним пользователем, и множество пользователей могут совместно использовать один сервер 1521-1525 приложения/игры. Ввод сигнала управления пользователя, отправленный из клиента 415, как описано ранее, принимается как входящий трафик 1501 в Internet и маршрутизируется через входящую маршрутизацию 1502 в сервер 1521 приложения/игры. Сервер 1521 приложения/игры использует ввод сигнала управления пользователя как ввод сигнала управления в игру или приложение, исполняющиеся на сервере, и вычисляет следующий

кадр видео и аудио, связанного с ним. Сервер 1521 приложения/игры после этого выводит несжатое видео/аудио 1529 в совместно используемое сжатие 1530 видео. Сервер приложения/игры может выводить несжатое видео посредством любых средств, включающих в себя одно или несколько соединений по технологии Gigabit Ethernet, но в одном варианте осуществления видео выводится через соединение DVI, а аудио и другие сжатые данные и информация о состоянии канала связи выводятся через соединение универсальной последовательной шины (USB).

Совместно используемое сжатие 1530 видео сжимает несжатое видео и аудио из серверов 1521-1525 приложения/игры. Сжатие может быть реализовано полностью в аппаратном обеспечении, или в аппаратном обеспечении, исполняющем программное обеспечение. Для каждого сервера 1521-1525 приложения/игры может (существовать) выделенное устройство сжатия, или если устройства сжатия являются достаточно быстрыми, то данное устройство сжатия можно использовать для сжатия видео/аудио из нескольких серверов 1521-1525 приложения/игры. Например, при 60 кадр/с время видеокadra равно 16,67 мс. Если устройство сжатия может сжимать один кадр в 1 мс, то это устройство сжатия можно использовать для сжатия видео/аудио из 16 серверов 1521-1525 приложения/игры посредством приема ввода из одного сервера за другим, причем устройство сжатия сохраняет состояние каждого процесса сжатия видео/аудио и переключает контекст по мере того, как оно циклически переключается между потоками видео/аудио из серверов. Это в результате приводит к существенной экономии затрат на аппаратное обеспечение для сжатия. Так как разные серверы будут завершать кадры в разное время, в одном варианте осуществления, ресурсы устройства сжатия находятся в совместно используемом пуле 1530 с совместно используемым средством хранения (например, RAM, флэш-память) для хранения состояния каждого процесса сжатия, и когда кадр сервера 1521-1525 закончен и готов для сжатия, средство управления определяет, какой ресурс сжатия является доступным в этот момент, обеспечивает ресурс сжатия состоянием процесса сжатия сервера и кадром несжатого видео/аудио для сжатия.

Отметим, что часть состояния процесса сжатия каждого сервера включает в себя информацию о самом сжатии, например, восстановленные данные буфера кадра предыдущего кадра, которые могут использоваться как опорный элемент для R-фрагментов, разрешающая способность видеовыхода, качество сжатия, мозаичная структура, выделение битов для каждого фрагмента, качество сжатия, аудиоформат (например, стерео, объемный звук, AC Dolby® 3). Но состояние процесса сжатия также включает в себя информацию о состоянии канала связи относительно пиковой скорости 941 передачи данных и того, выводится ли в настоящее время предыдущий кадр (как изображено на фиг.9b) (и в результате текущий кадр должен быть проигнорирован), и, возможно, того, существуют ли характеристики канала, которые должны учитываться при сжатии, например, чрезмерная потеря пакетов, которые влияют на принятие решений в отношении сжатия (например, с точки зрения частоты I-фрагментов и т.д.). По мере того, как пиковая скорость 941 передачи данных или другие характеристики канала изменяются со временем, как определяется сервером 1521-1525 приложения/игры, поддерживающим данные текущего контроля каждого пользователя, отправляемые из клиента 415, сервер 1521-1525 приложения/игры отправляет соответствующую информацию в совместно используемое аппаратное сжатие 1530.

Совместно используемое аппаратное сжатие 1530 также разбивает на пакеты сжатое видео/аудио с использованием средств, например, средств, описанных ранее, и если требуется, то с применением кодов FEC, дублированием определенных данных или

принятием других мер для соответствующего обеспечения возможности принятия клиентом 415 потока видео/аудио данных и восстановления сжатых данных с максимально возможным качеством и надежностью.

Некоторые приложения, например, описанные ниже, требуют, чтобы видео/аудиовыход данного сервера 1521-1525 приложения/игры был доступным со многими разрешающими способностями (или в других разнообразных форматах) одновременно. Если сервер 1521-1525 приложения/игры так уведомляет ресурс совместно используемого аппаратного сжатия 1530, то несжатые видео аудио 1529 этого сервера 1521-1525 приложения/игры будут одновременно сжаты в разных форматах, с разными разрешающими способностями и/или с разными структурами исправления ошибок/пакетов. В некоторых случаях, некоторые ресурсы сжатия могут совместно использоваться множеством процессов сжатия, сжимающих идентичное видео/аудио (например, во многих алгоритмах сжатия, существует этап, на котором изображение масштабируется во множество размеров перед применением сжатия. Если требуется вывод изображений разных размеров, то этот этап может использоваться для обслуживания нескольких процессов сжатия одновременно. В других случаях для каждого формата требуются отдельные ресурсы сжатия. В любом случае, сжатое видео/аудио 1539 со всеми требуемыми различными разрешающими способностями и во всех требуемых различных форматах для данного сервера 1521-1525 приложения/игр (будь то один или много) выводится одновременно в исходящую маршрутизацию 1540. В одном варианте осуществления вывод сжатого видео/аудио 1539 осуществляется в формате UDP, соответственно, он является однонаправленным потоком пакетов.

Сеть 1540 исходящей маршрутизации содержит набор коммутаторов и серверов маршрутизации, которые направляют каждый поток сжатого видео/аудио пользователю (ям), которому(ым) он предназначен, или в другие пункты назначения через интерфейс исходящего трафика 1599 из Internet (которые, как правило, подключены к волоконно-оптическому интерфейсу с Internet), и/или обратно в буфер 1515 задержки, и/или обратно во входящую маршрутизацию 1502, и/или через частную сеть (не изображена) для распределения видео. Отметим, что (как описывается ниже) исходящая маршрутизация 1540 может выводить данный видео/аудио поток во множество пунктов назначения одновременно. В одном варианте осуществления это реализовано с использованием групповой передачи по протоколу Internet (IP), в которой передается данный поток UDP, предназначенный для передачи потоком в множество пунктов назначения одновременно, и это ширококестание повторяется коммутаторами и серверами маршрутизации в исходящей маршрутизации 1540. Множеством пунктов назначения ширококестания могут быть клиенты 415 множества пользователей, (доступные) через Internet, множество серверов 1521-1525 приложения/игры, (доступные) через входящую маршрутизацию 1502, и/или один или несколько буферов 1515 задержки. Соответственно, выход данного сервера 1521-1522 сжимают в один или множество форматов, и каждый сжатый поток направляется в один или множество пунктов назначения.

Кроме того, в другом варианте осуществления, если множество серверов 1521-1525 приложения/игры одновременно используются одним пользователем (например, в конфигурации параллельной обработки для создания 3D вывода сложной сцены) и каждый сервер выводит часть получающегося в результате изображения, то видеовыход из множества серверов 1521-1525 может объединяться совместно используемым аппаратным сжатием 1530 в комбинированный кадр, и с этого момента и далее он обрабатывается так, как описано выше, как если бы он выходил из одного сервера 1521-1525 приложения/игры.

Отметим, что в одном варианте осуществления, копия (по меньшей мере, с разрешающей способностью видео, просматриваемого пользователем, или выше) всего видео, формируемого серверами 1521-1525 приложения/игры, записывается в буфер 1515 задержки, по меньшей мере, на некоторое количество минут (15 минут в одном варианте осуществления). Это обеспечивает возможность каждому пользователю "перематывать" видео из каждого сеанса для просмотра предыдущих действия или деяний (в случае игры). Соответственно, в одном варианте осуществления, в отношении каждого выходного 1539 потока сжатого видео/аудио, маршрутизируемого в клиент 415 пользователя, также осуществляется групповая передача в буфер 1515 задержки. Когда видео/аудио сохраняется в буфере 1515 задержки, каталог в буфере 1515 задержки обеспечивает перекрестную ссылку между сетевым адресом сервера 1521-1525 приложения/игры, который является источником задержанного видео/аудио, и местом в буфере 1515 задержки, где задержанное видео/аудио может быть найдено.

Игры в реальном масштабе времени, немедленно выводимые на экран, с возможностью немедленного ведения игры

Серверы 1521-1525 приложения/игры могут использоваться не только для исполнения данного приложения или видеоигры для пользователя, но они также могут использоваться для создания интерфейсных приложений пользователя для службы 210 хостинга, которые поддерживают навигацию в службе 210 хостинга и другие признаки. Моментальный снимок экрана одного такого интерфейсного приложения пользователя изображен на фиг.16, экран "Game Finder" ("средство поиска игры"). Этот конкретный экран интерфейса пользователя обеспечивает возможность пользователю смотреть 15 игр, которые ведутся в реальном масштабе времени (или отложены) другими пользователями. Каждое из "миниатюрных" видеоокон, например 1600, является окном с видео в реальном масштабе времени в движении, показывающим одно видео из игры одного пользователя. Вид, показанный в миниатюре, может быть видом, идентичным тому, который видит пользователь, или он может быть задержанным видом (например, если пользователь ведет боевую игру, то пользователю может потребоваться, чтобы другие пользователи не видели, где он скрывается, и он может решить задержать какой-либо вид его геймплей на некоторый период времени, скажем, на 10 минут). Вид также может быть полем зрения камеры игры, который отличается от вида любого пользователя. Посредством выбора пунктов меню (не изображено в этом примере) пользователь может отобразить выборку игр для одновременного просмотра, исходя из различных критериев. Как небольшое осуществление выборки иллюстративных отборов, пользователь может выбрать случайную выборку игр (например, изображенную на фиг.16), все игры одного вида (все, которые ведут разные игроки), только высоко котирующиеся игроки игры, игроки на данном уровне в игре, или низко котирующиеся игроки (например, если игрок изучает основы), игроки, которые являются "приятелями" (или соперниками), игры, у которых наибольшее количество зрителей и т.д.

Отметим, что, в общем, каждый пользователь принимает решение, могут ли другие смотреть видео его игры или приложения, и, если это так, то кто из них и когда может его смотреть, можно ли его смотреть только с задержкой.

Сервер 1521-1525 приложения/игры, который формирует экран интерфейса пользователя, изображенный на фиг.16, запрашивает 15 линий передач видео/аудио посредством отправки сообщения в сервер 1521-1525 приложения/игры для каждого пользователя, у которого он запрашивает игру. Сообщение отправляется через входящую маршрутизацию 1502 или другую сеть. Сообщение включает в себя размер и формат запрашиваемого видео/аудио и идентифицирует пользователя,

просматривающего экран интерфейса пользователя. Данный пользователь может решить выбрать режим "privacy" ("конфиденциальность") и не обеспечить возможность каким-либо другим пользователям просматривать видео/аудио его игры (или с его точки обзора или с другой точки обзора), или как описано в предыдущем абзаце, пользователь может решить обеспечить возможность просмотра видео/аудио из его игры, но задержать просматриваемое видео/аудио. Сервер 1521-1525 приложения/игры пользователя, принимающий запрос и соглашающийся обеспечить возможность просмотра его видео/аудио, отправляет такое подтверждение в запрашивающий сервер, и он также уведомляет совместно используемое аппаратное сжатие 1530 о необходимости формирования дополнительного потока сжатого видео в запрашиваемом формате или с запрашиваемым размером экрана (при предположении, что формат и размер экрана отличаются от тех, которые уже формируются), и он также указывает пункт назначения для сжатого видео (т.е. запрашивающий сервер). Если запрашиваемое видео/аудио только задержано, то запрашивающий сервер 1521-1525 приложения/игры уведомляют об этом, и он запрашивает задержанное видео/аудио из буфера 1515 задержки посредством поиска места расположения видео/аудио в каталоге, находящемся в буфере 1515 задержки, и сетевого адреса сервера 1521-1525 приложения/игры, который является источником задержанного видео/аудио. После формирования и обработки всех этих запросов, до 15 потоков видео миниатюрного размера в масштабе реального времени маршрутизируются из исходящей маршрутизации 1540 во входящую маршрутизацию 1502 в сервер 1521-1525 приложения/игры, формирующий экран интерфейса пользователя, и восстанавливаются и выводятся на экран сервером. Задержанные видео/аудио потоки могут быть со слишком большим размером экрана, и если это так, то сервер 1521-1525 приложения/игры восстановит сжатые потоки и уменьшит масштаб видеопотоков до размера миниатюры. В одном варианте осуществления запросы на аудио/видео отправляют (и она ими управляет) в центральную службу "управления", подобную системе управления службы хостинга по фиг.4а (не изображена на фиг.15), которая далее перенаправляет эти запросы в соответствующий сервер 1521-1525 приложения/игры. Кроме того, в одном варианте осуществления, может не требоваться запроса, потому что миниатюры "проталкиваются" в клиенты тех пользователей, которые дают разрешение на него.

Аудио из 15 игр, все смикшированное одновременно, может создать какофонию звуков. Пользователь может решить смикшировать все звуки вместе таким способом (возможно только для получения ощущения "шума", создаваемого всем просматриваемым действием), или пользователь может решить прослушивать только аудио из одной игры в данный момент времени. Выбор одной игры выполняют посредством перемещения желтой рамки 1601 выбора фрагмента в данную игру (перемещение желтой рамки может выполняться при использовании клавиш курсора на клавиатуре, при перемещении мыши, при перемещении джойстика или при нажатии на кнопки направления на другом устройстве, например, на мобильном телефоне). После выбора одной игры, из этой игры воспроизводится только аудио. Кроме того, показывается информация 1602 об игре. В случае этой игры, например, логотип издателя ("EA") и логотип игры "Need for Speed Carbon" ("Жажда скорости"), и оранжевая горизонтальная полоса указывает в относительном выражении количество людей, ведущих эту игру или наблюдающих за ней в этот конкретный момент (много, в этом случае, поэтому игра является "горячей"). Кроме того обеспечены "Статистические данные" ("Stats") с указанием того, что существуют 145 игроков, активно ведущих 80 разных экземпляров Need for Speed Game (т.е. она может вестись или как игра отдельного

игрока или как игра с несколькими участниками), и существует 680 зрителей (одним из которых является этот пользователь). Отметим, что эти статистические данные (и другие статистические данные) собираются системой 401 управления службы хостинга и сохраняются в дисковых массивах 1511-1512 типа RAID для ведения журналов

5 регистрации работы службы 210 хостинга и для надлежащего биллинга пользователей и выплат издателям, которые обеспечивают содержимое. Некоторые статистические данные записываются вследствие действий системы 401 управления службы, и некоторые сообщаются в систему 401 управления службы отдельным сервером 1521-1525

10 приложения/игры. Например, сервер 1521-1525 приложения/игры, исполняющий это приложение Game Finder (средство поиска игры), отправляет сообщения в систему 401 управления службы хостинга, когда игры просматриваются (и когда их прекращают для просмотра), чтобы она могла обновить статистические данные о том, сколько игр отображается на экране). Некоторые статистические данные доступны для интерфейсных приложений пользователя, например, для этого приложения Game Finder (средство

15 поиска игры).

Если пользователь щелкает по кнопке активации на своем устройстве ввода, то он видит, что миниатюрное видео в желтой раме увеличивается, в то время как оно остается в режиме реального времени, до размера во весь экран. Это воздействие изображено в процессе на фиг.17. Отметим, что видеоокно 1700 увеличилось в размере. Для

20 реализации этого воздействия, сервер 1521-1525 приложения/игры запрашивает сервер 1521-1525 приложения/игры, исполняющий выбранную игру, для получения копии видеопотока для размера во весь экран (с разрешающей способностью дисплея 422 пользователя) игры, маршрутизируемой в него. Сервер 1521-1525 приложения/игры, исполняющий игру, уведомляет совместно используемое аппаратное устройство 1530

25 сжатия, что копия миниатюрного размера игры больше не требуется (если другой сервер 1521-1525 приложения/игры не требует такую миниатюру), и после этого он предписывает, чтобы он отправлял копию видео размера во весь экран в сервер 1521-1525 приложения/игры с распакованием видео. Пользователь, ведущий игру, может иметь или может не иметь дисплей 422 с разрешающей способностью, идентичной

30 разрешающей способности дисплея пользователя, увеличивающего изображение игры. Кроме того, другие зрители игры могут иметь или могут не иметь дисплеи 422 с разрешающей способностью, идентичной разрешающей способности дисплея пользователя, увеличивающего изображение игры (и могут иметь другие средства воспроизведения аудио, например, стерео или объемный звук). Соответственно,

35 совместно используемое аппаратное устройство 1530 сжатия определяет то, формируется ли уже подходящий поток сжатого видео/аудио, который отвечает требованиям пользователя, запрашивающего упомянутый видео/аудио поток, и если он уже существует, то оно уведомляет исходящую маршрутизацию 1540, чтобы она маршрутизировала копию потока в сервер 1521-1525 приложения/игры с изменением

40 масштаба изображения видео, и если (оно) не сжимает другую копию видео, которая подходит для этого пользователя, (то) выдает команду в исходящую маршрутизацию отправить упомянутый поток обратно во входящую маршрутизацию 1502 и сервер 1521-1525 приложения/игры, изменяющий масштаб изображения видео. Этот сервер, теперь принимающий полноэкранную версию выбранного видео, восстанавливает его

45 и постепенно масштабирует его до полного размера.

На фиг.18 изображено то, как экран выглядит после окончательного увеличения масштаба изображения игры до полного экрана, и игра изображена с полной разрешающей способностью дисплея 422 пользователя, как показано посредством

изображения, на которое указывает стрелка 1800. Сервер 1521-1525 приложения/игры, исполняющий приложение средства поиска игры, отправляет сообщения в другие серверы 1521-1525 приложения/игры, которые обеспечивали миниатюры, о том, что они больше не требуются, и сообщения в сервер 401 управления службы хостинга, о том, что другие игры больше не просматриваются. В это момент единственным изображением, которое он формирует, является наложением 1801 другого графического изображения наверху экрана, которое предоставляет информацию и управляющие элементы меню пользователю. Отметим, что по мере продвижения этой игры, аудитория увеличилась до 2 503 зрителей. При таком большом количестве зрителей, должно существовать много зрителей с дисплеями 422, разрешающие способности которых идентичны или близки (каждый сервер 1521-1525 приложения/игры может масштабировать видео для корректировки соответствия).

Поскольку изображенная игра является игрой с несколькими участниками, пользователь может принять решение о присоединении к игре в некоторый момент времени. Служба 210 хостинга может обеспечить или может не обеспечить возможность пользователю присоединиться к игре по разным причинам. Например, пользователь, возможно, должен заплатить, чтобы вести игру, и он решил не платить, пользователь может не иметь достаточного рейтинга, чтобы присоединиться к этой конкретной игре (например, он не может конкурировать с другими игроками), или подключение к Internet пользователя может не иметь достаточно малого времени ожидания для обеспечения возможности пользователю вести игру (например, не существует ограничения по времени ожидания для просмотра игры, соответственно, игру, которую проводят далеко (безусловно, на другом континенте), можно смотреть, не заботясь о времени ожидания, но для того, чтобы вести игру, время ожидания должно быть достаточно мало, чтобы пользователь (а) получал удовольствие от игры и (б) находился в равных условиях с другими игроками, которые могут иметь соединения с меньшим временем ожидания). Если пользователю обеспечена возможность вести игру, то сервер 1521-1525 приложения/игры, который обеспечивает пользователю интерфейс пользователя Game Finder (средства поиска игры), посылает запрос, чтобы сервер 401 управления службы хостинга инициировал (т.е. определил местоположение и запустил) сервер 1521-1525 приложения/игры, который соответственно сконфигурирован для ведения конкретной игры, для загрузки игры из дискового массива 1511-1512 типа RAID, и после этого сервер 401 управления службы хостинга выдает команду во входящую маршрутизацию 1502 для передачи управляющих сигналов от пользователя в игровой сервер приложения/игры, который в настоящее время осуществляет хостинг игры, и он выдает команду в совместно используемое аппаратное сжатие 1530 для переключения от сжатия видео/аудио из сервера приложения/игры, который осуществляет хостинг приложения Game Finder (средства поиска игры) к сжатию видео/аудио из сервера приложения/игры, который в настоящее время осуществляет хостинг игры. Кадровый синхронизирующий импульс службы приложения/игры Game Finder (средство поиска игры) и нового сервера приложения/игры, который осуществляет хостинг игры, не синхронизированы, и в результате, вероятно, существует разность времен между двумя синхронизирующими импульсами. Поскольку совместно используемое аппаратное обеспечение 1530 сжатия видео начнет сжатие видео после того, как сервер 1521-1525 приложения/игры завершает видеокادر, то первый кадр из нового сервера может быть завершен раньше, чем (пройдет) полный период кадра старого сервера, что может произойти до того, как завершится передача предыдущего сжатого кадра (например, рассмотрим период 992 передачи по фиг.9b: если бы несжатый кадр 963 3 был закончен на пол периода кадра

раньше, то он столкнулся бы с периодом 992 передачи). В такой ситуации совместно используемое аппаратное обеспечение 1530 сжатия видео игнорирует первый кадр из нового сервера (например, подобно тому, как игнорируется 974 кадр 964 4), и клиент 415 удерживает последний кадр из старого сервера дополнительный период кадра, и совместно используемое аппаратное обеспечение 1530 сжатия видео начинает сжатие видео следующего периода кадра из нового сервера приложения/игры, который осуществляет хостинг игры. Визуально, для пользователя, переход от одного сервера приложения/игры на другой будет бесперебойным. Сервер 401 управления службы хостинга после этого уведомляет игровой сервер 1521-1525 приложения/игры, который осуществлял хостинг Game Finder (средство поиска игры), для переключения в состояние ожидания до тех пор, пока он не потребуетсся снова.

После этого пользователь может вести игру. Исключительным является то, что создается впечатление, что игру можно вести немедленно (так как она загружается на игровой сервер 1521-1525 приложения/игры из дискового массива 1511-1512 типа RAID со скоростью гигабит/секунда), и игра загружается на сервер, полностью пригодной для игры, вместе с операционной системой, полностью сконфигурированной для игры с идеальными драйверами, конфигурацией системного реестра (в случае Windows), и при этом другие приложения, который могут мешать функционированию игры, на сервере не исполняются.

Кроме того, по мере продвижения пользователя в игре, каждый из сегментов игры загружается в сервер со скоростью гигабит/секунда (т.е. 1 гигабайт загружается за 8 секунд) из дискового массива 1511-1512 типа RAID, и из-за огромной емкости памяти дискового массива 1511-1512 типа RAID (так как он является разделяемым ресурсом среди многих пользователей, он может быть очень большим и, тем не менее, быть эффективным по стоимости), настройка геометрии или настройка другого сегмента игры может предварительно вычисляться и сохраняться на дисковом массиве 1511-1512 типа RAID и загружаться чрезвычайно быстро. Кроме того, вследствие того, что конфигурация аппаратного обеспечения и вычислительные мощности каждого сервера 1521-1525 приложения/игры известны, могут быть предварительно проведены расчеты (с использованием) шейдеров вершин и пикселей.

Соответственно, игра запускается почти немедленно, она исполняется в идеальной среде, и последующие сегменты загружаются почти немедленно.

Но кроме этих преимуществ пользователь может смотреть, как другие ведут игру (посредством Game Finder (средство поиска игры), описанного ранее, и других средств), а (также) принимать решение о том, является ли игра интересной, и если это так, то изучать рекомендации при наблюдении за другими. И пользователь может испытать демонстрационную версию игры немедленно, и при этом нет необходимости ожидать длительной загрузки и/или инсталляции, и пользователь может вести игру немедленно, возможно, для испытания за небольшую плату, или на долгосрочной основе. И пользователь может вести игру на PC Windows, Macintosh, по телевизору, дома, во время поездок, и даже по мобильному телефону, посредством беспроводного соединения с достаточно малым временем ожидания. И это все можно выполнять, и при этом никогда не иметь физическую копию игры.

Как упоминалось ранее, пользователь может принять решение не обеспечивать другим (пользователям) возможность просмотра его геймплей, обеспечить возможность просмотра его игры после задержки, обеспечить возможность просмотра его игры выбранным пользователям или обеспечить возможность просмотра его игры всем пользователям. Не зависимо ни от чего, видео/аудио будут сохраняться, в одном

варианте осуществления, в течение 15 минут в буфере 1515 задержки, и пользователь сможет "перематывать" и просматривать свою предшествующую игру, и приостанавливать, воспроизводить ее медленно, быстро перематывать вперед и т.д., так, как он может делать при просмотре телевизора с цифровым видеомагнитофоном (DVR). Несмотря на то, что в этом примере пользователь ведет игру, идентичное средство "DVR" доступно, если пользователь использует приложение. Это может быть полезно при просмотре предшествующей работы и для других применений, как подробно описано ниже. Кроме того, если игра была разработана со средством перемотки на основе использования информации о состоянии игры так, что поле зрения камеры может быть изменено и т.д., то это средство "3D DVR" также будет поддерживаться, но потребуются, чтобы игра была разработана с его поддержкой. Ф Средство "DVR" с использованием буфера 1515 задержки работает с любыми игрой или приложением, конечно, с ограничениями, для видео, которое сформировано во время использования игры или приложения, но в случае игр со средством DVR 3D пользователь может управлять "сквозным пролетом" (fly through) в 3D ранее проведенного сегмента, и осуществлять запись получающегося в результате видео в буфер 1515 задержки, и осуществлять запись состояния игры для сегмента игры. Соответственно, конкретное "сквозное быстрое перемещение" записывается как сжатое видео, но так как состояние игры также записывается, то другое сквозное быстрое перемещение для идентичного сегмента игры впоследствии возможно.

Как описано ниже, каждый пользователь имеет на службе хостинга 210 страницу пользователя (User Page), куда он может помещать информацию о себе и другие данные. Помимо прочего пользователи могут помещать сегменты видео из геймплей (процесс игры), которые они сохранили. Например, если пользователь преодолел особенно трудное серьезное испытание в игре, то пользователь может "перематывать" до места непосредственно перед своим большим достижением в игре, и затем выдать команду в службу 210 хостинга для сохранения сегмента видео некоторой продолжительности (например, 30 секунд) на странице пользователя (User Page) этого пользователя для просмотра другими пользователями. Для реализации этого, это - просто вопрос сервера 1521-1525 приложения/игры, который пользователь использует, считать видео, сохраненное в буфере 1515 задержки, в дисковый массив 1511-1512 типа RAID и затем сделать ссылку на этот сегмент видео на странице пользователя (User Page) упомянутого пользователя.

Если в игре существует средство DVR 3D, как описано выше, то информация о состоянии игры, требуемая для DVR 3D также может быть записана пользователем и сделана доступной на странице пользователя (User Page) упомянутого пользователя.

В случае, когда игра разработана с возможностью наличия "посетителей" (т.е. пользователей, которые могут перемещаться через мир 3D и наблюдать за действием без участия в нем) наряду с активными игроками, то приложение Game Finder (средство поиска игры) обеспечивает возможность пользователям присоединяться к играм в качестве посетителей, а также игроков. С точки зрения реализации, для системы 210 хостинга не существует разницы, является ли пользователь посетителем или активным игроком. Игра загружается на сервер 1521-1525 приложения/игры, и пользователь управляет игрой (например, посредством управления виртуальной камерой, которая смотрит в мир). Единственным различием будет опыт использования игры пользователем.

Совместная работа множества пользователей

Другим признаком службы 210 хостинга является возможность совместной работы

множества пользователей, когда они просматривают видео в реальном масштабе времени, даже при использовании значительно различающихся устройств для просмотра. Это полезно и при ведении игры и при использовании приложений.

Многие РС и мобильные телефоны оснащены видеокамерами и в них существует средство для выполнения сжатия видео в реальном времени, в частности, когда изображение является маленьким. Кроме того, в продаже имеются маленькие камеры, которые могут быть подсоединены к телевизору, и не трудно реализовать сжатие в реальном времени или в программном обеспечении или с использованием одного из многих аппаратных устройств сжатия для сжатия видео. Кроме того, во многих РС и во всех мобильных телефонах существуют микрофоны, и в продаже имеются наушники с микрофонами.

Такие камеры и/или микрофоны, объединенные с локальным средством сжатия видео/аудио (в частности, с применением способов сжатия видео с малым временем ожидания, описанные в этом документе), обеспечивают пользователю возможность передачи видео и/или аудио с территории 211 пользователя в службу 210 хостинга вместе с данными управления устройства ввода. Когда применяются такие способы, то можно получить средство, иллюстрируемое на фиг.19: пользователь может выводить свое видео и аудио 1900 на экран внутри приложения или игры другого пользователя. Этот пример является игрой с несколькими участниками, где товарищи по команде совместно участвуют в автомобильной гонке. Видео/аудио пользователя может быть выборочно просматриваем/прослушиваемым только своими товарищами по команде. И, так как фактически время ожидания отсутствует при использовании способов, описанных выше, то игроки могут разговаривать друг с другом или показывать жестом друг другу в реальном масштабе времени без ощутимой задержки.

Эта интеграция видео/аудио выполняется посредством поступления сжатого видео и/или аудио из камеры/микрофона пользователя как входящий трафик 1501 в Internet. После этого входящая маршрутизация 1502 маршрутизирует видео и/или аудио в игровые серверы 1521-1525 приложения/игры, которым обеспечена возможность просмотра/прослушивания видео и/или аудио. Далее пользователи соответствующих игровых серверов 1521-1525 приложения/игры, которые решили использовать видео и/или аудио, восстанавливают его и интегрируют как требуется для вывода внутри игры или приложения, например, как изображено в позиции 1900.

В примере по фиг.19 показано, как такая совместная работа используется в игре, но такая совместная работа может быть мощным инструментом для приложений. Рассмотрим ситуацию, когда большое здание проектируется для Нью-Йорка архитекторами в Чикаго для застройщика, который находится в Нью-Йорке, но принятие решения подразумевает финансового инвестора, который путешествует и находится в аэропорту в Майами, и решение должно быть принято об определенных элементах проекта здания исходя из того, как оно согласуется со зданиями, находящимися около него, чтобы и инвестор и застройщик были удовлетворены. Предположим, что архитектурная фирма имеет монитор с высоким разрешением с камерой, подсоединенной к РС в Чикаго, застройщик имеет ноутбук с камерой в Нью-Йорке, а инвестор имеет мобильный телефон с камерой в Майами. Архитектурная фирма может использовать службу 210 хостинга для размещения мощного приложения для архитектурного проектирования с возможностью высоко реалистичной визуализации 3D и возможностью использования большой базы данных зданий в Нью-Йорке, а также базы данных проектируемых зданий. Приложение архитектурного проектирования выполняется на одном, или, если требуется большая вычислительная мощность, на

нескольких, из серверов 1521-1525 приложения/игры. Каждый из этих 3 пользователей, находящихся в различных местах, соединяется со службой 210 хостинга, и каждый имеет одновременный вид выхода приложения архитектурного проектирования, но размер его будет соответственно изменен совместно используемым аппаратным сжатием

5 1530 для данного устройства и характеристик сетевого соединения, которые имеет каждый пользователь (например, в архитектурной фирме можно видеть изображение, 2560×1440, 60 кадр/с, посредством коммерческого подключения к Internet 20 Мбит/с, застройщик в Нью-Йорке может видеть изображение, 1280×720, 60 кадр/с, по соединению DSL 6 Мбит/с на своем ноутбуке, а инвестор может видеть изображение, 320×180, 60

10 кадр/с, по сотовому информационному соединению 250 Кбит/с на своем мобильном телефоне. Каждый участник слышит голос других участников (проведение конференц-связи обрабатывается любым из множества широко доступных пакетов программ для проведения конференц-связи в сервере(ах) 1521-1525 приложения/игры) и, посредством активизации кнопки на устройстве ввода пользователя, пользователь сможет выводить

15 видео самого себя на экран с использованием своей локальной камеры. По ходу совещания, архитекторы смогут показывать то, как выглядит строение, когда они вращают его и далее быстро перемещаются вдоль него к другому зданию на этой территории, с чрезвычайно фотореалистичной визуализацией 3D, и все участники видят идентичное видео с разрешающей способностью дисплея каждого участника. Не имеет

20 значения то, что ни одно из локальных устройств, используемых любым участником, не может обрабатывать анимацию 3D с таким реализмом, уже не говоря о загрузке или даже хранении огромной базы данных, требуемой для визуализации близлежащих зданий в Нью-Йорке. С точки зрения каждого из пользователей, несмотря на то, что они находятся на расстоянии, и несмотря на различные локальные устройства, у них

25 просто будет бесперебойный опыт использования с невероятной степенью реализма. И когда одному участнику требуется, чтобы его лицо было видно с лучшей передачей его эмоционального состояния, он может это сделать. Кроме того, если или (застройщику) или инвестору требуется получить управление архитектурной программой и использовать свое собственное устройство ввода (независимо от того, является ли

30 оно клавиатурой, мышью, кнопочной панелью или сенсорным экраном), они могут это сделать, и она ответит без ощутимого времени ожидания (при предположении, что его сетевое соединение имеет приемлемое время ожидания). Например, в случае мобильного телефона, если мобильный телефон соединен с сетью WiFi в аэропорту, он будет иметь очень малое время ожидания). Но если он использует сотовые сети передачи

35 данных, доступные в настоящее время в США, то он, вероятно, будет испытывать заметное запаздывание. Однако для большинства целей совещания, когда инвестор наблюдает за тем, как архитекторы управляют быстрым перемещением вдоль здания, или для проведения видеотелеконференции, даже время ожидания для сотовой связи должно быть приемлемым.

40 Наконец, в конце совместной конференц-связи, застройщик и инвестор сделают свои комментарии и отсоединятся от службы хостинга, в архитектурной фирме смогут "перемотать" видео конференц-связи, которая была записана в буфер 1515 задержки и просмотреть комментарии, выражения лица и/или действия, относящихся к модели 3D здания, сделанные во время совещания. Если существуют конкретные сегменты, которые

45 им требуется сохранить, то эти сегменты видео/аудио могут быть перемещены из буфера 1515 задержки в дисковый массив 1511-1512 типа RAID для архивного хранения и воспроизведения впоследствии.

Кроме того, с точки зрения стоимости, если архитекторам требуется использование

только вычислительной мощности и большой базы данных Нью-Йорка в течение 15 минут конференц-связи, они должны заплатить только за время, в течение которого эти ресурсы используются, а не за обладание мощными рабочими станциями и покупку дорогостоящей копии большой базы данных.

5 **Обширные общественные видео услуги**

Служба 210 хостинга обеспечивает беспрецедентную возможность для установки обширных общественных видео услуг в Internet. На фиг.20 изображена иллюстративная страница пользователя (User Page) для игрока в службе 210 хостинга. Как в случае с приложением Game Finder (средство поиска игры), User Page (страница пользователя) является приложением, которое исполняется на одном из серверов 1521-1525 приложения/игры. Во всех миниатюрах и видеоокнах на этой странице постоянно показывается движущееся видео (если сегменты являются короткими, то они циклически повторяются).

С использованием видеокамеры или посредством выгрузки видео, пользователь (именем пользователя которого является "KILLHAZARD") может поместить видео самого себя 2000, которое могут просматривать другие пользователи. Видео хранится в дисковом массиве 1511-1512 типа RAID. Кроме того, когда другие пользователи переходят на страницу пользователя (User Page) KILLHAZARD, если KILLHAZARD использует службу 210 хостинга в это время, то будет показано видео 2001 в реальном масштабе времени того, что он делает (при предположении, что он обеспечивает возможность пользователям просматривать его страницу пользователя (User Page) для наблюдения за ним). Это выполняется сервером 1521-1525 приложения/игры, который осуществляет хостинг приложения User Page (страница пользователя), с запросом у системы 401 управления службы, является ли KILLHAZARD активным, и если это так, то сервер 1521-1525 приложения/игры, который он использует. После этого, с использованием идентичных способов, используемых Game Finder (средство поиска игры), сжатый видеопоток с соответствующими разрешающей способностью и форматом отправятся в сервер 1521-1525 приложения/игры, исполняющий приложение User Page (страница пользователя), и оно будет выведено на экран. Если пользователь выбирает окно геймплей в реальном масштабе времени KILLHAZARD, и затем надлежащим образом щелкает по своему устройству ввода, то окно увеличивается (и опять же с использованием идентичных способов, как в приложениях Game Finder (средства поиска игры), и видео в реальном масштабе времени заполняет экран с разрешающей способностью дисплея 422 наблюдающего пользователя, соответствующее характеристикам подключения к Internet наблюдающего пользователя.

Главным преимуществом этого подхода перед подходами предшествующего уровня техники является то, что пользователь, просматривающий User Page (страницу пользователя) может видеть игру, воспроизводящуюся в режиме реального времени, которая не принадлежит пользователю, и он может не иметь локального компьютера или игровой консоли с возможностью воспроизведения игры. То, что пользователь видит пользователя в User Page (страница пользователя) "в действии", ведущего игры, предоставляет ему большие возможности, и это является возможностью узнать об игре, которую просматривающему пользователю, возможно, требуется испытать или улучшить свои результаты в ней.

Записанные камерой или выгруженные видеоклипы от приятелей 2002 KILLHAZARD также показываются в User Page (на странице пользователя), и под каждым видеоклипом существует текст, который указывает, находится ли этот приятель в режиме онлайн и ведет игру (например, six_shot ведет игру "Eragon", и MrSnuggles99 находится в режиме

офлайн, и т.д.). При щелчке на пункте меню (не изображен) видеоклипы приятеля переключатся с показа записанных или выгруженных видео на видео в реальном масштабе времени того, что приятели, которые в настоящее время ведут игры в службе 210 хостинга, делают в этот момент в своих играх. Соответственно, это становится

5 Game Finder (средством поиска игры), выполняющим группировку по приятелям. Если выбрается игра приятеля, и пользователь щелкает по ней, то она увеличится до полного экрана, и пользователь сможет смотреть игру, воспроизводимую во весь экран в режиме реального времени.

И опять же, пользователю, просматривающему игру приятеля, не принадлежит копия этой (игры), а также не принадлежат локальные вычислительные ресурсы/ресурсы игровой консоли для ведения этой игры. Просмотр игры является фактически

10 незамедлительным.

Как ранее описано выше, когда пользователь ведет игру в службе 210 хостинга, пользователь может "перематывать" игру и найти сегмент видео, который ему требуется

15 сохранить, и затем сохраняет этот сегмент видео на своей странице пользователя (User Page). Их называют "Brag Clip" ("клипы хвастовства"). Все сегменты 2003 видео являются Brag Clip 2003, сохраненными KILLHAZARD из предыдущих игр, которые он вел. В позиции 2004 показывается, сколько раз Brag Clip был просмотрен, и когда Brag Clip просматривается, у пользователей есть возможность оценить его, и количество

20 оранжевых пиктограмм 2005 в виде замочной скважины указывает, насколько высоким является рейтинг. Brag Clip 2003 постоянно циклически повторяются, когда пользователь просматривает страницу пользователя (User Page), вместе с остальной частью видео на странице. Если пользователь выбирает и щелкает по одному из Brag Clip 2003, он увеличивается для представления Brag Clip 2003, причем элементы управления DVR

25 обеспечивают возможность воспроизведения клипа, его приостановки, перемотки, быстрой перемотки вперед, прохождения по этапам и т.д.

Воспроизведение Brag Clip 2003 реализуется сервером 1521-1525 приложения/игры, загружающим сегмент сжатого видео, сохраненный в дисковом массиве 1511-1512 типа RAID, когда пользователь записывал Brag Clip, и восстанавливающим его и

30 воспроизводящим его.

Brag Clip 2003 могут также быть сегментами видео "DVR 3D" (т.е. последовательностью состояний игры из игры, которая может быть повторно воспроизведена и обеспечивает возможность пользователю изменять точку обзора

камеры) из игр, которые поддерживают такое средство. В этом случае информация о

35 состоянии игры сохраняется, наряду с записью сжатого видео конкретного "сквозного пролета", которую сделал пользователь, когда записывался сегмент игры. Когда просматривается User Page (страница пользователя), и все миниатюры и видеоокна постоянно циклически повторяются, Brag Clip 2003 DVR 3D постоянно циклически повторяет Brag Clip 2003, который был записан как сжатое видео, когда пользователь

40 записывал "сквозной пролет" сегмента игры. Но когда пользователь выбирает Brag Clip 2003 DVR 3D и щелкает на нем, наряду с элементами управления DVR, обеспечивающими возможность воспроизведения сжатого видео Brag Clip, пользователь сможет щелкнуть на кнопке, которая предоставляет ему средство DVR 3D для сегмента игры. Он сможет управлять "сквозным пролетом" камеры в течение сегмента игры

45 самостоятельно, и, если ему потребуется (и пользователь, которому принадлежит эта страница пользователя, обеспечивает эту возможность), то он сможет записать альтернативный "сквозной пролет" Brag Clip в форме сжатого видео, (которое) после этого будет доступно другим зрителям страницы пользователя (или немедленно, или

после того, как у владельца страницы пользователя будет возможность просмотреть этот Brag Clip).

