РОССИЙСКАЯ ФЕДЕРАЦИЯ

(19) **RU** (11) **2 340 932** (13) **C2**

(51) МПК
G06F 9/06 (2006.01)
G06F 15/16 (2006.01)

ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ,
ПАТЕНТАМ И ТОВАРНЫМ ЗНАКАМ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21), (22) Заявка: **2004105877/09, 27.02.2004**

(24) Дата начала отсчета срока действия патента:
27.02.2004

(30) Конвенционный приоритет:
28.02.2003 US 60/451,151

(43) Дата публикации заявки: **10.08.2005**

(45) Опубликовано: **10.12.2008 Бюл. № 34**

(56) Список документов, цитированных в отчете о поиске: **US 2002/0174219 A1, 21.11.2002. US 2003/0018784 A1, 23.01.2003. RU 2134931 C1, 20.08.1999. RU 2141131 C1, 10.11.1999.**

Адрес для переписки:
129090, Москва, ул. Б.Спасская, 25, стр.3, ООО "Юридическая фирма Городисский и Партнеры", пат.пов. Ю.Д.Кузнецову, рег.№ 595

(72) Автор(ы):
МАРКАРЯН Марк (US),
КАКУРИН Дмитрий (US),
ОЛЬСОН Шон К. (US),
ШОРОФФ Срикант (US),
ИОНЕСКУ Раду (US)

(73) Патентообладатель(и):
МАЙКРОСОФТ КОРПОРЕЙШН (US)

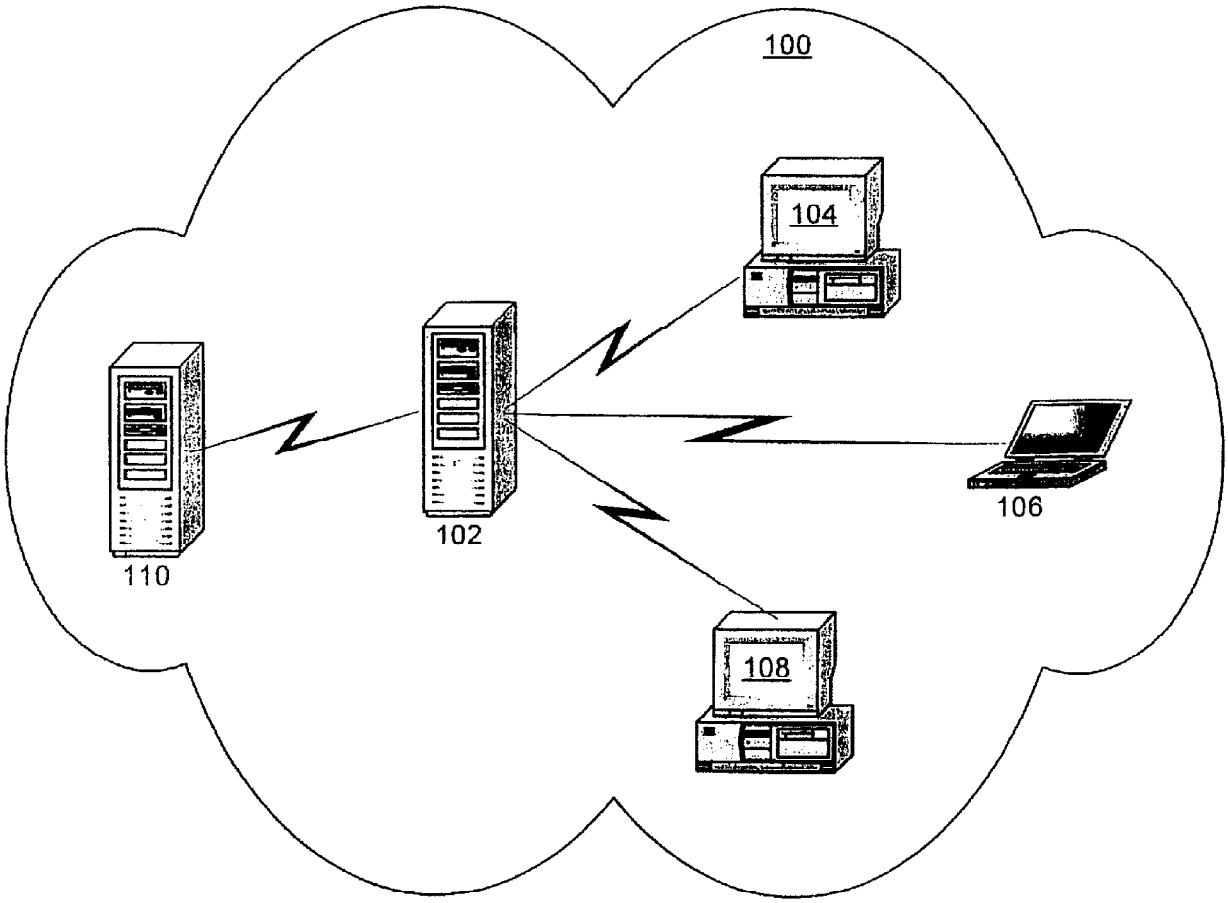
(54) СПОСОБ И СИСТЕМА ДЛЯ ОТСРОЧЕННОГО ВЫДЕЛЕНИЯ РЕСУРСОВ

(57) Реферат:

Настоящее изобретение относится к вычислительной технике. Техническим результатом является обеспечение серверу возможности отложить выделение ресурсов по запросу клиента. Когда клиент запрашивает признак, которому необходимы ресурсы сервера, сервер принимает запрос клиента и подтверждает его прием, но клиенту запрещают использовать запрашиваемый признак до следующего указания от сервера. Например, во время процесса авторизации сервер выделяет только минимальные ресурсы, необходимые для поддержания сеанса и авторизации клиента. После этого сервер выделяет ресурсы, необходимые для обеспечения запроса клиента, только тогда, когда ресурсы становятся доступными. До этих пор сервер поддерживает сеанс связи, не обеспечивая запрос. Таким образом, сервер следит за своими ресурсами вместо того, чтобы передавать их по желанию клиента. Также нет необходимости клиенту повторять свой запрос, если сервер не может немедленно удовлетворить его, вместо этого сервер принимает запрос и затем позже начинает обеспечивать его, когда соответствующие ресурсы становятся доступными. 6 н. и 34 з.п. ф-лы, 7 ил., 1 табл.

RU 2340932 C2

RU 2340932 C2



Фиг. 1

RU 2340932 C2

RU 2340932 C2

RUSSIAN FEDERATION

FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY,
PATENTS AND TRADEMARKS

(19) **RU** (11) **2 340 932** (13) **C2**

(51) Int. Cl.
G06F 9/06 (2006.01)
G06F 15/16 (2006.01)

(12) ABSTRACT OF INVENTION

(21), (22) Application: **2004105877/09, 27.02.2004**

(24) Effective date for property rights: **27.02.2004**

(30) Priority:
28.02.2003 US 60/451,151

(43) Application published: **10.08.2005**

(45) Date of publication: **10.12.2008 Bull. 34**

Mail address:
129090, Moskva, ul. B.Spasskaja, 25, str.3, OOO "Juridicheskaja firma Gorodisskij i Partnery", pat.pov. Ju.D.Kuznetsovu, reg.№ 595

(72) Inventor(s):
MARKARJaN Mark (US),
KAKURIN Dmitrij (US),
OL'SON Shon K. (US),
ShOROFF Srikant (US),
IONESKU Radu (US)

(73) Proprietor(s):
MAJKROSOFT KORPOREJShN (US)

(54) DELAYED ALLOCATION METHOD AND RELATED SYSTEM

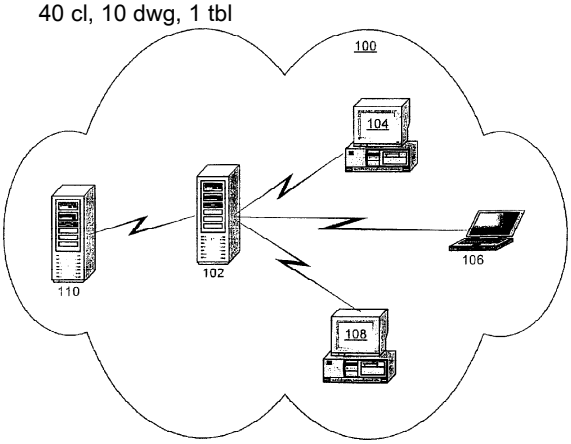
(57) Abstract:

FIELD: information technologies.