Это средство Brag Clip 2003 DVR 3D обеспечивается посредством активации игры, которая готова начать повторное воспроизведение записанной информации о состоянии 5 игры на другом сервере 1521-1525 приложения/игры. Так как игра может быть активирована почти мгновенно (как описано ранее), ее не трудно активировать, причем воспроизведение ее ограничивается до состояния игры, записанного сегментом Brag Clip, и после этого обеспечить возможность пользователю выполнять "сквозной пролет", причем камера записывает сжатое видео в буфер 1515 задержки. После завершения 10 пользователем выполнения "сквозного пролета" игра отключается.

С точки зрения пользователя активация "сквозного пролета" посредством Brag Clip 2003 DVR 3D не требует большего усилия, чем управление элементами управления DVR линейного Brag Clip 2003. Он может ничего не знать об игре или даже то, как вести игру. Он является только оператором виртуальной камеры, смотрящей в мир 3D в течение 15 сегмента игры, записанного другим.

Пользователи также могут накладывать их собственный аудио на Brag Clip, которое является или записанным с микрофонов или выгруженным. Следовательно, Brag Clip могут использоваться для создания пользовательской анимации с использованием персонажей и действий из игр. Этот способ анимации обычно известен как "машинима".

20 По мере продвижения пользователей в играх, они достигают отличающихся уровней мастерства. Воспроизведенные игры сообщают об этих достижениях в систему управления 401 службы, и эти уровни мастерства будут показаны на User Page (страницах пользователя).

Интерактивные анимированные рекламные объявления

25 Осуществлялся переход онлайнowych рекламных объявлений от текста к неподвижным изображениям, к видео и в настоящее время к интерактивным сегментам, как правило, реализуемым с использованием тонких клиентов анимации, подобных Adobe Flash. Причиной, по которой используются тонкие клиенты анимации, является то, что пользователи испытывают нетерпение при задержке (осуществления преимущественного 30 права получения) продукта или услуги, передаваемых им. Кроме того, тонкие клиенты исполняются на очень низкопроизводительных РС и, по существу, лицо, дающее объявление, может быть совершенно уверено, что интерактивное рекламное объявление будет работать должным образом. К сожалению, тонкие клиенты анимации, например Adobe Flash, ограничены по степени интерактивности и продолжительности опыта 35 использования (для уменьшения времени загрузки).

На фиг.21 изображена интерактивная реклама, в которой пользователь должен выбрать внешние и внутренние цвета автомобиля во время вращения автомобиля в демонстрационном зале, когда посредством трассировки лучей в реальном времени демонстрируется, как выглядит автомобиль. Далее пользователь выбирает "аватара" 40 для управления автомобилем, и после этого пользователь может взять автомобиль для поездки или по трассе для гонок, или по экзотической местности, например, Монако. Пользователь может выбрать двигатель с большим рабочим объемом или лучшие шины и после этого может проверить, как измененная конфигурация влияет на возможность автомобиля ускоряться или держать дорогу.

45 Несомненно, упомянутая реклама фактически является усложненной видеоигрой 3D. Но для воспроизведения такой рекламы на РС или видеоигровой консоли может потребоваться загрузка 100 мегабайт и, в случае РС, может потребоваться установка специальных драйверов, и может вообще оказаться невозможным ее исполнение, если

в PC отсутствуют отвечающая требованиям вычислительная возможность GPU или CPU. Соответственно, такие рекламные объявления практически невыполнимы в конфигурациях предшествующего уровня техники.

В службе 210 хостинга такие рекламные объявления запускаются почти мгновенно и исполняются превосходно, независимо от возможностей клиента 415 пользователя. Соответственно, они запускаются более быстро, чем интерактивные рекламные объявления тонкого клиента, со значительно более обширным опытом использования и с высокой степенью надежности.

Потоковая геометрия во время анимации в реальном времени

Дисковый массив 1511-1512 типа RAID и входящая маршрутизация 1502 могут обеспечивать скорости передачи данных, которые являются такими большими и с величинами времени ожидания такими малыми, что можно проектировать видеоигры и приложения, которые основаны на дисковых массивах 1511-1512 типа RAID и входящей маршрутизации 1502 для надежной доставки геометрии на лету в середине геймплея или в приложении во время анимации в реальном времени (например, "сквозной пролет" со сложной базой данных).

С системами предшествующего уровня техники, например, игровая видеосистема, изображенная на фиг.1, доступные устройства массовой памяти, в частности, в применяемых на практике домашних устройствах, являются слишком медленными для передачи потоком геометрии во время геймплея, за исключением ситуаций, когда требуемая геометрия является до некоторой степени предсказуемой. Например, в игре с вождением, где существует заданное шоссе, геометрия для зданий, которые появляются в поле зрения, может быть приемлемо предсказуемой, и устройства массовой памяти могут вести поиск заранее до места, где находится приближающаяся геометрия.

Но в сложной сцене с непредсказуемыми изменениями (например, в сцене сражения со сложными персонажами повсюду), если RAM в PC или игровой видеосистеме полностью заполнена геометрией для объектов, в настоящее время находящихся в поле зрения, и далее пользователь внезапно поворачивает своего персонажа, чтобы видеть то, что находится сзади него, если геометрия не была предварительно загружена в RAM, то может произойти задержка перед выводом ее на экран.

В службе 210 хостинга, дисковые массивы 1511-1512 типа RAID могут потоком передавать данные со скоростью, превышающей скорость по технологии Gigabit Ethernet, и в случае с сетью SAN, можно достичь скорости 10 гигабит/секунда по технологии 10 Gigabit Ethernet или по другим сетевым технологиям. При 10 гигабит/секунда гигабайт данных загружается менее, (чем) за секунду. В период кадра 60 кадр/с (16,67 мс), могут быть загружены приблизительно 170 мегабитов (21 мегабайт) данных. Вращающиеся носители информации, несомненно, даже в конфигурации RAID, по-прежнему, вызывают величины времени ожидания, превышающие период кадра, но запоминающее устройство RAID на основе флеш-памяти в итоге будет такого размера, как дисковые массивы типа RAID вращающихся носителей информации, и не будет вызывать такого большого времени ожидания. В одном варианте осуществления используется кэширование со сквозной записью в массивную RAM для обеспечения доступа с очень малым временем ожидания.

Соответственно, с достаточно высокой скоростью сети и достаточной массовой памятью с достаточно малым временем ожидания, геометрию можно передавать потоком в игровые серверы 1521-1525 приложения/игры с такой скоростью, с которой CPU и/или GPU могут обрабатывать данные 3D. Соответственно, в примере, приведенном ранее, где пользователь внезапно поворачивает своего персонажа и

смотрит назад, геометрия для всех персонажей, находящихся сзади, может быть загружена до того, как персонаж завершает поворот, и, соответственно, пользователю будет казаться, как будто он или она находятся в фотореалистичном мире, который настолько реальный как игра актеров.

- 5 Как обсуждалось ранее, одной из последних границ в фотореалистичной компьютерной анимации является человеческое лицо, и из-за чувствительности человеческого глаза к несовершенствам, малейшая ошибка в фотореальном лице может в результате привести к отрицательной реакции у зрителя. На фиг.22 изображено то, как живое исполнение, захваченное с использованием технологии Contour™ Reality
- 10 Capture (предмет заявок, находящихся в процессе одновременного рассмотрения: "Apparatus and method for capturing the motion of a performer" Ser. No. 10/942609, поданная 15 сентября 2004 г., "Apparatus and method for capturing the expression of a performer" Ser. No. 10/942413, поданная 15 сентября 2004 г., "Apparatus and method for improving marker identification within a motion capture system" Ser. No. 11/066954, поданная 25 февраля 2005 г.,
- 15 "Apparatus and method for performing motion capture using shutter synchronization" Ser. No. 11/077628, поданная 10 марта 2005 г., "Apparatus and method for performing motion capture using a random pattern on capture surfaces," Ser. No. 11/255854, поданная 20 октября 2005 г., "System and method for performing motion capture using phosphor application techniques," Ser. No. 11/449131, поданная 7 июня 2006 г., "System and method for performing motion capture
- 20 by strobing a fluorescent lamp," Ser. No. 11/449043, поданная 7 июня 2006 г., "System and method for three dimensional capture of stop-motion animated characters," Ser. No. 11/449127, поданная 7 июня 2006 г., права на каждую из которых принадлежат заявителю настоящей заявки СІР), в результате приводит к очень гладкой захваченной поверхности, затем к сетчатой поверхности, насчитывающей большое количество многоугольников, (т.е.
- 25 движение многоугольника точно следует за движением лица). Наконец, когда видео живого исполнения отображают на сетчатую поверхность для вывода текстурированной поверхности, выводится фотореальный результат.

- Несмотря на то, что посредством современной технологии GPU можно визуализировать количество многоугольников в сетчатой поверхности и текстурировать
- 30 и осветить поверхность в реальном времени, если многоугольники и текстуры изменяются каждый период кадра (что приводит к наиболее фотореальным результатам), то вся доступная RAM современного РС или видеоигровой консоли быстро израсходуется.

- С использованием способов потоковой геометрии, описанных выше, становится
- 35 возможным применение на практике непрерывной подачи геометрии в игровые серверы 1521-1525 приложения/игры, чтобы они непрерывно могли анимировать фотореальные лица с обеспечением возможности создания видеоигр с лицами, которые почти нельзя отличить от лиц при игре актеров.

Интеграция линейного содержимого с интерактивными признаками

- 40 Кинофильмы, телевизионные программы и аудио материал (обобщенно, "линейное содержимое") являются широко доступными для домашних и офисных пользователей во многих видах. Линейное содержимое может быть приобретено на физических носителях информации, подобных носителям информации CD, DVD, HD-DVD и Blu-ray. Оно также может быть записано посредством DVR из вещательной передачи кабельного
- 45 телевидения и спутниковой вещательной передачи. И оно доступно как содержимое с платой за просмотр (PPV) через спутниковое и кабельное телевидение и как видео по запросу (VOD) по кабельному телевидению.

Все больше линейное содержимое доступно через Internet, и как загружаемое и как

потокное содержимое. В настоящее время, на самом деле не существует ни одного места, где можно испытать все признаки, связанные с линейным мультимедиа. Например, DVD и другие оптические носители видеoinформации, как правило, имеют интерактивные признаки, которые не доступны где-либо еще, подобные комментариям

5 кинорежиссера, короткометражным фильмам "мнение о" и т.д. Онлайн-музыкальные сайты охватывают художественную и песенную информацию, как правило, не доступную на CD, но не все CD доступны в режиме онлайн. И на Web-сайтах, связанных с телевизионными программами, часто существуют дополнительные признаки, блоги и иногда комментарии актеров или творческого персонала.

10 Кроме того, в отношении многих кинофильмов или спортивных событий, часто существуют видеоигры, которые выпускают (в случае кинофильмов) часто вместе с линейным мультимедиа, или (в случае спортивных состязаний) которые могут быть близко связаны с реальными событиями (например, торговля игроками).

Служба 210 хостинга хорошо подходит для доставки линейного содержимого в компоновке разрозненных форм взаимосвязанного содержимого. Конечно, доставка кинофильмов требует не больше усилий, (чем) доставка в высокой степени интерактивных видеоигр, и служба 210 хостинга может доставлять линейное содержимое в широкий диапазон устройств, (находящихся) в домах или офисах, или в мобильные устройства. На фиг.23 изображена иллюстративная страница интерфейса пользователя

15 для службы 210 хостинга, на которой изображена выборка линейного содержимого.

Но в отличие от большинства систем доставки линейного содержимого, служба 210 хостинга также может доставлять взаимосвязанные интерактивные компоненты (например, меню и признаки на DVD, интерактивные наложения одного графического изображения на другое на HD-DVD и анимацию Adobe Flash (как объясняется ниже) на

20 Web-сайтах. Соответственно, ограничения клиентского устройства 415 больше не вводят ограничений относительно того, какие признаки являются доступными.

Кроме того, система 210 хостинга может динамически и в реальном времени компоновать линейное содержимое с содержимым видеоигры. Например, если пользователь смотрит матч Quidditch в фильме Гарри Поттер, и решает, что она бы

30 хотела попробовать сыграть в Quidditch, то она может просто щелкнуть на кнопке, и фильм будет приостановлен, и она немедленно переместится в сегмент Quidditch видеоигры Гарри Поттер. После проведения матча Quidditch, еще один щелчок кнопкой, и фильм немедленно будет продолжен.

С фотореальной графикой и технологией производства, при которой фотографически захваченное видео нельзя отличить от персонажей с игрой актеров, когда пользователь осуществляет переход от игры Quidditch в фильме с игрой актеров к игре Quidditch в видеоигре на службе хостинга, как описано в этом документе, эти две сцены фактически

35 нельзя отличить. Это обеспечивает полностью новыми творческими возможными вариантами для режиссеров и линейного содержимого и интерактивного содержимого (например, видеоигры), поскольку границы между этими двумя мирами становятся неразличимыми.

С использованием архитектуры службы хостинга, изображенной на фиг.14, зрителю может быть предоставлено управление виртуальной камерой в фильме 3D. Например, в сцене, которая имеет место внутри вагон поезда, можно обеспечить возможность

45 зрителю управлять виртуальной камерой и осматривать вагон по ходу рассказа. При этом предполагается, что все объекты 3D ("ресурсы") в вагоне являются доступными, а также отвечающий требованиям уровень вычислительных возможностей с возможностью визуализации сцен в реальном времени, а также оригинал фильма.

И даже для развлечений, формируемых не на компьютере, могут быть предоставлены очень захватывающие интерактивные признаки. Например, в кинофильме "Гордость и Предубеждение", 2005 г., существует много сцен в богато украшенных старых английских особняках. Для сцен в определенном особняке, пользователь может приостановить видео и затем управлять камерой, чтобы отправиться в путешествие по особняку, или, возможно, по окрестностям. Для реализации этого, камеру можно носить по особняку с линзой типа "рыбий глаз", в то время как она ведет запись своего положения, во многом подобно тому, как осуществлен QuickTime VR предшествующего уровня техники корпорации Apple, Inc. Тогда различные кадры могут быть преобразованы так, что изображения не будут искажены, и после этого сохранены в дисковом массиве 1511-1512 типа RAID вместе с фильмом, и воспроизведены, когда пользователь решит отправиться в виртуальное путешествие.

В отношении спортивных событий, спортивное событие, происходящее в реальном масштабе времени, например, игра в баскетбол, может передаваться через службу 210 хостинга, чтобы пользователи смотрели ее так, как они могут смотреть ее по обычному телевизору. После того, как пользователи посмотрели конкретную игру, видеоигра этой игры (в итоге, с баскетболистами, которые выглядят также фотореально, как реальные игроки) может появиться на экране, причем игроки начинают с идентичной позиции, и пользователи (возможно, каждый при этом получает управление одним игроком) могут восстановить игру и посмотреть, могут ли они добиться большего успеха, чем эти игроки.

Служба 210 хостинга, описанная в этом документе, очень хорошо подходит для поддержки этого мира будущего, потому что она может привлекать вычислительные возможности и ресурсы массовой памяти, которые практически нецелесообразно устанавливать в домашней обстановке или в большинстве офисных обстановок, и также ее вычислительные ресурсы всегда являются современными, с последним вычислительным доступным аппаратным обеспечением, тогда как в домашней обстановке, всегда будут существовать дома с видеоиграми и РС предыдущих поколений. И, в службе 210 хостинга, вся эта вычислительная сложность скрыта от пользователя, поэтому, несмотря на то, что они могут использовать очень усложненные системы, с точки зрения пользователя, это является простым, как переключение каналов на телевизоре. Кроме того, пользователи могут получать доступ ко всей вычислительной мощности и опыт использования, который может предоставить эта вычислительная мощность из любого клиента 415.

Игры с несколькими участниками

В тех случаях, когда игра является игрой с несколькими участниками, тогда она может осуществлять связь и с игровыми серверами 1521-1525 приложения/игры через сеть входящей маршрутизации 1502 и посредством сетевого моста с Internet (не изображен) с серверами или игровыми машинами, которые не функционируют в службе 210 хостинга. При ведении игры с несколькими участниками посредством компьютеров в общем Internet, игровые серверы 1521-1525 приложения/игры будут иметь преимущество с чрезвычайно быстрым доступом к Internet (по сравнению с тем, если игра исполняется на сервере дома), но они будут ограничены возможностями других компьютеров, которые ведут игру на более медленных соединениях, и также, возможно, ограничены тем, что игровые серверы в Internet предназначены для обеспечения наименьшего общего знаменателя, который может быть домашними компьютерами на относительно медленных потребительских подключениях к Internet.

Но когда игра с несколькими участниками полностью ведется внутри серверного

центра службы 210 хостинга, тогда можно достичь большого различия. Каждый игровой сервер 1521-1525 приложения/игры, который осуществляет хостинг игры для пользователя, будет соединен с другими игровыми серверами 1521-1525 приложения/игры, а также с любыми серверами, которые осуществляют хостинг центрального управления для игры с несколькими участниками с чрезвычайно высокой скоростью, с соединением с чрезвычайно малым временем ожидания и с огромными, очень быстрыми массивами запоминающих устройств. Например, если используется технология Gigabit Ethernet для сети входящей маршрутизации 1502, то игровые серверы 1521-1525 приложения/игры осуществляют связь друг с другом и осуществляют связь с любыми серверами, которые осуществляют хостинг центрального управления для игры с несколькими участниками на скорости гигабит/секунда с, возможно, временем ожидания только 1 мс или меньше. Кроме того, дисковые массивы 1511-1512 типа RAID смогут отвечать очень быстро и затем передать данные со скоростями гигабит/секунда. В качестве примера, если пользователь настраивает самостоятельно персонажа с точки зрения внешнего вида и снаряжения так, что этот персонаж имеет большой объем геометрии и линий поведения, которые являются уникальными для персонажа, то с системами предшествующего уровня техники, которые ограничены клиентом игры, исполняющимся дома на PC или игровой консоли, если этот персонаж должен появиться в поле зрения другого пользователя, то этот пользователь должен ждать, пока не завершится длинная, медленная загрузка, чтобы все данные линии поведения и геометрии загрузились в его компьютер. В пределах службы 210 хостинга, идентичная загрузка может осуществляться по Gigabit Ethernet с обслуживанием из дискового массива 1511-1512 типа RAID со скоростью гигабит/секунда. Даже если домашний пользователь имеет подключение к Internet 8 Мбит/с (которое является чрезвычайно быстрым по современным стандартам), то Gigabit Ethernet является в 100 раз быстрее. Соответственно, то, что занимает одну минуту по быстрому подключению к Internet, займет меньше чем, одна секунда по Gigabit Ethernet.

Группы лучших игроков и соревнования

Служба 210 хостинга очень хорошо подходит для соревнований. Поскольку игра не выполняется в локальном клиенте, для пользователей не существует возможности обманывать. Кроме того, из-за того, что выходная маршрутизация 1540 может осуществлять групповую передачу потоков UDP, Служба 210 хостинга может транслировать основные соревнования тысячам людей в аудитории одновременно.

Фактически, когда существуют определенные видеопотоки, которые настолько популярны, что тысячи пользователей принимают идентичный поток (например, показ видов основного соревнования), может оказаться более эффективным отправить этот видеопоток в сеть доставки контента (CDN), например, Akamai или Limelight для массового распределения во многие клиентские устройства 415.

Аналогичный уровень эффективности может быть получен, когда CDN используют для показа страниц Game Finder (средство поиска игры) групп лучших игроков.

Для основных соревнований, можно использовать реального спортивного комментатора знаменитостей для обеспечения комментариев во время определенных матчей. Хотя большое количество пользователей будут смотреть основное соревнование, и относительно маленькое количество пользователей будут играть на соревновании. Аудио от спортивного комментатора знаменитостей может маршрутизироваться в игровые серверы 1521-1525 приложения/игры, осуществляющие хостинг пользователей, играющих на соревновании, и хостинг любых копий режимов посетителя игры во время соревнования, и это аудио может быть наложено поверх аудио игры. Видео спортивного

комментатора знаменитостей может также быть наложено на игры, возможно, только на отображения на экране посетителя.

Ускорение загрузки web-страницы

Исходный транспортный протокол всемирной паутины, протокол передачи гипертекстовых файлов (HTTP), был задуман и определен в период, когда только в компаниях были высокоскоростные подключения к Internet, и потребители, которые были в режиме онлайн, использовали модемы коммутируемой линии передачи или ISDN. В это время, "золотым стандартом" для высокоскоростного соединения была линия T1, которая обеспечивала скорость передачи данных 1,5 Мбит/с симметрично (т.е. с равной скоростью передачи данных в обоих направлениях).

В настоящее время ситуация является совершенно другой. Средняя скорость домашнего соединения через соединения кабельной модемной связи или DSL в большей части развитых стран имеет гораздо более высокую скорость передачи данных нисходящего потока данных, чем линия T1. Фактически, в некоторых частях мира, посредством технологии ввода в здания оптического кабеля (fiber-to-the-curb) в дом подаются скорости передачи данных от 50 до 100 Мбит/с.

К сожалению, HTTP не был спроектирован (он также не был реализован) для фактического использования этих существенных увеличений скорости. Web-сайт является совокупностью файлов на удаленном сервере. В самых простых словах, HTTP запрашивает первый файл, ожидает его загрузки, и после этого запрашивает второй файл, ожидает его загрузки и т.д. Фактически, HTTP обеспечивает возможность нескольких "открытых соединений", т.е. запрос нескольких файлов за один раз, но из-за установленных стандартов (и стремления предотвратить перегрузку web-серверов) обеспечивается возможность только очень малого количества открытых соединений. Кроме того, из-за способа создания Web-страниц, браузерам часто не известно, что множество страниц могут быть одновременно доступны для немедленной загрузки (т.е. только после анализа (парсинга) страницы становится очевидно, что должен быть загружен новый файл, например, изображение). Соответственно, файлы на web-сайте по существу загружаются один за другим. И, из-за протокола запрос-и-ответ, используемого HTTP, время ожидания, связанное с каждым загружаемым файлом, равно примерно (получение доступа к типичным web-серверам в США) 100 мс.

В отношении соединений с относительно низкой скоростью, это не представляет большую проблему, потому что время загрузки самих файлов доминирует над временем ожидания для веб-страниц. Но, поскольку скорости соединения растут, особенно в отношении сложных веб-страниц, начинают возникать проблемы.

В примере, изображенном на фиг.24, представлен типичный коммерческий web-сайт (этот конкретный web-сайт являлся сайтом от лидирующей торговой марки спортивной обуви). На этом web-сайте существует 54 файла. Файлы включают в себя HTML, CSS, JPEG, PHP, JavaScript и Flash файлы, и включают в себя содержимое видео. В общей сложности должно быть загружено 1,5 мегабайта до того, как страница станет активной (т.е. пользователь может щелкнуть по ней и начать использовать ее). Существует несколько причин для большого количества файлов. Во-первых, эта web-страница является сложной и находится на уровне современных требований, и, во-вторых, она является web-страницей, которая ассемблируется динамически на основе информации о пользователе, получающем доступ к этой странице, (например, из какой страны пользователь, какой язык, делал ли пользователь раньше покупки, и т.д.) и в зависимости от всех этих факторов, загружаются разные файлы. Однако, она является очень типичной коммерческой веб-страницей.

На фиг.24 представлено количество времени, которое протекает до того, как веб-страница становится активной, по мере того, как скорость соединения возрастает. При скорости 2401 соединения 1,5 Мбит/с, с использованием обычного web-сервера посредством обычного web-браузера, требуется 13,5 секунд, чтобы веб-страница стала активной. При скорости 2402 соединения 12 Мбит/с, время загрузки уменьшается до 6,5 секунд, или почти в два раза быстрее. Но при скорости 2403 соединения 96 Мбит/с, время загрузки уменьшается только приблизительно до 5,5 секунд. Причина, по которой это происходит, заключается в том, что при такой высокой скорости загрузки время загрузки самих файлов является минимальным, но время ожидания для каждого файла, примерно 100 мс, по-прежнему, остается, что в результате приводит к 54 файла×100 мс=5,4 секунд времени ожидания. Соответственно, независимо от скорости соединения до дома, время для того, чтобы этот веб-сайт стал активным, всегда будет равно, по меньшей мере, 5,4 секунды. Другим фактором является организация очереди со стороны сервера. Каждый запрос HTTP добавляется в конец очереди, соответственно, на загруженном сервере это оказывает существенное влияние, потому что, чтобы получить из web-сервера каждый маленький элемент, запросы HTTP должны ждать своей очереди.

Одним способом решения этой проблемы является отказ от HTTP или его переопределение. Или, возможно, лучше, чтобы владелец web-сайта объединил свои файлы в единый файл (например, в формате Adobe Flash). Но, на практике, эта компания, а также многие другие (компании) много инвестировали в архитектуру своего веб-сайта. Кроме того, в то время как в некоторых домах существуют соединения 12-100 Мбит/с, в большинстве домов все еще существуют более медленные скорости, а HTTP хорошо работает при медленной скорости.

Одним альтернативным вариантом является хостинг web-браузеров на серверах 1521-1525 приложения/игры и хостинг файлов для web-серверов на дисковых массивах 1511-1512 типа RAID (или, возможно, в RAM или на локальном запоминающем устройстве на серверах 1521-1525 приложения/игры, осуществляющих хостинг web-браузеров). Вследствие очень быстрой взаимосвязи через входящую маршрутизацию 1502 (или с локальным запоминающим устройством), вместо 100 мс времени ожидания при использовании HTTP для каждого файла, будет существовать незначительное время ожидания при использовании HTTP для каждого файла. Тогда, вместо того, чтобы пользователь в своем доме получал доступ к веб-странице через HTTP, пользователь может получать доступ к веб-странице через клиента 415. Тогда, даже с соединением 1,5 Мбит/с (потому что эта веб-страница не требует большой полосы пропускания для своего видео), веб-страница будет активной меньше, чем через 1 секунду для каждой линии 2400 связи. По существу, не будет времени ожидания до вывода на экран активной страницы web-браузером, исполняющимся на сервере 1521-1525 приложения/игры, и не будет заметного времени ожидания до вывода на экран клиентом 415 видеовыхода из web-браузера. По мере того, как пользователь водит мышью по веб-странице и/или набирает текст на ней, информация ввода пользователя отправляется в web-браузер, исполняющийся на сервере 1521-1525 приложения/игры, и web-браузер отвечает соответственно.

Одним недостатком этого подхода является то, что, если устройство сжатия постоянно передает видеоданные, то полоса пропускания используется, даже если веб-страница становится неподвижной. Это можно исправить посредством выполнения устройства сжатия с возможностью передавать данные только тогда, когда (и если) веб-страница изменяется, и тогда передавать данные только в части страницы, которые изменились. Несмотря на то, что существуют некоторые веб-страницы с флэш-баннерами

и т.д., которые постоянно изменяются, такие веб-страницы, как правило, раздражают, и обычно веб-страницы являются неподвижными, если не существует причины для какого-либо движения (например, видеоклип). Для таких веб-страниц, это, вероятно, тот случай, когда с использованием службы 210 хостинга будет передаваться меньше данных, чем для обычного web-сервера, потому что будут передаваться только фактически выводимые на экран изображения, никакого исполняемого кода тонкого клиента и никаких больших объектов, которые могут никогда не просматриваться, например, переключающиеся картинки.

Соответственно, с использованием службы 210 хостинга для осуществления хостинга унаследованных веб-страниц, время загрузки веб-страниц может быть уменьшено настолько, что открытие веб-страницы будет похоже на переключение телевизионных каналов: веб-страница становится активной фактически немедленно.

Обеспечение отладки игр и приложений

Как упоминалось ранее, видеоигры и приложения с графикой в реальном времени являются очень сложными приложениями и, как правило, когда их выпускают в реальные условия эксплуатации, они содержат ошибки. Хотя разработчики программного обеспечения получают с обратной связью от пользователей (сообщения) об ошибках, и у них могут быть некоторые средства для обратной передачи машинного состояния после аварийных отказов, очень трудно точно идентифицировать то, что вызвало аварийный отказ или ненадлежащее выполнение игры или приложения реального времени.

Когда игра или приложение исполняются в службе 210 хостинга, видео/аудиовыход игры или приложения постоянно записывается в буфер 1515 задержки. Кроме того, сторожевой процесс выполняется каждым сервером 1521-1525 приложения/игры, который регулярно сообщает в систему 401 управления службой хостинга о том, что сервер 1521-1525 приложения/игры исполняется гладко. Если сторожевой процесс не передает сообщение, то система 401 управления сервером пытается связаться с сервером 1521-1525 приложения/игры, и в случае успеха, собирает все доступное машинное состояние. Вся доступная информация, вместе с видео/аудио, записанным буфером 1515 задержки, будет отправлена разработчику программного обеспечения.

Соответственно, когда разработчик прикладного программного обеспечения или игры получает уведомление об аварийном отказе из службы 210 хостинга, (он) получает кадровую запись того, что привело к этому аварийному отказу. Эта информация может быть очень ценной в обнаружении ошибок и исправлении их.

Также отметим, что, когда происходит аварийный отказ сервера 1521-1525 приложения/игры, сервер повторно запускается с самой последней точки перезапуска, и пользователю передается сообщение с извинениями за техническую проблему.

Совместное использование ресурсов и снижение расходов

Система, изображенная на фиг.4а и фиг.4b, обеспечивает множество преимуществ как для конечных пользователей, так и для разработчиков приложений и игр. Например, как правило, домашние и офисные клиентские системы (например, PC или игровые консоли) используются только в течение небольшого количества времени в неделю. Согласно пресс-релизу, опубликованному 5 октября 2006 г., Nielsen Entertainment "Active Gamer Benchmark Study" (<http://www.prnewswire.com/cgi-bin/stories.pl?ACCT=104&STORY=/www/story/10-05-2006/0004446115&EDATE=>) активные игроки тратят в среднем 14 часов в неделю на игру на видеоигровых консолях и приблизительно 17 часов в неделю на карманных компьютерах. В этом сообщении также утверждается, что вся игровая активность (в том числе ведение игр на PC, карманном компьютере, консоли) (активных

игроков) в среднем составляет 13 часов в неделю. С учетом более высокий показатель времени ведения видеоигр на консоли, $24 \times 7 = 168$ часов в неделю, которая подразумевает, что в доме активного игрока, видеоигровая консоль используется только $17/168 = 10\%$ времени в неделю. Или 90% времени видеоигровая консоль не используется. С учетом

5 высокой стоимости видеоигровых консолей, и того, что производители финансируют такие устройства, это является очень неэффективным использованием дорогостоящего ресурса. РС, находящиеся в компаниях, также, как правило, используются только часть времени в неделю, особенно стационарные настольные РС, часто требуемые для приложений с широкими функциональными возможностями, например, Maya Autodesk.

10 Несмотря на то, что некоторые компании работают все время и в праздники, и некоторые РС (например, ноутбуки, принесенные домой для работы вечером), используются все время и в праздники, большая часть коммерческой деятельности, как правило, сосредоточена примерно от 9 часов до 17 часов, в часовом поясе данной компании, с понедельника по пятницу, кроме выходных и перерывов (например, обед),

15 и так как большинство РС используются, когда пользователь активно работает с РС, из этого следует, что настольные РС, как правило, используются в это время работы. Если предположить, что РС постоянно используются от 9 часов до 17 часов, 5 дней в неделю, то это означает, что РС используются $40/168 = 24\%$ времени в неделю.

Высокопроизводительные настольные РС являются очень дорогостоящими

20 инвестициями для компаний, и это отражается в очень низком уровне использования. Школы, в которых обучение ведется на настольных компьютерах, могут использовать компьютеры в течение еще меньшей части недели, и, несмотря на то, что она изменяется в зависимости от времени обучения, большая часть обучения происходит в дневное время с понедельника по пятницу. Соответственно, в общем, РС и видеоигровые консоли

25 используются только небольшую часть времени в неделю.

То есть, так как многие люди работают в компаниях или в школе в дневное время, с понедельника по пятницу, кроме выходных, эти люди, как правило, не играют в видеоигры в это время, и поэтому, на самом деле, они играют в видеоигры, как правило, в другое время, например, по вечерам, в выходные дни и в праздники.

30 С учетом конфигурации службы хостинга, изображенной на фиг.4а, схемы использования, описанные в двух абзацах выше, в результате приводят к очень эффективному использованию ресурсов. Очевидно, что существует ограничение на количество пользователей, которые могут обслуживаться службой 210 хостинга в данное время, особенно, если пользователи требуют быстрого реагирования в реальном

35 времени для сложных приложений, аналогичных усложненным видеоиграм 3D. Но, в отличие от видеоигровой консоли в доме или РС, используемом компанией, которые, как правило, не используются большую часть времени, серверы 402 могут быть повторно использованы разными пользователями в разное время. Например,

40 высокопроизводительный сервер 402 с высокопроизводительными двоянным CPU и двоянными GPU и большим количеством RAM может использоваться компаниями и школами с 9 часов до 5 часов по рабочим дням, но использоваться игроками, ведущими усложненные видеоигры по вечерам, в выходные дни и в праздники. Аналогично, приложения с низкими характеристиками могут использоваться компаниями и школами на низкопроизводительном сервере 402 с CPU Celeron, без GPU (или с очень

45 низкопроизводительным GPU) и с ограниченной RAM в течение рабочего времени, и игра с низкими характеристиками может использовать низкопроизводительный сервер 402 в течение нерабочего времени.

Кроме того, с компоновкой службы хостинга, описанной в этом документе, ресурсы

совместно используются фактически тысячами, если не миллионами, пользователей. В общем, только небольшой процент от общего количества пользователей онлайн-услуг используют службу в данный момент. Если учитывать, статистику Nielsen использования видеоигр, перечисленную ранее, то легко понять почему. Если активные игроки играют в консольные игры только 17 часов в неделю, и если предположить, что пиковое время использования игры обычно приходится на нерабочее время, нерабочее время по вечерам (с (17) часов до (24) часов, 7×5 дней = 35 часов/неделя) и по выходным дням (с 8 часов до (24) часов, $16 \times 2 = 32$ часов/неделя), то есть $35 + 32 = 65$ пиковых часов в неделю для 17 часов геймплея. Точную пиковую пользовательскую нагрузку в системе трудно оценить по многим причинам: некоторые пользователи играют в не пиковые периоды времени, могут существовать периоды дня, когда существуют пики разбиения на группы пользователей, причем на эти пиковые периоды может влиять тип ведущейся игры (например, в детские игры будут, вероятно, играть вечером в более раннее время), и т.д. Но, с учетом того, что среднее количество часов, которые игрок проводит за игрой, является гораздо меньшим, чем количество часов дня, когда игрок, вероятно, играет в игру, только часть от количества пользователей службы 210 хостинга будут использовать ее в данный момент. Для этого анализа предположим, что пиковая нагрузка составляет 12,5%. Соответственно, только 12,5% ресурсов полосы пропускания, сжатия и вычисления используются в данный момент, что в результате приводит только к 12,5% от стоимости аппаратного обеспечения для поддержки данного пользователя для ведения данного уровня выполнения игры благодаря повторному использованию ресурсов.

Кроме того, с учетом того, что некоторые игры и приложения требуют большей вычислительной мощности, чем другие, ресурсы могут выделяться динамически на основе ведущейся игры или приложений, исполняемых пользователями. Соответственно, пользователю, выбирающему приложение или игру с низкими характеристиками, будет выделен низкопроизводительный (менее дорогой) сервер 402, а пользователю, выбирающему приложения или игру с высокими характеристиками, будет выделен высокопроизводительный (более дорогой) сервер 402. Безусловно, данная игра или приложение могут иметь секции с более высокими характеристиками и более низкими характеристиками игры или приложений, и пользователи могут переключать с одного сервера 402 на другой сервер 402 между секциями игры или приложения для поддержки работы пользователя на сервере 402 с самой низкой стоимостью, который удовлетворяет потребностям приложения или игры. Отметим, что дисковые массивы 405 типа RAID, которые намного быстрее, чем один диск, доступны даже низкопроизводительным серверам 402, которые будут иметь преимущество более быстрых дисковых скоростей передачи. Соответственно, средняя стоимость каждого сервера 402 по всем ведущимся играм или используемым приложениям намного меньше, чем стоимость наиболее дорогого сервера 402, на котором исполняются приложения или игра с самыми высокими характеристиками, однако даже низкопроизводительные серверы 402, извлекают преимущества производительности диска из дисковых массивов 405 типа RAID.

Кроме того, сервер 402 в службе 210 хостинга может быть не чем иным как системной платой PC без интерфейсов периферийных или дисковых запоминающих устройств, за исключением сетевого интерфейса, и со временем, может быть интегрирован в однокристалльную интегральную схему только с быстрым сетевым интерфейсом с SAN 403. Кроме того, дисковые массивы 405 типа RAID, вероятно, будут совместно использоваться намного большим количеством пользователей, чем существует дисков, поэтому стоимость дискового запоминающего устройства для каждого активного

пользователя будет намного меньше, чем один накопитель на магнитных дисках. Все это оборудование, вероятно, будет находиться в стойке в обстановке помещения для установки серверов с регулируемыми характеристиками окружающей среды. Если произойдет отказ сервера 402, то он может быть быстро исправлен или заменен в службе 210 хостинга. Напротив, РС или игровая консоль в доме или офисе должны быть жестким, автономным аппаратом, который должен выдерживать приемлемый износ из-за ударов или падений, требует корпуса, имеет, по меньшей мере, один накопитель на магнитных дисках, должен выдерживать неблагоприятные окружающие условия (например, будучи втиснутым в перегретый аудио- и видеокорпус вместе с другим оборудованием), требует гарантийного обслуживания, должен быть упакован и отправлен, и продается розничным торговцем, который, вероятно, получает розничную наценку. Кроме того, РС или игровая консоль должны быть выполнены в соответствии с пиковыми характеристиками ожидаемых приложения или игры, требующих наиболее большого объема вычислений, которые будут использоваться в некоторый момент в будущем, даже несмотря на то, что приложение или игры с более низкими характеристиками (или секции игр или приложений) могут исполняться большую часть времени. И, если произойдет отказ РС или консоли, то их починка является дорогостоящим процессом, отнимающим много времени (с неблагоприятным влиянием на производителя, пользователя и разработчика программного обеспечения).

Соответственно, с учетом того, что система, изображенная на фиг.4а, обеспечивает пользователю опыт использования, сопоставимый с опытом использования локального вычислительного ресурса, для пользователя в доме, офисе или школе, чтобы испытывать данный уровень вычислительных возможностей, намного дешевле обеспечивать эти вычислительные возможности посредством архитектуры, изображенной на фиг.4а.

Устранение необходимости модернизации

Кроме того, пользователи больше не должны заботиться о модернизации РС и/или консолей для игры в новые игры или управления новыми приложениями с более высокими характеристиками. Любая игра или приложения в службе 210 хостинга, независимо от того, какой тип сервера 402 требуется для этой игры или приложений, доступны пользователю, и все игры и приложения исполняются почти немедленно (т.е. с быстрой загрузкой из дисковых массивов 405 типа RAID или локального запоминающего устройства на серверах 402) и должным образом с последними обновлениями и исправлениями ошибок (т.е. разработчики программного обеспечения могут выбрать идеальную конфигурацию сервера для сервера(ов) 402, который(ые) исполняет(ют) данную игру или приложение, и затем сконфигурировать сервер(ы) 402 с оптимальными драйверами, и впоследствии со временем разработчики могут обеспечивать обновления, исправления ошибок и т.д. для всех копий игры или приложения в службе 210 хостинга одновременно). Безусловно, после того, как пользователь начинает использовать службу 210 хостинга, пользователь, вероятно, обнаружит, что игры и приложения продолжают обеспечивать лучший опыт использования (например, посредством обновлений и/или исправлений ошибок, и может оказаться так, что пользователь обнаружит год спустя, что новая игра или приложение стали доступны в службе 210, которая использует технологию вычислений (например, более высокопроизводительное GPU), которая даже не существовала за год до этого, соответственно, за год до этого было невозможно, чтобы пользователь купил упомянутую технологию, которая могла бы исполнять игру или приложения год спустя. Так как вычислительный ресурс, на котором ведется игра, или который исполняет приложение, является невидимым для пользователя (т.е. с точки зрения пользователя

пользователь просто выбирает игру или приложение, которые начинают исполняться почти немедленно - почти так же, как если бы пользователь переключал телевизионные каналы), аппаратное обеспечение пользователя будет "модернизировано" без пользователя, даже если ему известно о модернизации.

Устранение необходимости резервных копий

Другой важной проблемой для пользователей в компаниях, школах и домах являются резервные копии. Информация, хранящаяся в локальном РС или видеоигровой консоли (например, в случае консоли, рейтинг и достижения в игре пользователя), может быть утрачена в случае сбоя диска или случайного стирания. Существует много доступных приложений, которые обеспечивают ручное или автоматическое резервное копирование для РС, и состояние игровой консоли может выгружаться на онлайнный сервер для резервирования, но локальные резервные копии обычно копируются на другой локальный диск (или другое запоминающее устройство), который должен храниться в каком-нибудь безопасном месте и быть организован, и резервирование в онлайнные службы часто ограничено из-за низкой скорости восходящего потока данных, доступной через типичные низкочастотные подключения к Internet. Посредством службы 210 хостинга по фиг.4а, данные, которые хранятся в дисковых массивах 405 типа RAID, могут быть сконфигурированы с использованием таких способов конфигурации RAID предшествующего уровня техники, известных специалистам в данной области техники, образом, что, если произойдет сбой диска, то данные не будут утрачены, и техник в серверном центре, в котором находится поврежденный диск, будет уведомлен, и тогда (он) заменит диск, который после этого будет автоматически обновлен так, чтобы дисковый массив типа RAID снова был устойчивым к сбоям. Кроме того, так как все накопители на магнитных дисках находятся рядом друг с другом с (обеспечением) быстрых локальных сетей между ними посредством SAN 403, то в серверном центре не трудно организовать, чтобы регулярно создавались резервные копии всех дисковых систем на вспомогательном запоминающем устройстве, которое может храниться в серверном центре или перемещено за его пределы. С точки зрения пользователей службы 210 хостинга, их данные просто все время защищены, и они никогда не должны думать о резервных копиях.

Доступ к демонстрационным версиям

Пользователям часто требуется испытать игры или приложения перед их покупкой. Как описано ранее, существуют средства предшествующего уровня техники для испытания демонстрационной версии (глагольная форма "demo" означает испытывать демонстрационную версию, которая также называется "demo", но как существительное) игр и приложений, но в каждом из них существуют ограничения и/или недостатки. С использованием службы 210 хостинга пользователям легко и удобно испытать демонстрационные версии. На самом деле, все, что пользователь делает, - это выбирает демонстрационную версию посредством интерфейса пользователя (например, интерфейса, описанного ниже) и испытывает ее. Демонстрационная версия загружается на сервер 402, подходящий для демонстрационной версии, почти немедленно, и она исполняется подобно любой другой игре или приложению. Независимо от того, требуется ли для демонстрационной версии очень высокопроизводительный сервер 402 или низкопроизводительный сервер 402, и независимо от того, какого типа домашний или офисный клиент 415 использует пользователь, с точки зрения пользователя, демонстрационная версия просто работает. Издатель программного обеспечения демонстрационной версии или игры или приложения может регулировать то, какую именно демонстрационную версию предоставить пользователю для испытания и на

какое время, и, конечно, эта демонстрационная версия может включать в себя элементы интерфейса пользователя, которые предоставляют пользователю возможность получить доступ к полной версии демонстрируемой игры или приложения.

Так как демонстрационные версии, вероятно, будут предложены за меньшую цену или бесплатно, то некоторые пользователи могут попытаться использовать демонстрационные версии повторно (особенно демонстрационные версии игр, повторная игра в которые может являться развлечением). Служба 210 хостинга может применять различные способы для ограничения использования демонстрационной версии для данного пользователя. Самым прямым подходом является установка ID пользователя для каждого пользователя и ограничение количества испытаний демонстрационной версии, обеспечиваемого для данного ID пользователя. Пользователь, однако, может установить множество ID пользователя, особенно, если они являются бесплатными. Одним способом, направленным на решение этой проблемы является ограничение количества испытаний демонстрационной версии, обеспечиваемого для данного клиента 415. Если клиентом является автономное устройство, то у этого устройства существует серийный номер, и служба 210 хостинга может ограничить количество раз, которое клиент может получать доступ к демонстрационной версии с этим серийным номером. Если клиент 415 исполняется как программное обеспечение на РС или другом устройстве, то серийный номер может быть назначен службой 210 хостинга и сохранен на РС и использоваться для ограничения использования демонстрационной версии, но с учетом того, что РС могут быть перепрограммированы пользователями, и серийный номер может быть стерт или изменен, другим возможным вариантом для службы 210 хостинга является ведение учета адреса протокола управления доступом к среде (MAC) сетевого адаптера РС (и/или другие машинно-зависимые идентификаторы, например, серийные номера накопителей на жестких дисках и т.д.) и ограничение использования демонстрационной версии для него. С учетом того, что MAC-адреса сетевых адаптеров могут быть изменены, однако, это не является надежным способом. Еще одним подходом является ограничение количества испытаний демонстрационной версии для данного IP-адреса. Несмотря на то, что IP-адреса могут периодически переназначаться провайдерами DSL и кабельной модемной связи, на практике это не происходит очень часто, и если можно определить (например, связавшись с ISP) то, что данный IP находится в блоке IP-адресов для доступов к DSL или кабельной модемной связи, связанных с жилыми домами, то для данного дома, как правило, может быть установлено небольшое количество использований демонстрационной версии. Кроме того, может быть множество устройств в доме за маршрутизатором NAT, совместно использующих идентичный IP-адрес, но, как правило, в жилом окружении существует ограниченное количество таких устройств. Если IP-адрес находится в блоке, обслуживаемом компанией, то для компании может быть установлено большее количество демонстрационных версий. Но, в конечном счете, комбинация всех ранее упомянутых подходов является лучшим способом ограничения количества демонстрационных версий для РС. Несмотря на то, что не может существовать надежного способа, посредством которого полного решимости и технически искусного пользователя можно ограничить в количестве повторных использований демонстрационных версий, с созданием большого количества преград можно создать такое достаточное сдерживающее средство, что нарушение режима эксплуатации системы демонстрационной версии станет нецелесообразным для большинства пользователей РС, и скорее они будут использовать демонстрационные версии так, как подразумевалось испытание новых игр и приложений.

Преимущества для школ, компаний и других учреждений

Значительные преимущества, в частности, получают компании, школы и другие учреждения, которые используют систему, изображенную на фиг.4а. В компаниях и школах существуют существенные затраты, связанные с установкой, поддержанием в рабочем состоянии и модернизацией РС, особенно, когда дело доходит до РС для исполнения приложений с высокими характеристиками, например, Maya. Как утверждалось ранее, РС, как правило, используются только часть времени в неделю, и как в доме, стоимость РС с данным уровнем рабочих характеристик является намного выше в окружающей обстановке школы или офиса, чем в окружающей обстановке серверного центра.

В случае больших компаний или школ (например, большие университеты), может оказаться целесообразным, чтобы отделы ИТ таких организаций устанавливали серверные центры и поддерживали компьютеры, которые получают доступ удаленно через соединения уровня LAN. Существует несколько решений для удаленного доступа компьютеров по LAN или через частное широкополосное соединение между офисами. Например, посредством терминального сервера Microsoft Windows Terminal Server или через приложения удаленного доступа к рабочему столу компьютера (virtual network computing), подобные VNC, от RealVNC, Ltd., или через средство тонкого клиента от Sun Microsystems, пользователи могут получать удаленный доступ к РС или серверам, с диапазоном качества по времени отклика графических устройств и практическому опыту пользователя. Кроме того, такие самоуправляемые серверные центры обычно закрепляются за одной компанией или школой и, по существу, не могут использовать преимущества совмещения использования, которое является возможным, когда различные приложения (например, развлекательные приложения и бизнес-приложения) используют идентичные вычислительные ресурсы в разное время недели. Соответственно, у многих компаний и школ отсутствует масштаб, ресурсы или специальные знания для установки серверного центра самостоятельно, в котором существует сетевое соединение со скоростью LAN с каждым пользователем. На самом деле, большой процент школ и компаний имеет идентичные подключения к Internet (например, DSL, кабельные модемы), как и в домах.

Однако в таких организациях может, тем не менее, существовать потребность в очень высокопроизводительных вычислениях, или постоянно или периодически. Например, в маленькой архитектурной фирме может быть только небольшое количество архитекторов, с относительно умеренными вычислительными потребностями при выполнении проектных работ, но ей могут периодически требоваться очень высокопроизводительные вычисления 3D (например, при создании 3D "сквозного пролета" нового архитектурного проекта для клиента). Система, изображенная на фиг.4а, очень хорошо подходит для таких организаций. Упомянутым организациям требуется не что иное, как сетевое соединение вида, идентичного тем, которые предлагаются для домов (например, DSL, кабельные модемы) и, как правило, являются очень экономичными. Они могут или использовать экономичные РС как клиент или совершенно обходиться без РС и использовать экономичные специализированные устройства, которые просто реализуют логику управляющего сигнала и восстановление сжатого видео с малым временем ожидания. Эти признаки, в частности, привлекательны для школ, у которых могут быть проблемы с воровством РС или повреждением компонентов, требующих осторожного обращения, внутри РС.

Такая компоновка решает несколько проблем для таких организаций (и многие из этих преимуществ также совместно используются домашними пользователями,

выполняющими универсальные вычисления. Например, эксплуатационные расходы (которые, в конечном счете, должны возвращаться в некоторой форме к пользователям, чтобы иметь жизнеспособный бизнес) могут быть намного меньше, потому что (а) вычислительные ресурсы совместно используются с другими приложениями, которые имеют разные пиковые периоды использования в течение недели, (б) организации могут получать доступ к (и нести расходы по) высокопроизводительным вычислительным ресурсам только тогда, когда требуется, (с) организации не должны обеспечивать ресурсы для создания резервных копий или каким-либо иным способом поддерживать высокопроизводительные вычислительные ресурсы.

Исключение пиратства

Кроме того, игры, приложения, интерактивные фильмы и т.д., больше нельзя будет незаконно использовать, как в настоящее время. Поскольку игра исполняется в центре обслуживания, пользователям не обеспечивается доступ к лежащей в основе управляющей программы, соответственно, нет ничего, что можно незаконно использовать. Даже если пользователь может скопировать исходный код, то он не может исполнить его на стандартной игровой консоли или домашнем компьютере. Это открывает рынки в таких местах мира, как Китай, где стандартные видеоигры не доступны. Перепродажа использованных игр также не возможна.

Для разработчиков игр, существует меньше сосредоточенных неоднородностей рынка, как имеет место в настоящее время. Служба 210 хостинга может постепенно обновляться с течением времени, по мере изменения требований к играм, в отличие от текущей ситуации, когда совершенно новое поколение технологии вынуждает пользователей и разработчиков осуществлять модернизацию, и разработчик игр зависит от своевременной поставки аппаратной платформы.

Интерактивное потоковое видео

Вышеупомянутые описания обеспечивают широкий диапазон приложений, возможность которых обеспечивается новой лежащей в основе идее интерактивного потокового видео, основанного на общих Internet-технологиях с малым временем ожидания, (которое также неявно включает в себя аудио вместе с видео, как используется в этом описании). Системы предшествующего уровня техники, которые обеспечивают потоковое видео через Internet, обеспечивают возможность только приложениям, которые могут быть реализованы с взаимодействиями с большим временем ожидания. Например, основные управляющие элементы воспроизведения для линейного видео (например, пауза, перемотка, быстрая перемотка вперед) работают, соответственно, с высоким временем ожидания, и можно выбирать между линиями передачи линейного видео. И, как утверждалось ранее, характер некоторых видеоигр обеспечивает возможность их ведения с большим временем ожидания. Но большое время ожидания (или низкий коэффициент сжатия) подходов предшествующего уровня техники для потокового видео жестко ограничивает возможные приложения потокового видео или сужает их применения до специализированных сетевых сред, и даже в таких средах, способы предшествующего уровня техники вводят существенные нагрузки в сети. Технология, описанная в этом документе, открывает дверь для широкого диапазона приложений, возможных с интерактивным потоковым видео с малым временем ожидания через Internet, в частности, для тех, которые обеспечиваются через подключения к Internet потребительского уровня.

На самом деле, с такими маленькими клиентскими устройствами, как клиент 465 по фиг.4с, которого достаточно для обеспечения расширенного практического опыта пользователя с фактически произвольным количеством вычислительной мощности,

произвольным объемом быстрой памяти и с чрезвычайно быстрой работой в сети среди мощных серверов, это обеспечивает возможность новой эры вычислений. Кроме того, так как требования к полосе пропускания не возрастают, по мере роста вычислительной мощности системы (т.е., так как требования к полосе пропускания привязаны только к разрешающей способности дисплея, качеству и частоте кадров), то после

5 к разрешающей способности дисплея, качеству и частоте кадров), то после повсеместного распространения высокоскоростного подключения к Internet (например, посредством всеобщего охвата беспроводной связью с малым временем ожидания), надежного и с достаточно высокой пропускной способностью для удовлетворения потребностей дисплеев 422 всех пользователей, встанет вопрос, являются ли

10 необходимыми толстые клиенты (например, PC или мобильные телефоны, работающие под управлением Windows, Linux, OSX и т.д.), или даже тонкие клиенты (например, Adobe Flash или Java) для типичных потребительских и бизнес-приложений.

Появление интерактивного потокового видео в результате приводит к пересмотру предположений об архитектурах вычислительной системы. Примером этого является

15 вариант осуществления серверного центра службы 210 хостинга, изображенный на фиг.15. Путь видео для буфера задержки и/или групповое видео 1550 является контуром обратной связи, в котором передаваемый группе пользователей интерактивный потоковый видеовыход серверов 1521-1525 приложения/игры подается обратно в серверы 1521-1525 приложения/игры или в реальном времени по пути 1552 или согласно

20 выбираемой задержке по пути 1551. Это обеспечивает возможность широкого диапазона практических приложений (например, таких, которые изображены на фиг.16, фиг.17 и фиг.20), которые были бы или невозможными или неосуществимыми посредством локальных архитектур вычислительной системы или сервера предшествующего уровня техники. Но, в качестве более общего архитектурного признака, то, что обеспечивает

25 контур 1550 обратной связи, является рекурсией на уровне интерактивного потокового видео, так как видео может возвращаться обратно неограниченно, когда приложение требует его. Это обеспечивает широкий диапазон возможностей приложения, которые раньше не были доступны.

Другим ключевым архитектурным признаком является то, что видеопотоки являются

30 однонаправленными потоками UDP. Это обеспечивает возможность фактически произвольного уровня групповой передачи интерактивного потокового видео (для сравнения, двусторонние потоки, например, потоки TCP/IP, создавали бы все больше заторов трафика в сетях из-за передачи информации туда и обратно по мере увеличения количества пользователей). Групповая передача является важным средством в серверном

35 центре, потому что она обеспечивает возможность системе быстро реагировать на возрастающие потребности пользователей Internet (и, на самом деле, населения мира) в осуществлении связи на основе "один ко многим", или даже "многие ко многим". И опять же, примеры, обсуждаемые в этом описании, например, по фиг.16, на которой изображено использование и рекурсии интерактивного потокового видео и групповой

40 передачи, являются только видимой частью огромного айсберга возможностей.

Нетранзитный пиринг

В одном варианте осуществления служба 210 хостинга имеет одно или несколько соединений согласно пирингу с одним или несколькими провайдерами Internet-услуг (ISP), которые также оказывают Internet-услуги пользователям, и, следовательно, служба

45 хостинга 210 может обмениваться информацией с пользователем через нетранзитный маршрут, который остается внутри сети этого ISP. Например, если интерфейс 441 WAN службы 210 хостинга непосредственно соединен с сетью Comcast Cable Communications, Inc, и на территории 211 пользователя обеспечена услуга широкополосного доступа с

кабельным модемом Comcast, то маршрут между службой 210 хостинга и клиентом 415 может быть установлен полностью внутри сети Comcast. Потенциально возможные преимущества этого включают в себя меньшую стоимость передачи информации (так как можно избежать затрат на IP-транзит между двумя или несколькими сетями ISP),
 5 потенциально более надежное соединение (в случае перегрузки или других разрывов транзита между сетями ISP) и меньшее время ожидания (в случае перегрузки, неэффективных маршрутов или других задержек между сетями ISP).

В этом варианте осуществления, когда клиент 415 первоначально связывается со службой 210 хостинга в начале сеанса, служба 210 хостинга принимает IP-адрес
 10 территории 211 пользователя. После этого она использует доступные таблицы IP-адресов, например, от ARIN (Американский реестр адресов в Internet), чтобы проверить, является ли IP-адрес адресом, выделенным конкретному ISP, соединенному со службой 210 хостинга, который может маршрутизировать до территории 211 пользователя без IP-транзита через другой ISP. Например, если IP-адрес находится между 76.21.0.0 и
 15 76.21.127.255, то IP-адрес назначен Comcast Cable Communications, Inc. В этом примере, если служба 210 хостинга поддерживает соединения с ISP Cox, AT&T и Comcast, то она, наиболее вероятно, выберет Comcast в качестве ISP для обеспечения оптимального маршрута к конкретному пользователю.

Сжатие видео с использованием обратной связи

20 В одном варианте осуществления обеспечивают обратную связь из клиентского устройства в службу хостинга для указания успешной (или не успешной) доставки кадра и/или фрагмента. Информация обратной связи, обеспечиваемая из клиента, далее используется для корректировки операций сжатия видео в службе хостинга.

Например, на фиг.25a-фиг.25b изображен один вариант осуществления изобретения, в котором канал обратной 2501 связи установлен между клиентским устройством 205
 25 и службой 210 хостинга. Канал 2501 обратной связи используется клиентским устройством 205 для отправки квитанций в виде пакетов успешно принятых фрагментов/кадров и/или указаний о неуспешно принятых фрагментах/кадрах.

В одном варианте осуществления, после успешного приема каждого фрагмента/
 30 кадра, клиент передает сообщение-подтверждение в службу 210 хостинга. В этом варианте осуществления служба 210 хостинга выявляет потерю пакетов, если она не принимает квитанцию после заданного периода времени, и/или если она принимает квитанцию о том, что клиентское устройство 205 приняло более поздний фрагмент/кадр чем тот, который отправлен. В качестве альтернативы или дополнительно,
 35 клиентское устройство 205 может выявлять потерю пакетов и передавать указание о потере пакетов в службу 210 хостинга вместе с указанием потерянных пакетов фрагментов/кадров. В этом варианте осуществления не требуется непрерывное квитирование успешно доставленных фрагментов/кадров.

Независимо от того, как выявляют потерю пакетов, в варианте осуществления, изображенном на фиг.25a-фиг.25b, после формирования начального набора I-фрагментов для изображения (не изображено на фиг.25a), кодер впоследствии формирует только R-фрагменты до тех пор, пока не выявлена потеря пакетов. Отметим, что на фиг.25a, каждый кадр, например 2510, изображен как 4 вертикальных фрагмента. Кадр может фрагментироваться в разной конфигурации, например, 2x2, 2x4, 4x4 и т.д., или кадр
 45 может быть закодирован целиком без фрагментов (т.е. как 1 большой фрагмент). Вышеуказанные примеры конфигураций фрагментации кадра обеспечены для иллюстрации этого варианта осуществления изобретения. Лежащие в основе принципы изобретения не ограничиваются какой-либо конкретной конфигурацией фрагментации

кадра.

Передача только Р-фрагментов уменьшает требования к ширине полосы канала по всем вышеизложенным причинам (т.е. Р-фрагменты обычно меньше, чем I-фрагменты). Когда выявляют потерю пакетов посредством канала 2501 обратной связи, кодер 2500 формирует новые I-фрагменты, как изображено на фиг.25b, для повторной инициализации состояния декодера 2502 на клиентском устройстве 205. Как изображено, в одном варианте осуществления, I-фрагменты распространяются на несколько закодированных кадров для ограничения ширины полосы, используемой каждым отдельным закодированным кадром. Например, на фиг.25, на которой каждый кадр включает в себя 4 фрагмента, один I-фрагмент передается в разных позициях в пределах 4 последовательных закодированных кадров.

Кодер 2500 может комбинировать способы, описанные в отношении этого варианта осуществления, с другими способами кодирования, описанными в этом документе. Например, в дополнение к формированию I-фрагментов в ответ на выявленную потерю пакетов кодер 2500 может формировать I-фрагменты при других обстоятельствах, при которых I-фрагменты могут быть полезными для надлежащей визуализации последовательности изображений (например, в ответ на внезапные монтажные переходы).

На фиг.26a изображен другой вариант осуществления изобретения, который основывается на канале 2601 обратной связи между клиентским устройством 205 и службой 210 хостинга. Вместо формирования новых I-фрагментов/кадров в ответ на выявленную потерю пакетов, кодер 2600 этого варианта осуществления корректирует зависимости Р-фрагментов/кадров. Сначала следует отметить, что конкретные детали, изложенные в этом примере, не требуются для выполнения лежащих в основе принципов изобретения. Например, несмотря на то, что этот пример описывается с использованием Р-фрагментов/кадров, лежащие в основе принципы изобретения не ограничиваются конкретным форматом кодирования.

На Фиг.26a, кодер 2600 кодирует множество несжатых фрагментов/кадров 2605 во множество Р-фрагментов/кадров 2606 и передает эти Р-фрагменты/кадры по каналу связи (например, по Internet) в клиентское устройство 205. Декодер 2602 на клиентском устройстве 205 декодирует Р-фрагменты/кадры 2606 для формирования множества восстановленных из сжатых данных фрагментов/кадров 2607. Прошлые(ые) состояние(я) 2611 кодера 2600 сохраняются на запоминающем устройстве 2610 в службе 210 хостинга, и прошлые(ые) состояние(я) 2621 декодера 2602 сохраняются на запоминающем устройстве 2620 в клиентском устройстве 205. "Состояние" декодера является известным термином в области техники систем кодирования видеосигнала, например, MPEG-2 и MPEG-4. В одном варианте осуществления прошлое "состояние", сохраняемое в блоках памяти, содержит комбинированные данные из предшествующих Р-фрагментов/кадров. Блоки памяти 2611 и 2621 могут быть интегрированы внутри кодера 2600 и декодера 2602, соответственно, а не отделены от кодера 2600 и декодера 2602, как изображено на фиг.26a. Кроме того, различные типы памяти могут использоваться с включением, например, оперативного запоминающего устройства.

В одном варианте осуществления, когда не происходит потеря пакетов, кодер 2600 кодирует каждый Р-фрагмент/кадр так, что он зависит от предыдущего Р-фрагмента/кадра. Соответственно, как обозначено посредством системы обозначений, используемой на фиг.26a, Р-фрагмент/кадр 4 зависит от Р-фрагмента/кадра 3 (идентифицируется с использованием обозначения 4₃), Р-фрагмент/кадр 5 зависит от Р-фрагмента/кадра 4 (идентифицируется с использованием обозначения 5₄), и Р-

фрагмент/кадр 6 зависит от Р-фрагмента/кадра 5 (идентифицируется с использованием обозначения 6₅). В этом примере во время передачи между кодером 2600 и декодером 2602 потерян Р-фрагмент/кадр 4₃. Потеря может быть передана в кодер 2600 различными способами, включающими в себя, например, описанные выше. Например, каждый раз, когда декодер 2606 успешно принимает и/или декодирует фрагмент/кадр, эта информация может быть передана из декодера 2602 в кодер 2600. Если по истечении периода времени кодер 2600 не принимает указание на то, что конкретный фрагмент/кадр принят и/или декодирован, то кодер 2600 предполагает, что этот фрагмент/кадр не был успешно принят. В качестве альтернативы или в дополнение, декодер 2602 может уведомлять кодер 2600, когда конкретный фрагмент/кадр не был успешно принят.

В одном варианте осуществления, независимо от того, как выявляют потерянный фрагмент/кадр, как только это происходит, кодер 2600 кодирует следующий фрагмент/кадр с использованием последнего фрагмента/кадра, о котором известно, что он успешно принят декодером 2602. В примере, изображенном на фиг.26а, считается, что фрагменты/кадры 5 и 6 не были “успешно приняты”, потому что они не могут быть должным образом декодированы декодером 2602 из-за потери фрагмента/кадра 4 (т.е. декодирование фрагмента/кадра 5 зависит от фрагмента/кадра 4, и декодирование фрагмента/кадра 6 зависит от фрагмента/кадра 5). Соответственно, в примере, изображенном на фиг.26а, кодер 2600 кодирует фрагмент/кадр 7, который зависит от фрагмента/кадра 3 (последний успешно принятый фрагмент/кадр), а не от фрагмента/кадра 6, который декодер 2602 не может должным образом декодировать. Несмотря на то, что не изображено на фиг.26а, фрагмент/кадр 8 впоследствии кодируется так, что он зависит от фрагмента/кадра 7, и фрагмент/кадр 9 кодируется так, что он зависит от фрагмента/кадра 8, с предположением о том, что дополнительные потери пакетов не выявлены.

Как упоминалось выше, как кодер 2600, так и декодер 2602 поддерживают прошлые состояния, 2611 и 2621, кодера и декодера в блоках памяти 2610 и 2620, соответственно. Соответственно, при кодировании фрагмента/кадра 7, кодер 2600 извлекает предшествующее состояние кодера, связанное с фрагментом/кадром 3 из памяти 2610. Аналогично в памяти 2620, связанной с декодером 2602, сохраняется, по меньшей мере, последнее заведомо исправное состояние декодера (состояние, связанное с Р-фрагментом/кадром 3 в примере). Следовательно, декодер 2602 извлекает прошлую информацию о состоянии, связанную с фрагментом/кадром 3, для возможности декодирования фрагмента/кадра 7.

В результате способов, описанных выше, интерактивное видео в реальном времени с малым временем ожидания может кодироваться и передаваться потоком с использованием относительно небольшой ширины полосы, потому что никогда не требуются I-фрагменты/кадры (за исключением случаев инициализации декодера и кодера в начале потока). Кроме того, в то время как видеоизображение, выводимое декодером, может временно включать в себя нежелательное искажение, получающееся в результате потери фрагмента/кадра 4 и фрагментов/кадров 5 и 6 (которые не могут быть должным образом декодированы из-за потери фрагмента/кадра 4), это искажение визуализируется в течение очень короткого времени. Кроме того, если используются фрагменты (а не полные видеокадры), то искажение ограничивается до конкретной области визуализированного видеоизображения.

На фиг.26b изображен способ согласно одному варианту осуществления изобретения. В позиции 2650, формируется фрагмент/кадр на основе ранее сформированного фрагмента/кадра. В позиции 2651, выявляют потерянный фрагмент/кадр. В одном

варианте осуществления потерянный фрагмент/кадр выявляют на основе информации, передаваемой из кодера в декодер, как описано выше. В позиции 2652, следующий фрагмент/кадр формируется на основе фрагмента/кадра, о котором известно, что он успешно принят и/или декодирован в декодере. В одном варианте осуществления, кодер формирует следующий фрагмент/кадр с загрузкой состояния, связанного с успешно принятым и/или декодированным фрагментом/кадром, из памяти. Аналогично, когда декодер принимает новый фрагмент/кадр, он декодирует этот фрагмент/кадр с загрузкой состояния, связанного с успешно принятым и/или декодированным фрагментом/кадром, из памяти.

В одном варианте осуществления следующий фрагмент/кадр формируется на основе последнего фрагмента/кадра, успешно принятого и/или декодированного в кодере. В другом варианте осуществления следующий сформированный фрагмент/кадр является I-фрагментом/кадром. В еще одном варианте осуществления, выбор того, формировать ли следующий фрагмент/кадр на основе ранее успешно принятого фрагмента/кадра или как I-кадр, основан на том, сколько фрагментов/кадров были потеряны, и/или на задержке канала. В ситуации, когда потеряно относительно небольшое количество (например, 1 или 2) фрагментов/кадров, и время ожидания (передачи сигнала) туда и обратно относительно мало (например, 1 или 2 периода кадра), то оптимальным может являться формирование P-фрагмента/кадра, так как разность между последним успешно принятым фрагментом/кадром и вновь формируемым кадром может являться относительно небольшой. Если потеряно несколько фрагментов/кадров, или время ожидания (передачи сигнала) туда и обратно является большим, то оптимальным может являться формирование I-фрагмента/кадра, так как разность между последним успешно принятым фрагментом/кадром и вновь формируемым фрагментом/кадром может являться большой. В одном варианте осуществления, устанавливается значение порога времени ожидания и/или порога потери фрагмента/кадра для определения того, передавать I-фрагмент/кадр или P-фрагмент/кадр. Если количество потерянных фрагментов/кадров ниже порога потери фрагмента/кадра, и/или, если время ожидания (передачи сигнала) туда и обратно ниже значения порога времени ожидания, то формируется новый I-фрагмент/кадр, в противном случае формируется новый P-фрагмент/кадр.

В одном варианте осуществления, кодер всегда пытается сформировать P-фрагмент/кадр относительно последнего успешно принятого фрагмента/кадра, и, если в процессе кодирования кодер определит, что P-фрагмент/кадр вероятно больше, чем I-фрагмент/кадр (например, если он сжал 1/8-ую фрагмента/кадра, и сжатый размер больше, чем 1/8-ая размера среднего I-фрагмента/кадра, сжатого ранее), то тогда кодер отказывается от сжатия P-фрагмента/кадра, и вместо этого сжимает I-фрагмент/кадр.

Если потеря пакетов происходит не часто, то системы, описанные выше, с использованием обратной связи для сообщения о сброшенном фрагменте/кадре, как правило, в результате приводят к очень небольшому разрыву в видеопотоке для пользователя, потому что фрагмент/кадр, который был разрушен потерянным пакетом, заменяется примерно через промежуток времени в один (проход сигнала) туда и обратно между клиентским устройством 205 и службой 210 хостинга при предположении о том, что кодер 2600 сжимает фрагмент/кадр за короткое время. И, (вследствие) нового фрагмента/кадра, который сжимают на основе более позднего кадра в несжатом видеопотоке, видеопоток не отстает от несжатого видео потока. Но, если пакет, содержащий новый фрагмент/кадр, также потерян, то это в результате приводит к задержке, по меньшей мере, в два (прохода сигнала) туда и обратно для еще одного

повторного запроса и отправки другого нового фрагмента/кадра, которая, во многих практических ситуациях, в результате приводит к заметному разрыву в видеопотоке. Как следствие, очень важно, чтобы вновь кодируемый фрагмент/кадр, отправляемый после сброшенного фрагмента/кадра, был успешно отправлен из службы 210 хостинга в клиентское устройство 205.

В одном варианте осуществления, используются способы кодирования с прямой коррекцией ошибок (FEC), например, способы, описанные ранее, и изображенные на фиг.11a, фиг.11b, фиг.11c и фиг.11d, для уменьшения вероятности потери вновь кодируемого фрагмента/кадра. Если кодирование FEC уже используется при передаче фрагментов/кадров, то используется более мощный код FEC для вновь кодируемого фрагмента/кадра.

Одной потенциально возможной причиной сброшенных пакетов является внезапная потеря в ширине полосы канала, например, если некоторый другой пользователь широкополосного соединения на территории 211 пользователя начал использование большого объема ширины полосы. Если вновь формируемый фрагмент/кадр также потерян из-за сброшенных пакетов (даже если используется FEC), то в одном варианте осуществления, когда служба 210 хостинга уведомляется клиентом 415 о том, что сброшен второй вновь закодированный фрагмент/кадр, устройство 404 сжатия видео уменьшает скорость передачи данных при кодировании последующего вновь кодируемого фрагмента/кадра. В разных вариантах осуществления скорость передачи данных уменьшают с использованием разных способов. Например, в одном варианте осуществления, это уменьшение скорости передачи данных достигается с понижением качества кодируемого фрагмента/кадра, с увеличением коэффициента сжатия. В другом варианте осуществления, скорость передачи данных уменьшается посредством понижения частоты кадров видео (например, с 60кадр/с до 30кадр/с) и с соответствующим замедлением скорости передачи данных. В одном варианте осуществления, используются оба способа уменьшения скорости передачи данных (например, и уменьшение частоты кадров, и увеличение коэффициента сжатия). Если эта меньшая скорость передачи данных является успешной в уменьшении сброшенных пакетов, то в соответствии со способами регулировки и выявления скорости передачи данных по каналу, описанными ранее, служба 210 хостинга продолжает кодировать с меньшей скоростью передачи данных и затем постепенно корректирует скорость передачи данных вверх или вниз согласно предоставляемым возможностям канала. Непрерывный прием данных обратной связи, относящихся к сброшенным пакетам и/или времени ожидания, позволяет службе 210 хостинга динамически корректировать скорость передачи данных на основе текущих условий на канале.

Управление состоянием в системе онлайн-игр

В одном варианте осуществления изобретения используются способы для эффективного хранения и переноса текущего состояния активной игры между серверами. Несмотря на то, что варианты осуществления, описанные в этом документе, относятся к онлайн-играм, лежащие в основе принципы изобретения могут использоваться для различных других типов приложений (например, приложений для проектирования, текстовых процессоров, связанного программного обеспечения, например, электронной почты или мгновенного обмена сообщениями и т.д.). На фиг.27a изображена иллюстративная архитектура системы для реализации этого варианта осуществления, и на фиг.27b изображен иллюстративный способ. Несмотря на то, что способ и архитектура системы описываются параллельно, способ, изображенный на фиг.27b, не ограничивается конкретной архитектурой системы.

В позиции 2751 на фиг.27b, пользователь инициирует новую онлайн-игру в службе 210a хостинга из клиентского устройства 205. В ответ, в позиции 2752, из запоминающего устройства загружается образ "хорошего качества" игры 2702a (например, с жесткого диска, независимо от того, при непосредственном соединении с сервером, исполняющем игру, или при соединении с сервером через сеть) в память (например, RAM) в службе 210a хостинга. Образ "хорошего качества" содержит данные и код программы времени прогона для игры до инициирования какого-либо воспроизведения игры (например, как тогда, когда игра исполняется впервые). Пользователь далее воспроизводит игру, в позиции 2753, и вызывает изменение образа "хорошего качества" на образ плохого качества (например, исполняющаяся игра, представленная "Состоянием А" на фиг.27a). В позиции 2754, игра приостанавливается или заканчивается, или пользователем или службой 210a хостинга. В позиции 2755, логика 2700a управления состоянием в службе 210a хостинга определяет разности между образом "хорошего качества" игры и текущим состоянием игры ("Состояние А"). Для вычисления разности между двумя двоичными образами могут использоваться различные известные способы, включающие в себя, например, способы, используемые в известной утилите "diff", доступной в операционной системе Unix. Само собой разумеется, что лежащие в основе принципы изобретения не ограничиваются какими-либо конкретными способами вычисления разности.

Независимо от того, как вычисляются разности, после их вычисления разностные данные хранятся локально в запоминающем устройстве 2705a и/или передаются в другую службу 210b хостинга. Если разностные данные передаются в другую службу 210b хостинга, то они могут сохраняться на запоминающем устройстве (не изображено) в этой новой службе 210b хостинга. В любом случае, разностные данные связываются с учетной записью пользователя в службах хостинга для возможности его идентификации в следующий раз, когда пользователь входит в службы хостинга и инициирует игру. В одном варианте осуществления, вместо немедленной передачи разностных данных, их не передают в новую службу хостинга до следующего раза, когда пользователь попытается воспроизвести игру (и другую службу хостинга идентифицируют как лучший выбор для размещения игры).

Возвращаясь к способу, изображенному на фиг.27b, в позиции 2757, пользователь повторно инициирует игру из клиентского устройства, которое может быть идентичным клиентским устройством 205, из которого пользователь первоначально воспроизводил игру, или другим клиентским устройством (не изображено). В ответ, в позиции 2758, логика 2700b управления состоянием в службе 210b хостинга извлекает образ "хорошего качества" игры из запоминающего устройства и разностные данные. В позиции 2759, логика 2700b управления состоянием комбинирует образ хорошего качества и разностные данные для восстановления состояния, в котором была игра в исходной службе 210a хостинга ("Состояние А"). Для восстановления данных состояния двоичного образа с использованием разностных данных могут использоваться различные известные способы, включающие в себя, например, способы, используемые в известной утилите "patch", доступной в операционной системе Unix. Также могут использоваться способы вычисления разности, используемые в известных программах резервного копирования, например, Backup PC. Лежащие в основе принципы изобретения не ограничиваются конкретными способами использования разностных данных для восстановления данных двоичного образа.

Кроме того, в позиции 2760, зависящие от платформы данные 2710 включаются в конечный образ 2701b игры. Зависящие от платформы данные 2710 могут включить в

себя любые данные, которые являются уникальными для платформы сервера-адресата. Например, зависящие от платформы данные 2710 могут включать в себя адрес Управления доступом к среде (MAC) новой платформы, адреса TCP/IP, время дня, серийные номера аппаратных средств (например, для жесткого диска и центрального процессора), адреса сетевого сервера (например, серверов DHCP/Win) и серийный(ые) номер(а)/код(ы) активизации программного обеспечения (включающие в себя серийный (ые) номер(а)/код(ы) активизации Операционной системы).

Другие зависящие от платформы данные, относящиеся к клиенту/пользователю, могут включать в себя, например, следующие:

1. Разрешающая способность экрана пользователя. Когда пользователь возобновляет игру, он может использовать другое устройство с другой разрешающей способностью.

2. Конфигурация контроллера пользователя. Когда игра возобновляется, пользователь может переключаться с игрового контроллера на клавиатуру/мышь.

3. Пользовательские права, например, истек ли (срок) тарифа со скидкой (например, если пользователь играл в игру во время льготного периода и теперь играет во время обычного периода за более высокую плату), или, существуют ли у пользователя или устройства определенные ограничения по возрасту (например, родители пользователя могут изменить параметры настройки для ребенка так, что ребенку не позволяют смотреть материал для взрослых, или, существуют ли у устройства, воспроизводящего игру, (например, у компьютера в публичной библиотеке) определенные ограничения на то, может ли материал для взрослых выводиться на экран).

4. Рейтинг пользователя. Пользователю могли позволить играть в игру с несколькими участниками в определенной лиге, но, так как у некоторых других пользователей рейтинг превысил рейтинг этого пользователя, то пользователь мог быть понижен до низшей лиги.

Вышеуказанные примеры зависящих от платформы данных 2710 обеспечены для иллюстрации этого варианта осуществления изобретения. Лежащие в основе принципы изобретения не ограничиваются каким-либо конкретным набором зависящих от платформы данных.

На фиг.28 наглядно изображено то, как логика 2700a управления состоянием в первой службе хостинга извлекает разностные данные 2800 из исполняющейся игры 2701a. Логика 2700b управления состоянием во второй службе хостинга далее комбинирует образ 2702b хорошего качества с разностными данными 2800 и зависящими от платформы данными 2710 для восстановления состояния исполняющейся игры 2701b. Как, в общем, изображено на фиг.28, размер разностных данных значительно меньше, чем размер всего образа 2701a игры, и, следовательно, с сохранением/передачей только разностных данных сохраняется значительный объем пространства памяти и ширины полосы. Несмотря на то, что не изображено на фиг.28, зависящие от платформы данные 2700 могут перезаписывать некоторые из разностных данных, когда они являются включенными в конечный образ 2701b игры.

Несмотря на то, что выше описана реализация онлайн-видеоигр, лежащие в основе принципы изобретения не ограничиваются видеоиграми. Например, вышеизложенные способы управления состоянием могут быть осуществлены в контексте любого типа размещаемого онлайн приложения.

Способы поддержания клиентского декодера

В одном варианте осуществления изобретения, служба 210 хостинга передает новый декодер в клиентское устройство 205 каждые раз, когда пользователь запрашивает соединение со службой 210 хостинга. Следовательно, в этом варианте осуществления,

декодер, используемый клиентским устройством, всегда относится к данному моменту времени и однозначно специально приспособляется для аппаратных средств/программного обеспечения, реализованных на клиентском устройстве.

Как изображено на фиг.29, в этом варианте осуществления, приложение, которое постоянно установлено на клиентском устройстве 205, не включает в себя декодер. Наоборот, клиентское приложение 2903 загрузчика управляет загрузкой и установкой временного декодера 2900 каждые раз, когда клиентское устройство 205 соединяется со службой 210 хостинга. Приложение 2903 загрузчика может быть реализовано в аппаратных средствах, программном обеспечении, программно-аппаратных средствах или любой их комбинации. В ответ на запрос пользователя на новый сеанс в режиме онлайн, приложение 2903 загрузчика передает информацию, относящуюся к клиентскому устройству 205 по сети (например, по Internet). Информация может включать в себя идентификационные данные, идентифицирующие клиентское устройство, и/или конфигурацию аппаратных средств/программного обеспечения клиентского устройства (например, процессора, операционной системы и т.д.).

На основе этой информации, приложение 2901 загрузчика в службе 210 хостинга выбирает соответствующий временный декодер 2900 для использования на клиентском устройстве 205. Приложение 2901 загрузчика в службе хостинга после этого передает временный декодер 2900, и приложение 2903 загрузчика на клиентском устройстве верифицирует и/или устанавливает декодер на клиентском устройстве 205. Кодер 2902 после этого кодирует содержимое аудио/видео с использованием какого-либо из способов, описанных в этом документе, и передает содержимое 2910 в декодер 2900. После установки нового декодера 2900, он декодирует содержимое для текущего сеанса в режиме онлайн (т.е. с использованием одного или нескольких способов восстановления из сжатых данных аудио/видео, описанных в этом документе). В одном варианте осуществления, когда сеанс завершен, декодер 2900 удаляется (например, деинсталлируется) из клиентского устройства 205.