SUBSTANCE: when the client requests attribute requiring server resources, the server receives client request and confirms the reception, but the client is disabled to use required attribute till the next server guide. E.g. during authorisation the server allocates minimum resources only required to maintain session and client authorisation. Thereafter the server allocates resources required to maintain client request, as soon as resources are available. Heretofore the server maintains communication session without ensuring request. Thus the server follows the resources instead of transmitting them respecting client interests. Besides it is not required to reapply client request if the server cannot solve it immediately, instead the server receives request and then starts to ensure it later as soon as related resources are available.

EFFECT: server is optional to maintenance to delay allocation by client request.

40 cl, 10 dwg, 1 tbl



Фиг. 1

RU 2340932 C2

RU 2340932 C2

Родственная заявка

По настоящей заявке испрашивается приоритет предварительной патентной заявки США 60/451151, поданной 28 февраля 2003, которая полностью включена в материалы заявки посредством ссылки.

Область техники, к которой относится изобретение

Изобретение относится в общем случае к обмену информацией по сети и, более конкретно, к распределению ресурсов среди клиентов и серверов в сети.

Предшествующий уровень техники

Быстрый рост компьютерных сетей, и общественных, и частных, в последние годы в значительной степени стимулировался «вычислениями клиент-сервер». В такой модели одно вычислительное устройство, клиент, запрашивает, чтобы другое вычислительное устройство, сервер, представило ему услуги или функциональные возможности. Следует обратить внимание, что «клиент» и «сервер» используются исключительно для обозначения сторон в транзакции запроса. Хотя некоторые вычислительные устройства реализованы как выделенные серверы, которые могут обслуживать многочисленных клиентов, клиент и сервер могут менять роли от одной транзакции к другой. В «одноранговой» сети (обычной, например, среди устройств, поддерживающих связь с помощью радиосвязи ближнего действия) каждое вычислительное устройство имеет возможность быть и клиентом, и сервером последовательно или одновременно.

Серверы часто должны выделять ценные ресурсы для выполнения запроса на функциональную возможность или услугу. После приема запроса от клиента сервер проверяет доступность своих ресурсов. Традиционно, если сервер не имеет ресурсов для выполнения запроса, то сервер отклоняет запрос. Если клиент может продолжать работу без запрашиваемой функциональной возможности или услуги, то он это делает и повторно представляет запрос позднее, когда сервер, возможно, будет иметь в наличии необходимые ресурсы для выполнения запроса.

Чтобы гарантировать, что ценные ресурсы сервера будут выделены только тем клиентам, которые авторизованы их использовать, серверы часто выполняют процедуру аутентификации (проверки подлинности) клиента, делающего запрос. Если клиент не может пройти процедуру аутентификации, чтобы удовлетворить сервер, то сервер отклоняет запрос.

Однако эта защита от неавторизованных клиентов не совершенна. Некоторые типы запросов выполняют прежде, чем процесс авторизации закончен. Обработка этих запросов, даже если они в конечном счете будут отклонены, потребляет некоторый уровень ресурсов сервера. Например, недобросовестный клиент может выполнить атаку на сервер типа «отказ в обслуживании» (OBO, DOS), неоднократно делая запросы к серверу. Хотя этот клиент будет не в состоянии пройти процедуру аутентификации, и его запросы будут в конечном счете отклонены, сервер может в это время использовать так много ресурсов, пытаясь выполнить процедуру аутентификации клиента во время каждого запроса, что сервер исчерпает совокупность своих ресурсов до такого состояния, что сервер окажется неспособным выполнять любые запросы, даже сделанные авторизованными клиентами.

Сущность изобретения

Ввиду вышеизложенного настоящее изобретение позволяет серверу откладывать выделение ресурсов по запросу клиента. Когда клиент запрашивает услугу или функциональную возможность, которые требуют ресурсы сервера (такие, например, как шифрование или сжатие сообщений между клиентом и сервером), сервер принимает запрос клиента и подтверждает его прием, но клиенту запрещают использовать запрашиваемую функциональную возможность до следующего извещения от сервера. Например, во время процесса авторизации сервер выделяет только минимальные ресурсы, требуемые для поддержания сеанса и авторизации клиента. После этого сервер выделяет ресурсы, необходимые для обеспечения запроса клиента, только тогда, когда эти ресурсы становятся доступными. До этих пор сервер поддерживает сеанс связи, не обеспечивая запрос. Таким образом, сервер следит за своими ресурсами вместо того, чтобы

передавать их по желанию возможно злонамеренного, неправильно работающего или неправильно сконфигурированного клиента. Кроме того, для легального клиента не требуется повторять свой запрос, если сервер не может немедленно удовлетворить его; вместо этого сервер принимает запрос и затем позднее начинает обеспечивать его, когда соответствующие ресурсы становятся доступными.

Согласно одному из вариантов осуществления после приема от клиента запроса на сжатие данных сервер принимает запрос и подтверждает его прием, но откладывает выделение ресурсов, необходимых для сжатия передаваемых данных. На самом деле сервер может даже не выяснять, доступны ли ресурсы, пока клиент успешно не выполнит процедуру аутентификации для сервера. Даже при том, что запрос на сжатие был принят, клиент и сервер осуществляют обмен данными, не сжимая свои данные. Это продолжается до тех пор, пока и в том случае, если ресурсы, необходимые для сжатия, становятся доступными серверу. Тогда сервер выделяет необходимые ресурсы и указывает клиенту, что сжатие теперь обеспечивается. Сервер может сигнализировать об этом, например, посылая клиенту сжатые данные. После приема такого сигнала (например, приема сжатых данных) клиент понимает, что теперь разрешено осуществлять передачу данных со сжатием. Клиент отвечает, начиная передавать сжатые данные на сервер.

Сжатие - только один из примеров функциональной возможности обмена данными, который может запросить клиент. Другие примеры включают в себя широкий диапазон функциональных возможностей, обычно называемых «качество обслуживания» (QOS). Функциональные возможности QOS соответствуют, в общем случае, пропускной способности, гарантированному времени ответа, устойчивости к ошибкам, целостности последовательности сообщений и отсутствию дублирования, максимально допустимому уровню потерь и т.п. Функциональные возможности QOS обеспечивают примеры, в которых согласно одному из вариантов осуществления настоящего изобретения сервер может выделять ресурсы уровень за уровнем, а не все за один раз. Например, клиент запрашивает большую гарантированную пропускную способность. Сервер сначала принимает запрос, но выделяет ресурсы, достаточные для обеспечения только небольшой гарантированной пропускной способности. Клиент распознает это и использует только небольшую пропускную способность. Позже сервер выделяет большую пропускную способность этому клиенту (в ответ, например, на освобождение другим клиентом пропускной способности), и клиент начинает использовать большую пропускную способность.