В одном варианте осуществления приложение 2903 загрузчика получает характеристики канала, поскольку временный декодер 2900 загружается с выполнением оценок канала, например, скорости передачи данных, которую можно достичь на канале, (например, посредством определения времени загрузки данных, частоты потери пакетов на канале и задержки канала. Приложение 2903 загрузчика формирует данные характеристики канала, описывающие оценки канала. Эти данные характеристики канала после этого передаются из клиентского устройства 205 в загрузчик 2901 службы хостинга, который использует эти данные характеристики канала для определения того, как лучше всего использовать канал для передачи мультимедиа в клиентское устройство 205.

Клиентское устройство 205 обычно отправляет сообщения обратно в службу 205 хостинга во время загрузки временного декодера 2900. Эти сообщения могут включать в себя сообщения квитирования, указывающие на то, приняты ли пакеты без ошибок или с ошибками. Кроме того, сообщения обеспечивают обратную связь с загрузчиком 2901 относительно скорости передачи данных (вычисляемой на основе скорости, с которой принимаются пакеты), частоты ошибок пакета (на основе процентного отношения пакетов, о которых сообщалось, что они приняты с ошибками), и времени ожидания для (передачи сигнала) туда и обратно по каналу (на основе времени, которое проходит до того, как загрузчик 2901 примет обратную связь о данном пакете, который был передан).

Например, если определено, что скорость передачи данных равна 2 Мбит/с, то

загрузчик может выбрать меньшую разрешающую способность видеокна для кодера 2902 (например Ж, 640×480 при 60кадр/с), чем в случае определения того, что скорость передачи данных равна 5 Мбит/с (например, 1280×720 в 60кадр/с). Могут быть выбраны другие структуры пакета или прямой коррекции ошибок (FEC), в зависимости от частоты

5 потери пакетов.

Если потеря пакетов находится на очень низком уровне, то сжатые аудио и видео могут передаваться без какого-либо исправления ошибок. Если потеря пакетов находится на среднем уровне, то сжатые аудио и видео могут передаваться посредством способов кодирования с исправлением ошибок (например, описанных ранее и

10 изображенных на фиг.11a, фиг.11b, фиг.11c и фиг.11d). Если потеря пакетов находится на очень высоком уровне, то может быть определено, что аудиовизуальный поток соответствующего качества не может быть передан, и клиентское устройство 205 может или уведомлять пользователя о том, что служба хостинга не доступна через этот канал

15 связи (т.е. "линию связи"), или оно может попытаться установить другой маршрут в службу хостинга, который имеет более низкий уровень потери пакетов (как описано ниже).

Если время ожидания является малым, то сжатые аудио и видео могут быть переданы с малым временем ожидания, и сеанс может быть установлен. Если время ожидания является слишком большим (например, более 80мс), то для игр, которые требуют малого

20 времени ожидания, клиентское устройство 205 может или уведомлять пользователя о том, что служба хостинга не доступна через эту линию связи, что линия связи доступна, но отклик на ввод пользователя будет медленным или "с запаздыванием", или, что пользователь может попытаться установить другой маршрут к службе хостинга, который имеет меньшее время ожидания (как описано ниже).

Клиентское устройство 205 может попытаться соединиться со Службой 210 хостинга через другой маршрут через сеть (например, Internet), чтобы увидеть, уменьшились ли

искажения (например, уменьшилась потеря пакетов, уменьшилось время ожидания, или даже повысилась ли скорость передачи данных). Например, Служба 210 хостинга может подключаться к сети Internet из множества географических местоположений

30 (например, хостинг-центр в Лос-Анджелесе и хостинг-центр в Денвере), и может существовать высокий уровень потери пакетов из-за перегрузки в Лос-Анджелесе, но в Денвере не существует перегрузки. Кроме того, Служба 210 хостинга может подключаться к сети Internet через множество провайдеров Internet-услуг (например, AT&T и Comcast).

Перегрузка или другие проблемы между клиентским устройством 205 и одним из поставщиков услуг (например, AT&T) могут в результате приводить к потере пакетов и/или большому времени ожидания и/или ограниченной скорости передачи данных. Однако если Клиентское устройство 205 соединяется со службой 210 хостинга через

другого поставщика услуг (например, Comcast), то оно может соединиться без проблем

40 перегрузки, и/или с меньшей потерей пакетов, и/или с меньшим временем ожидания, и/или с более высокой скоростью передачи данных. Соответственно, если в клиентском устройстве 205 потеря пакетов выше заданного порога (например, конкретное количество сброшенных пакетов за заданный отрезок времени), время ожидания выше заданного порога, и/или скорость передачи данных ниже заданного порога, то при

45 загрузке временного декодера 2900, в одном варианте осуществления, оно пытается повторно соединиться со службой 210 хостинга через альтернативный маршрут (обычно посредством соединения с другим IP-адресом или другим именем домена) для определения того, можно ли получить лучшее соединение.

Если после того, как альтернативные варианты соединения исчерпаны, при соединении все еще существуют недопустимые искажения, то это может происходить из-за того, что искажения существуют в локальном подключении клиентского устройства 205 к Internet, или что оно находится слишком далеко от службы 210 хостинга для достижения соответствующего времени ожидания. В таком случае клиентское устройство 205 может уведомлять пользователя о том, что Служба хостинга не доступна через эту линию связи, или о том, что она доступна только с искажениями, и/или доступны только определенные типы игр/приложений с малым временем ожидания.

Так как эта оценка и потенциально возможное улучшение характеристик линии связи между Службой 210 хостинга и Клиентским устройством 205 происходит во время загрузки временного декодера, то это уменьшает время, необходимое клиентскому устройству 205 для выполнения отдельно загрузки временного декодера 2900 и оценки характеристик линии связи. Тем не менее, в другом варианте осуществления, оценка и потенциально возможное улучшение характеристик линии связи выполняется клиентским устройством 205 отдельно от загрузки временного декодера 2900 (например, при использовании данных модельных испытаний, а не кода программы декодера). Существует несколько причин, по которым это может являться предпочтительной реализацией. Например, в некоторых вариантах осуществления, клиентское устройство 205 реализовано частично или полностью в аппаратных средствах. Соответственно, для этих вариантов осуществления, не существует программного декодера, по сути необходимого для загрузки.

Сжатие с использованием размеров фрагмента на основе стандартов

Как упоминалось выше, когда используется сжатие на основе фрагмента, лежащие в основе принципы изобретения не ограничиваются конкретной ориентацией, формой и размером фрагмента. Например, в системе сжатия на основе DCT, например, MPEG-2 и MPEG-4, фрагменты могут иметь размер макроблоков (компонентов, используемых в сжатии видео, которые обычно представляют блок 16 на 16 пикселей). В этом варианте осуществления обеспечивает очень высокий уровень степени детализации для работы с фрагментами.

Кроме того, независимо от размера фрагмента, могут использоваться различные типы схем циклического сдвига. Например, на фиг.30 изображен вариант осуществления, в котором в каждом R-кадре 3001-3004 используется множество I-фрагментов. Используется схема циклического сдвига, в которой I-фрагменты рассредоточены по всему кадру в каждом R-кадре так, что полный I-кадр формируется каждым набором из четырех R-кадров. Рассредоточение I-фрагментов таким способом уменьшает воздействия потери пакетов (с ограничением потери до небольшой области дисплея).

Размер фрагментов может также задаваться согласно встроенной родной структуре лежащего в основе алгоритма сжатия. Например, если используется алгоритм сжатия H.264, в одном варианте осуществления, то размер фрагментов устанавливается равным размеру "слайсов" H.264. Это обеспечивает возможность способам, описанным в этом документе, легко интегрироваться в контекст различных других стандартных алгоритмов сжатия, например, H.264 и MPEG-4. После установки размера фрагмента в родную структуру сжатия, могут быть реализованы способы, идентичные описанным выше.

Способы для потоковых операций перемотки назад и воспроизведения

Как описано ранее согласно Фиг.15, несжатый видео/аудио поток 1529, формируемый сервером 1521-1525 приложения/игры, может быть сжат совместно используемым аппаратным сжатием 1530 с множеством разрешающих способностей с одновременным

получением в результате множества сжатых видео/аудио потоков 1539. Например, видео/аудио поток, формируемый сервером 1521 приложения/игры, может быть сжат с 1280×720×60 кадр/с совместно используемым аппаратным сжатием 1530 и передан пользователю через исходящую маршрутизацию 1540 как исходящий трафик 1599 из Internet. Идентичный видео/аудио поток может быть одновременно уменьшен в масштабе до размера миниатюры (например, 200×113) совместно используемым аппаратным сжатием 1530 через путь 1552 (или через буфер 1515 задержки) в сервер 1522 приложения/игры, который выводится на экран как одна миниатюра 1600 из совокупности миниатюр на фиг.16. Когда масштаб миниатюры 1600 изменяют через промежуточный размер 1700 на фиг.17 до размера 1800 (1280×720×60кадр/с) на фиг.18, то вместо восстановления из сжатых данных потока миниатюр, сервер 1522 приложения/игры может восстанавливать из сжатых данных копию потока 1280×720×60кадр/с, отправляемую пользователю сервера 1521 приложения/игры, и масштабировать видео с более высокой разрешающей способностью, когда его размер изменяют в масштабе от миниатюры до размера 1280×720. У этого подхода существует преимущество повторного использования сжатого потока 1280×720 дважды. Но у него существует несколько недостатков: (а) качество изображения сжатого видеопотока, отправляемого пользователю, может меняться, если пропускная способность соединения с Internet пользователя меняется, что в результате приводит к меняющемуся качеству изображения, которое видит "являющийся зрителем" пользователь сервера 1522 приложения/игры, даже если это соединение с Internet не меняется, (b) сервер 1522 приложения/игры должен использовать ресурсы обработки для восстановления из сжатых данных всего изображения 1280×720 и после этого масштабировать это изображение (и вероятно применить фильтр повторной выборки) для вывода на экран намного меньшие размеры (например, 640×360) во время изменения масштаба изображения, (с) если кадры сброшены из-за ограниченной ширины полосы соединения с Internet и/или потерянных/искаженных пакетов, и являющийся зрителем пользователь "перематывает" и "приостанавливает" видео, записанное в буфере 1515 задержки, являющийся зрителем пользователь обнаружит, что сброшенные кадры отсутствуют в буфере задержки (это будет особенно очевидно, если пользователь "пошагово проходит" кадр за кадром), и (d) если являющийся зрителем пользователь осуществляет перемотку назад для нахождения конкретного кадра в видео, записанного в буфере задержки, то сервер 1522 приложения/игры должен искать I-кадр или I-фрагменты до того, как искомый кадр в видео потоке записан в буфер задержки, и после этого восстанавливать из сжатых данных все P-кадры/фрагменты до тех пор, пока не будет достигнут требуемый кадр. Идентичные ограничения относятся не только к пользователям, "являющимся зрителями" видео/аудио потока в режиме реального времени, но и к пользователям (включающим в себя пользователя, который сформировал видео/аудио поток), которые смотрят архивную копию (например, "Brag Clip") видео/аудио потока.

Альтернативный вариант осуществления изобретения решает эти проблемы посредством сжатия видеопотока в нескольких размерах и/или структурах. Один поток (поток "в прямом эфире") оптимально сжимают для передачи потоком конечному пользователю, как описано в этом документе, на основе характеристик сетевого соединения (например, ширины полосы данных, надежности передачи пакета) и функциональных возможностей локального клиента пользователя (например, функциональных возможностей восстановления из сжатых данных, разрешающей способности дисплея). Другие потоки (называемые в этом документе "Hq"-потоками)

сжимают с высоким качеством, с одной или несколькими разрешающими способностями и со структурой, поддающейся воспроизведению видео, и такие HQ-потoki маршрутизируются и сохраняются в серверном центре 210. Например, в одном варианте осуществления, сжатые HQ-потoki сохраняются в дисковом массиве 1515 типа RAID и используются для обеспечения функций, например, паузы, перемотки назад и других функций воспроизведения (например, "Brag Clips", которые могут распределяться другим пользователям для просмотра).

Как изображено на фиг.31a, один вариант осуществления изобретения содержит кодер 3100, который может сжимать видеопоток, по меньшей мере, в двух форматах: один, который периодически включает в себя I-фрагменты или I-кадры 3110, и один, который включает в себя I-фрагменты или I-кадры 3111, если только это необходимо из-за разрыва потока или потому, что определено, что I-фрагмент или I-кадр, вероятно, являются меньшими, чем (P)-фрагмент или (P)-кадр (как описано выше). Например, поток 3111 "в прямом эфире", передаваемый пользователю при воспроизведении видеоигры может быть сжат с использованием только P-кадров (если только I-фрагменты или I-кадры не являются необходимыми или меньшими, как описано выше). Кроме того, кодер 3100 этого варианта осуществления одновременно сжимает видеопоток 3111 в прямом эфире во втором формате, который, в одном варианте осуществления, периодически включает в себя I-фрагменты или I-кадры (или в аналогичном типе формата изображения).

Несмотря на то, что варианты осуществления, описанные выше, используют I-фрагменты, I-кадры, P-фрагменты и P-кадры, лежащие в основе принципы изобретения не ограничиваются каким-либо конкретным алгоритмом сжатия. Например, вместо P-фрагментов или P-кадров может использоваться любой тип формата изображения, в котором кадры зависят от предыдущих или последующих кадров. Аналогично, вместо I-фрагментов или I-кадров, описанных выше, может быть подставлен любой тип формата изображения, который не зависит от предыдущих или последующих кадров.

Как упоминалось выше, HQ-поток 3110 включает в себя периодические I-кадры (например, в одном варианте осуществления, каждые примерно 12 кадров). Это важно, так как, если пользователь когда-либо захочет быстро перемотать назад сохраненный видеопоток до конкретной точки, то потребуются I-фрагменты или I-кадры. Со сжатым потоком только из P-кадров (т.е. без первого кадра последовательности, являющегося I-кадром), декодер должен возвращаться к первому кадру последовательности (который может находиться на расстоянии нескольких часов) и восстанавливать из сжатых данных P-кадры до точки, до которой пользователь хочет перемотать назад. С I-кадром каждые 12 кадров, сохраненных в HQ-потке 3110, пользователь может решить перемотать назад до конкретного места, и ближайший предыдущий I-кадр HQ-потка находится не дальше, чем на расстоянии 12 кадров перед требуемым кадром. Даже если максимальная скорость декодирования декодера равна реальному времени (например, 1/60-ая секунды для потока 60 кадров/сек), то $12 \text{ (кадров)} / 60 \text{ (кадров/сек)} = 1/5 \text{ секунды}$ от I-кадра. И, во многих случаях, декодеры могут работать намного быстрее, чем в реальном времени так, например, со скоростью в 2 раза быстрее реального времени декодер может декодировать 12 (кадров) за 6 кадров, что является только 1/10-ой секунды задержки для "перемотки назад". Само собой разумеется, что даже быстрый декодер (например, в 10 раз быстрее реального времени) имеет неприемлемую задержку, если ближайший предыдущий I-кадр находится на расстоянии большого количества кадров перед точкой перемотки назад (например, требуется 1 час/10=6 минут для "перемотки назад"). В другом варианте осуществления, используются периодические

I-фрагменты, и в этом случае, когда пользователь ведет поиск для перемотки назад, декодер найдет ближайший предыдущий I-фрагмент до точки перемотки назад, и после этого начнет декодирование этого фрагмента от этой точки до тех пор, пока все фрагменты до точки перемотки назад не будут декодированы. Несмотря на то, что

5 периодические I-фрагменты или I-кадры в результате приводят к менее эффективному сжатию, чем полное исключение I-кадров, в службе 210 хостинга обычно существует более чем достаточно локально доступной ширины полосы и емкости памяти для управления HQ-поток.

В другом варианте осуществления кодер 3100 кодирует HQ-поток с периодическим

10 I-фрагментом или I-кадрами, за которыми следуют P-фрагменты или P-кадры, как описано ранее, но также и которым предшествуют B-фрагменты или B-кадры. B-кадры, как описано ранее, являются кадрами, которые предшествуют I-кадру и основаны на разностях кадров с I-кадром с действием в обратном направлении во времени. B-фрагменты являются дополнением фрагмента, предшествующими I-фрагменту и

15 основанными на разностях кадров с действием в обратном направлении от I-фрагмента. В этом варианте осуществления, если требуемой точкой перемотки назад является B-кадр (или она содержит B-фрагменты), то декодер находит ближайший последующий I-кадр или I-фрагмент и декодирует в обратном направлении во времени до тех пор, пока требуемая точка перемотки назад не будет декодирована, и после этого по мере

20 того, как воспроизведение видео продолжается от этой точки и далее, декодер декодирует B-кадры, I-кадры и P-кадры (или их дополнения фрагментов) в последовательных кадрах, продвигающихся вперед. Преимуществом использования B-кадров или B-фрагментов в дополнение к I- и P-типам является то, что часто может быть достигнуто более высокое качество при данном коэффициенте сжатия.

В еще одном варианте осуществления, кодер 3100 кодирует весь HQ-поток как I-кадры. Преимуществом этого подхода является то, что каждая точка перемотки назад является I-кадром, и в результате не требуется декодирование других кадров для

25 достижения точки перемотки назад. Недостатком является то, что скорость передачи сжатых данных является очень высокой по сравнению с кодированием I-, P- или I-, P-,

30 B-потока.

Другие действия по воспроизведению видеопотока (например, быстрая или медленная перемотка назад, быстрая или медленная перемотка вперед и т.д.) обычно на практике являются намного более завершенными с периодическими I-кадрами или I-фрагментами (одними или в комбинации с P- и/или B-дополнениями), так как в каждом случае поток

35 воспроизводится в порядке кадров, отличном от покадрового вперед во времени, и в результате декодеру требуется найти и декодировать конкретный, часто произвольный, кадр в последовательности. Например, в случае очень быстрой перемотки вперед (например, скорость 100х), каждый последующий выведенный на экран кадр находится на расстоянии 100 кадров после предшествующего кадра. Даже с декодером, который

40 работает со скоростью в 10 раз превышающей реальное время и декодирует 10 кадров за 1 период кадра, это тем не менее будет в 10 раз медленнее, чем требуется для достижения быстрой перемотки вперед (со скоростью) 100х. Тогда как, с периодическими I-кадрами или I-фрагментами, как описано выше, декодер может вести поиск ближайших соответствующих I-кадра или I-фрагментов по отношению к кадру, который он должен

45 вывести на экран в следующий раз, и декодировать только промежуточные кадры или фрагменты до точки целевого кадра.

В другом варианте осуществления I-кадры кодируются в HQ-потоке с постоянной периодичностью (например, всегда каждые 8 кадров), и множители скорости, сделанные

доступными пользователю для ускоренной перемотки вперед и перемотки назад, которые являются более быстрыми, чем скорость I-кадра, являются числами, точно кратными периодичности I-кадра. Например, если периодичность I-кадра равна 8 кадров, то скорости быстрой перемотки вперед или перемотки назад, сделанные

5 доступными пользователю, могут быть равными 1X, 2X, 3X, 4X, 8X, 16X, 64X и 128X и 256X. Для скорости более быстрой, чем периодичность I-кадра, декодер сначала перейдет вперед к самому близкому I-кадру, который находится на расстоянии

10 количества кадров вперед при этой скорости (например, если в настоящее время выводимый на экран кадр находится на расстоянии 3 кадра перед I-кадром, то при 128X, декодер перейдет к кадру, который находится на расстоянии 128+3 кадра впереди), и после этого для каждого последующего кадра, декодер пропускает точное количество

кадров, равное выбранной скорости (например, при выбранной скорости 128X, декодер пропускает 128 кадров), при этом он каждый раз будет попадать точно на I-кадр. Соответственно, с учетом того, что все скорости, быстрее, чем периодичность I-кадра,

15 являются числами, точно кратными периодичности I-кадра, декодер никогда не должен декодировать какие-либо предшествующие или последующие кадры для поиска требуемого кадра, и должен декодировать только один I-кадр для каждого выводимого на экран кадра. Для скоростей, медленнее, чем периодичность I-кадра, (например 1X, 2X, 3X, 4X) или для более быстрых скоростей, которые не являются числами, кратными

20 периодичности I-кадра, для каждого выводимого на экран кадра, декодер ведет поиск, тех кадров, которые требуют наименьшего количества дополнительных вновь декодируемых кадров для вывода на экран требуемого кадра, независимо от того, является ли он не декодированным I-кадром или уже декодированным кадром, все еще доступным в декодированном виде (в RAM или в другом быстродействующем

25 запоминающем устройстве), и после этого декодируют промежуточные кадры, по мере необходимости, до тех пор, пока требуемый кадр не будет декодирован и выведен на экран. Например, при ускоренной перемотке вперед со скоростью 4X, в I-, P-закодированной последовательности с периодичностью I-кадра 8X, если текущий кадр является P-кадром, который является кадром 1 после I-кадра, то требуемый кадр,

30 который должен быть выведен на экран, находится на расстоянии 4 кадра дальше, который является 5-ым P-кадром после предыдущего I-кадра. Если в настоящее время выводимый на экран кадр (который только что был декодирован) используется как начальная точка, то декодер должен декодировать еще 4 P-кадра для вывода на экран требуемого кадра. Если использоваться предыдущий I-кадр, то декодер должен

35 декодировать 6 кадров (I-кадр и следующие 5 P-кадров) для вывода на экран требуемого кадра. (Ясно, в этом случае, что предпочтительно использовать в настоящее время выводимый на экран кадр для минимизации дополнительных кадров, которые должны быть декодированы). Далее, следующий кадр, который должен быть декодирован, 4 кадра вперед, является 1-ым P-кадром после I-кадра. В этом случае, если в настоящее

40 время декодируемый кадр используется как начальная точка, то декодер должен декодировать еще 4 кадра (2 P-кадра, I-кадр и P-кадр). Но, если вместо этого используется следующий I-кадр, то декодер должен декодировать только I-кадр и последующий P-кадр. (Ясно, в этом случае, что предпочтительно использовать следующий I-кадр как начальную точку для минимизации дополнительных кадров,

45 которые должны быть декодированы). Соответственно, в этом примере, декодер чередует использование в настоящее время декодируемого кадра в качестве начальной точки и использование последующего I-кадра в качестве начальной точки. В качестве общего принципа, независимо от режима воспроизведения видеопотока (ускоренная

перемотка вперед, перемотка назад или пошагово) и скорости, декодер начинает с того кадра, который, независимо от того, является ли он I-кадром или ранее декодированным кадром, требует наименьшего количества вновь декодируемых кадров для вывода на экран требуемого кадра, для каждого последующего кадра, выводимого на экран для

5 этих режима воспроизведения и скорости.

Как изображено на фиг.31b, один вариант осуществления службы 210 хостинга включает в себя логику 3112 повторного воспроизведения потока для управляющих запросов пользователя на повторное воспроизведение HQ-потока 3110. Логика 3112 воспроизведения потока принимает клиентские запросы, содержащие команды

10 воспроизведения видео (например, пауза, перемотка назад, воспроизведение от заданной точки и т.д.), интерпретирует команды и декодирует HQ-поток 3110 от заданной точки (с началом или в I-кадре, или в ранее декодированном кадре, как требуется, и далее продолжает в прямом направлении или в обратном направлении до заданной точки). В одном варианте осуществления, декодированный HQ-поток обеспечивается в кодер

15 3100 (потенциально идентичный кодер 3100, если он может кодировать несколько потоков одновременно, или отдельный кодер 3100) для возможности его повторного сжатия (с использованием способов, описанных в этом документе) и передачи в клиентское устройство 205. Декодер 3102 в клиентском устройстве далее декодирует и визуализирует поток, как описано выше.

В одном варианте осуществления, логика 3112 воспроизведения потока не декодирует HQ-поток и далее вызывает повторное кодирование потока кодером 3100. Наоборот, она просто передает потоком HQ-поток 3110 непосредственно в клиентское устройство 205 от заданной точки. Декодер 3102 в клиентском устройстве 205 после этого декодирует HQ-поток. Поскольку у функций воспроизведения, описанных в этом

25 документе, обычно не существует требований малого времени ожидания, идентичных требованиям при воспроизведении видеоигры в реальном времени (например, если игрок просто просматривает предшествующий геймплей, без активного воспроизведения), дополнительное время ожидания, как правило, присущее HQ-потoku, обычно более высокого качества, может в результате привести к приемлемому

30 восприятию конечным пользователем (например, при более высоком времени ожидания, но с более высококачественным видео).

Например, если пользователь воспроизводит видеоигру, то кодер 3100 обеспечивает поток в прямом эфире по существу всех P-кадров, оптимизированный для соединения пользователя и локального клиента (например, приблизительно 1,4 Мбит/с при

35 разрешающей способности 640×360). Одновременно, кодер 3100 также сжимает видеопоток как HQ-поток 3110 в службе 310 хостинга и сохраняет этот HQ-поток в локальном дисковом массиве типа RAID декодера цифрового видео, например, 1280×720 при 10 Мбит/с с I-кадрами каждые 12 кадров. Если пользователь нажимает кнопку "Пауза", то игра приостанавливается на последнем декодированном кадре клиента, и

40 экран зависает. После этого, если пользователь нажимает кнопку "Перемотка назад", то логика 3112 воспроизведения потока считывает HQ-поток 3110 из RAID DVR, начинающийся с самого близкого I-кадра или доступного уже декодированного кадра, как описано выше. Логика 3112 воспроизведения потока восстанавливает из сжатых данных промежуточные P- или B-кадры, в случае необходимости, повторно

45 упорядочивает кадры в случае необходимости так, чтобы последовательность воспроизведения была в обратном направлении при требуемой скорости перемотки назад, и затем изменяет размер (с использованием способов масштабирования изображения, известных в данной области техники) требуемого декодированного

(кадра), который должен быть выведен на экран, с 1280×720 на 640×360, и кодер 3100 потока в прямом эфире повторно сжимает повторно упорядоченный поток с разрешающей способностью 640×360 и передает его пользователю. Если пользователь снова сделает паузу и после этого продвигается одиночными шагами внутри видео для внимательного просмотра последовательности, то в HQ-потоке 3110 в RAID DVR каждый кадр будет доступен для продвижения одиночными шагами просмотра (даже при том, что в исходном потоке в прямом эфире могли быть пропущены кадры по любой из множества причин, описанных в этом документе). Кроме того, качество воспроизведения видео является довольно высоким в каждой точке в HQ-потоке, тогда как могут быть точки в потоке в прямом эфире, в которых, например, ширина полосы была уменьшена, что в результате привело к временному снижению качества сжатого изображения. В то время как ухудшенное качество изображения в течение краткого периода времени, или в движущемся изображении, может быть приемлемым для пользователя, если пользователь остановил на конкретном кадре (или медленно продвигается одиночными шагами) и внимательно рассматривает кадры, то ухудшенное качество может быть неприемлемым. Пользователю также обеспечивается возможность ускоренной перемотки вперед или перехода к конкретному месту посредством задания точки в HQ-потоке (например, на 2 минуты раньше). Все эти операции являются практически невыполнимыми в своей полной общности и при высоком качестве с потоком видео в прямом эфире, который состоит только из Р-кадров или имеет редкие (или не поддающиеся прогнозированию) I-кадры.

В одном варианте осуществления, пользователю обеспечивается видеоокно (не изображено), например, видеоокно Adobe Flash или Apple QuickTime с "элементом для ручного управления воспроизведением" (т.е. управляющий элемент-движок влево/вправо), который обеспечивает возможность пользователю перемещаться вперед и назад внутри видеопотока, назад до того места, до которого HQ-поток сохранил видео. Несмотря на то, что пользователю очевидно то, что, когда он или она "вручную управляет воспроизведением" внутри потока в прямом эфире, фактически он или она вручную управляет воспроизведением внутри сохраненного HQ-потока 3110, который тогда изменен и повторно сжат как поток в прямом эфире. Кроме того, как упоминалось ранее, если HQ-поток одновременно смотрит кто-либо еще, или пользователь в другое время, то его могут смотреть с более высокой (или более низкой) разрешающей способностью, чем разрешающая способность потока в прямом эфире, когда HQ-поток кодируется одновременно, и качество будет столь же высоким, как качество потока в прямом эфире зрителя, потенциально до качества HQ-потока.

Соответственно, при одновременном кодировании как потока в прямом эфире (как описано в этом документе, соответствующим способом для его малого времени ожидания, ширины полосы и требований к устойчивости к ошибкам пакета), так и HQ-потока с его требованиями к действию воспроизведения потока, высокому качеству, пользователю тем самым обеспечивают требуемую конфигурацию обоих сценариев. И, фактически, это практически очевидно пользователю, что существуют два разных потока, закодированные по-разному. С точки зрения пользователя, он ощущает очень быструю реакцию с малым временем ожидания, несмотря на работу с очень изменчивым и с малой шириной полосы соединением с Internet, функциональность цифровой видеозаписи (DVR) является очень высококачественной, с гибкими действиями и гибкими скоростями.

В результате способов, описанных выше, пользователь получает преимущества как видеопотока в прямом эфире, так и HQ-видеопотока во время геймплея в режиме онлайн

или другого взаимодействия в режиме онлайн без каких-либо ограничений потока в прямом эфире или HQ-потока.

На фиг.31с изображен один вариант осуществления архитектуры системы для выполнения вышеупомянутых операций. Как изображено, в этом варианте осуществления, кодер 3100 кодирует ряд потоков 3121L, 3122L и 3125L "в прямом эфире" и соответствующий ряд "HQ"-потоков 3121H1-H3, 3122H1-H3 и 3125H1-H3, соответственно. Каждый HQ-поток H1 кодируется с полной разрешающей способностью, в то время как каждый кодер H2 и H3 масштабирует до видеопотока меньшего размера для кодирования. Например, если разрешающая способность видеопотока равна 1280×720, то H1 кодирует с разрешающей способностью 1280×720, в то время как H2 может масштабировать до 640×360 и кодировать с этой разрешающей способностью, и H3 может масштабировать до 320×180 и кодировать с этой разрешающей способностью. Одновременно может использоваться любое количество преобразователей масштаба/кодеров Hn с обеспечением множества одновременных HQ-кодирований с множеством разрешающих способностей.

Каждый из потоков в прямом эфире запускается в ответ на сигналы 3161, 3162 и 3165 обратной связи канала, принимаемые через входящее соединение 3101 с Internet, как описано выше (см., например, обсуждение сигналов 2501 и 2601 обратной связи по фиг.25-фиг.26). Потоки в прямом эфире передаются по Internet (или другой сети) через логику 3140 исходящей маршрутизации. Устройства сжатия 3121L-3125L в прямом эфире включают в себя логику для адаптации сжатых видеопотоков (включающую в себя масштабирование, сброс кадров и т.д.) на основе обратной связи канала.

HQ-потоки маршрутизируются логикой 3141 и 1502 входящей маршрутизации во внутренние буферы задержки (например, в дисковый массив 3115 типа RAID) или в другие запоминающие устройства по пути 3151 сигнала, и/или возвращаются в качестве обратной связи по пути 3152 сигнала в серверы приложения/игры и кодер 3100 для дополнительной обработки. Как описано выше, HQ-потоки 3121Hn-3125Hn впоследствии передаются потоком конечным пользователям по запросу (см., например, фиг.31b и связанный с ней текст).

В одном варианте осуществления, кодер 3100 реализован с логикой 1530 совместно используемого аппаратного сжатия, изображенной на фиг.15. В другом варианте осуществления некоторые или все кодеры и преобразователи масштаба являются индивидуальными подсистемами. Лежащие в основе принципы изобретения не ограничиваются каким-либо конкретным совместным использованием ресурсов сжатия или масштабирования или аппаратной/программной конфигурации.

Преимущество конфигурации по фиг.31с состоит в том, что Серверы 3121-3125 приложения/игры, которые требуют меньших, (чем) полноразмерные, видеоокон, не должны обрабатывать и восстанавливать из сжатых данных полноразмерное окно. Кроме того, Услуги 3121-3125 приложения/игры требуют, чтобы окна переходных размеров могли принимать сжатый поток, размер окна которого примерно равен требуемому размеру окна, и после этого увеличивать или уменьшать масштаб до требуемого размера окна. Кроме того, если множество Серверов 3121-3125 приложения/игры запрашивают видеопоток идентичного размера из другого Сервера 3121-3125 приложения/игры, то Входящая маршрутизация 3141 может реализовывать способы групповой передачи по протоколу IP, например, те, которые известны в данной области техники, и передавать в широковещательном режиме запрошенный поток одновременно во множество Серверов 3121-3125 приложения/игры, при этом не требуется независимый поток для каждого Сервера приложения/игры, сделавшего запрос. Если сервер

приложения/игры, принимающий передачу в широковещательном режиме, изменяет размер видеоокна, то он может переключиться на передачу в широковещательном режиме видео другого размера. Соответственно, произвольно большое количество пользователей может одновременно смотреть видеопоток из Сервера приложения/

5 игры, причем каждый с гибкостью масштабирования своего видеоокна, и всегда с получением преимущества видеопотока, масштабированного близко к требуемому размеру окна.

Одним недостатком подхода, изображенного на фиг.31с, является то, что во многих практических реализациях Службы 210 хостинга, никогда не существует момента

10 времени, когда все сжатые HQ-потoki, не говоря обо всех размерах всех сжатых HQ-потокoв, просматривают одновременно. Когда кодер 3100 реализуется как совместно используемый ресурс (например, преобразователь масштаба/устройство сжатия, или реализуется в программном обеспечении или аппаратных средствах), эта неэкономность уменьшается. Но могут существовать практические проблемы в соединении большого

15 количества несжатых потоков в общий совместно используемый ресурс из-за данной ширины полосы. Например, каждый поток 1080×60 равен почти 3 Гбит/с, что намного превышает даже Gigabit Ethernet. Следующие альтернативные варианты осуществления предназначены для решения этой проблемы.

На фиг.31d изображен альтернативный вариант осуществления Службы 210 хостинга,

20 в котором в каждом Сервере 3121-3125 приложения/игры существует два выделенных ему устройства сжатия: (1) устройство 3121L-3125L сжатия потока в прямом эфире, которое адаптирует сжатый видеопоток на основе обратной связи 3161-3165 канала, и (2) устройство сжатия HQ-потока, которое выводит HQ-поток полной разрешающей способности, как описано выше. А именно, устройство сжатия в прямом эфире является

25 динамичным и адаптивным с использованием двухсторонней передачи информации с клиентом 205, в то время как HQ-поток является неадаптивным и односторонним. Другими различиями между (потоками) является то, что качество потока в прямом эфире может существенно меняться, в зависимости от условий на канале и характера видеоматериала. Некоторые кадры могут иметь низкое качество, и могут существовать

30 сброшенные кадры. Кроме того, поток в прямом эфире может почти полностью состоять из Р-кадров или Р-фрагментов, причем I-кадры или I-фрагменты появляются изредка. HQ-поток обычно имеет намного более высокую скорость передачи данных, чем поток в прямом эфире, и он обеспечивает постоянное высокое качество без сброса каких-либо кадров. HQ-поток может весь состоять из I-кадров, или он может иметь частые

35 и/или регулярные I-кадры или I-фрагменты. HQ-поток также может включать в себя В-кадры или В-фрагменты.

В одном варианте осуществления, совместно используемые масштабирование и повторное сжатие 3142 видео (подробно описанные ниже) выбирают только определенные HQ-видеопотоки 3121H1-3125H1, которые должны быть масштабированы

40 и повторно сжаты с одной или несколькими разными разрешающими способностями перед отправкой во входящую маршрутизацию 3141 для маршрутизации, как описано ранее. Другие HQ-видеопотоки или пропускают в полном объеме во входящую маршрутизацию 3141 для маршрутизации, как описано ранее, или вообще не пропускают. В одном варианте осуществления решение, на основе которого HQ-потоки

45 масштабируются и повторно сжимаются, и/или на основе которого HQ-потоки вообще пропускают, определяется на основе того, существует ли Сервер 3121-3125 приложения/игры, который запрашивает этот конкретный HQ-поток с конкретной разрешающей способностью (или с разрешающей способностью, близкой к масштабированной или

полной разрешающей способности). Благодаря этому средству единственными HQ-потоками, которые масштабируются и повторно сжимаются (или потенциально вообще пропускаются), являются HQ-потоки, которые фактически необходимы. Во многих приложениях Службы 210 хостинга, это в результате приводит к значительным сокращениям ресурсов сжатия и масштабирования. Кроме того, с учетом того, что каждый HQ-поток, по меньшей мере, сжимается с полной разрешающей способностью устройствами 3121Н1-3125Н1 сжатия, ширина полосы, необходимая для маршрутизации в Совместно используемое масштабирование и повторное сжатие 3142 видео и внутри него, значительно уменьшается, по сравнению с тем, если оно принимает несжатое видео. Например, несжатый поток 1080на60 3 Гбайт/с может быть сжат до 10 Мбит/с и, тем не менее, сохранить очень высокое качество. Соответственно, с возможностью подключения к Gigabit Ethernet, вместо отсутствия возможности транспортировать даже один несжатый видеопоток 3 Гбит/с, существует возможность транспортировать десятки видеопотоков 10 Мбит/с с небольшим видимым уменьшением качества.

На фиг.31f изображены детали Совместно используемого масштабирования и повторного сжатия 3142 видео, вместе с большим количеством устройств сжатия HQ-видео HQ 3121Н1-3131Н1. Внутренняя маршрутизация 3192, согласно запросам на конкретные видеопотоки, масштабированные до конкретных размеров, из Серверов 3121-3125 приложения/игры, обычно выбирает подмножество сжатых HQ-потков из устройств сжатия HQ-видео HQ 3121Н1-3131Н1. Поток в этом выбранном подмножестве потоков маршрутизируется или через Устройство 3161-3164 восстановления сжатых данных, если запрашиваемый поток должен быть масштабирован, или он маршрутизируется по пути 3196 не масштабированного видео, если запрашивается поток с полной разрешающей способностью. Потоки, которые должны быть масштабированы, восстанавливаются в несжатое видео Устройствами 3161-3164 восстановления сжатых данных, после этого каждый масштабируется до запрошенного размера Преобразователями 3171-3174 масштаба, после этого каждый сжимается Устройством 3181-3184 сжатия. Отметим, что, если конкретный HQ-поток запрашивается с несколькими разрешающими способностями, то Внутренняя маршрутизация 3192 осуществляет групповую передачу этого потока (с использованием технологии групповой передачи по протоколу IP, которая является известной специалистам-практикам в данной области техники) в одно или несколько Устройств 3161-3164 восстановления сжатых данных и (если один запрошенный размер (является) полной разрешающей способностью) в Исходящую маршрутизацию 3193. Все запрошенные потоки, независимо от того, являются ли они масштабированными (из Устройств 3181-3184 сжатия) или нет (из Внутренней маршрутизации 3192), после этого отправляют в Исходящую маршрутизацию 3193. Маршрутизация 3193 после этого отправляет каждый запрошенный поток в Сервер 3121-3125 приложения/игры, который запрашивал его. В одном варианте осуществления, если несколько серверов приложения/игры запрашивают идентичный поток с идентичной разрешающей способностью, то Исходящая маршрутизация 3193 осуществляет групповую передачу этого потока во все серверы 3121-3125 приложения/игры, которые делают этот запрос.

В настоящее время предпочтительном варианте осуществления Совместно используемого масштабирования и повторного сжатия 3142 видео, маршрутизация реализуется с использованием коммутаторов Gigabit Ethernet, и восстановление сжатых данных, масштабирование и сжатие реализуется дискретными специализированными полупроводниковыми устройствами, реализующими каждую функцию. Идентичная функциональность может быть реализована с более высоким уровнем интеграции в

аппаратных средствах или очень быстрыми процессорами.

На фиг.31e изображен другой вариант осуществления Службы 210 хостинга, в котором функция Буфера 3115 задержки, описанного ранее, реализована в Совместно используемой подсистеме 3143 восстановления сжатых данных и масштабирования с

буфером задержки видео. Детали подсистемы 3143 изображены на фиг.31g. Функционирование подсистемы 3143 аналогично функционированию подсистемы 3142, изображенной на фиг.31f, за исключением того, что сначала в позиции 3191 выбирают то, какие HQ-видеопотоки должны маршрутизироваться, согласно запросам из Серверов 3121-3125 приложения/игры, и далее, HQ-потки, в отношении которых запрашивают, чтобы они были задержаны, маршрутизируются через Буфер 3194 задержки, реализованный в настоящее время предпочтительном варианте осуществления как дисковый массив типа RAID (но может быть реализован в любом носителе информации с достаточной шириной полосы и пропускной способностью), и потоки, в отношении которых не запрашивают, чтобы они были задержаны, маршрутизируются по пути 3195 Видео без задержки. Вывод и Буфера 3194 задержки, и Видео 3195 без задержки далее маршрутизируется Внутренней маршрутизацией 3192 на основе того, должны ли запрошенные потоки быть масштабированными или не масштабированными. Масштабированные потоки маршрутизируются через Устройства 3161-3164 восстановления сжатых данных, Преобразователи 3171-3174 масштаба и Устройства 3181-3184 сжатия в Исходящую маршрутизацию 3193, и Не масштабированное видео 3196 также отправляют в Исходящую маршрутизацию 3193, и после этого Исходящая маршрутизация 3193 отправляет видео в индивидуальном или групповом режиме на Серверы приложения/игры способом, идентичным описанному ранее в подсистеме 3142 по фиг.31f.