Также согласно изобретению сервер или клиент (или оба) поддерживают информацию о запрашиваемой функциональной возможности и о фактическом уровне обеспечиваемой услуги. Сервер отслеживает эту информацию для каждого клиента и выделяет дополнительные ресурсы клиентам, когда эти ресурсы становятся доступными, чтобы более полно обеспечить запросы клиентов.

Клиент может отображать своему пользователю состояния запросов, такие как «принят и обеспечивается», «принят, но еще не обеспечивается» и «отклонен». Сервер может обеспечивать передачу подобной информации администратору или в файл журнала регистрации.

Перечень фигур чертежей

Хотя прилагаемая формула изобретения конкретно формулирует признаки настоящего изобретения, данное изобретение вместе с его задачами и преимуществами может быть лучше всего понято из последующего подробного описания, рассматриваемого вместе с сопроводительными чертежами, на которых:

фиг.1 - структурная схема примерной компьютерной сетевой среды, в которой может осуществляться настоящее изобретение;

фиг.2 - схематическое представление, в общем случае показывающее примерную компьютерную систему, которая поддерживает настоящее изобретение;

фиг.3а и 3В вместе формируют последовательность операций, показывающую примерный обмен сообщениями между клиентом и сервером во время согласования

запроса клиента на функциональную возможность (ФВ) обмена данными;

фиг.4 - схема структуры данных примерного сообщения, которым обмениваются между собой клиент и сервер во время выполнения сценария, показанного на фиг.3a и 3b;

фиг.5a и 5b вместе формируют последовательность операций, иллюстрирующую примерный способ выделения ресурсов, выполняемый сервером;

фиг.6 - схема структуры данных списка состояния запросов, используемого сервером;

фиг.7a и 7b вместе формируют последовательность операций, показывающую примерный способ выполнения запроса функциональной возможности, выполняемого клиентом.

Подробное описание изобретения

На чертежах, на которых подобные ссылочные номера относятся к подобным элементам, настоящее изобретение показано как реализуемое в соответствующей вычислительной среде. Последующее описание основано на вариантах осуществления изобретения, и оно не должно использоваться в качестве ограничения изобретения по отношению к альтернативным вариантам осуществления, которые явно здесь не описаны.

В последующем описании настоящее изобретение описано по отношению к действиям и символическим представлениям операций, которые выполняются одним или более вычислительными устройствами, если не обозначено иначе. Также следует понимать, что такие действия и операции, которые время от времени упоминаются как выполняемые компьютером, включают в себя обработку процессором вычислительного устройства электрических сигналов, представляющих данные в структурированной форме. Эта обработка преобразовывает данные или сохраняет их в ячейках в системе памяти вычислительного устройства, которое реконфигурирует или иным образом изменяет работу устройства способом, хорошо понятным специалистам в данной области техники. Структуры данных, где хранятся данные, являются физическими ячейками памяти, которые имеют определенные свойства, которые определяются форматом данных. Однако хотя изобретение описывается в указанном контексте, не подразумевается, что оно ограничено им, поскольку специалисты в данной области техники должны признать, что различные описанные в дальнейшем действия и операции могут быть также реализованы в аппаратных средствах.

Настоящее изобретение позволяет серверу принимать запрос клиента, но откладывать выделение ресурсов, необходимых для обслуживания этого запроса. Фиг.1 показывает пример компьютерной сетевой среды 100, в которой изобретение может использоваться. Примерная сеть 100 включает в себя вычислительное устройство-сервер 102 и три вычислительных устройства-клиента 104, 106 и 108. Сеть 100 может быть корпоративной локальной сетью (ЛС, LAN), беспроводной сетью, Интернет или занимать промежуточное положение между ними, и она может включать в себя множество известных компонентов, таких как маршрутизаторы, шлюзы, концентраторы и т.д. В примерной транзакции клиент 104 запрашивает услугу или функциональную возможность обмена данными у сервера 102. Сервер 102 предварительно принимает запрос, но не выделяет ресурсы для обеспечения запрашиваемой функциональной возможности до того, например, пока клиент 104 не выполнит процедуру аутентификации для сервера 102, или пока ресурсы не станут доступными. Пока ресурсы не будут выделены и пока сервер 102 не сообщит клиенту 104 об этом факте, клиент 104 и сервер 102 осуществляют обмен данными, не используя запрашиваемую функциональную возможность. Таким образом, сервер 102 следит за своими ресурсами вместо того, чтобы передавать их по желанию потенциально злонамеренного, неправильно работающего или неправильно сконфигурированного клиента.

В другой транзакции клиент 104 и сервер 102 могут меняться ролями, когда «сервер» 102 запрашивает услугу у «клиента» 104. В одноранговой сети каждое вычислительное устройство может быть и клиентом, и сервером последовательно или одновременно. Соответственно, варианты осуществления изобретения могут применяться в клиентах, серверах, одноранговых объектах сети или любых их комбинациях.

Вычислительное устройство 110 является другим сервером, но таким, которое непосредственно осуществляет связь только с сервером 102 для предоставления ему ресурсов. Его присутствие иллюстрирует, что, следуя способу настоящего изобретения, сервер 102 следит не только за своими собственными ресурсами, но и за ресурсами сетевой среды 100 вообще.

Вычислительные устройства 102 и 104 по фиг.1 могут иметь любую архитектуру. Фиг.2 - структурная схема, в общем случае иллюстрирующая примерную компьютерную систему, которая поддерживает настоящее изобретение. Компьютерная система по фиг.2 - только один из примеров соответствующей среды, и она не может использоваться в качестве какого-либо ограничения объема использования или функций изобретения. И при этом вычислительное устройство 102 не должно интерпретироваться как имеющее какие-либо зависимости или требования, касающиеся любого компонента или их комбинации, показанных на фиг.2. Изобретение может работать с многочисленными другими вычислительными средами или конфигурациями, универсальными или специального назначения. Примеры известных вычислительных систем, сред и конфигураций, которые могут использоваться с изобретением, включают в себя персональные компьютеры, серверы, карманные или портативные компьютеры, планшетные компьютеры, многопроцессорные системы, системы на основе микропроцессора, абонентские устройства, программируемую бытовую электронику, сетевые персональные компьютеры, миникомпьютеры, универсальные компьютеры и распределенные вычислительные среды, которые включают в себя любое из вышеупомянутых систем или устройств, но не ограничены ими. В основной конфигурации вычислительное устройство 102 обычно включает в себя по меньшей мере один процессор 200 и память 202. Память 202 может быть энергозависимой (такой как оперативная память (ОП, RAM)), энергонезависимой (такой как постоянное запоминающее устройство (ПЗУ, ROM) или флэш-память) или некоторой их комбинацией. Эта основная конфигурация показана на фиг.2 пунктирной линией 204. Вычислительное устройство 102 может иметь дополнительные характеристики и функциональные возможности. Например, устройство 102 может содержать дополнительное запоминающее устройство (съемное и несъемное), которое включает в себя магнитные и оптические диски и ленту, но не ограничено ими. Такое дополнительное запоминающее устройство проиллюстрировано на фиг.2 съемным запоминающим устройством 206 и несъемным запоминающим устройством 208. Компьютерные носители данных включают в себя энергозависимые и энергонезависимые, съемные и несъемные носители, воплощенные любым способом или технологией для хранения информации, такой как машиночитаемые команды, структуры данных, модули программ или другие данные. Память 202, съемное запоминающее устройство 206 и несъемное запоминающее устройство 208 являются примерами компьютерных носителей данных. Компьютерные носители данных включают в себя ОП, ПЗУ, электрически стираемое программируемое постоянное запоминающее устройство (ЭСППЗУ, EEPROM), флэш-память, память другой технологии, ПЗУ на компакт-диске (CD-ROM), цифровые универсальные диски, другие оптические запоминающие устройства, магнитные кассеты, магнитную ленту, запоминающее устройство на магнитном диске, другие магнитные запоминающие устройства и любые другие носители, которые могут использоваться для хранения необходимой информации, и к которым вычислительное устройство 102 может осуществить доступ, но не ограничены перечисленным. Устройство 102 может также содержать каналы 210 связи, которые позволяют компьютеру связываться с другими устройствами. Каналы 210 связи являются примерами сред передачи информации. Среды передачи информации обычно воплощают машиночитаемые команды, структуры данных, модули программ или другие данные в сигнале, модулированном данными, таком как несущая, или используют другой механизм транспортировки и включают в себя любые среды передачи информации. Термин «сигнал, модулированный данными» означает сигнал, который имеет одну или более из своих характеристик, которые устанавливаются или изменяются таким образом, чтобы кодировать информацию в сигнале. Для примера, а

не в качестве ограничения, среды передачи информации включают в себя проводные среды, такие как проводные сети или прямое проводное подключение, и беспроводные среды, такие как акустические, радиочастотные (РЧ, RF), инфракрасные и другие беспроводные среды передачи информации. Термин «машиночитаемые носители» включает в себя и носители данных, и среды передачи информации. Вычислительное устройство 102 может также иметь устройства 212 ввода данных, такие как клавиатура, мышь, перо, устройство речевого ввода, графический планшет, сенсорное устройство ввода данных и т.д. Оно также может включать в себя устройства 214 вывода, такие как дисплей (который может быть интегрирован с сенсорным устройством ввода данных), динамики и принтер. Все эти устройства известны из уровня техники, и их подробное обсуждение не требуется.

Фиг.3a и 3b совместно показывают примерный обмен сообщениями, когда клиент 104 запрашивает функциональную возможность у сервера 102. Фиг.5a, 5b, 7a и 7b, описываемые ниже, представляют дополнительные подробности возможного обмена сообщениями. На этапе 300 по фиг.3a клиент 104 запрашивает функциональную возможность. Функциональная возможность может быть любого типа, она соответствует сжатию данных, шифрованию данных и многочисленным функциональным возможностям QOS. Протокол сообщений может также быть любого типа, такой как, например, SIP (протокол инициирования сеанса). Следует обратить внимание, что запрос функциональной возможности на этапе 300 не обязательно должен быть явным: вместо этого он может подразумеваться в соответствии с протоколом сообщений, используемым между клиентом 104 и сервером 102.

На этапе 302 сервер 102 принимает запрос функциональной возможности и решает, будет ли он обеспечивать эту функциональную возможность. Если нет, то сервер 102 использует способы, определенные в протоколе, для отклонения запроса (не показан). Если сервер 102 собирается обеспечивать запрашиваемую функциональную возможность и готов сделать это немедленно, то сервер 102 выделяет ресурсы, требующиеся для обеспечения этой функциональной возможности, и принимает запрос (также не показан). Сценарий, изображенный на фиг.3a и 3b, касается третьей возможности для сервера 102: он может намереваться обеспечить запрашиваемую функциональную возможность в будущем, но еще не готов сделать это. Одним из примеров, который приводит к такому сценарию, является случай, когда сервер 102 в текущий момент не имеет в наличии ресурсы для обеспечения функциональной возможности, но ожидает скоро получить эти ресурсы. В другом примере сервер 102 еще не доверяет клиенту 104 в той степени, чтобы выделять ценные ресурсы по его запросу. Сервер 102 пока не выделяет ресурсы, а ждет, когда клиент 104 успешно пройдет процедуру аутентификации. (См. обсуждение этапов 310 и 312 по фиг.3b ниже.) В сценарии по фиг.3a сервер 102 на этапе 302 посылает сообщение клиенту 104, указывая, что запрос был принят, но также указывая, что запрашиваемая функциональная возможность еще не обеспечивается.

Существует множество способов, которыми сервер 102 может указать, что запрашиваемая функциональная возможность еще не обеспечивается. В SIP, например, когда сжатие данных разрешено в линии связи, к полям данных добавляются метки. (См. фиг.4 и сопровождающее ее обсуждение.) Не все информационные сообщения сжимаются, даже когда сжатие допускается (например, данное сообщение может быть слишком коротким, чтобы получить преимущество от сжатия), так что флажок в метке указывает, сжаты ли сопровождаемые данные. Варианты осуществления настоящего изобретения могут использовать эту метку и флажок на этапе 302: сообщение о приеме помечают (добавляют метку), указывая, что запрос на сжатие данных удовлетворен, но данные в этом сообщении не сжаты, что указано флажком. На этапе 304 клиент 104 принимает сообщение о приеме и отмечает, что запрашиваемая функциональная возможность еще не обеспечивается. В примере сжатия данных метка указывает на то, что запрос принят, но отсутствие сжатия указывает на то, что сервер не готов к сжатию данных.

На этапах 306 и 308 клиент 104 и сервер 102 осуществляют обмен данными, не

используя запрашиваемую функциональную возможность. В зависимости от обстоятельств эти этапы могут продолжаться в течение длительного времени (до тех пор, пока, например, сервер 102 не получит необходимые ресурсы) или может быть очень короток (например, только до тех пор, пока клиент 104 успешно не пройдет процедуру аутентификации для сервера 102).

Этапы 310 и 312 на фиг.3b в некотором смысле являются необязательными, но способ включает их в себя, потому что они иллюстрируют сценарий, в котором способы настоящего изобретения являются очень полезными. В течение этих этапов клиент 104 проходит процедуру аутентификации для сервера 102, используя способы, установленные в соответствии с протоколом, который они используют. (Множество таких способов известно в уровне техники.) Понятно, что сервер 102 отказывается выделять ценные ресурсы, пока эти этапы не закончены. Хотя этот сценарий не является единственным, в котором отсроченное выделение ресурсов оказывается полезным, он является одним из сценариев, тесно связанным с предотвращением атак «отказ в обслуживании» (DOS).

Наконец, на этапе 314 сервер 102 решает выделить ресурсы для обеспечения запроса клиента 104. На этапе 316 сервер 102 указывает клиенту 104, что функциональная возможность теперь обеспечивается. Так же, как и с многочисленными возможными указаниями, обсуждаемыми выше относительно этапа 302, существует множество способов, которыми сервер 102 может указывать, что функциональная возможность теперь обеспечивается. Используя пример сжатия данных, сервер 102 может просто послать сжатые данные клиенту 104. После приема указания независимо от того, каким оно будет, клиент 104 на этапе 318 отмечает, что функциональная возможность теперь обеспечивается. С этого момента клиент 104 и сервер 102 могут осуществлять обмен данными или с использованием, или без использования запрашиваемой функциональной возможности, в зависимости от ситуации.