Другой вариант осуществления подсистемы 3143 восстановления сжатых данных и масштабирования с буфером задержки видео изображен на фиг.31h. В этом варианте осуществления для каждого HQ-потока обеспечен отдельный Буфер HQ 3121D-HQ 3131D Задержки. С учетом быстро снижающейся стоимости RAM и флэш-памяти, которые могут использоваться для задержки отдельного сжатого видеопотока, в результате они могут оказаться более дешевыми и/или более гибким, чем совместно используемый Буфер 3194 задержки. Или, в еще одном варианте осуществления, один Буфер 3197 задержки (изображенный внутри пунктирной линии) может обеспечивать задержку для всех HQ-потоков по отдельности в высокопроизводительном коллективном ресурсе (например, очень быстродействующие RAM, флэш-память или диск). В любом сценарии, каждый Буфер HQ 3121D-3131D задержки может на переменный период задерживать поток из источника HQ-видео или пропускать поток без задержки. В другом варианте осуществления, каждый буфер задержки может обеспечивать множество потоков с разными величинами задержки. Все задержки или не задержки запрашиваются Сервисами 3121-3125 приложения/игры. Во всех этих случаях задержанные и не задержанные видеопотоки 3198 отправляют во Внутреннюю 3192 маршрутизацию, и далее через остальную часть подсистемы 3143, как описано ранее в отношении фиг.31g.

В предыдущих вариантах осуществления, относящихся к различным фиг.31h, отметим, что поток в прямом эфире использует двухстороннее соединение и формируется для конкретного пользователя с минимальным временем ожидания. HQ-потоки используют односторонние соединения и являются как индивидуальными, так и групповыми. Отметим, что, несмотря на то, что на этих чертежах групповая функция изображена как один блок так, как может быть реализовано в коммутаторе Gigabit Ethernet, в

большой системе, групповая функция, вероятно, будет реализована посредством дерева множества коммутаторов. Действительно, в случае видеопотока от признанного видеоигрока, вполне может случиться так, что HQ-поток игрока смотрят миллионы пользователей одновременно. В таком случае, вероятно, существует большое количество индивидуальных коммутаторов на последовательных этапах передачи в ширококовещательном режиме HQ-потока, передаваемого группе пользователей.

И с целью диагностики, и для обеспечения обратной связи пользователю (например, для сообщения пользователю о том, насколько является популярным его выполнение геймплея), в одном варианте осуществления, служба 210 хостинга отслеживает количество зрителей, которые одновременно смотрят видеопоток каждого Сервера 3121-3125 приложения/игры. Это может быть достигнуто при выполнении непрерывного подсчета количества активных запросов серверами приложения/игры конкретного видеопотока. Соответственно, игроку, у которого существует 100 000 зрителей одновременно, будет известно, что его или ее геймплей является очень популярным, и это будет стимулировать игроков в компьютерные игры к лучшему выполнению и привлечению зрителей. Когда существует очень большая зрительская аудитория видеопотоков (например, матч чемпионата по видеоиграм), то может потребоваться, чтобы комментаторы говорили во время матча по видеоигре так, чтобы некоторые или все пользователи, смотрящие групповую передачу, могли слышать их комментариев.

Приложения и игры, выполняющиеся на серверах приложения/игры, обеспечиваются интерфейсом прикладного программирования (API), в котором приложение и/или игра могут передавать запросы на конкретные видеопотоки с конкретными характеристиками (например, разрешающая способность и величина задержки). Кроме того, эти API, передаваемые в рабочую среду, выполняющуюся на Сервере приложения/игры, или в Систему 401 управления Службой хостинга по фиг.4а, могут отклонять такие запросы по множеству причин. Например, у запрашиваемого видеопотока могут существовать определенные ограничения на лицензионные права (например, его может смотреть только один зритель, не может передаваться в ширококовещательном режиме другим), могут быть ограничения на подписку (например, пользователь, возможно, должен заплатить за право просмотра потока), могут быть ограничения по возрасту (например, для просмотра потока зрителю, возможно, должно быть 18 лет), могут быть ограничения по секретности (например, человек, использующий Приложение, или играющий в игру, может разрешить просмотр только выбранному количеству или классу зрителей (например, его или ее "друзьям"), или может вообще запретить просмотр), и могут быть ограничения с требованием, чтобы материал был задержан (например, если пользователь играет в игру с незаметным действием, где его или ее позиция может быть раскрыта). Существует любое количество других ограничений, которые ограничивают просмотр потока. В любом из этих случаев, запрос сервером приложения/игры отклоняется по некоторой причине для отклонения, и в одном варианте осуществления, с альтернативными вариантами, при которых запрос принимается (например, с началом с того, какая плата должна быть внесена за подписку).

HQ-видеопотоки, которые сохраняются в Буферах задержки в любом из предыдущих вариантов осуществления, могут быть экспортированы в другие пункты назначения за пределами Службы 210 хостинга. Например, в частности, интересный видеопоток может быть запрошен сервером приложения/игры (как правило, по запросу пользователя) для экспорта в YouTube. В таком случае, видеопоток передается через Internet в формате, согласованном с YouTube, вместе с соответствующей описательной информацией (например, имя играющего пользователя, игра, время, оценка и т.д.). Это

может быть реализовано посредством групповой передачи в отдельном потоке аудио с комментарием во все серверы 3121-3125 игры/приложения, запрашивающие такой комментарий. Серверы игры/приложения сливают аудио с комментарием с использованием способов смещения аудио, известных специалистам-практикам в данной области техники, в аудиопоток, отправляемый на территорию 211 пользователя. Вполне может существовать множество комментаторов (например, с разными точками зрения или на разных языках), и пользователи могут выбирать среди них.

Аналогичным способом отдельные аудиопотоки могут быть смешаны с аудиодорожкой конкретных видеопотоков (или отдельных потоков) в Службе 210 хостинга или служить заменой для нее, либо посредством смешения или посредством замены аудио из видео, передающегося потоком в реальном времени или из Буфера задержки. Такое аудио может быть комментарием или повествованием, или оно может обеспечивать голоса для персонажей в видеопотоке. Это обеспечивает возможность создания без труда Машины (мультипликации, создаваемые пользователем из видеопотоков видеоигры) пользователями.

Видеопотоки, описанные в этом документе, изображены как захватываемые из видеовыхода серверов приложения/игры, и далее передаваемые потоком и/или (задерживаемые) и повторно используемые или распределяемые множеством способов. Идентичные Буферы задержки могут использоваться для хранения видеоматериала, который получен из источников-серверов не приложения/игры, и обеспечивают идентичную степень гибкости для воспроизведения и распределения, с соответствующими ограничениями. Такие источники включают в себя линии передачи в прямом эфире из телевизионных станций (или беспроводные, или проводные, например, CNN (Си-эн-эн), и или платные, например, НВО (Эйч-би-оу), или бесплатные). Такие источники также включают в себя предварительно записанные кинофильмы или телепередачи, домашние кинофильмы, рекламные объявления, а также линии передачи видеотелеконференции в прямом эфире. Линии передачи в прямом эфире обрабатываются аналогично выходу в прямом эфире Сервера игры/приложения. Предварительно записанный материал обрабатывается аналогично выходу Буфера задержки.

В одном варианте осуществления различные функциональные модули, иллюстрированные в этом описании, и связанные с ними этапы, могут быть выполнены конкретными аппаратными компонентами, которые содержат "защиту" логику для выполнения этих этапов, например, специализированной интегральной схемой ("ASIC") или любой комбинацией программируемых компьютерных компонентов и заказных аппаратных компонентов.

В одном варианте осуществления упомянутые модули могут быть реализованы на программируемом цифровом сигнальном процессоре ("DSP"), например, архитектуры TMS320x компании Texas Instruments (например, TMS320C6000, TMS320C5000... и т.д.). Могут использоваться различные другие DSP, хотя, по-прежнему, с выполнением этих лежащих в основе принципов.

Варианты осуществления могут включать в себя различные этапы, как изложено выше. Этапы могут быть воплощены в машинно-исполнимых командах, которые вызывают выполнение определенных этапов универсальным или специализированным процессором. Различные элементы, которые не относятся к этим лежащим в основе принципам, например, память компьютера, накопитель на жестких дисках, устройства ввода, не были включены в некоторые или все чертежи во избежание затруднения понимания относящихся к делу аспектов.

Элементы раскрытого предмета изобретения могут также быть обеспечены как машиночитаемый носитель информации для хранения машинно-исполнимых команд. Машиночитаемый носитель информации может включать в себя, например, флэш-память, оптические диски, CD-ROM, ROM DVD, RAM, EPROM, EEPROM, магнитные
 5 или оптические карточки, среды распространения или машиночитаемые носители информации другого типа, подходящие для хранения компьютерных команд. Например, настоящее изобретение может быть загружено как компьютерная программа, которая может быть передана из удаленного компьютера (например, сервера) в запрашивающий компьютер (например, клиент) посредством сигналов данных, воплощенных в несущей
 10 или другой среде распространения через линию связи (например, модем или сетевое соединение).

Также следует понимать, что элементы раскрытого предмета изобретения могут также быть обеспечены как компьютерный программный продукт, который может включать в себя машиночитаемый носитель информации, на котором сохранены
 15 команды, которые могут использоваться для программирования компьютера (например, процессора или другого электронного устройства) для выполнения последовательности операций. В качестве альтернативы, операции могут быть выполнены комбинацией аппаратного обеспечения и программного обеспечения. Машиночитаемый носитель информации может включить в себя, например, гибкие дискеты, оптические диски, CD-
 20 ROM и магнитооптические диски, ROM, RAM, EPROM, EEPROM, магнитные или оптические карточки, среды распространения или среды/машиночитаемый носитель информации другого типа, подходящие для хранения компьютерных команд. Например, элементы раскрытого предмета изобретения могут быть загружены как компьютерный программный продукт, причем эта программа может передаваться из удаленного
 25 компьютера или электронного устройства в запрашивающий процесс посредством сигналов данных, воплощенных в несущей или другой среде распространения через линию связи (например, модем или сетевое соединение).

Кроме того, несмотря на то, что раскрытый предмет изобретения был описан вместе с конкретными вариантами осуществления, существует достаточно многочисленных
 30 модификаций и изменений в рамках настоящего раскрытия предмета изобретения.

Формула изобретения

1. Машинореализуемый способ потоковой передачи видео от сервера к клиенту, содержащий этапы, на которых:

35 принимают в сервере запрос на игру в видеоигру с малым временем ожидания или на исполнение приложения из клиента по сети Интернет,

в ответ на этот запрос устанавливают с клиентом сеанс видеоигры или приложения, определяют конфигурацию аппаратных средств/программного обеспечения клиента и исполняют запрошенную видеоигру или приложение на сервере,

40 формируют и/или выбирают временный декодер на основе конфигурации аппаратных средств/программного обеспечения клиента,

передают временный декодер к клиенту, причем клиент устанавливает этот временный декодер,

45 принимают сигналы управления, переданные от клиента, которые указывают ввод пользователя в клиент, так как пользователь играет в видеоигру или использует приложение, и исполняют в ответ видеоигру или приложение на сервере, чтобы сформировать видеопоток с малым временем ожидания, содержащий последовательность видеоизображений, выводимых из видеоигры или приложения;

кодируют и осуществляют потоковую передачу запрошенного видеосодержимого из сервера в клиент, причем это видеосодержимое кодируется на основе функциональных возможностей временного декодера, причем видеосодержимое декодируется временным декодером и визуализируется в клиенте,

5 причем операции приема сигналов управления, переданных от клиента, исполнения видеоигры или приложения, кодирования и потоковой передачи видеопотока с малым временем ожидания к клиенту по сети Интернет, и декодирования видеопотока с малым временем ожидания выполняются таким образом, чтобы у пользователя было ощущение, что выбранная видеоигра или приложение отвечают незамедлительно на сигналы

10 управления, принятые от клиента,

выявляют, что клиент закончил сеанс с сервером, и
в ответ на выявление того, что клиент закончил сеанс, временно отключают и/или удаляют временный декодер из клиента.

2. Способ по п.1, дополнительно содержащий этапы, на которых:

15 выявляют одну или более характеристик канала связи, когда временный декодер передается из сервера в клиент, и

кодируют запрошенное видеосодержимое на сервере на основе выявленных характеристик канала связи.

3. Способ по п.2, в котором одна из характеристик канала связи содержит время ожидания прохождения сигнала в прямом и обратном направлении по каналу связи.

4. Способ по п.2, в котором одна из характеристик канала связи содержит скорость передачи данных по каналу связи.

5. Способ по п.2, в котором одна из характеристик канала связи содержит надежность канала связи.

25 6. Способ по п.5, в котором надежность канала связи характеризуют на основе процентного содержания потерянных пакетов или пакетов, принятых с ошибками.

7. Способ по п.6, в котором в ответ на выявление того, что процентное содержание потерянных пакетов или пакетов, принятых с ошибками, ниже первого заданного порогового значения, кодируют видеосодержимое без исправления ошибок и в ответ

30 на выявление того, что процентное содержание потерянных пакетов или пакетов, принятых с ошибками, выше первого заданного порогового значения, кодируют видеосодержимое с исправлением ошибок.

8. Способ по п.7, в котором в ответ на выявление того, что процентное содержание потерянных пакетов или пакетов, принятых с ошибками, выше второго заданного порогового значения, осуществляют попытку установить другой канал связи с клиентом.

35 9. Способ по п.8, в котором другой канал связи находится между клиентом и другим сервером.

10. Способ по п.3, в котором в ответ на выявление того, что время ожидания прохождения сигнала в прямом и обратном направлении выше заданного порогового значения, осуществляют попытку установить другой канал связи с клиентом с

40 относительно меньшим временем ожидания прохождения сигнала в прямом и обратном направлении.

11. Способ по п.1, дополнительно содержащий этапы, на которых:

45 выявляют одну или более характеристик канала связи, по которому запрошенный видеопоток с малым временем ожидания передается от сервера к клиенту; и

динамически изменяют сжатие запрошенного видеосодержимого на основании выявленных характеристик канала связи.

12. Способ по п.11, дополнительно содержащий этап, на котором:

задают требование по максимальному времени ожидания для видеоигры с малым временем ожидания, причем требование по максимальному времени ожидания указывает максимально допустимое время ожидания между вводом первого пользователя и визуализацией видеоизображения в клиенте в результате ввода первого пользователя.

5 13. Система для хостинга потоковых интерактивных видеоигр или приложений с малым временем ожидания, содержащая:

множество серверов для установления сеанса видеоигры или приложения с клиентом и исполнения потоковых интерактивных видеоигр или приложений с малым временем ожидания, причем один или более серверов дополнительно формирует и/или выбирает
10 временный декодер для декодирования видео потоковых интерактивных видеоигр с малым временем ожидания на основании конфигурации аппаратных средств/ программного обеспечения клиента и передает временный декодер к клиенту, причем клиент устанавливает этот временный декодер;

входную маршрутизирующую сеть для приема потоков пакетов по сети Интернет
15 от пользователей, играющих в видеоигры или приложения в клиенте, и направляющая потоки пакетов в один или более серверов, причем пакеты включают в себя входной сигнал управления пользователя для видеоигр, один или более серверов могут вычислять выходные видео и аудио данные видеоигры или приложения в ответ на входной сигнал управления пользователя, содержащие последовательности видеоизображений,
20 выводимых из видеоигры или приложения;

блок сжатия, соединенный с возможностью приема выходных видео и аудио данных видеоигры или приложения из одного или более серверов и формирования потоковых сжатых аудио/видео (A/V) данных с малым временем ожидания из них в формате, декодируемом с помощью временного декодера; и

25 выходную маршрутизирующую сеть для маршрутизации потоковых сжатых A/V данных с малым временем ожидания по сети Интернет к клиенту через соответствующий канал связи;

причем временный декодер в клиенте в ответ декодирует потоковые сжатые A/V данные в реальном времени для визуализации видео и аудио в клиенте, представляющем
30 выходные видео и аудио данные видеоигры или приложения;

причем операции приема потоков пакетов по сети Интернет от клиента, исполнения интерактивных видеоигр или приложений, формирования и маршрутизации потоковых сжатых A/V данных с малым временем ожидания по сети Интернет к клиенту и декодирования потоковых сжатых A/V данных с малым временем ожидания выполняют
35 таким образом, чтобы у пользователя было ощущение, что выбранная видеоигра или приложение отвечают незамедлительно на сигналы управления, принятые от клиента;

выявляют, что клиент закончил сеанс с сервером, и

в ответ на выявление того, что клиент закончил сеанс, временно отключают и/или удаляют временный декодер из клиента.

40 14. Система по п.13, в которой блок сжатия может выполнять межкадровое кодирование и смягчение пиков полосы пропускания канала путем распределения больших пиков полосы пропускания сжатого видео по последовательности из последовательных кадров.

15. Система по п.14, в которой распределение больших пиков полосы пропускания
45 сжатого видео по последовательности из последовательных кадров дополнительно содержит передачу сжатых кадров, содержащих больше данных, чем может быть передано при пиковой скорости передачи данных за один период кадра, в течение одного или нескольких дополнительных периодов кадра.

16. Система по п.15, в которой во время упомянутых дополнительных периодов кадра временный декодер продолжает отображать предварительно развернутый кадр.

17. Система по п.16, в которой во время упомянутых дополнительных периодов кадра блок сжатия игнорирует принятые кадры.

5 18. Система по п.15, в которой скорость передачи видеокадров временно снижается во время дополнительных периодов кадра.

19. Система по п.15, в которой дополнительный период кадра является одним периодом кадра.

10 20. Система по п.15, в которой дополнительный период кадра является более чем одним периодом кадра.

21. Система по п.16, в которой во время дополнительных периодов кадра аудио участок A/V потока необходимо продолжать сжимать и разворачивать в клиентском устройстве полностью.

15 22. Система по п.13, в которой блок сжатия может динамически регулировать скорость передачи данных потоковых сжатых A/V данных с малым временем ожидания.

23. Система по п.22, в которой скорость передачи данных регулируют в ответ на выявленную максимальную скорость передачи данных канала.

20

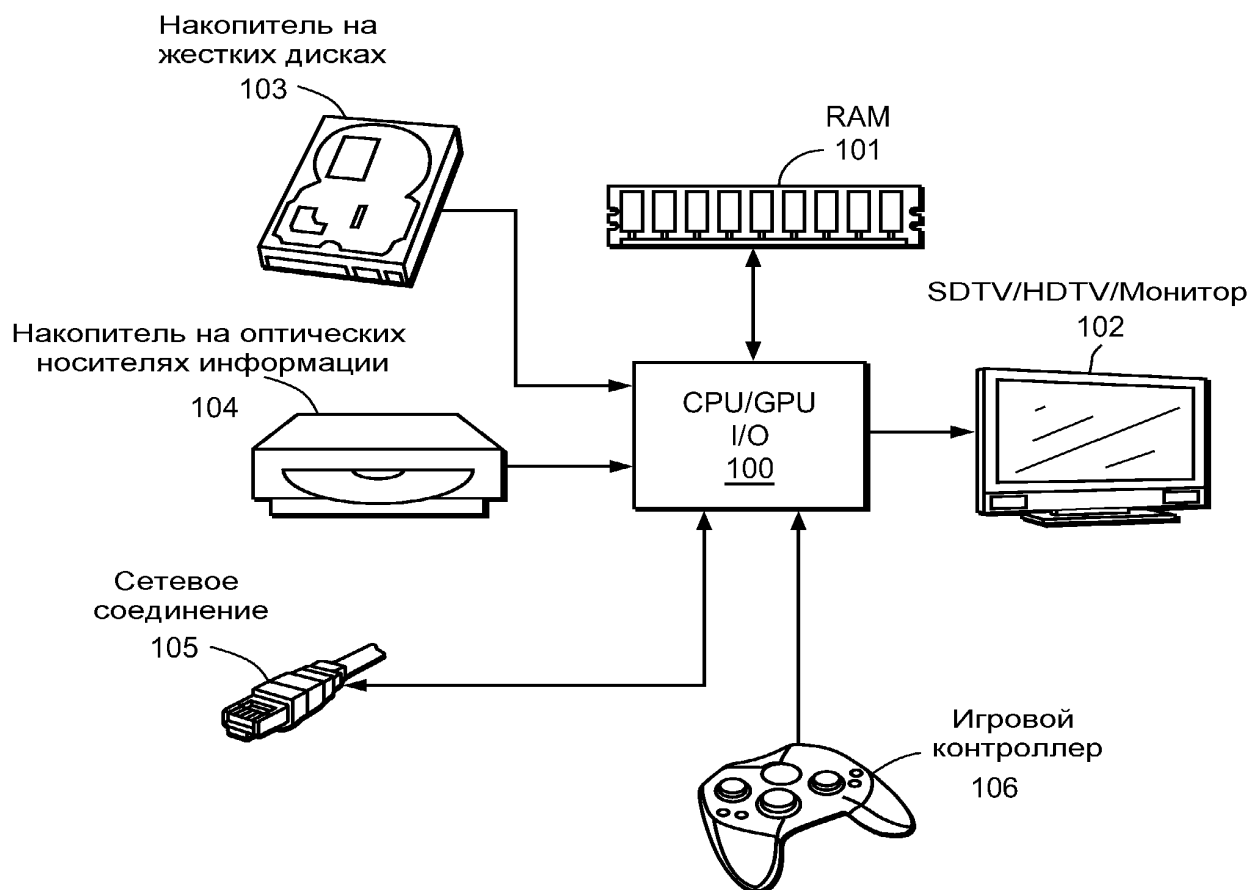
25

30

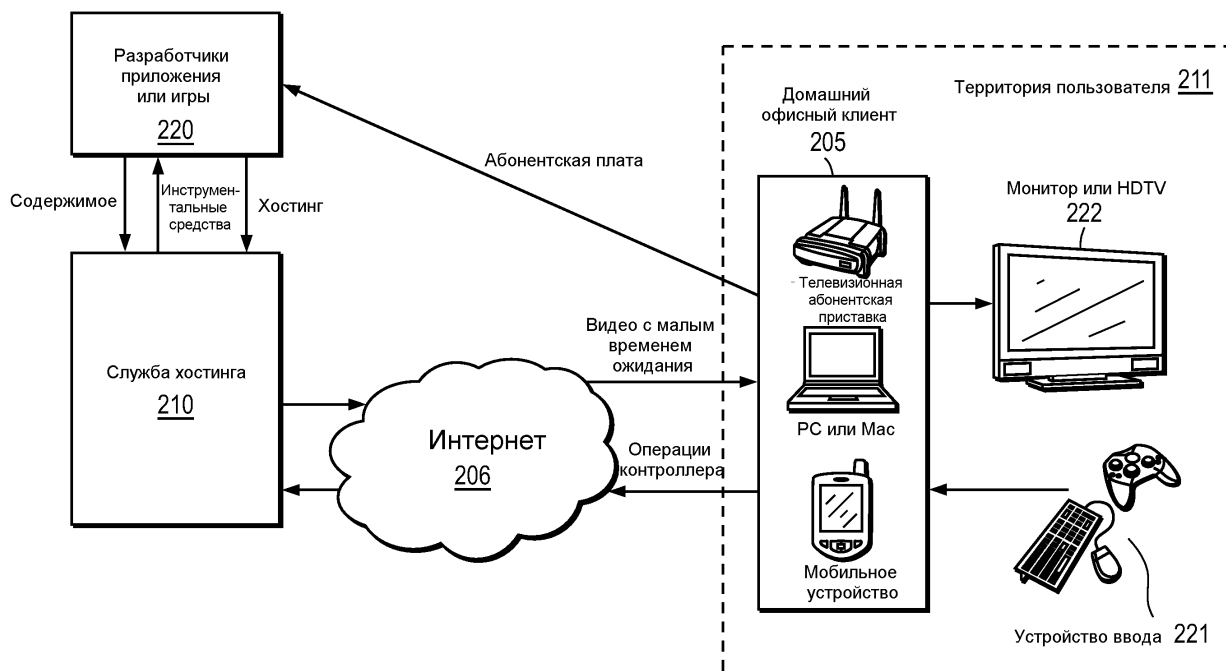
35

40

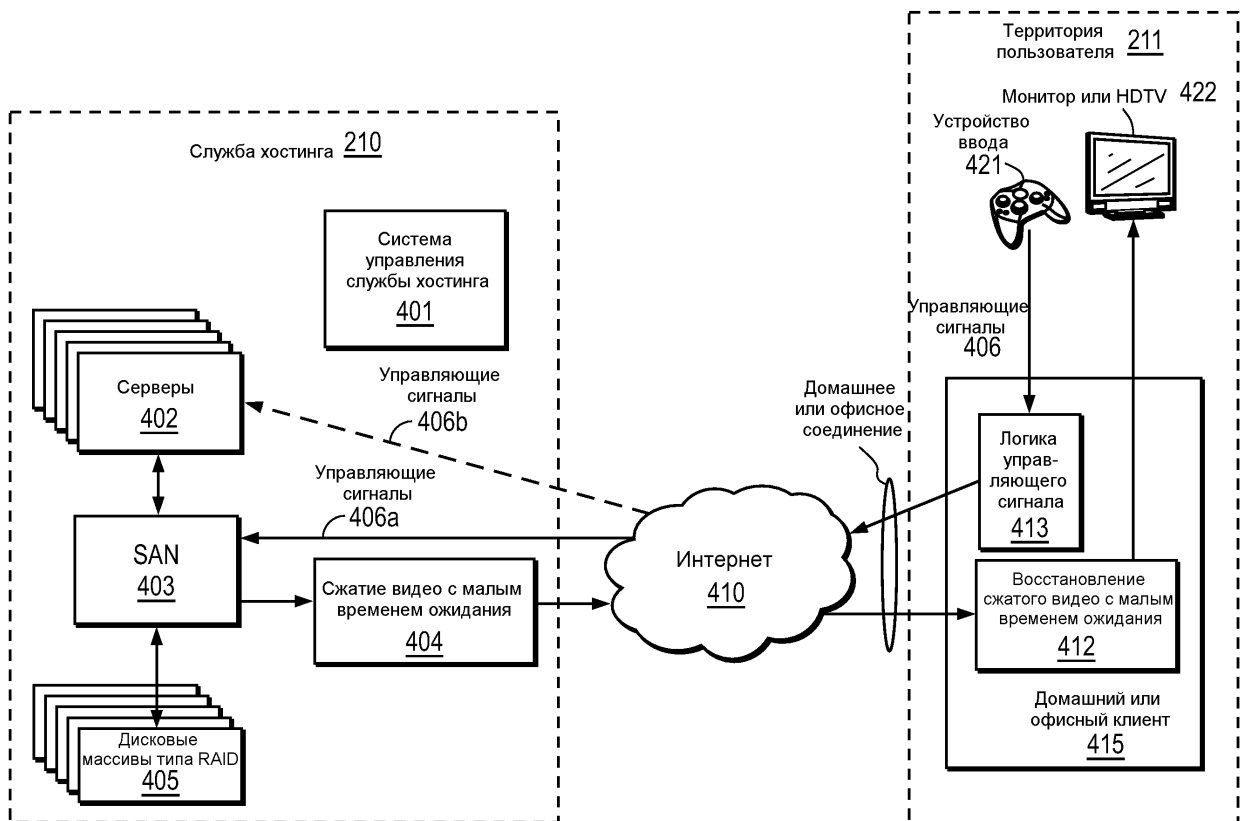
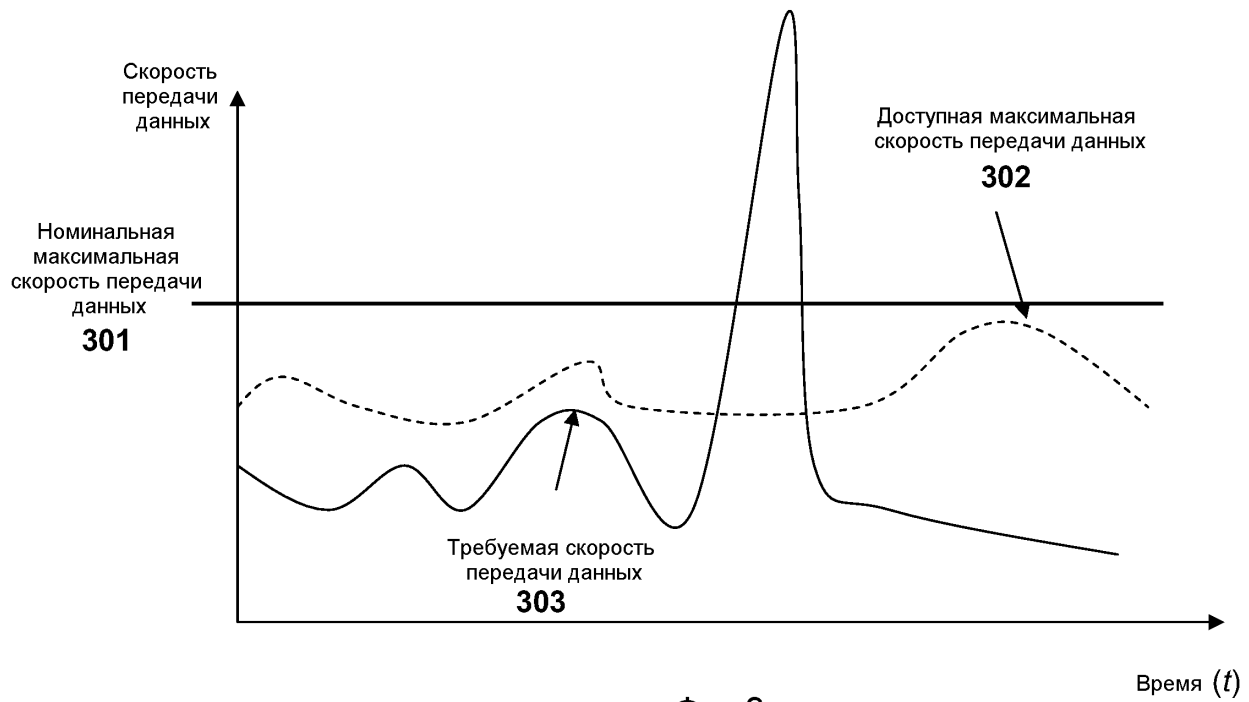
45