Фиг.4 показывает структуру 400 данных сообщения, используемого для того, чтобы послать сжатые или несжатые данные. Структура 400 данных включает в себя три поля меток. Первое поле 402 метки предназначено для флажков («метка флажков»). Поле 402 метки флажков используют для указания формата данных в поле 408, а именно сжаты ли данные. В данном варианте реализации поле флажков включает в себя взаимоисключающие биты. Например, бит 0×80 используется для указания, что данные не являются сжатыми, а бит 0×20 указывает, что данные являются сжатыми.

В некоторых вариантах осуществления существует по меньшей мере три типа пакетов данных: (1) непомеченные данные, указывающие, что сжатие данных не доступно для текущего соединения; (2) помеченные данные, указывающие, что сжатие возможно, но флажок для данных в поле 408 установлен, как для несжатых; и (3) помеченные данные, указывающие, что сжатие возможно, и данные в поле 408 сжаты. На этапах 304 и 318 на фиг.3a и 3b соответственно клиент 104 определяет тип пакета данных, который он принимает от сервера 102, чтобы знать, обеспечивается или нет сжатие данных.

На фиг.5a и 5b показана последовательность операций, иллюстрирующая примерные этапы, выполняемые сервером 102. На этапе 500 сервер 102 принимает от клиента 104 запрос на услугу или функциональную возможность передачи данных. Как указано выше, этот запрос может быть в форме явного сообщения, посланного клиентом 104, или он может неявно указываться в протоколе передачи данных, используемом между клиентом 104 и сервером 102. На этапе 502 сервер 102 проверяет свою собственную конфигурацию, чтобы определить, может ли он обеспечить запрашиваемую функциональную возможность. Может случиться, что клиент 104 запрашивает функциональную возможность, которую сервер 102 не может обеспечить. В этом случае способ продолжается на этапе 510, где сервер 102 отклоняет запрос.

Если сервер 102 может по меньшей мере теоретически обеспечить запрашиваемую функциональную возможность, то на этапе 504 он принимает запрос, но сообщает клиенту 104, что клиент 104 не может пока использовать эту функциональную возможность.

Существуют некоторые функциональные возможности, которые сервер 102

обеспечивает только клиентам, прошедшим процедуру аутентификации. Если клиент 104 затребовал такую функциональную возможность, то на этапе 506 выполняют процесс аутентификации. Если на этапе 508 клиент 104 не проходит процедуру аутентификации, то на этапе 510 сервер 102 может отклонить запрос даже при том, что он предварительно принял запрос ранее на этапе 504. Следует обратить внимание, что неспособность пройти процедуру аутентификации не обязательно подразумевает, что клиент 104 должен закончить свой сеанс связи с сервером 102. Хотя это является возможным результатом, для настоящего обсуждения последствием неспособности пройти процедуру аутентификации является невозможность клиента 104 использовать запрашиваемую функциональную возможность.

Если клиент 104 успешно проходит процедуру аутентификации для сервера 102 (или если такая аутентификация не является необходимой), то клиент 104 и сервер 102 начинают обмениваться данными друг с другом, но не используя запрашиваемую функциональную возможность. В случае необходимости на этапе 512 сервер 102 проверяет доступность достаточного количества ресурсов, и когда на этапе 514 по фиг.5b такие ресурсы становятся доступными, сервер 102 выделяет их для обеспечения функциональной возможности, которую запрашивает клиент 104. Как упомянуто выше относительно фиг.1, эти ресурсы не обязательно должны находиться непосредственно на сервере 102. Они могут быть предоставлены другим сервером 110. В некоторых сценариях ресурсы могут стать доступными на этапе 514, когда другой клиент освобождает их. В других сценариях ресурсы всегда доступны, но сервер 102 отказывается передавать их клиенту 104, пока клиент 104 успешно не выполнит процедуру аутентификации на этапе 508 по фиг.5a.

На этапе 516 по фиг.5b сервер 102 указывает, что теперь готов обеспечить запрашиваемую функциональную возможность. Некоторые функциональные возможности могут обеспечиваться на различных уровнях. Например, клиент 104 запрашивает минимальную гарантированную пропускную способность 512 кбит/с.Если сервер 102 не имеет ресурсов, чтобы полностью обеспечить этот запрос, он может просто отклонить его. Альтернативно, сервер 102 может принять запрос, но сообщить клиенту 104, что сервер 102 может обеспечить только гарантированную пропускную способность 128 кбит/с. Клиент 104 решает, приемлемо ли более низкое гарантированное значение или нет, и реагирует соответственно.

По всей этой процедуре сервер 102 отслеживает уровни и выделение своих ресурсов, как обозначено на этапе 518. Сервер 102 использует эту информацию при принятии решения, имеет ли он достаточно ресурсов для обеспечения запрашиваемой функциональной возможности. Системные администраторы используют эту информацию при принятии решения относительно того, сконфигурирован ли сервер 102 оптимально.

На фиг.6 приведен пример журнала регистрации ресурсов сервера 102. Журнал 600 регистрации распределения ресурсов содержит четыре строки записей, каждая из которых относится к одному запросу функциональной возможности. В журнале 600 регистрации отмечено, что клиент 104 (поле 602) запросил сжатие данных (поле 604), и этот запрос был принят (поле 606). Запрос клиента 106 о сжатии данных был отклонен возможно потому, что клиент 106 не смог пройти процедуру аутентификации для сервера 102. Запрос клиента 108 на сжатие данных был предварительно принят, но эта функциональная возможность еще не обеспечивается, клиент 108 сделал другой запрос на этот раз на гарантированную пропускную способность 512 кбит/с. Запрос был принят, но функциональная возможность в настоящее время обеспечивается только на более низком уровне 128 кбит/с.

На этапе 520 по фиг.5b клиент 104 и сервер 102 могут использовать запрашиваемую функциональную возможность при обмене данными между собой. Однако они не обязаны использовать эту функциональную возможность. Например, даже когда сжатие обеспечивается, некоторые сообщения слишком коротки, чтобы получить преимущество от сжатия.

Другое использование журнала 600 регистрации выделения ресурсов сервера 102 показано на этапе 522. На нем освобождаются некоторые ресурсы (вероятно, другим клиентом), и сервер 102 проверяет журнал 600 регистрации распределения своих ресурсов. Он обращает внимание, например, на то, что клиент 108 запрашивал гарантированную пропускную способность 512 кбит/с, но ему было предоставлено только 128 кбит/с. Если сервер 102 может и хочет обеспечить запрос клиента 108 на более высоком уровне, то теперь он может сделать это. Для некоторых средств сервер 102 может даже использовать этот способ для уменьшения обеспечиваемого уровня. Другие функциональные возможности этого не предусматривают, и обеспечиваемый уровень нужно повторно согласовывать.

Сторона клиента 104 в транзакции запроса функциональной возможности проиллюстрирована в последовательности операций по фиг.7a и 7b. Поскольку большая часть процедуры для клиента 104 очевидна в свете приведенного выше обсуждения процедуры для сервера 102, необходимо обсудить только несколько аспектов. Клиент 104 может поддерживать журнал регистрации своих собственных запросов, подобный журналу 600 регистрации выделения ресурсов сервера 102 по фиг.6. Состояние запросов функциональной возможности, которое включает в себя обеспечиваемый уровень, если он установлен, может отображаться пользователю клиента 104, как указано на этапах 716 и 720 по фиг.7b.

Приведенное выше обсуждение сосредотачивается на ожидаемом порядке обмена между сервером 102 и клиентом 104. Следующая таблица показывает некоторые из неожиданностей, которые могут произойти, и как клиент 104 должен реагировать.

Возможные ответы, которые клиент должен быть готов обрабатывать при запросе функциональной возможности		
Ответ	Значение	Соответствующая обработка
• Превышение лимита времени транзакции. • Недопустимый ответ. • Ответ без указания запрашиваемой функциональной возможности. • Ответ с недопустимым указанием запрашиваемой функциональной возможности.	Согласование завершилось неудачно.	Вернуться к неиспользованию запрашиваемой функциональной возможности в данной линии связи.
400	Сервер не поддерживает способ согласования в этот момент времени или не в состоянии распознать способ как допустимый.	Вернуться к неиспользованию запрашиваемой функциональной возможности в данной линии связи.
405, 501	Сервер не поддерживает способ согласования.	Вернуться к неиспользованию запрашиваемой функциональной возможности в данной линии связи.
488, 606	Сервер не обеспечивает запрашиваемую функциональную возможность.	Вернуться к неиспользованию запрашиваемой функциональной возможности в данной линии связи.

403	Сервер отклоняет запрос	Закрыть соединение. Открыть новое соединение и не запрашивать данную функциональную возможность. Не использовать запрашиваемую функциональную возможность в этой линии связи.
408, 480, 504	Превышение лимита времени.	Повторить после соответствующей задержки. Многочисленные превышения лимита времени должны привести к закрытию соединения и возникновению соответствующего сигнала тревоги. Это указывает на потерю связи с сервером.
1xx	Предварительный ответ	Игнорировать.
2xx	Успешное завершение	Разрешить запрашиваемую функциональную возможность для данной линии связи.
3xx	Сервер переадресовывает запрос.	Игнорировать. Вернуться к неиспользованию запрашиваемой функциональной возможности в данной линии связи.
4xx, 5xx, 6xx	Другие ошибки	Игнорировать. Вернуться к неиспользованию запрашиваемой функциональной возможности в данной линии связи.

Ввиду многих возможных вариантов осуществления, к которым могут применяться принципы настоящего изобретения, следует признать, что описанные относительно фигур чертежей варианты осуществления предназначены только для иллюстративных целей и не должны рассматриваться как ограничение объема изобретения. Например, специалисты в данной области техники должны признать, что показанные варианты осуществления могут быть изменены в плане компоновки и деталей без выхода за рамки объема изобретения. Хотя изобретение описано в терминах программных модулей или компонентов, специалисты в данной области техники должны признать, что они могут быть эквивалентно заменены аппаратными компонентами. Поэтому описанное здесь изобретение рассматривает все такие варианты осуществления, как находящиеся в пределах объема, определяемого следующей формулой изобретения и ее эквивалентами.

Формула изобретения

1. Реализуемый в среде передачи данных, содержащей вычислительное устройство-сервер и вычислительное устройство-клиент, способ отсроченного обеспечения сервером функциональной возможности для клиента, содержащий этапы, на которых принимают запрос функциональной возможности, указывают клиенту, что запрос удовлетворен, указывают клиенту, что запрашиваемая функциональная возможность еще не обеспечивается, после указания клиенту, что запрашиваемая функциональная возможность еще не обеспечивается, указывают клиенту, что запрашиваемая функциональная возможность обеспечивается, и обеспечивают запрашиваемую функциональную возможность для клиента.

2. Способ по п.1, в котором запрашиваемая функциональная возможность является

функциональной возможностью обмена данными между клиентом и сервером, выбранной из группы, состоящей из сжатия, качества обслуживания, пропускной способности, гарантированного времени ответа, устойчивости к ошибкам, целостности последовательности сообщений и отсутствия дублирования и максимально допустимого уровня потерь.

3. Способ по п.1, в котором прием запроса функциональной возможности содержит этап, на котором принимают от клиента явный запрос функциональной возможности.

4. Способ по п.1, в котором прием запроса функциональной возможности содержит этап, на котором принимают от клиента неявный запрос функциональной возможности.

5. Способ по п.4, в котором неявный запрос функциональной возможности определяют в соответствии с протоколом передачи данных, используемым клиентом и сервером.

6. Способ по п.1, в котором прием запроса функциональной возможности содержит этап, на котором принимают запрос о первом уровне обеспечения запрашиваемой функциональной возможности, и в котором указание клиенту, что запрашиваемая функциональная возможность обеспечивается, содержит этап, на котором указывают клиенту, что запрашиваемая функциональная возможность обеспечивается на втором уровне обеспечения запрашиваемой функциональной возможности.

7. Способ по п.6, в котором второй уровень обеспечения запрашиваемой функциональной возможности ниже первого уровня обеспечения запрашиваемой функциональной возможности.

8. Способ по п.6, в котором обеспечение запрашиваемой функциональной возможности для клиента содержит этап, на котором обеспечивают запрашиваемую функциональную возможность на втором уровне обеспечения запрашиваемой функциональной возможности.

9. Способ по п.6, дополнительно содержащий этап, на котором после указания клиенту, что запрашиваемая функциональная возможность обеспечивается на втором уровне обеспечения запрашиваемой функциональной возможности, указывают клиенту, что запрашиваемая функциональная возможность обеспечивается на первом уровне обеспечения запрашиваемой функциональной возможности.

10. Способ по п.1, в котором одно указание содержит указание клиенту, что запрос удовлетворен, и указание клиенту, что запрашиваемая функциональная возможность еще не обеспечивается.

11. Способ по п.1, в котором указание клиенту, что запрашиваемая функциональная возможность еще не обеспечивается, содержит этап, на котором не используют запрашиваемую функциональную возможность при обмене данными с клиентом.

12. Способ по п.1, в котором указание клиенту, что запрашиваемая функциональная возможность обеспечивается, содержит этап, на котором используют запрашиваемую функциональную возможность при обмене данными с клиентом.

13. Способ по п.1, в котором обеспечение запрашиваемой функциональной возможности для клиента содержит выделение ресурсов.

14. Способ по п.13, в котором выделение ресурсов содержит выделение ресурсов, предоставленных другим вычислительным устройством, отличным от сервера.

15. Способ по п.13, дополнительно содержащий этапы, на которых проверяют доступность ресурсов для обеспечения запрашиваемой функциональной возможности и откладывают выполнение указания клиенту, что запрашиваемая функциональная возможность обеспечивается, до тех пор, пока ресурсы для обеспечения запрашиваемой функциональной возможности не будут доступны.

16. Способ по п.15, дополнительно содержащий этапы, на которых поддерживают информацию об используемых ресурсах и запросах функциональных возможностей, которые еще не обеспечиваются.

17. Способ по п.16, в котором сервер поддерживает множество клиентов, и в котором ресурсы становятся доступными для обеспечения запрашиваемой функциональной возможности, когда они освобождаются другими клиентами.

18. Способ по п.1, дополнительно содержащий этапы, на которых перед указанием клиенту, что запрашиваемая функциональная возможность обеспечивается, выполняют аутентификацию клиента.

19. Способ по п.18, в котором указание клиенту, что запрашиваемая функциональная возможность обеспечивается, происходит во взаимосвязи с успешной аутентификацией клиента по отношению к серверу.

20. Способ по п.1, дополнительно содержащий этап, на котором регистрируют состояния запросов функциональной возможности, как «отклонен», «принят, но еще не обеспечивается» или «принят и обеспечивается».

21. Способ по п.20, дополнительно содержащий этапы, на которых регистрируют уровни обеспечения запросов функциональной возможности и регистрируют ресурсы, выделенные для обеспечения запросов функциональной возможности.