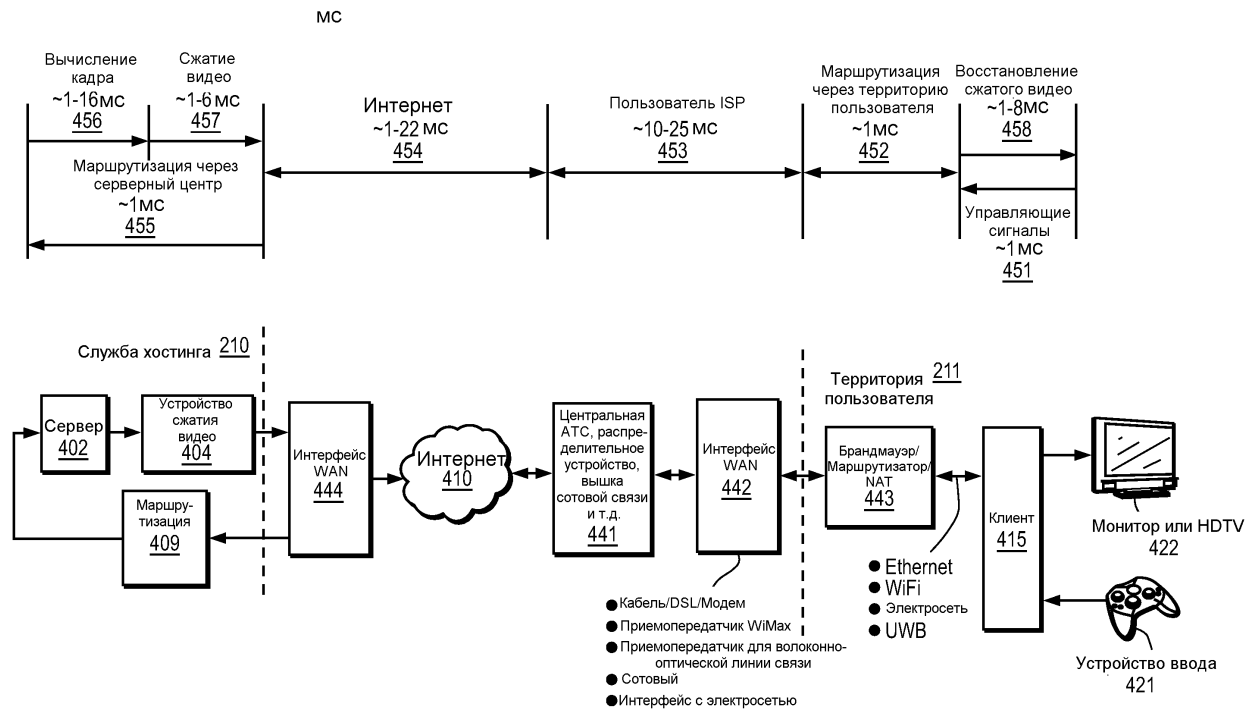


Фиг. 1

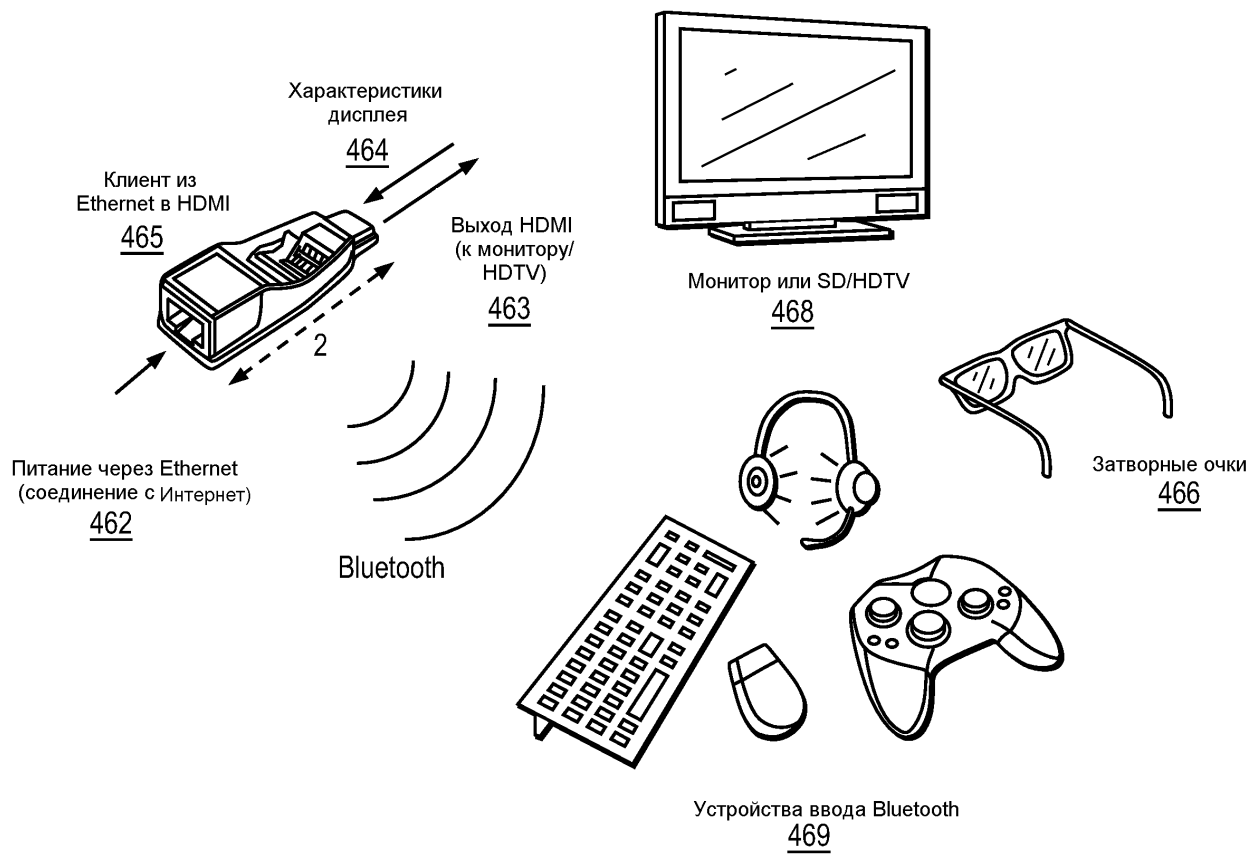


Фиг.2b

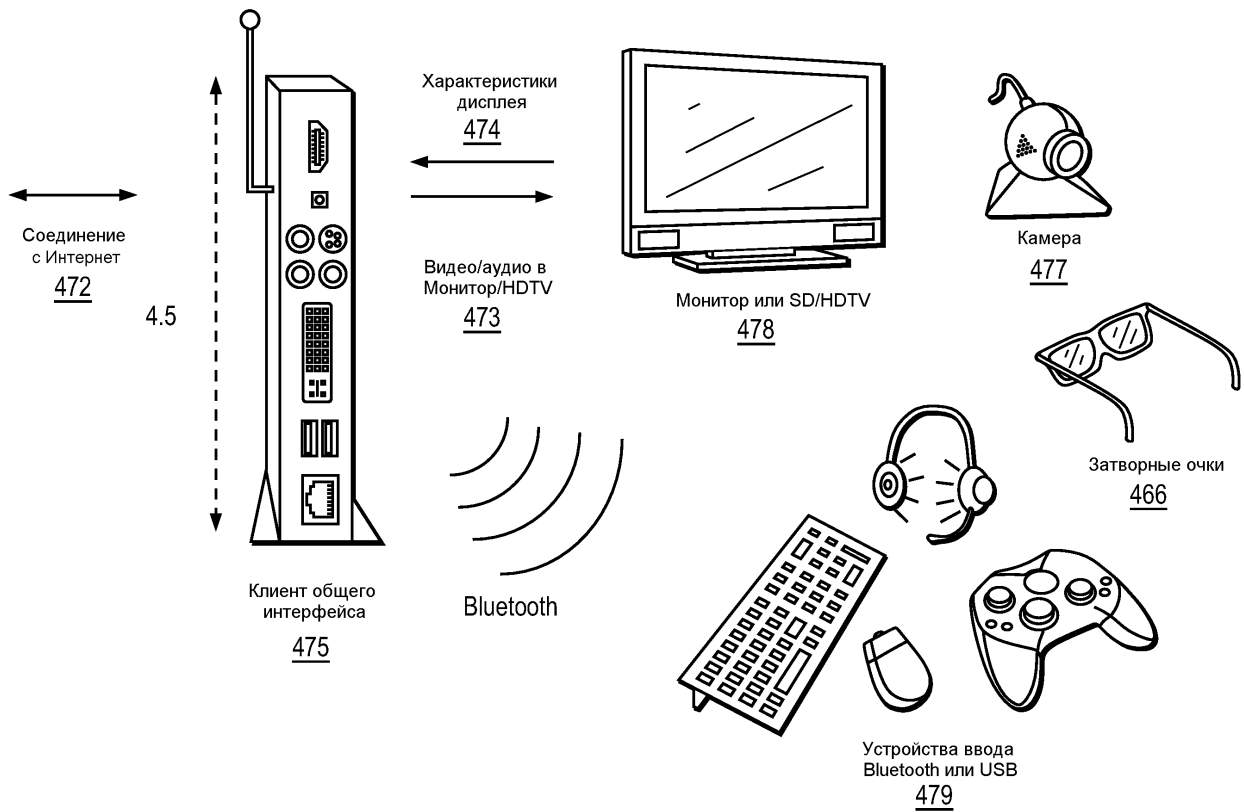




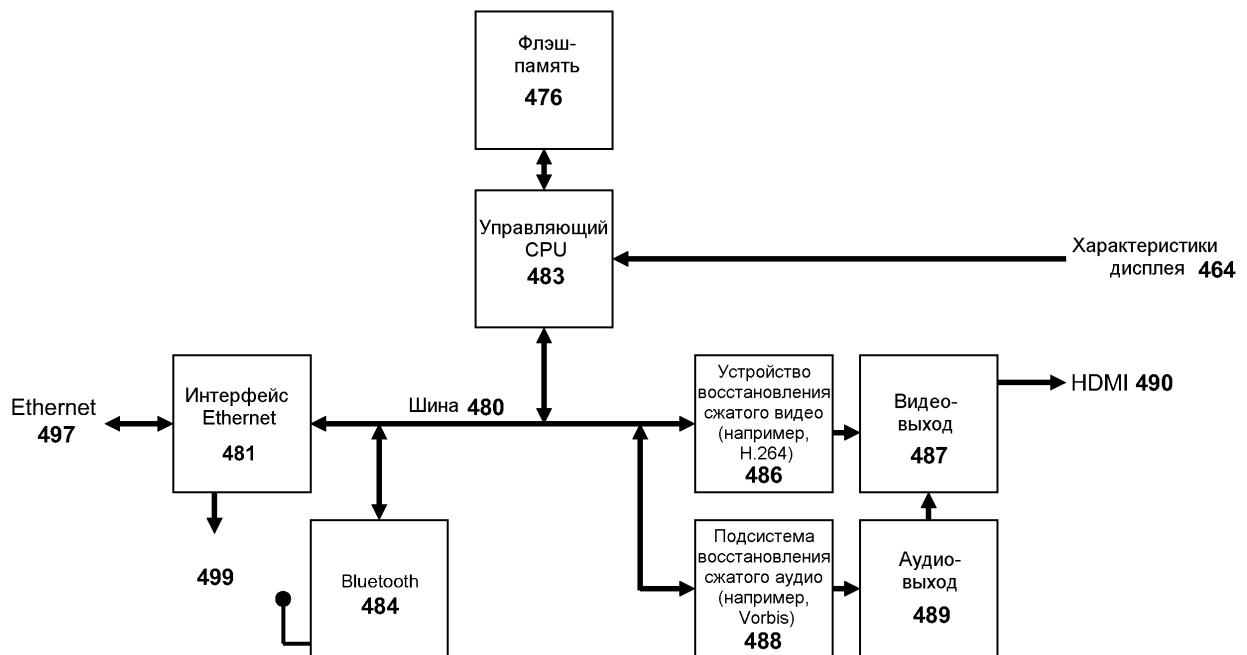
Фиг.4b



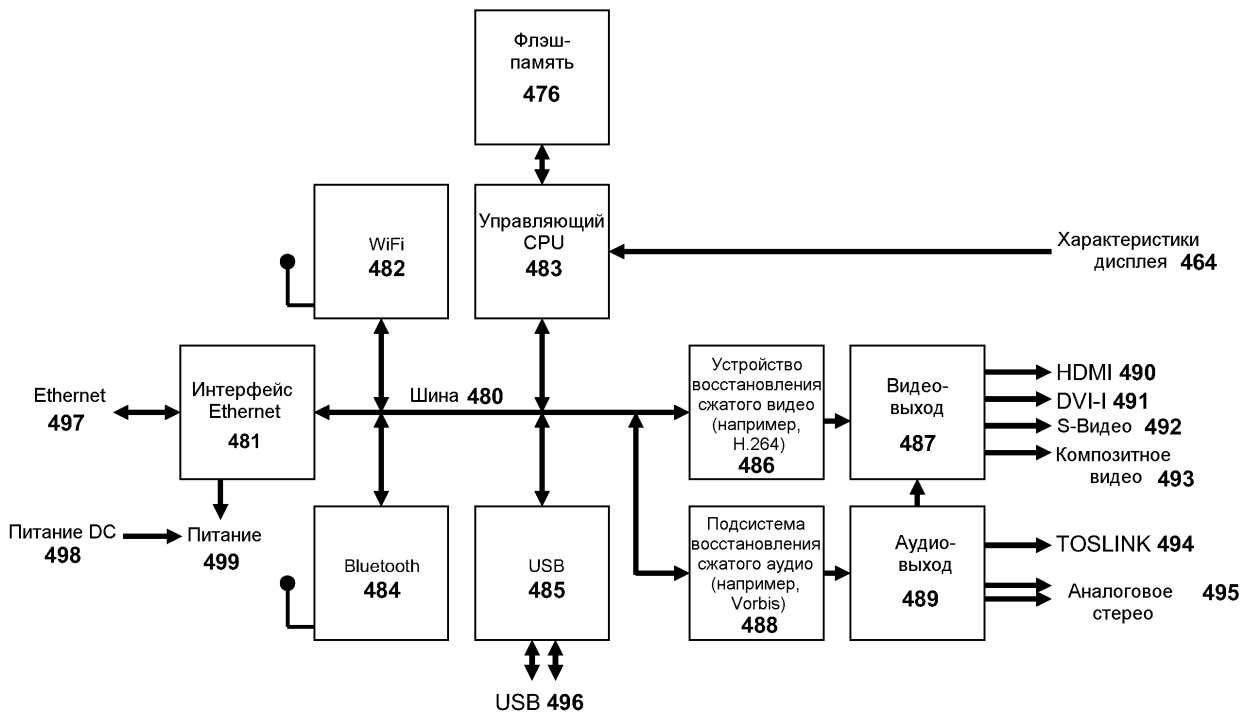
Фиг.4с



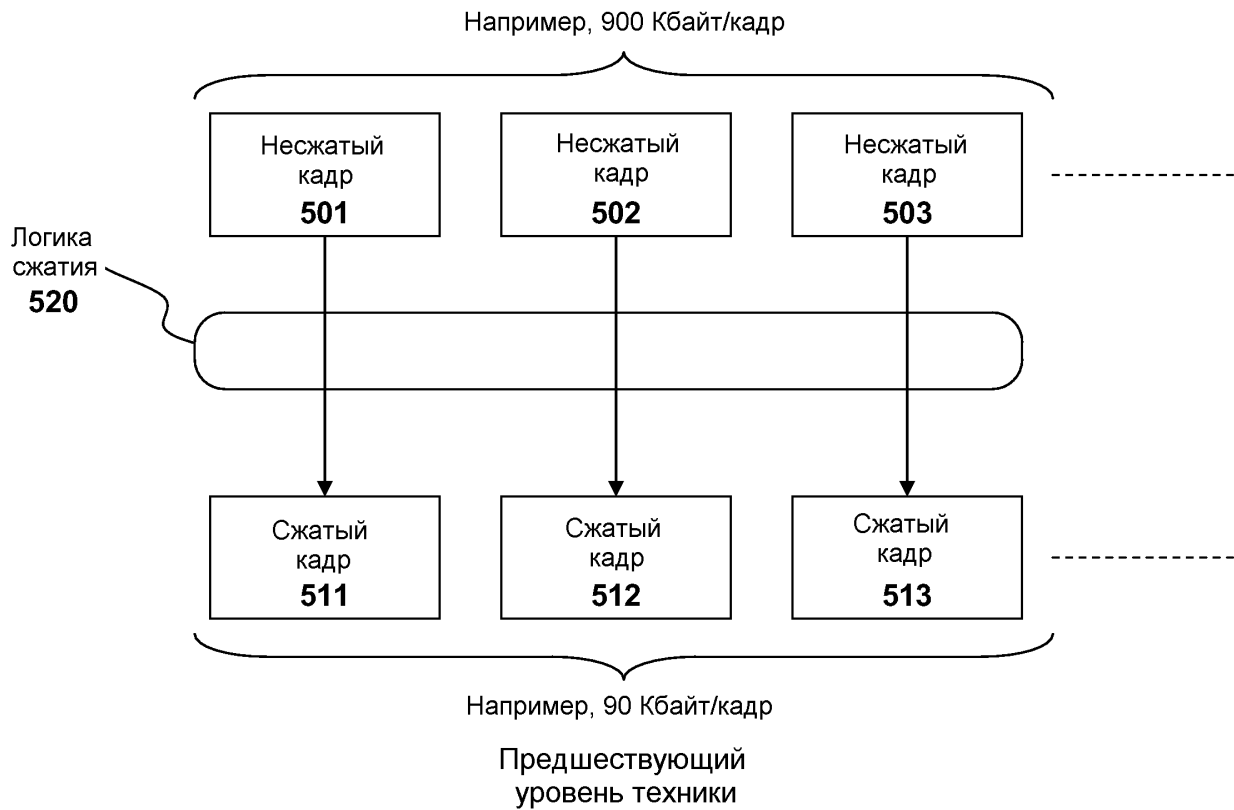
Фиг.4d



Фиг.4е

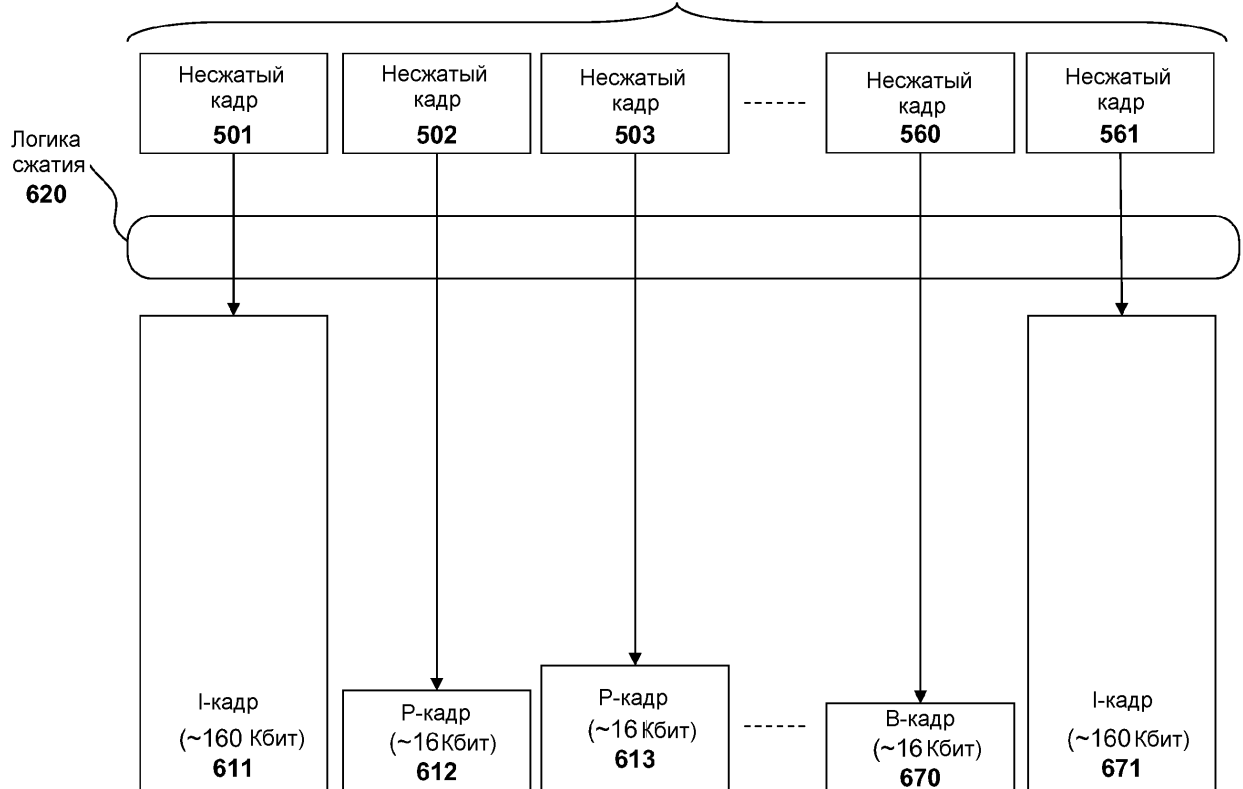


Фиг.4f



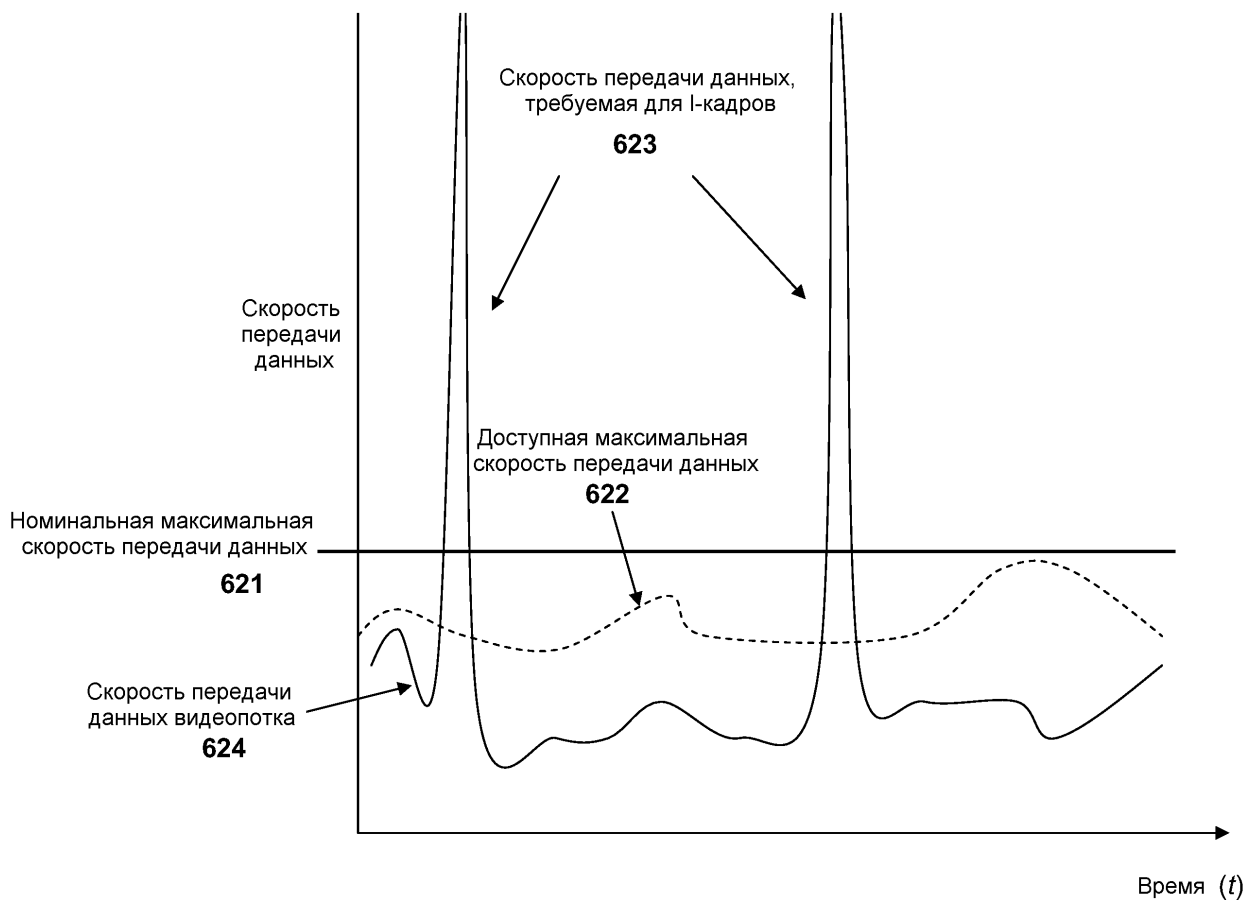
Фиг.5

Например, 900 Кбайт/кадр

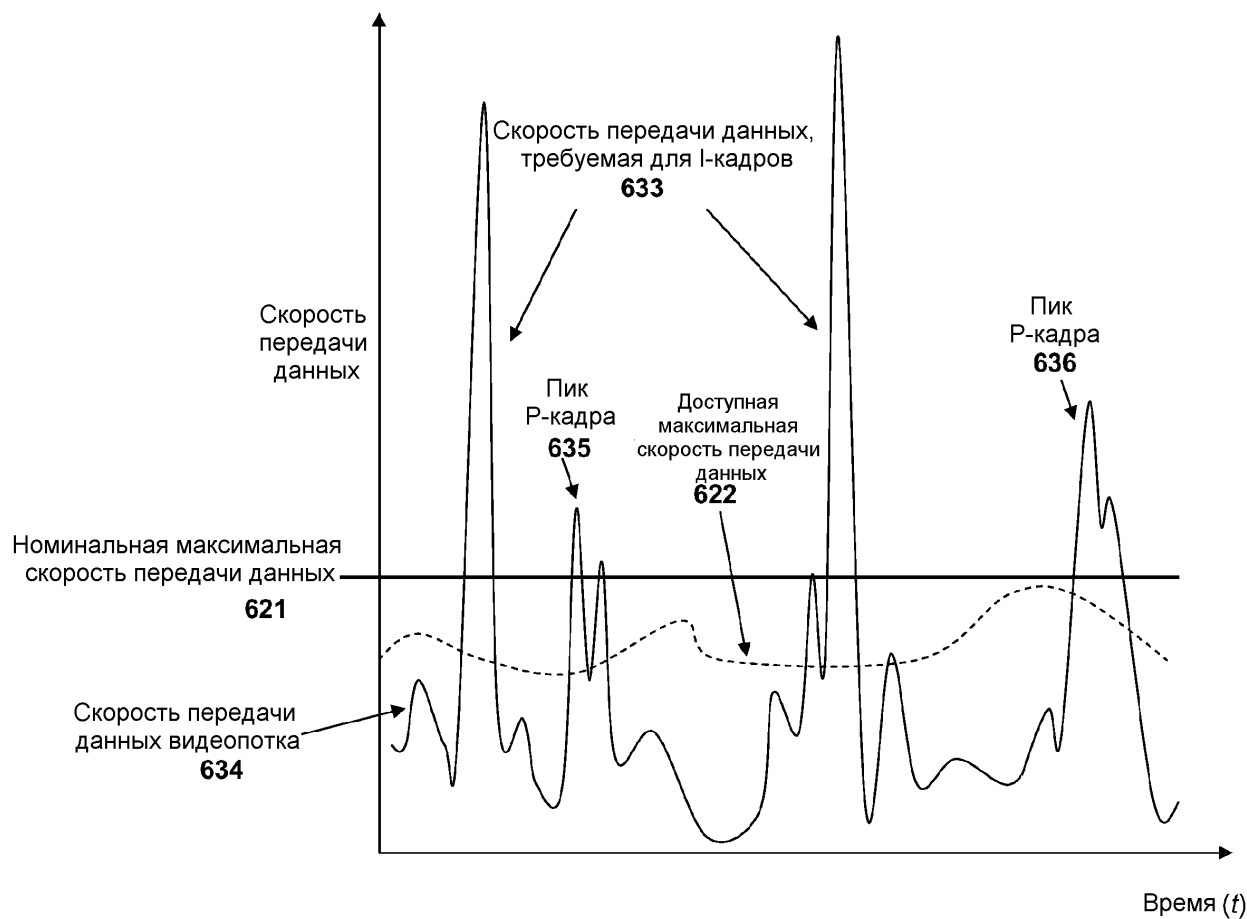


Предшествующий уровень техники

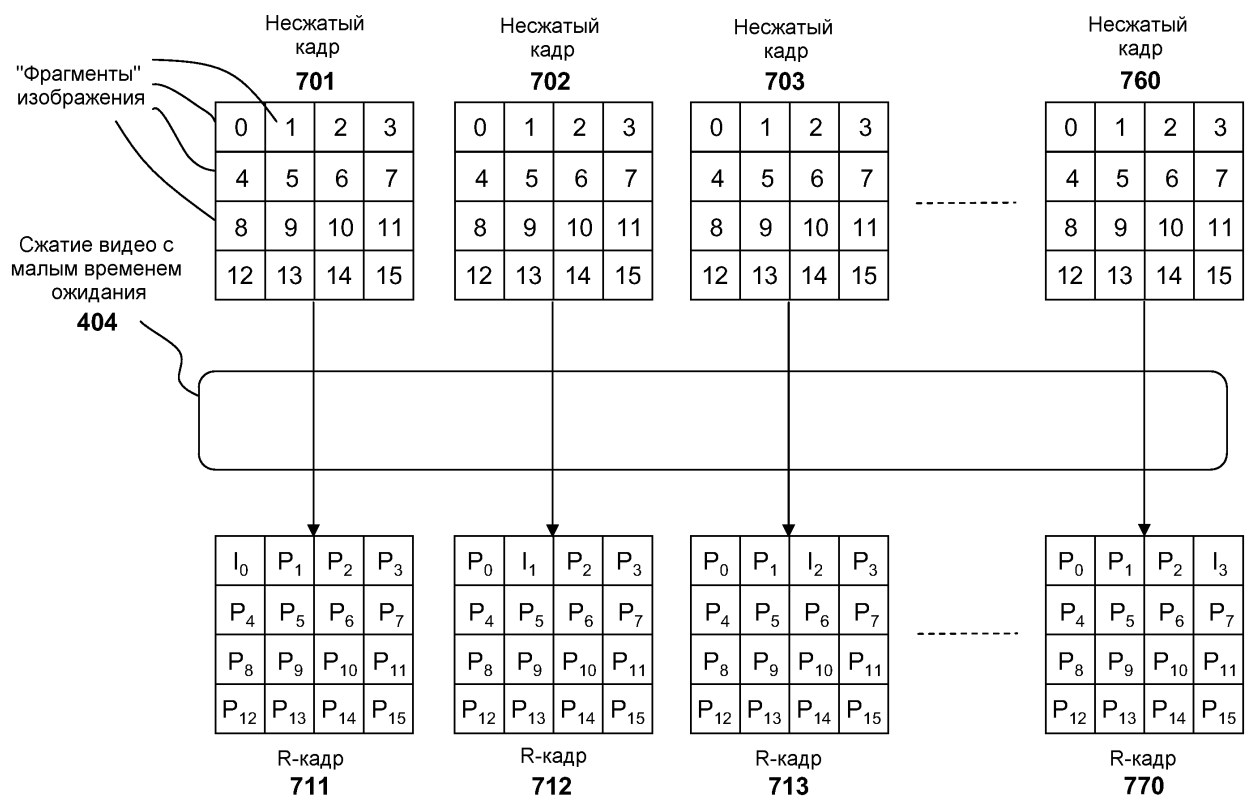
Фиг.6а



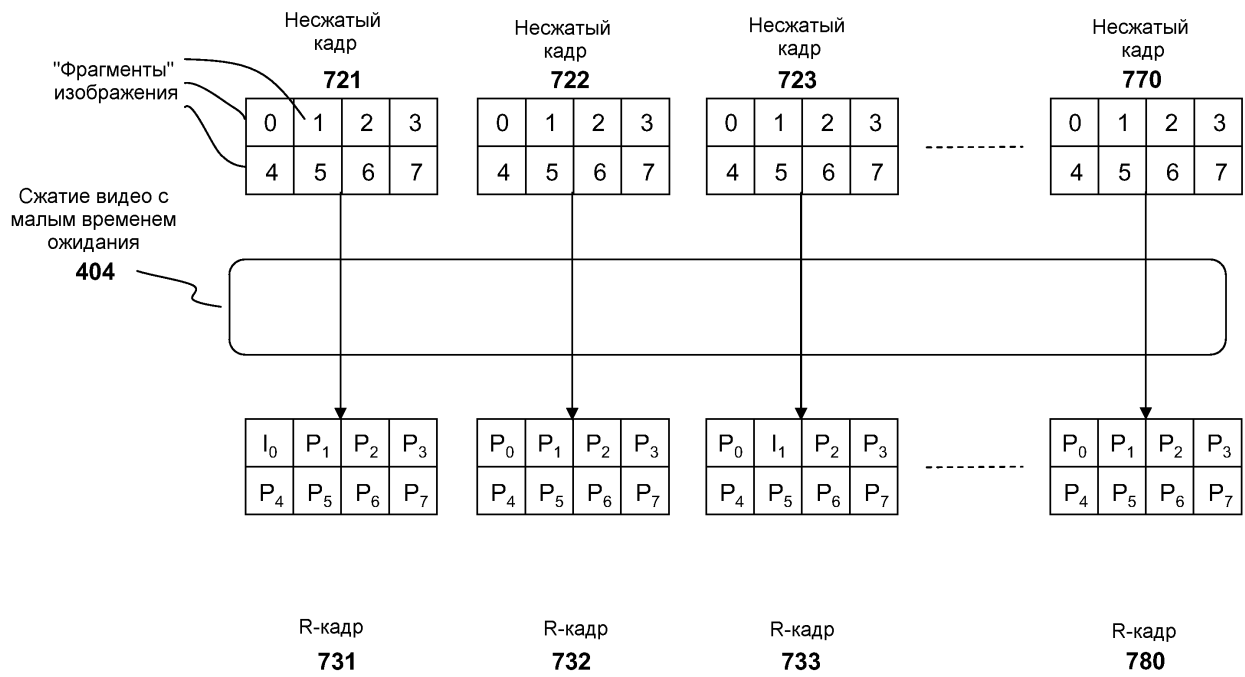
Фиг.6b



Фиг.6с

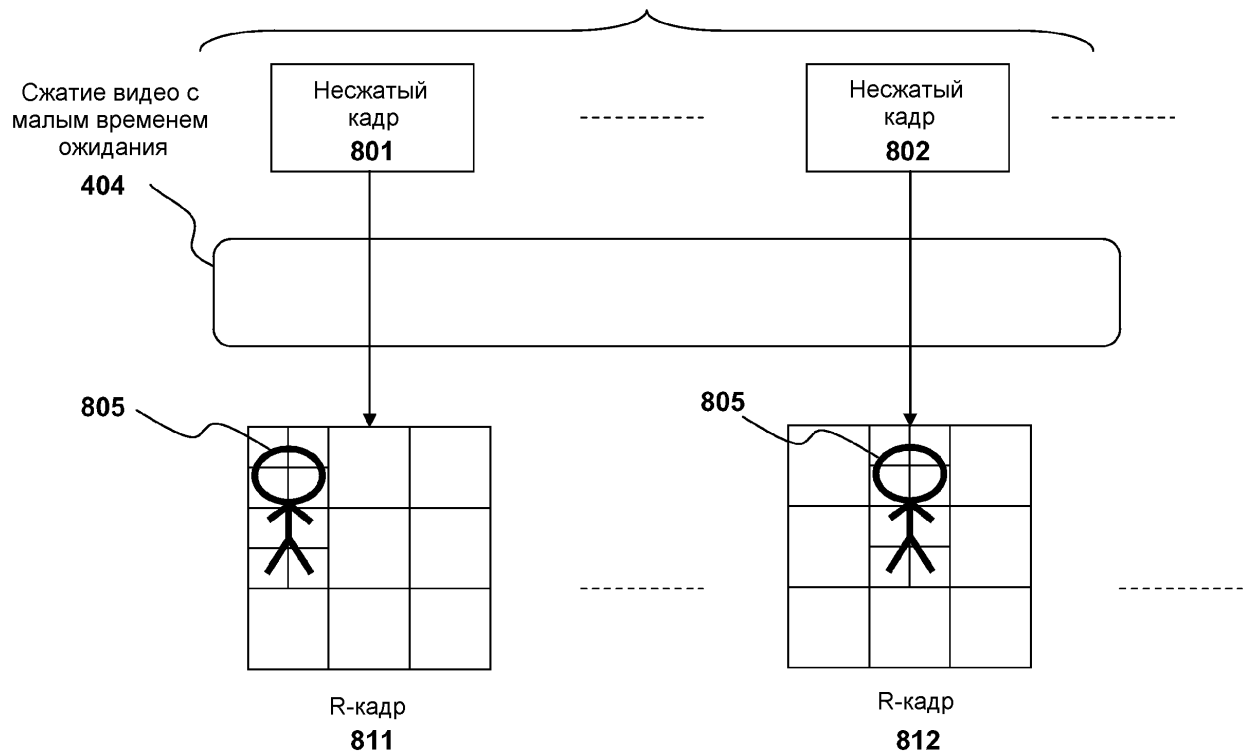


Фиг.7а

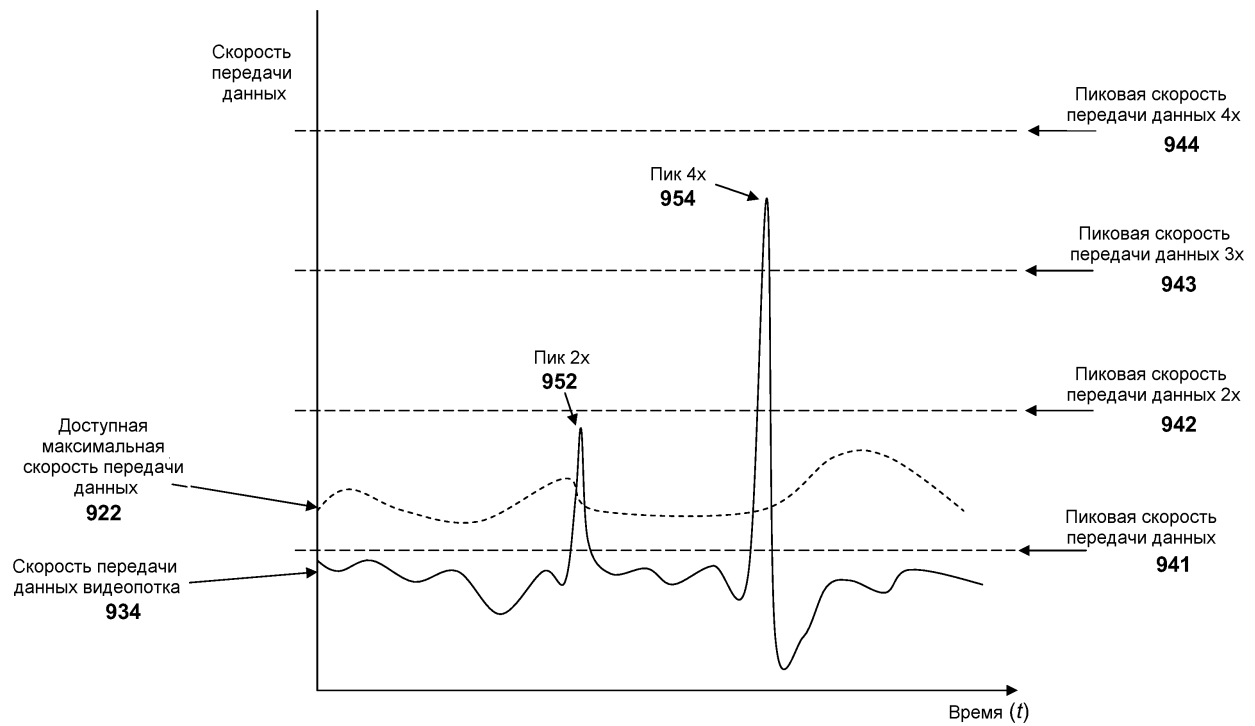


Фиг. 7b

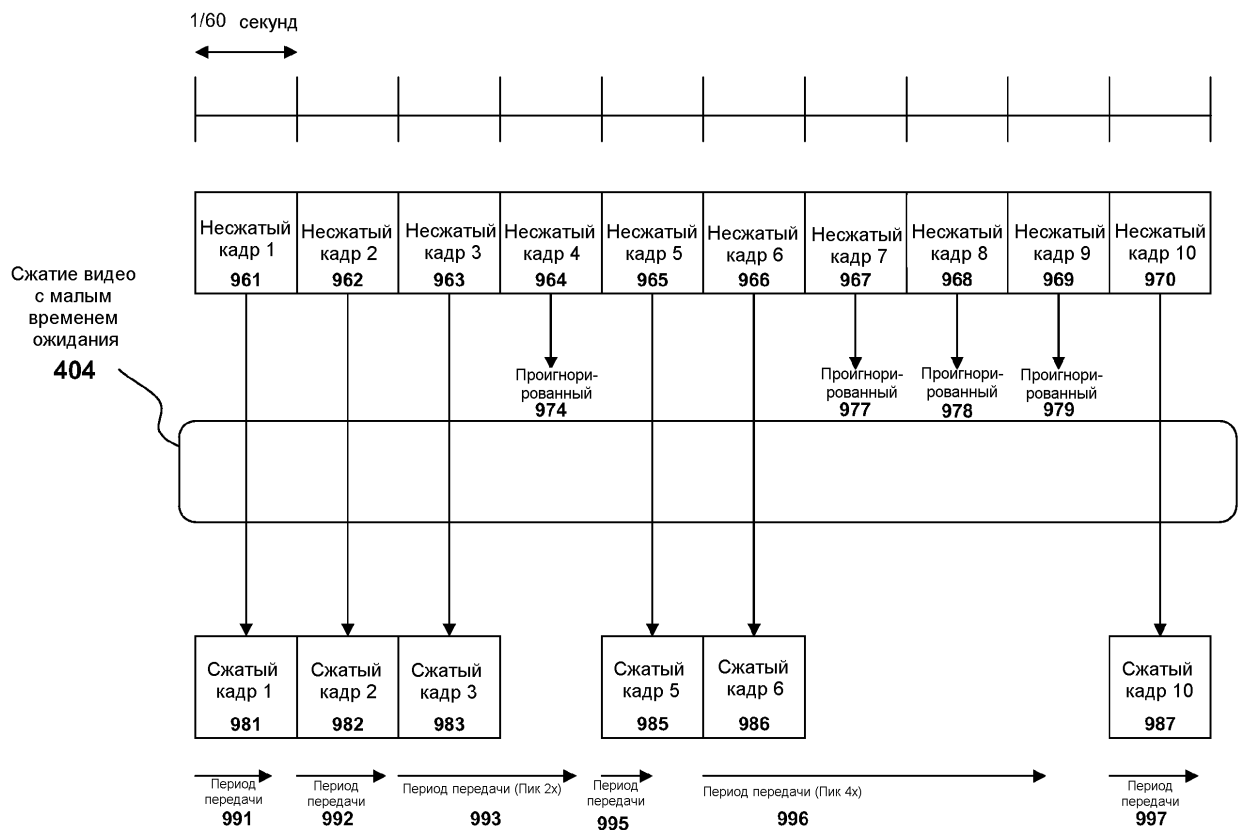
Например, 900 Кбайт/кадр



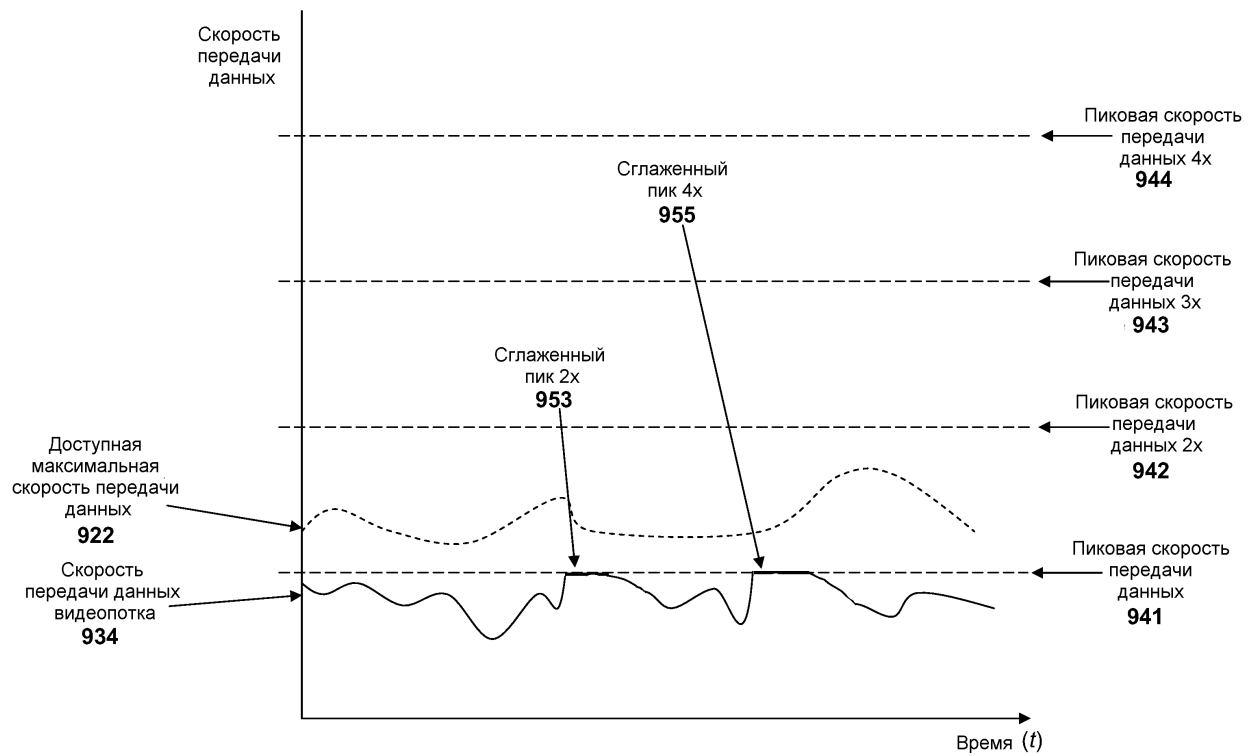
Фиг. 8



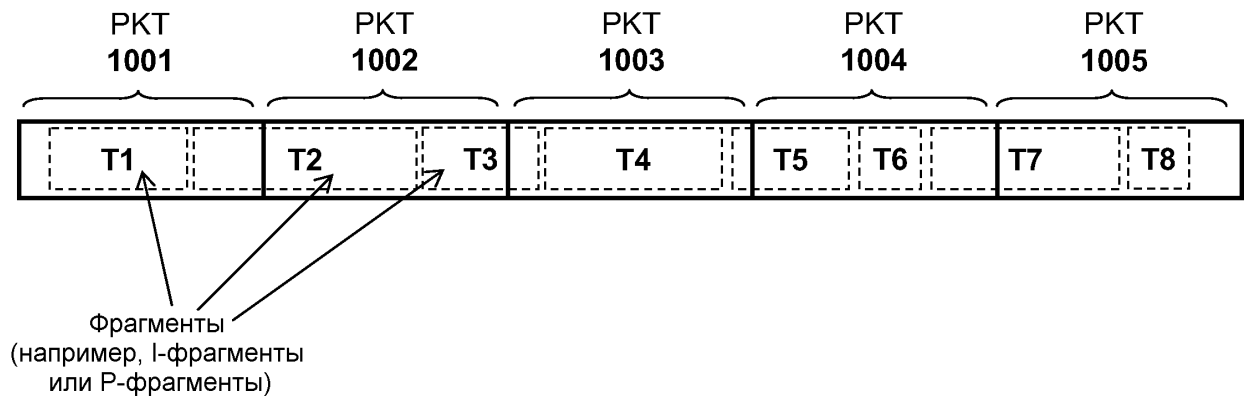
Фиг.9а



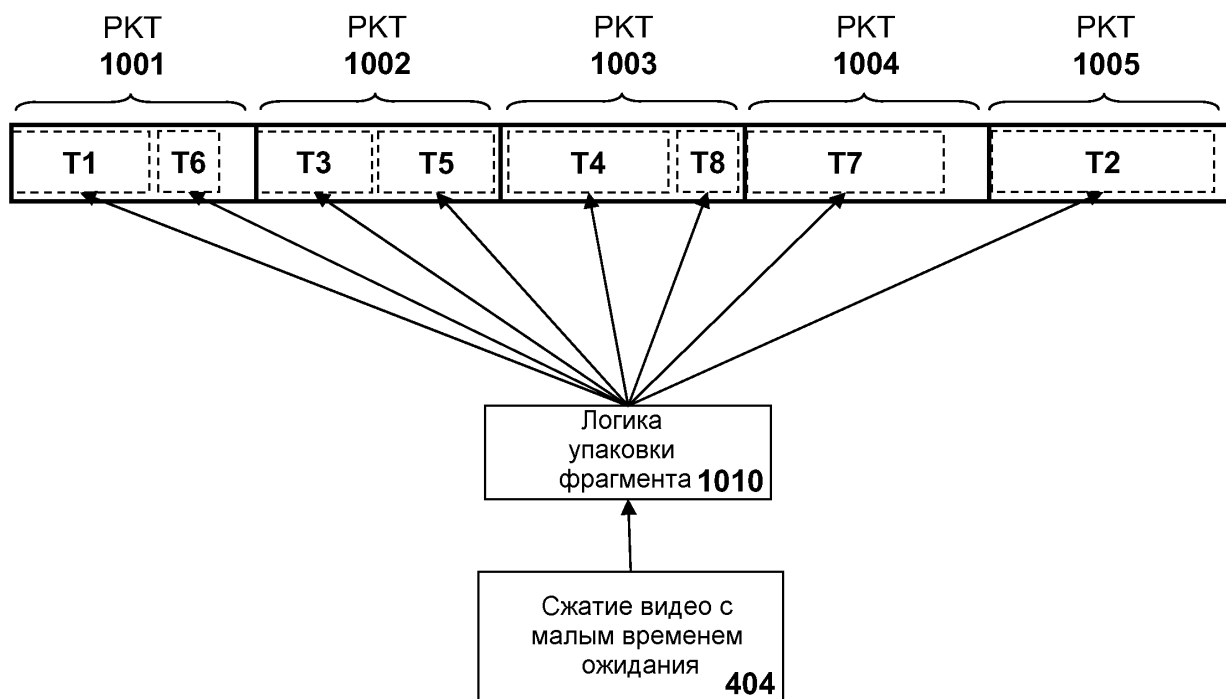
Фиг.9b



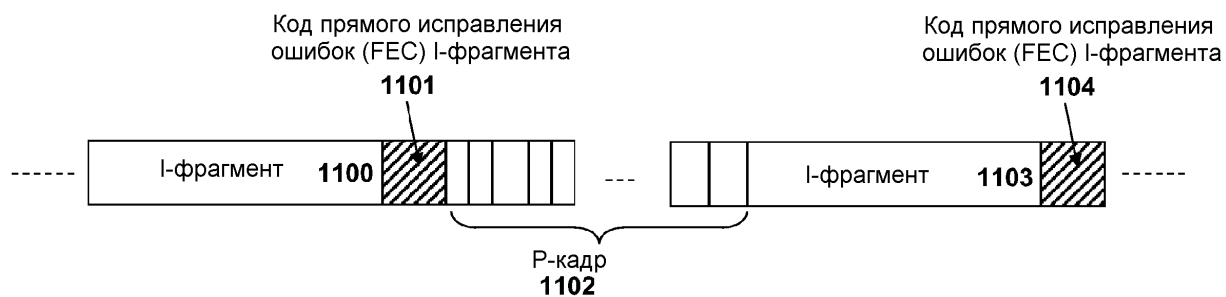
Фиг.9с



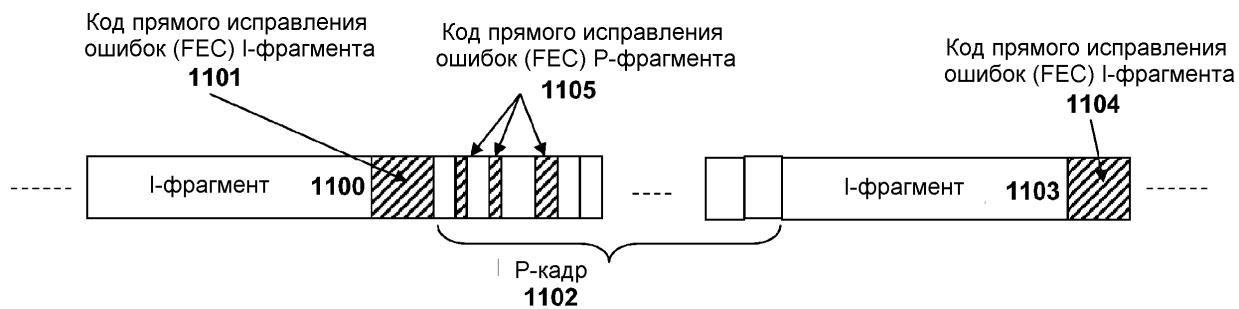
Фиг.10а



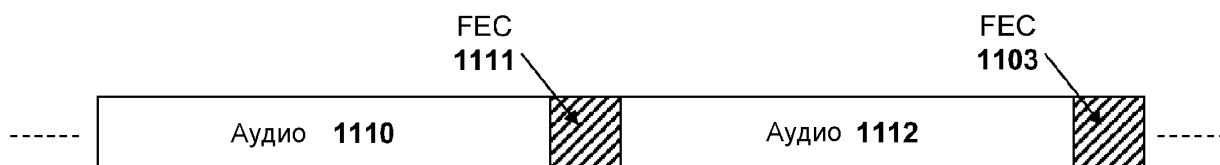
Фиг.10b



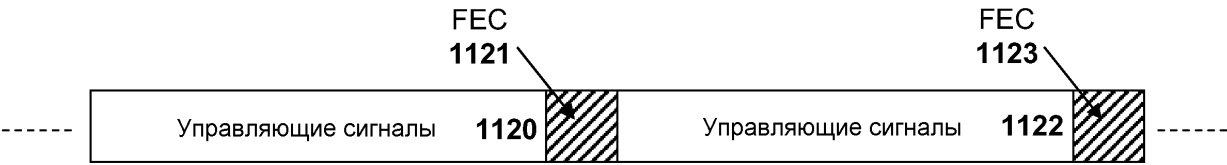
Фиг.11a



Фиг.11b

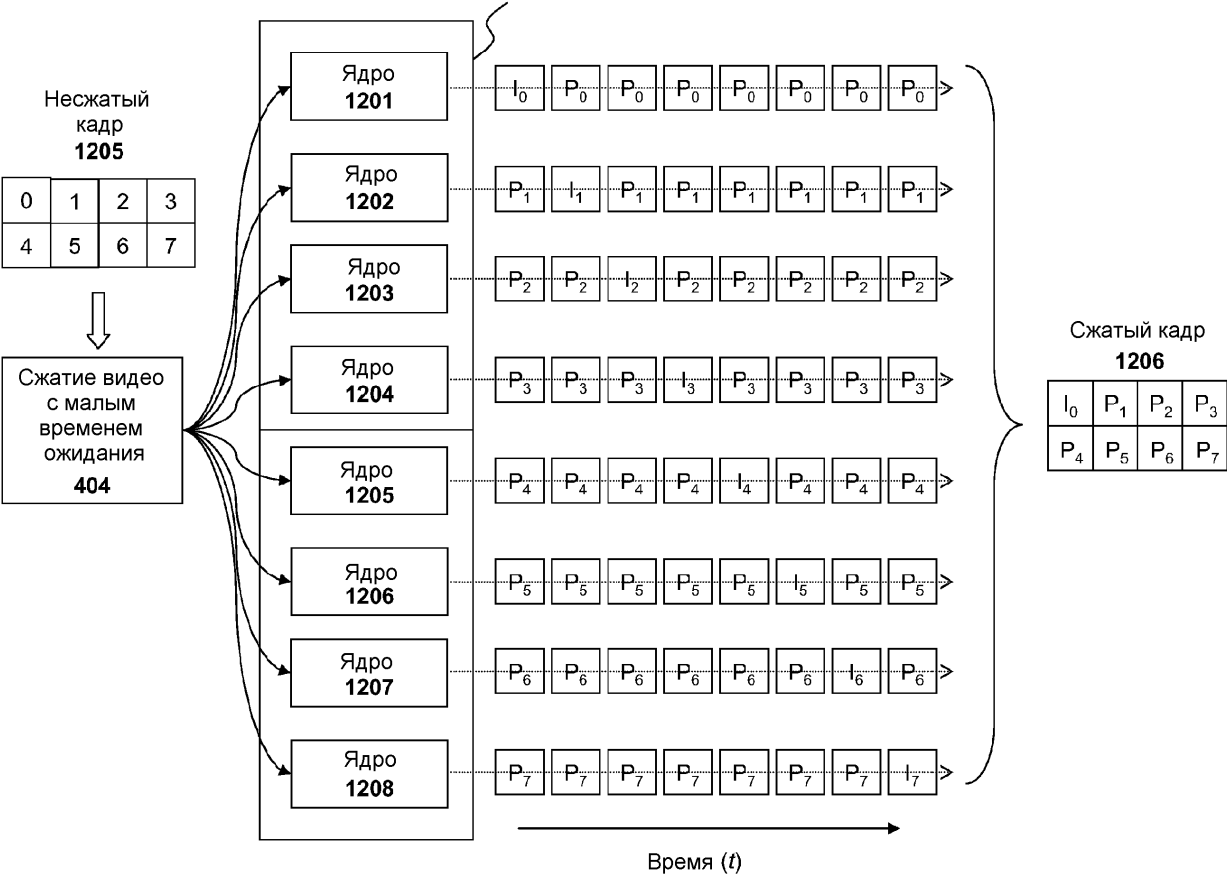


Фиг.11c

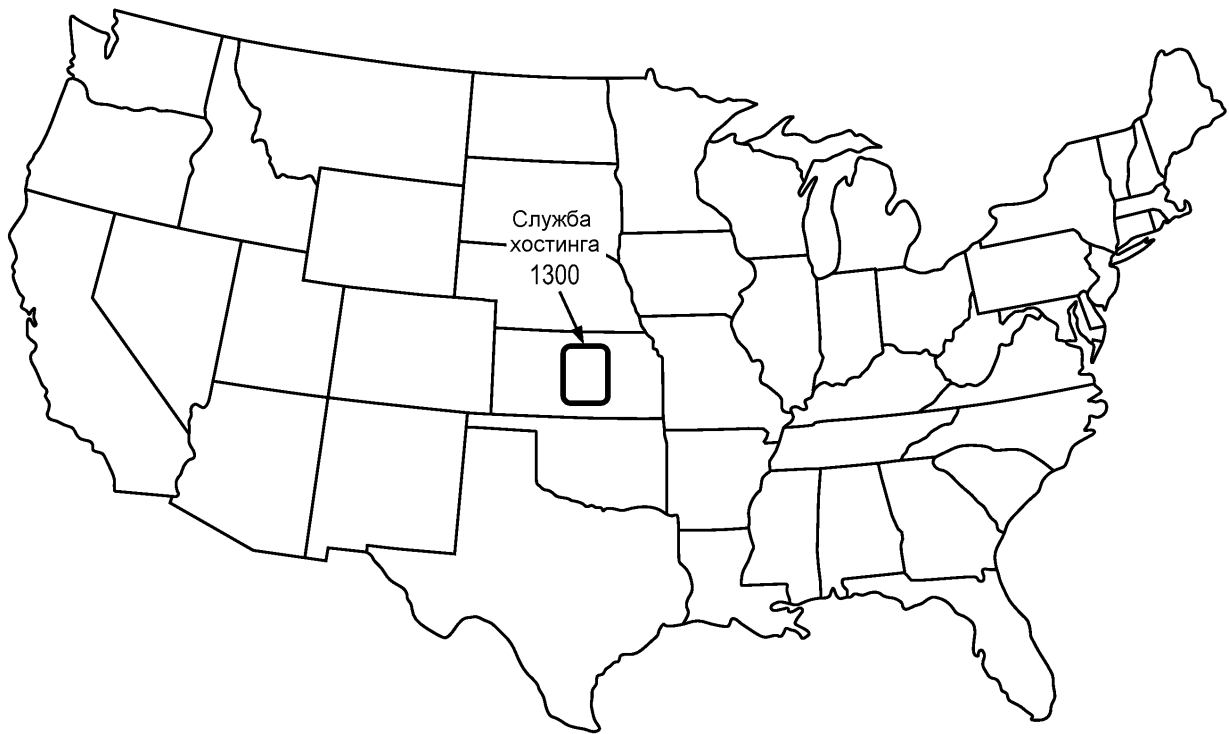


Фиг.11d

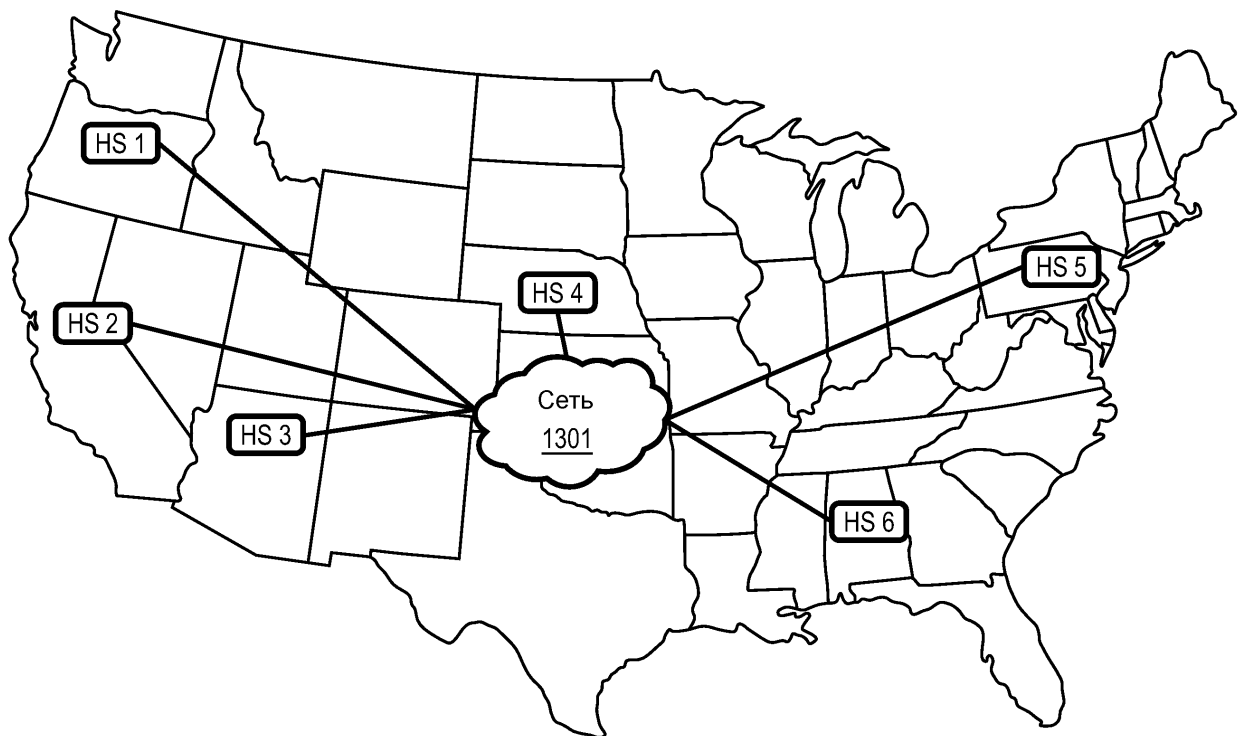
Сдвоенные
четырёхядерные CPU
1200



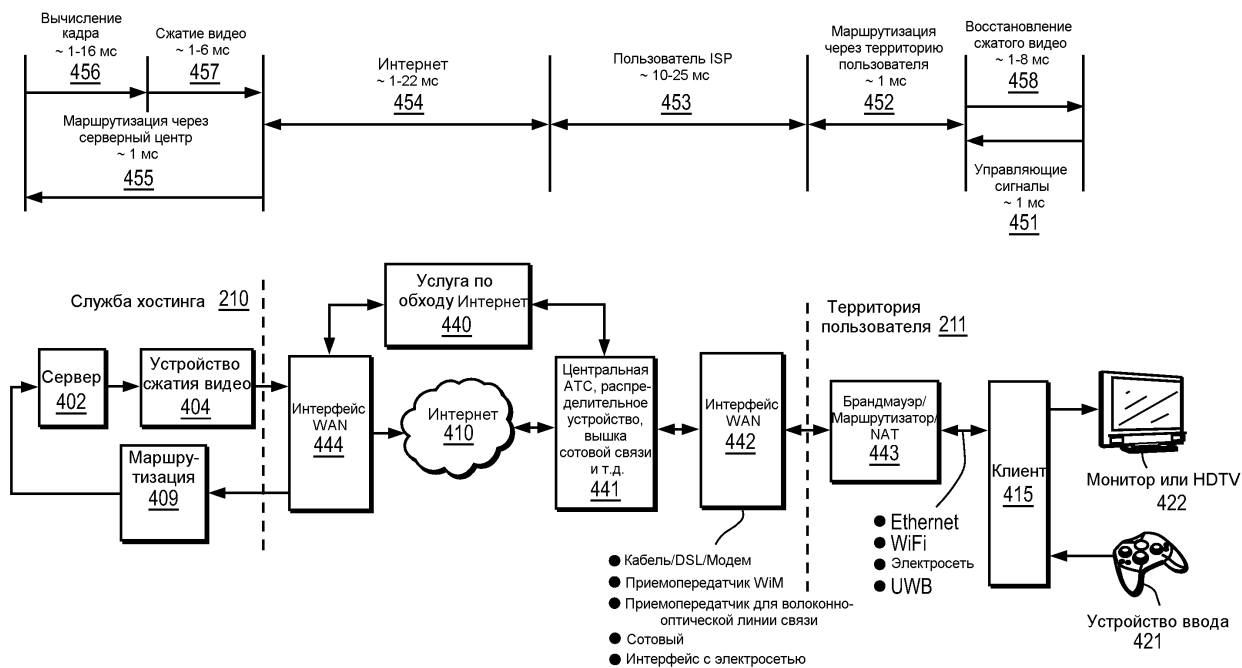
Фиг.12



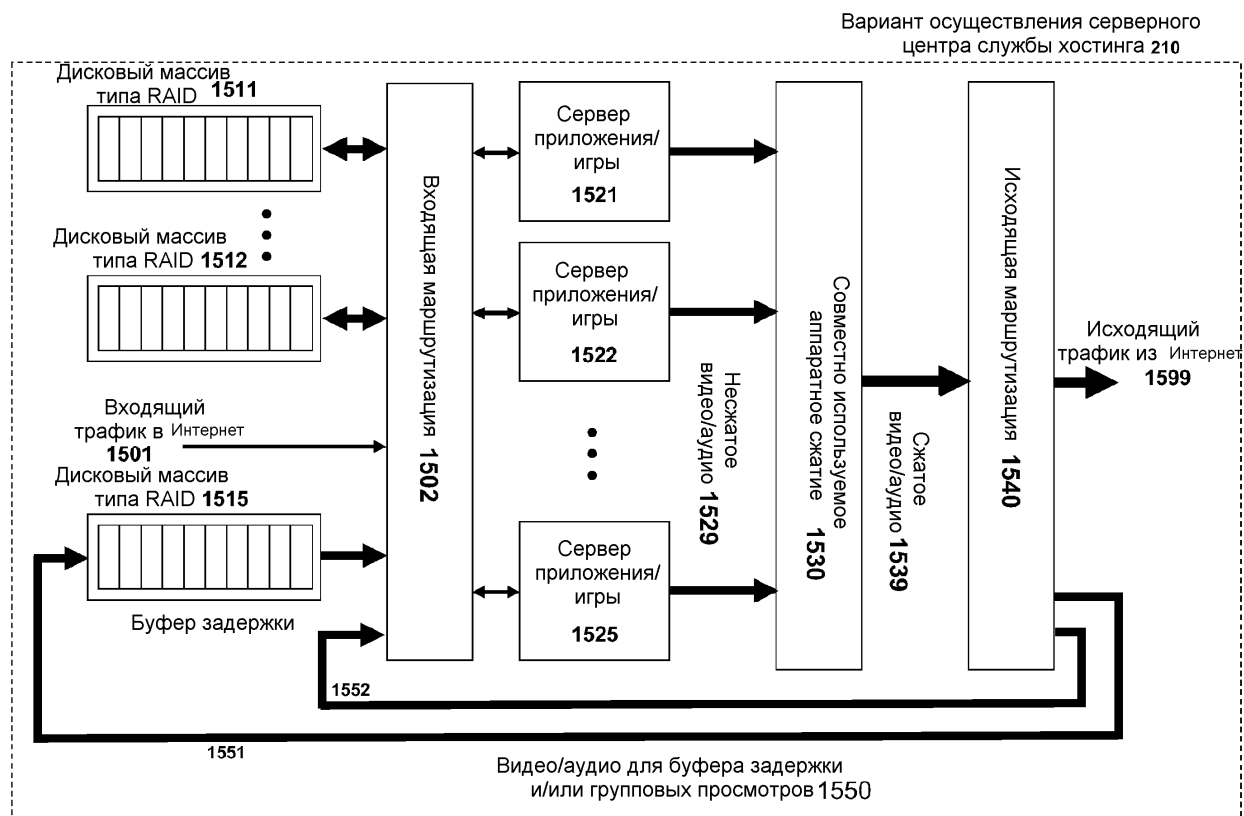
Фиг.13а



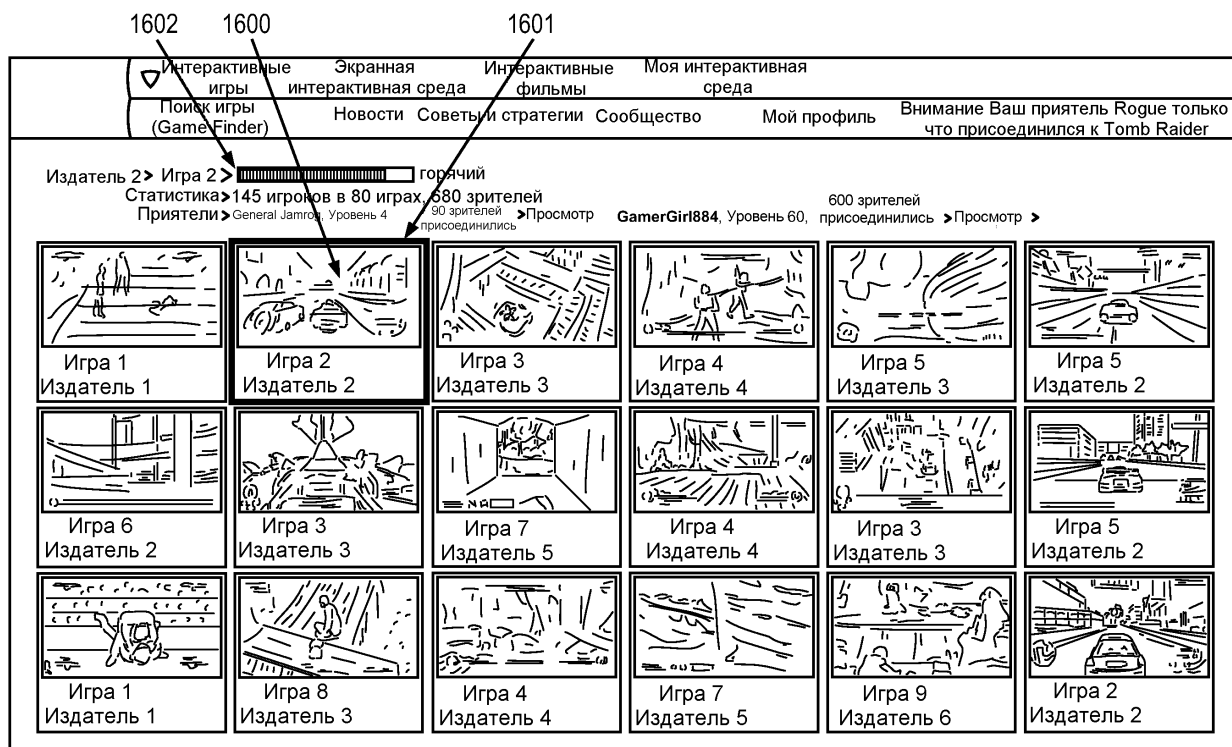
Фиг.13b



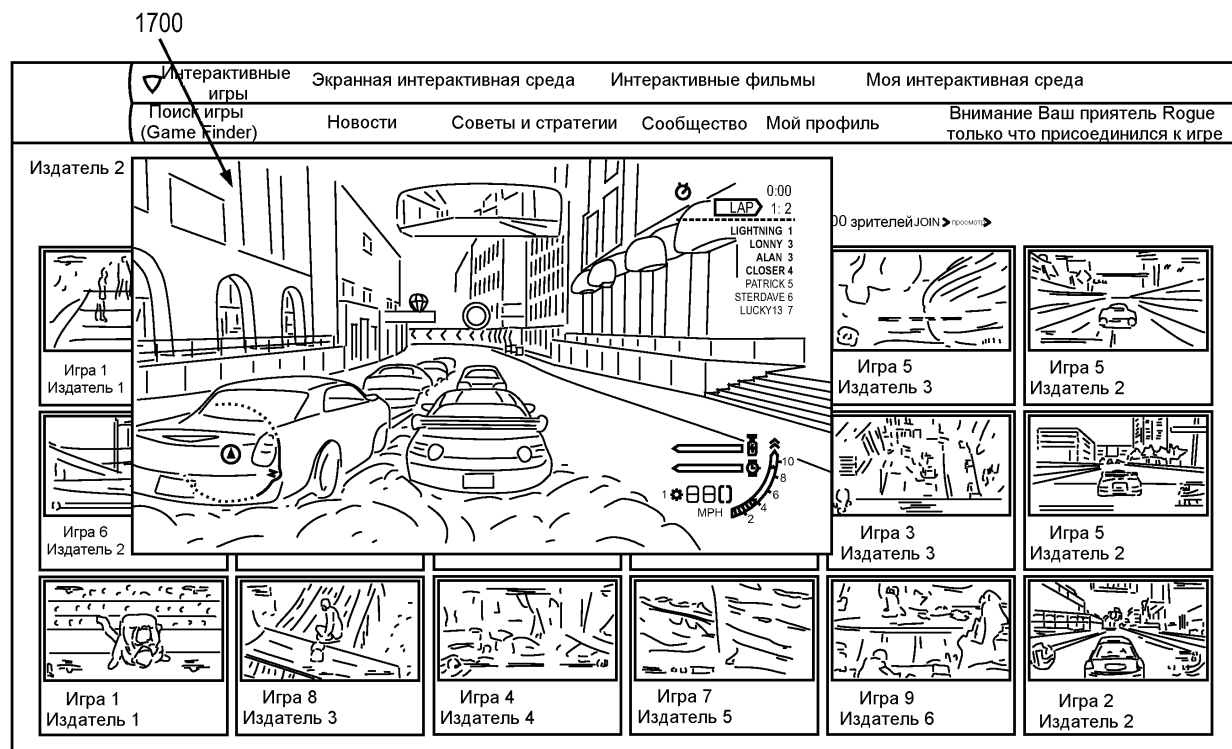
Фиг. 14



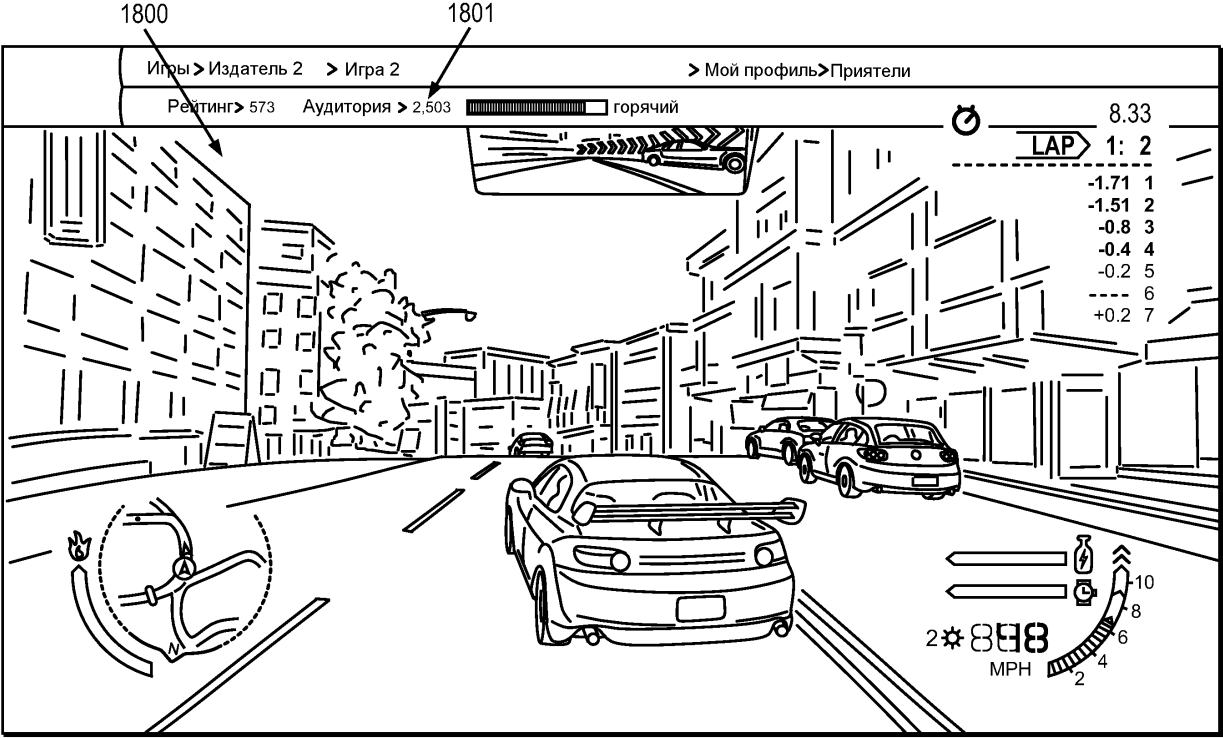
Фиг. 15



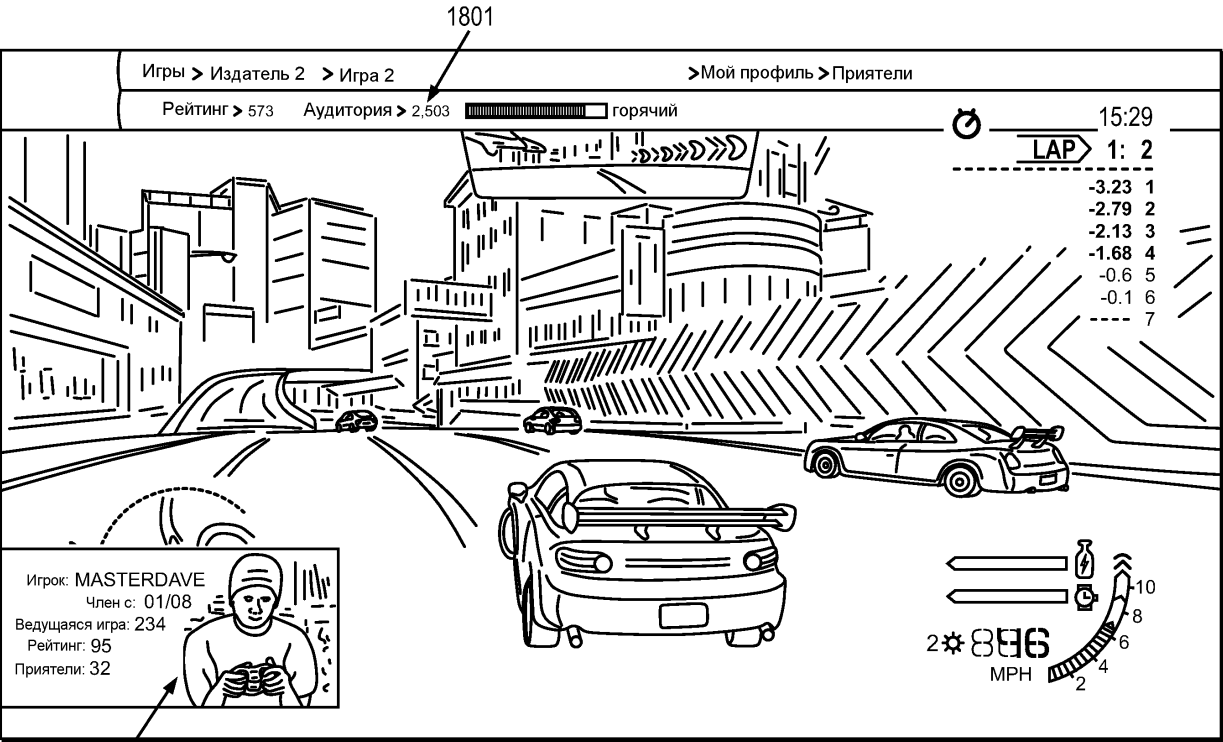
Фиг.16



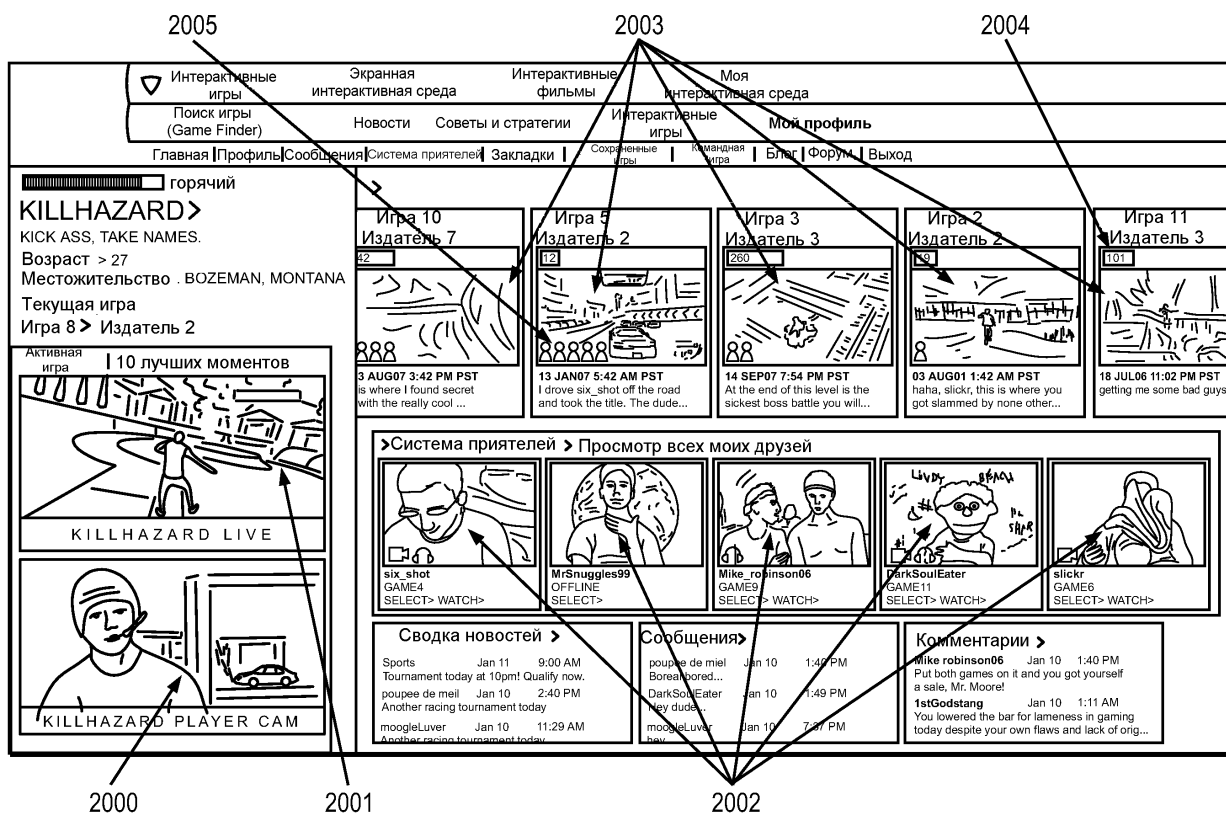
Фиг.17



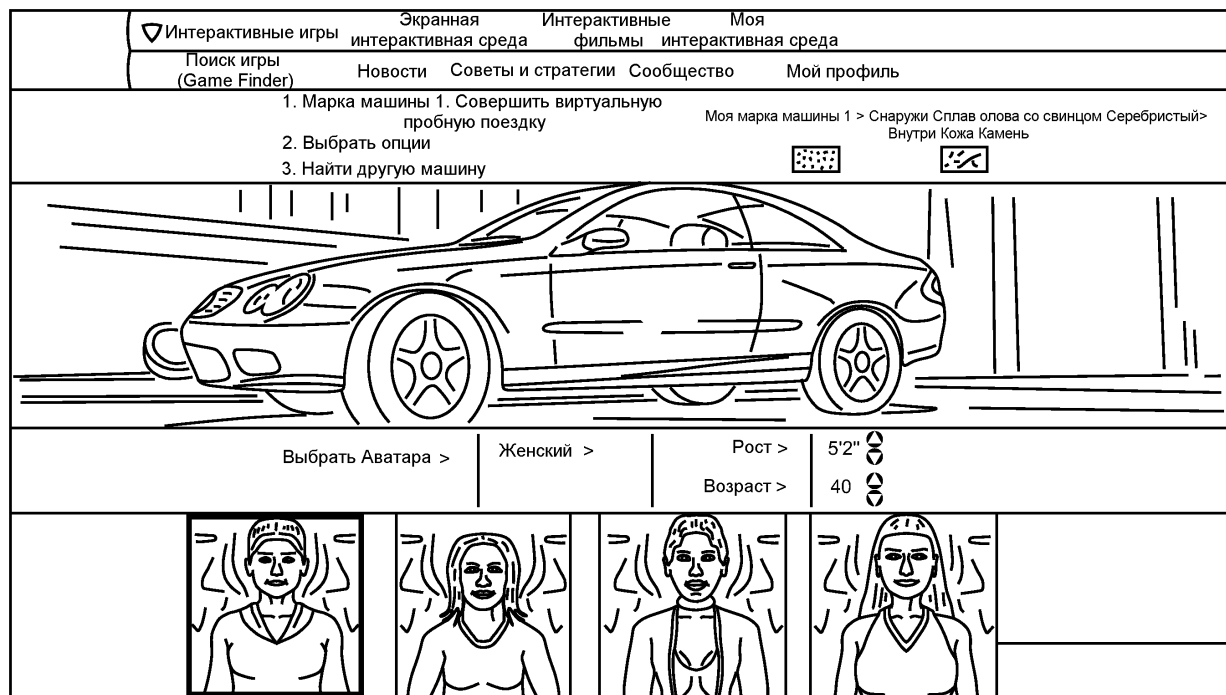
Фиг.18



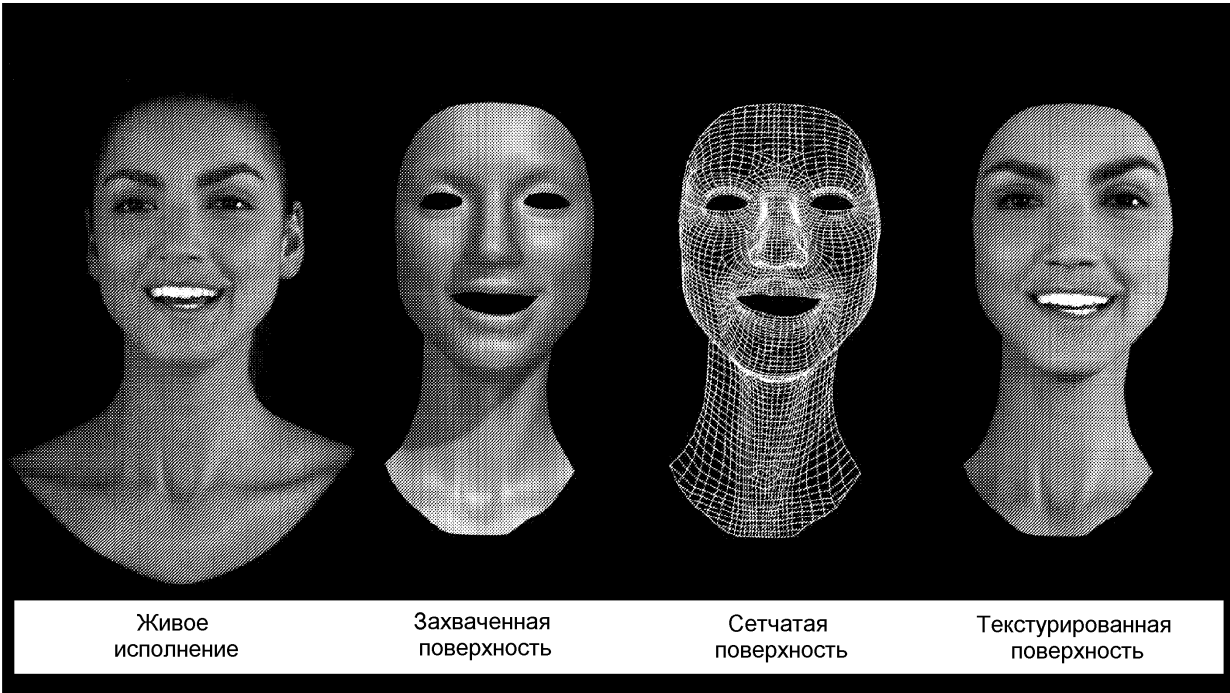
Фиг.19