22. Машиночитаемый носитель, содержащий машиноисполняемые команды для выполнения способа для вычислительного устройства-сервера с целью отсрочки обеспечения функциональной возможности для вычислительного устройства-клиента, причем данный способ содержит этапы, на которых принимают запрос функциональной возможности, указывают клиенту, что запрос удовлетворен, указывают клиенту, что запрашиваемая функциональная возможность еще не обеспечивается, после указания клиенту, что запрашиваемая функциональная возможность еще не обеспечивается, указывают клиенту, что запрашиваемая функциональная возможность обеспечивается, и обеспечивают запрашиваемую функциональную возможность для клиента.

23. Реализуемый в среде передачи данных, содержащей вычислительное устройство-сервер и вычислительное устройство-клиент, способ получения клиентом обеспечиваемой сервером функциональной возможности, содержащий этапы, на которых запрашивают функциональную возможность у сервера, принимают указание, что запрос удовлетворен, принимают указание, что запрашиваемая функциональная возможность еще не обеспечивается, воздерживаются от использования запрашиваемой функциональной возможности, после приема указания, что запрашиваемая функциональная возможность еще не обеспечивается, принимают указание, что запрашиваемая функциональная возможность обеспечивается, и используют запрашиваемую функциональную возможность.

24. Способ по п.23, в котором запрашиваемая функциональная возможность является функциональной возможностью обмена данными между клиентом и сервером, выбранным из группы, состоящей из сжатия, качества обслуживания, пропускной способности, гарантированного времени ответа, устойчивости к ошибкам, целостности последовательности сообщений и отсутствия дублирования и максимально допустимого уровня потерь.

25. Способ по п.23, в котором выдача запроса функциональной возможности содержит этап, на котором посылают явный запрос функциональной возможности на сервер.

26. Способ по п.23, в котором выдача запроса функциональной возможности содержит этап, на котором посылают неявный запрос функциональной возможности.

27. Способ по п.26, в котором неявный запрос функциональной возможности определяют в соответствии с протоколом передачи данных, используемым клиентом и сервером.

28. Способ по п.23, в котором выдача запроса функциональной возможности содержит выдачу запроса о первом уровне обеспечения запрашиваемой функциональной возможности, и в котором прием указания, что запрашиваемая функциональная возможность обеспечивается, содержит прием указания, что запрашиваемая функциональная возможность обеспечивается на втором уровне обеспечения запрашиваемой функциональной возможности.

29. Способ по п.28, в котором второй уровень обеспечения запрашиваемой функциональной возможности ниже первого уровня обеспечения запрашиваемой функциональной возможности.

30. Способ по п.28, в котором использование запрашиваемой функциональной возможности содержит этап, на котором используют запрашиваемую функциональную возможность на втором уровне обеспечения запрашиваемой функциональной возможности.

31. Способ по п.28, дополнительно содержащий этап, на котором после приема указания, что запрашиваемая функциональная возможность обеспечивается на втором уровне обеспечения запрашиваемой функциональной возможности, принимают указание, что запрашиваемая функциональная возможность обеспечивается на первом уровне обеспечения запрашиваемой функциональной возможности.

32. Способ по п.23, в котором прием одного указания содержит этапы, на которых принимают указание, что запрос удовлетворен, и принимают указание, что запрашиваемая функциональная возможность еще не обеспечивается.

33. Способ по п.23, в котором прием указания, что запрашиваемая функциональная возможность еще не обеспечивается, содержит этап, на котором принимают данные от сервера без использования запрашиваемой функциональной возможности.

34. Способ по п.23, в котором прием указания, что запрашиваемая функциональная возможность обеспечивается, содержит этап, на котором принимают данные от сервера с использованием запрашиваемой функциональной возможности.

35. Способ по п.23, дополнительно содержащий этап, на котором перед приемом указания, что запрашиваемая функциональная возможность обеспечивается, выполняют процедуру аутентификации клиента по отношению к серверу.

36. Способ по п.23, дополнительно содержащий этап, на котором отображают пользователю клиента состояние запроса функциональной возможности, как «отклонен», «принят, но еще не обеспечивается» или «принят и обеспечивается».

37. Способ по п.36, дополнительно содержащий этап, на котором отображают пользователю клиента уровень обеспечения запроса функциональной возможности.

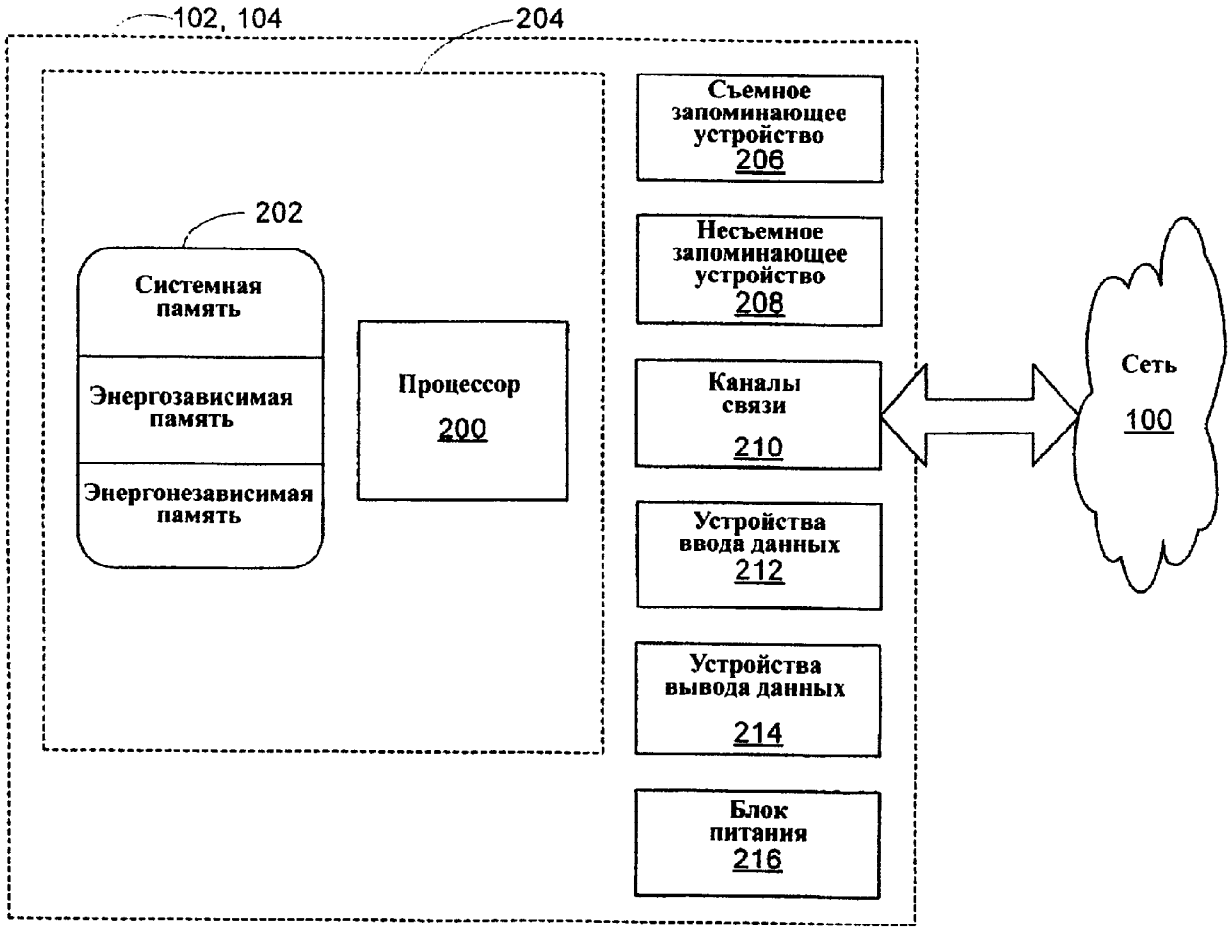
38. Машиночитаемый носитель, содержащий машиноисполняемые команды для выполнения способа для вычислительного устройства-клиента с целью получения обеспечиваемой вычислительным устройством-сервером функциональной возможности, причем данный способ содержит этапы, на которых запрашивают функциональную возможность у сервера, принимают указание, что запрос удовлетворен, принимают указание, что запрашиваемая функциональная возможность еще не обеспечивается, воздерживаются от использования запрашиваемой функциональной возможности,