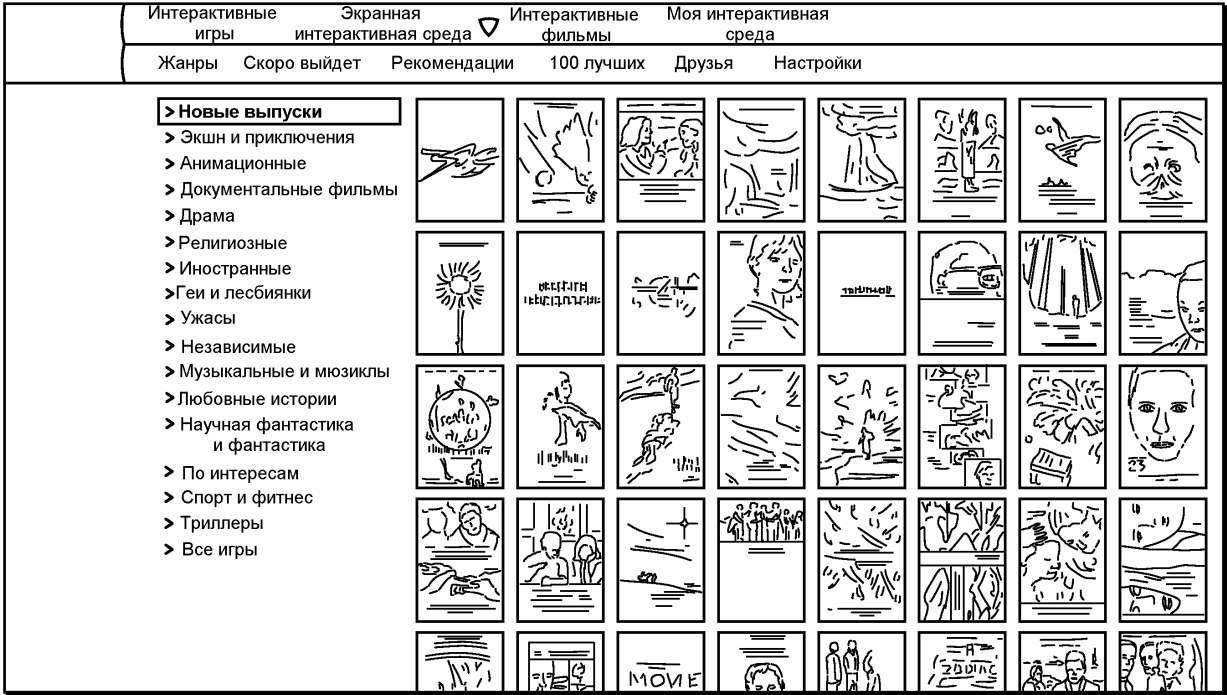
Фиг.20



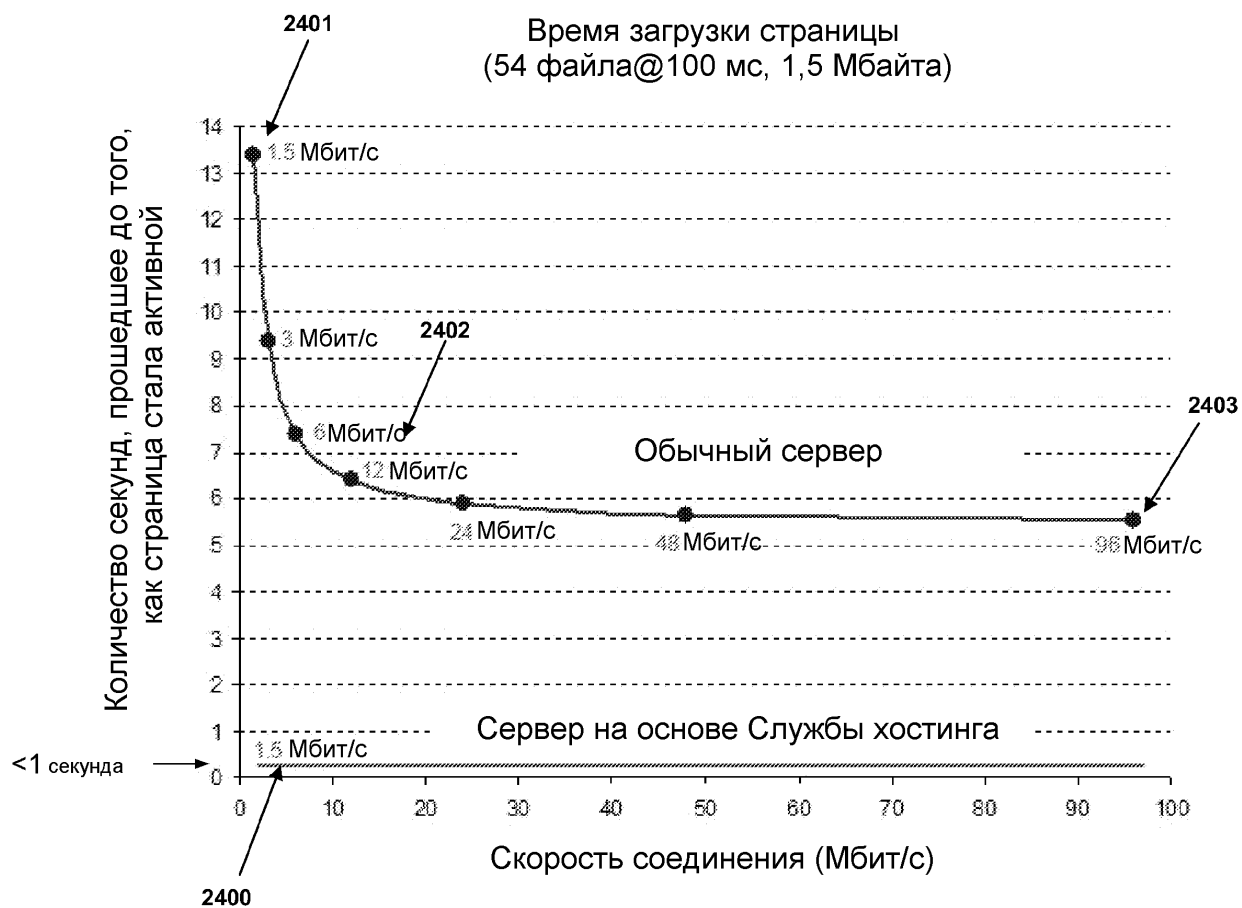
Фиг.21



Фиг.22



Фиг.23



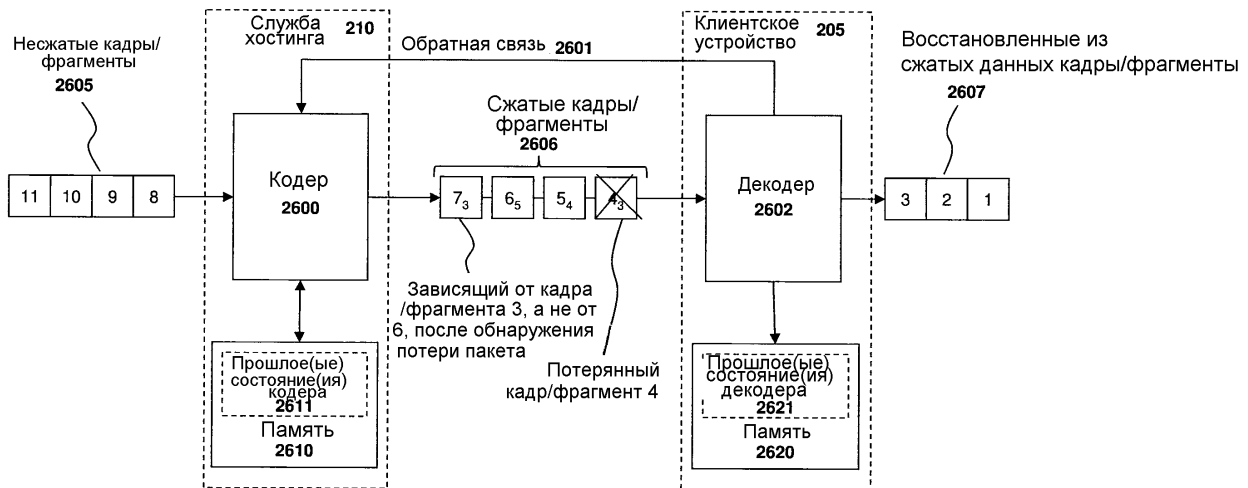
Фиг.24



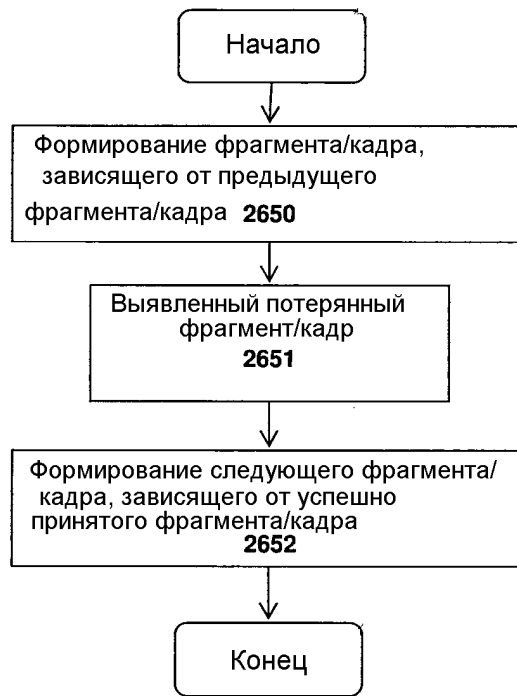
ФИГ.25а



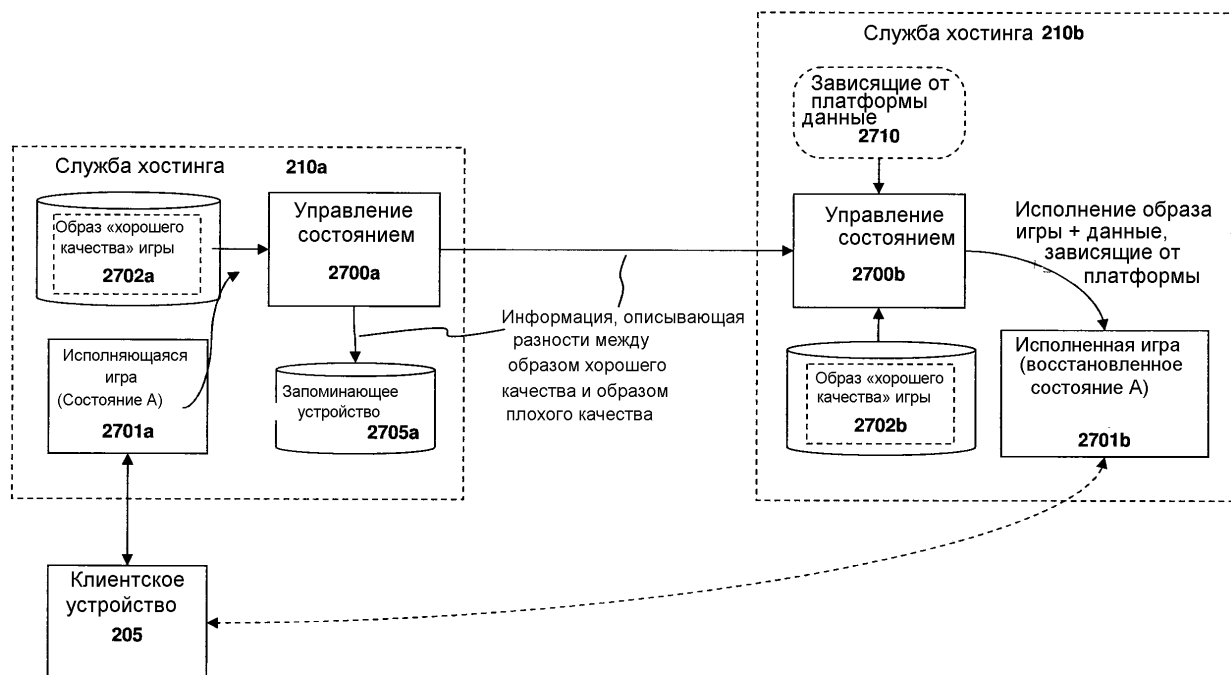
ФИГ.25b



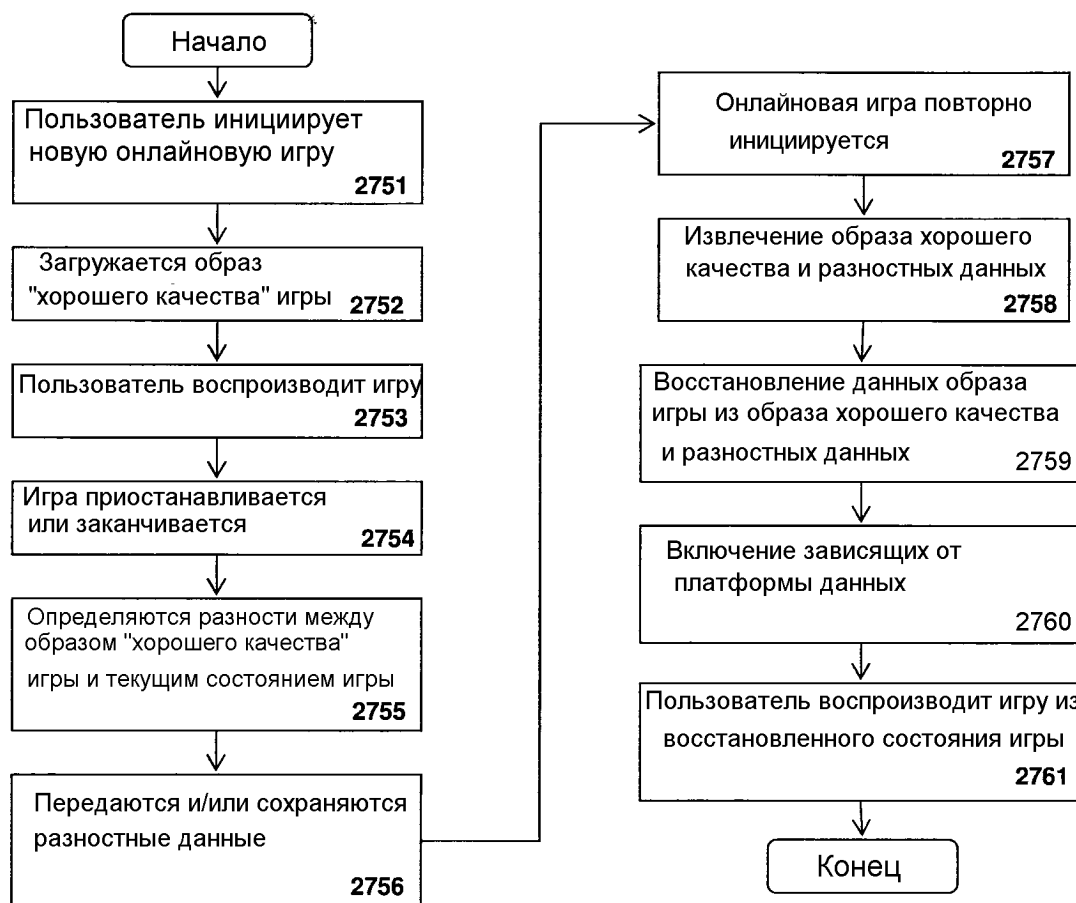
ФИГ.26a



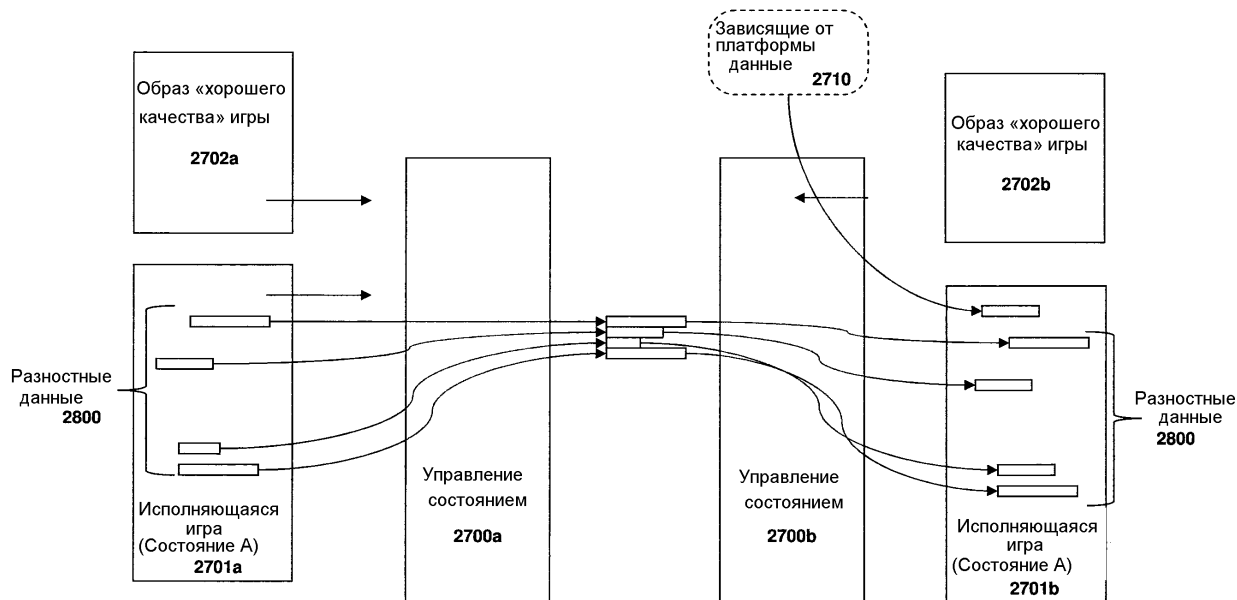
ФИГ.26b



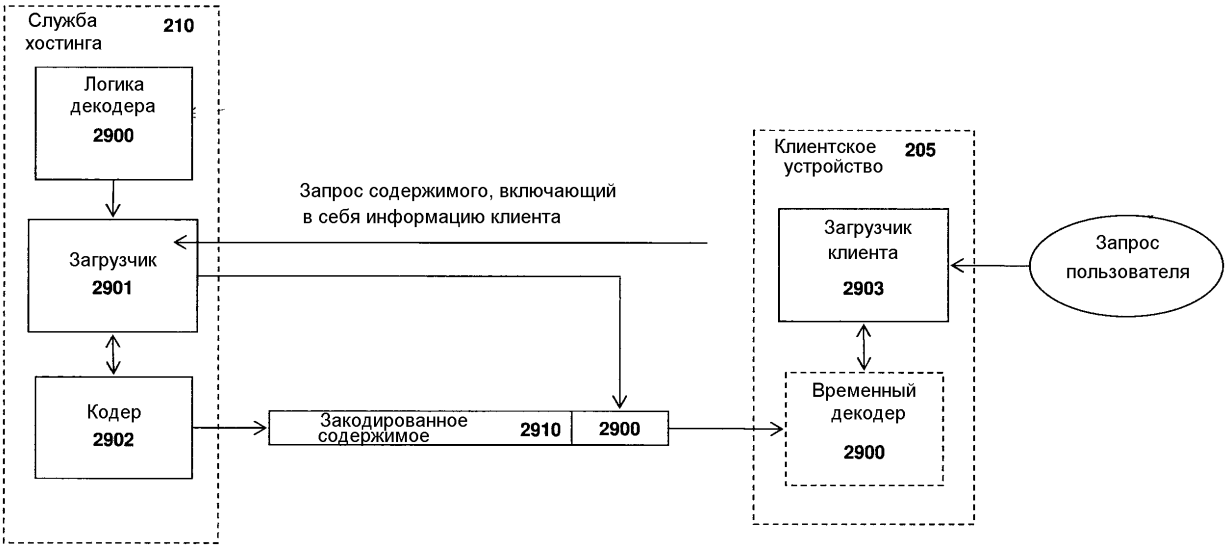
ФИГ.27a



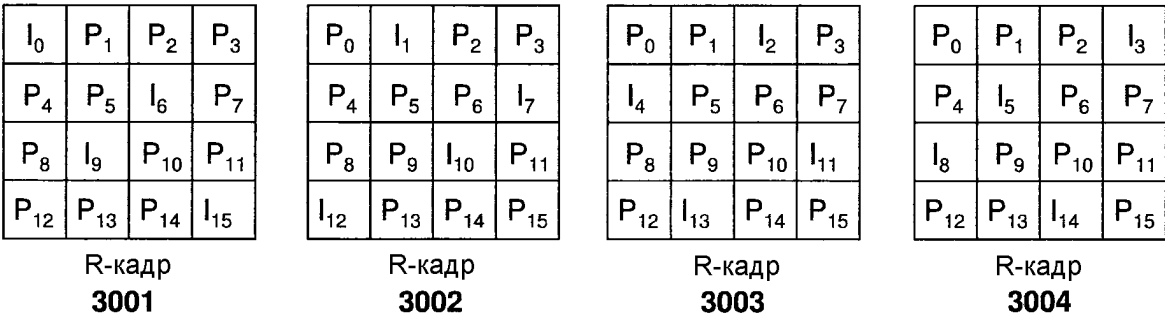
ФИГ.27b



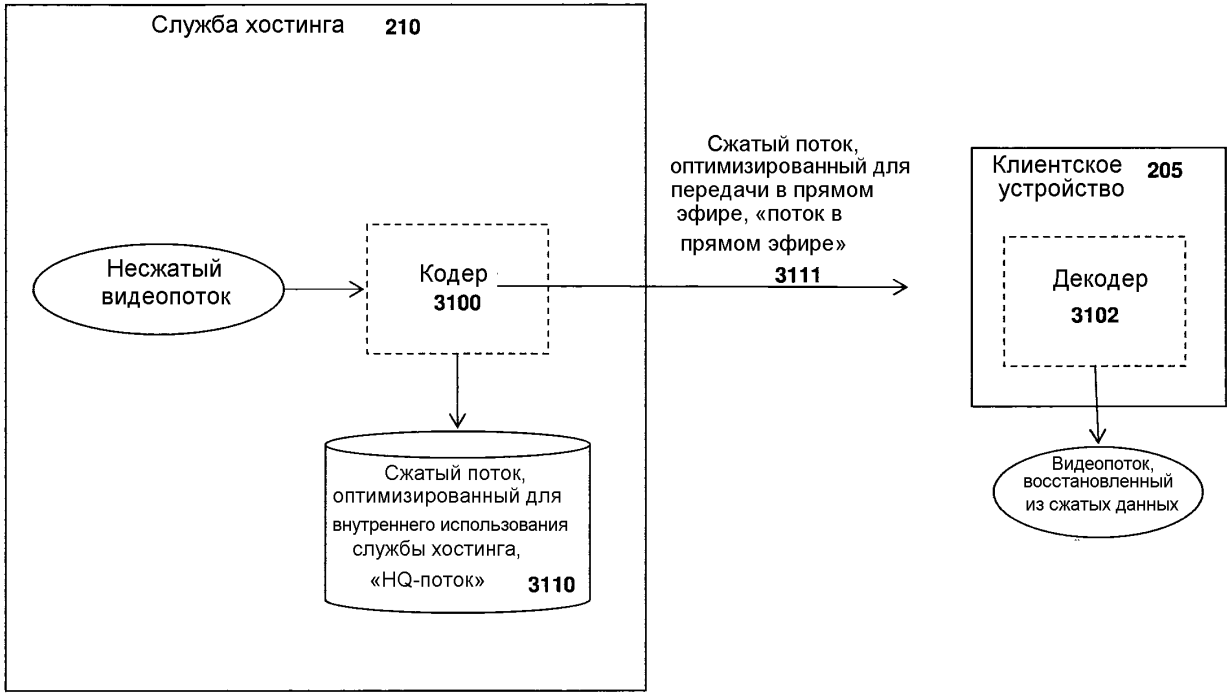
ФИГ.28



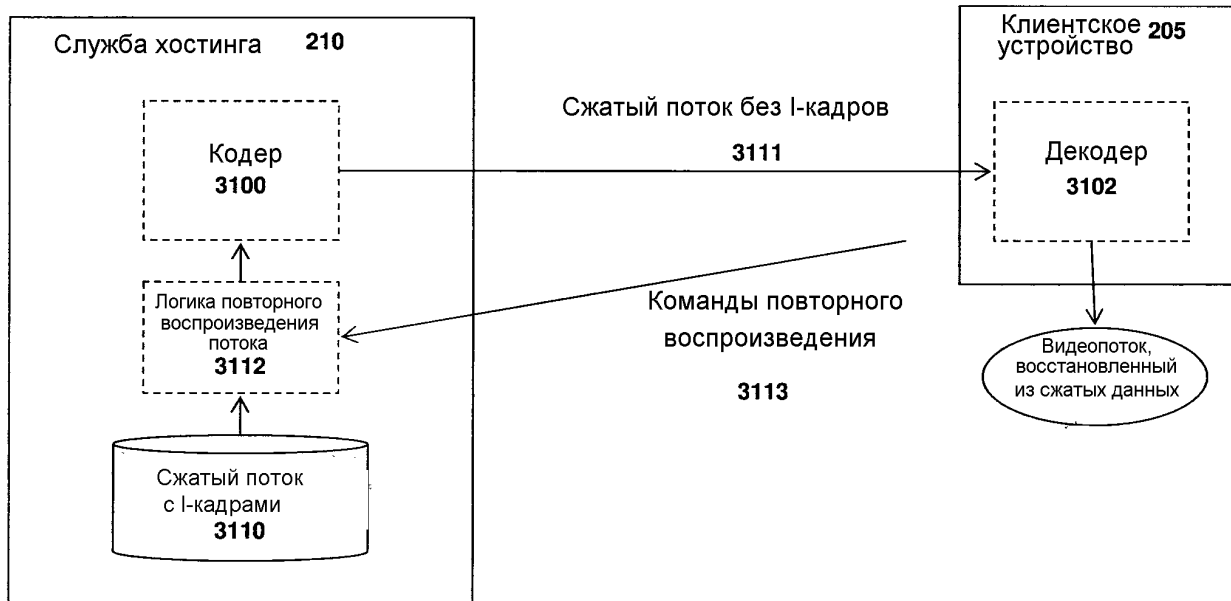
ФИГ.29



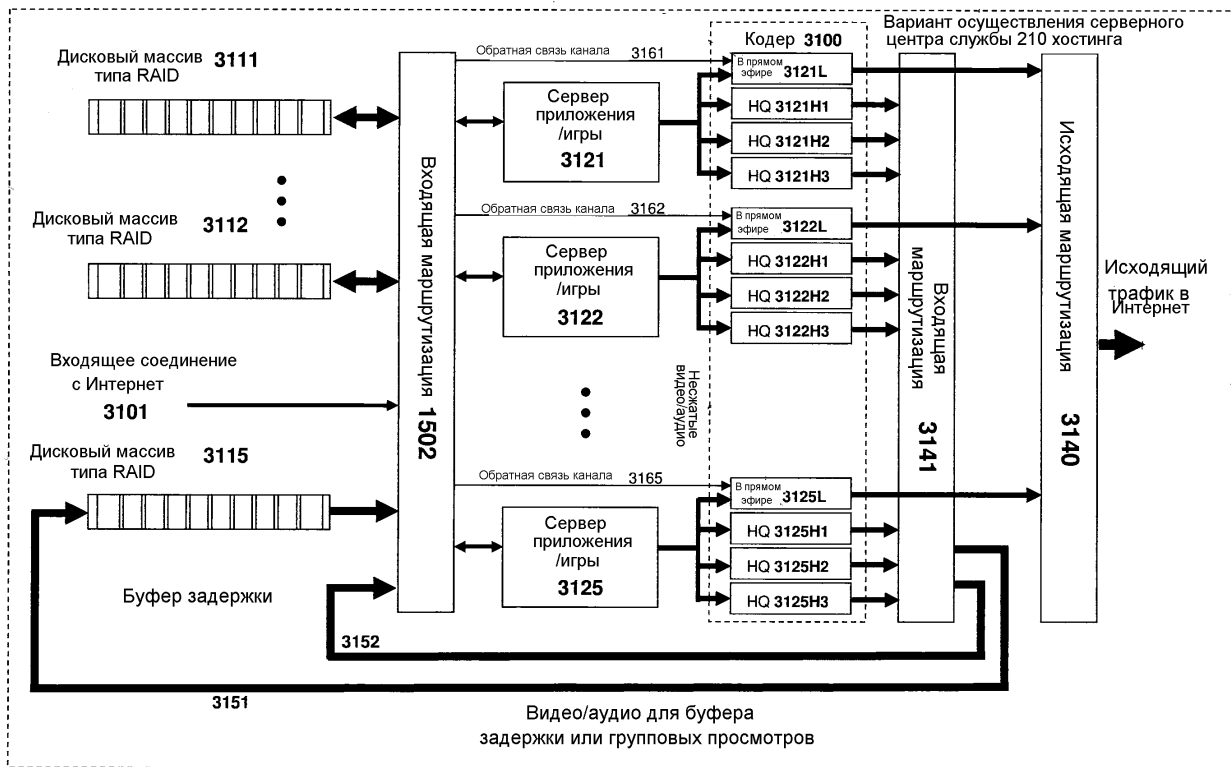
ФИГ.30



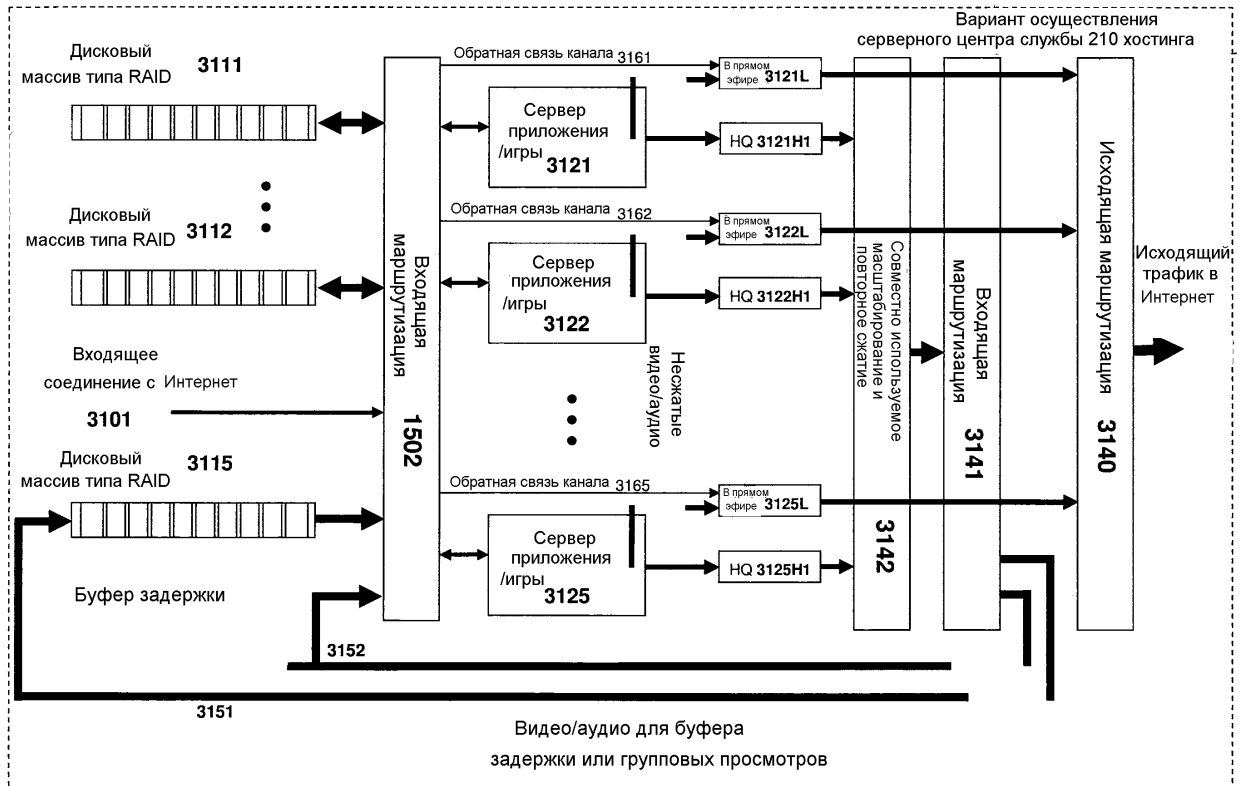
ФИГ.31а



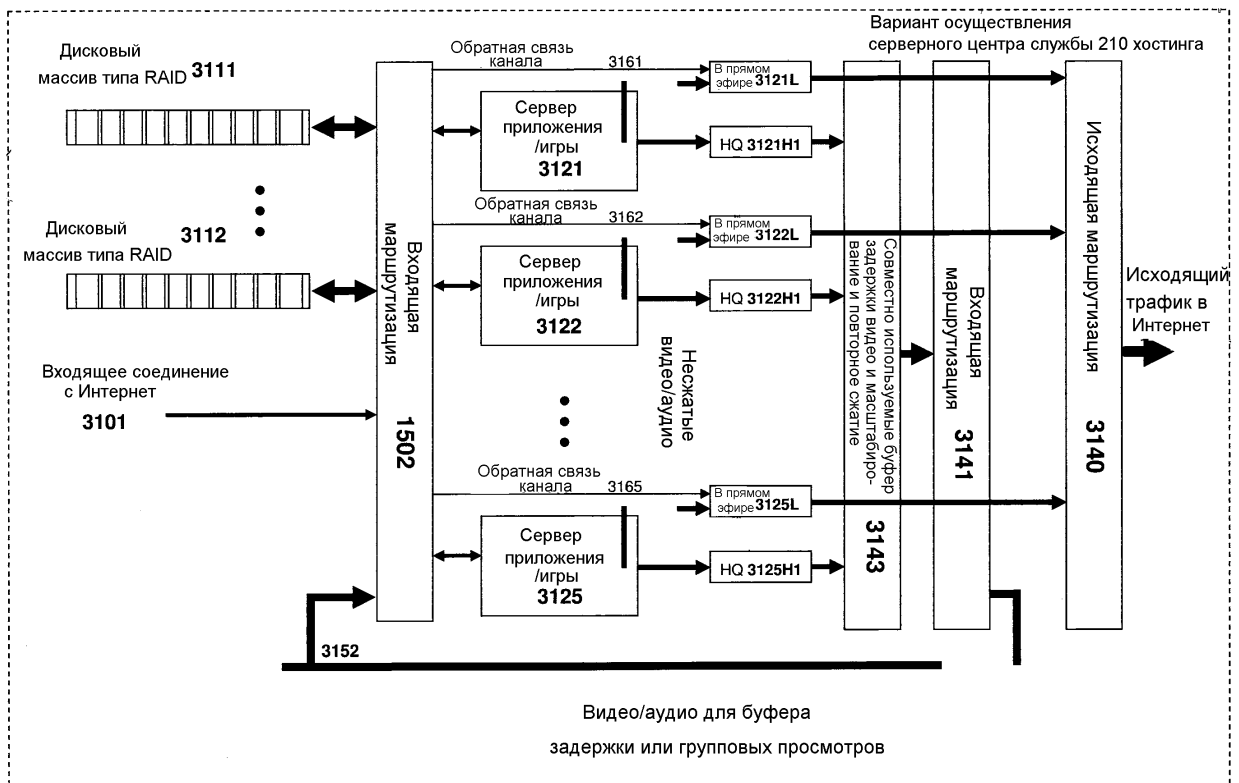
ФИГ.31b



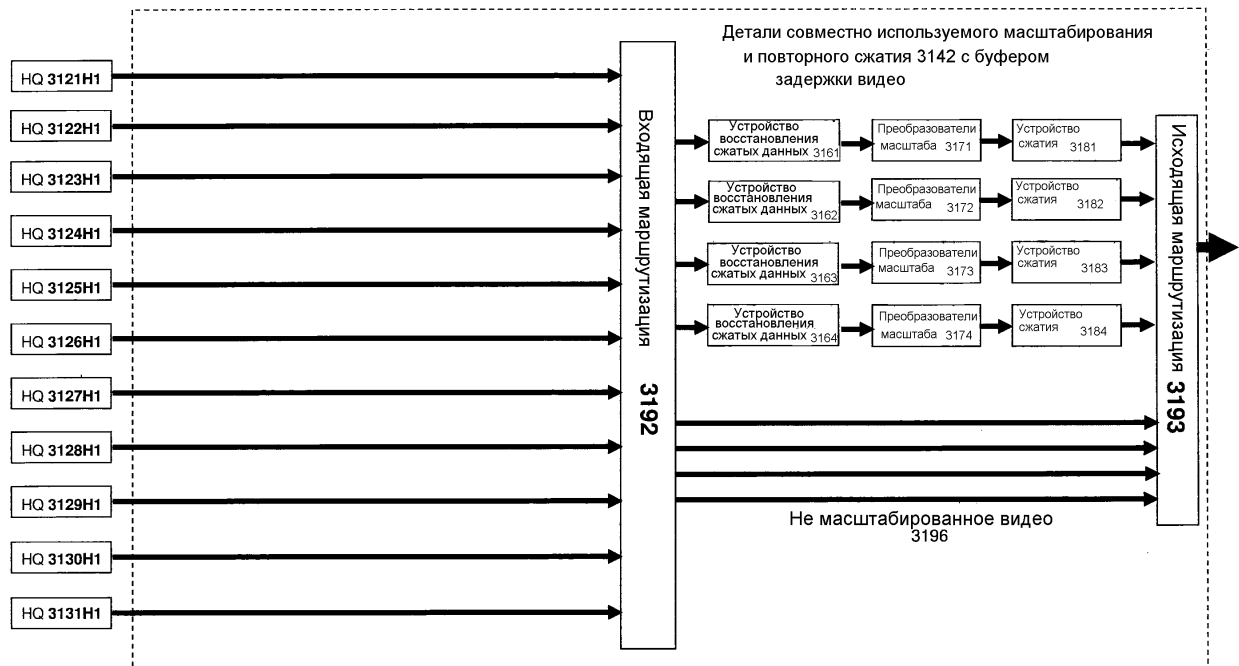
ФИГ.31c



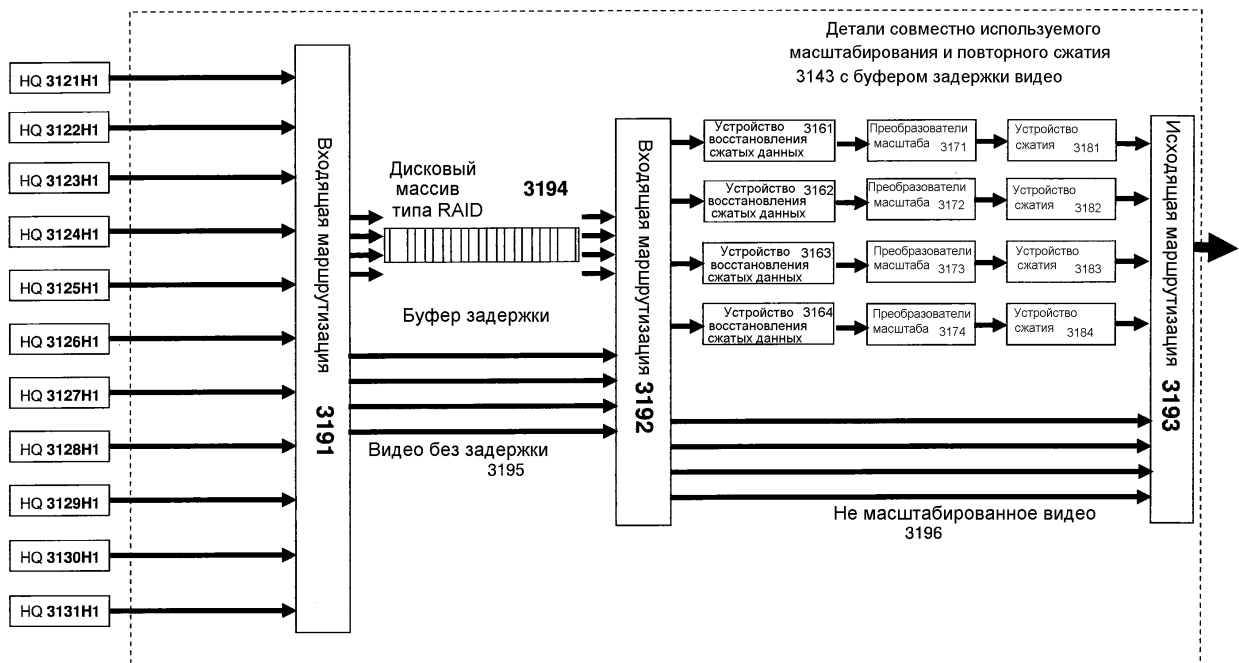
ФИГ.31d



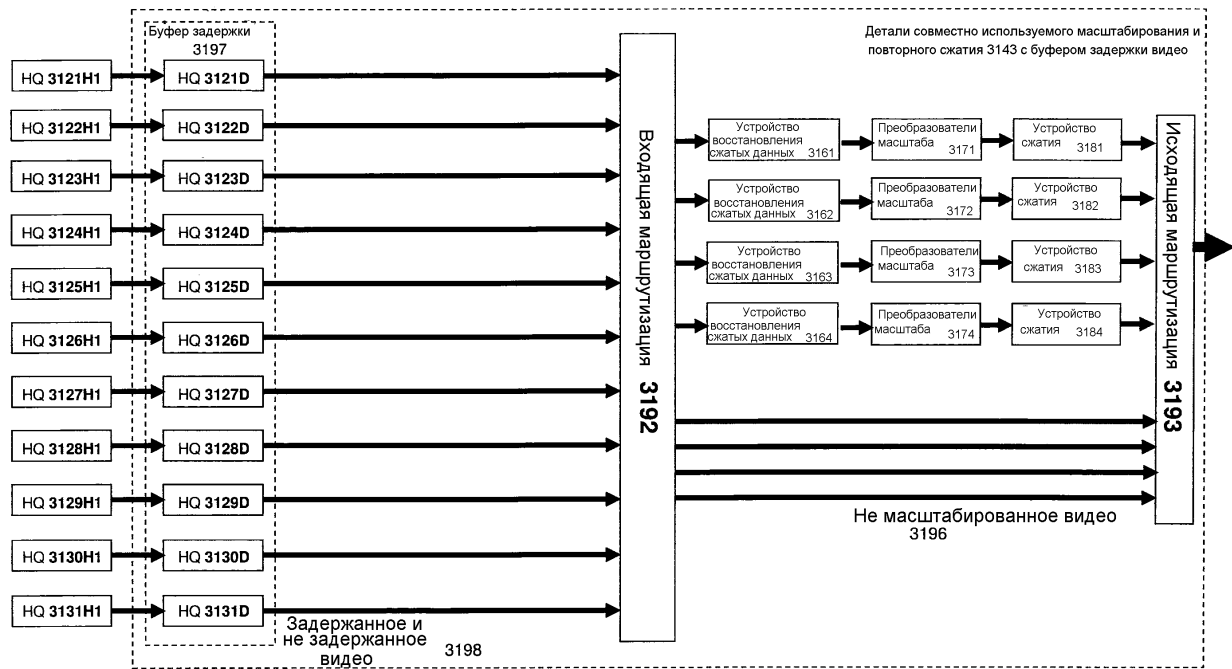
ФИГ.31e



ФИГ.31f



ФИГ.31g



ФИГ.31h