после приема указания, что запрашиваемая функциональная возможность еще не обеспечивается, принимают указание, что запрашиваемая функциональная возможность обеспечивается, и используют запрашиваемую функциональную возможность.

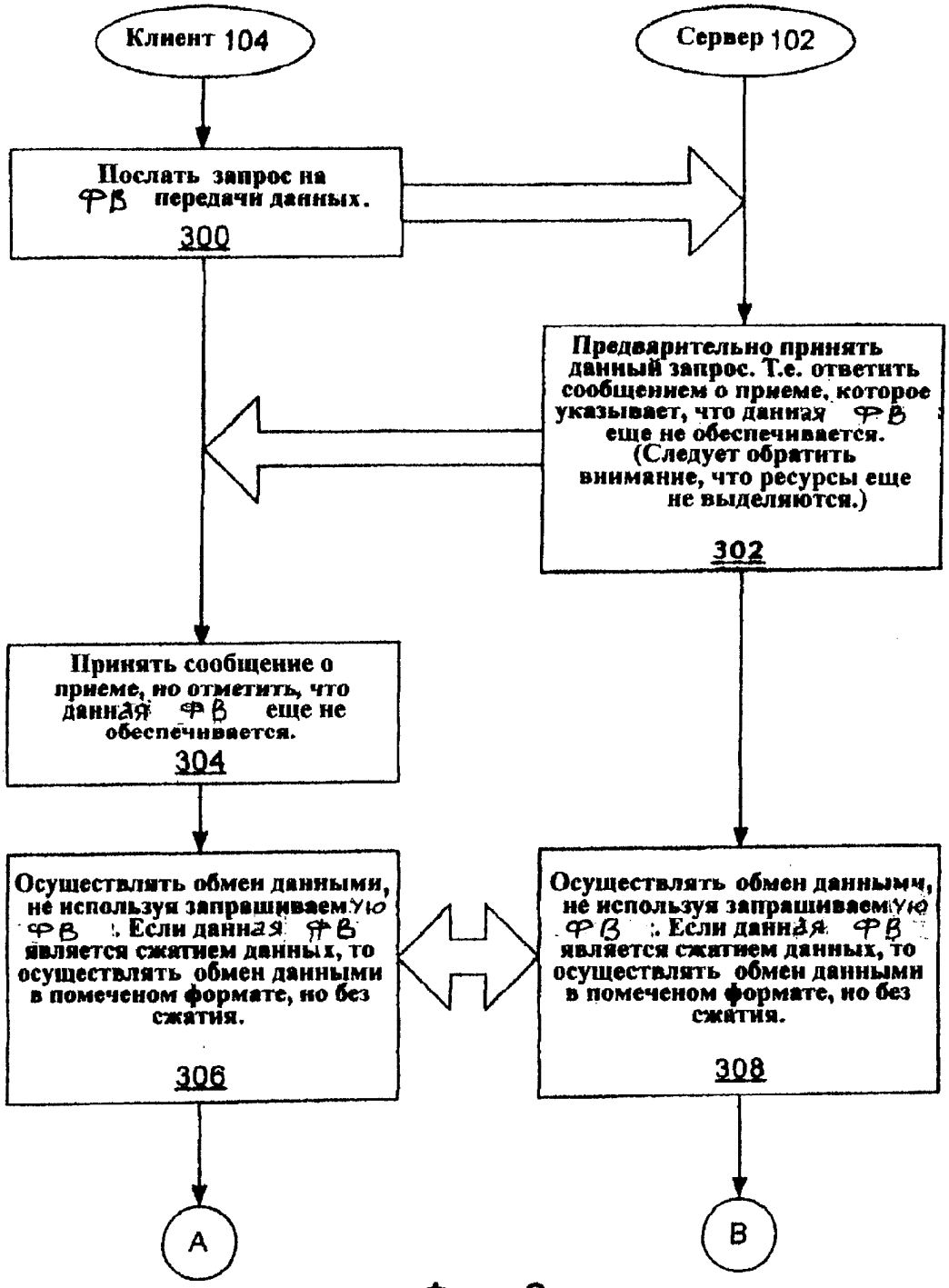
39. Реализуемый в среде передачи данных, содержащей вычислительное устройство-сервер и вычислительное устройство-клиент, способ отсроченного обеспечения сервером функциональной возможности для клиента и получения клиентом обеспечиваемой сервером функциональной возможности, при этом данный способ содержит этапы, на которых клиент запрашивает функциональную возможность у сервера, сервер принимает запрос функциональной возможности, сервер указывает клиенту, что запрос удовлетворен, клиент принимает указание, что запрос удовлетворен, сервер указывает клиенту, что запрашиваемая функциональная возможность еще не обеспечивается, клиент принимает указание, что запрашиваемая функциональная возможность еще не обеспечивается, клиент воздерживается от использования запрашиваемой функциональной возможности, после указания сервером клиенту, что запрашиваемая функциональная возможность еще не обеспечивается, сервер указывает клиенту, что запрашиваемая функциональная возможность обеспечивается, после приема клиентом указания, что запрашиваемая функциональная возможность еще не обеспечивается, клиент принимает указание, что запрашиваемая функциональная возможность обеспечивается, сервер обеспечивает запрашиваемую функциональную возможность для клиента, и клиент использует запрашиваемую функциональную возможность.

40. Машиночитаемый носитель, содержащий машиноисполняемые команды для выполнения способа для вычислительного устройства-сервера с целью отсроченного обеспечения функциональной возможности для вычислительного устройства-клиента и для клиента с целью получения обеспечиваемой сервером функциональной возможности, причем данный способ содержит этапы, на которых клиент запрашивает функциональную возможность у сервера, сервер принимает запрос функциональной возможности, сервер указывает клиенту, что запрос удовлетворен, клиент принимает указание, что запрос удовлетворен, сервер указывает клиенту, что запрашиваемая функциональная возможность еще не обеспечивается, клиент принимает указание, что запрашиваемая функциональная возможность еще не обеспечивается, клиент воздерживается от использования запрашиваемой функциональной возможности, после указания сервером клиенту, что запрашиваемая функциональная возможность еще не обеспечивается, сервер указывает клиенту, что запрашиваемая функциональная возможность обеспечивается, после приема клиентом указания, что запрашиваемая функциональная возможность еще не обеспечивается, клиент принимает указание, что запрашиваемая функциональная возможность обеспечивается, сервер обеспечивает запрашиваемую функциональную возможность для клиента,

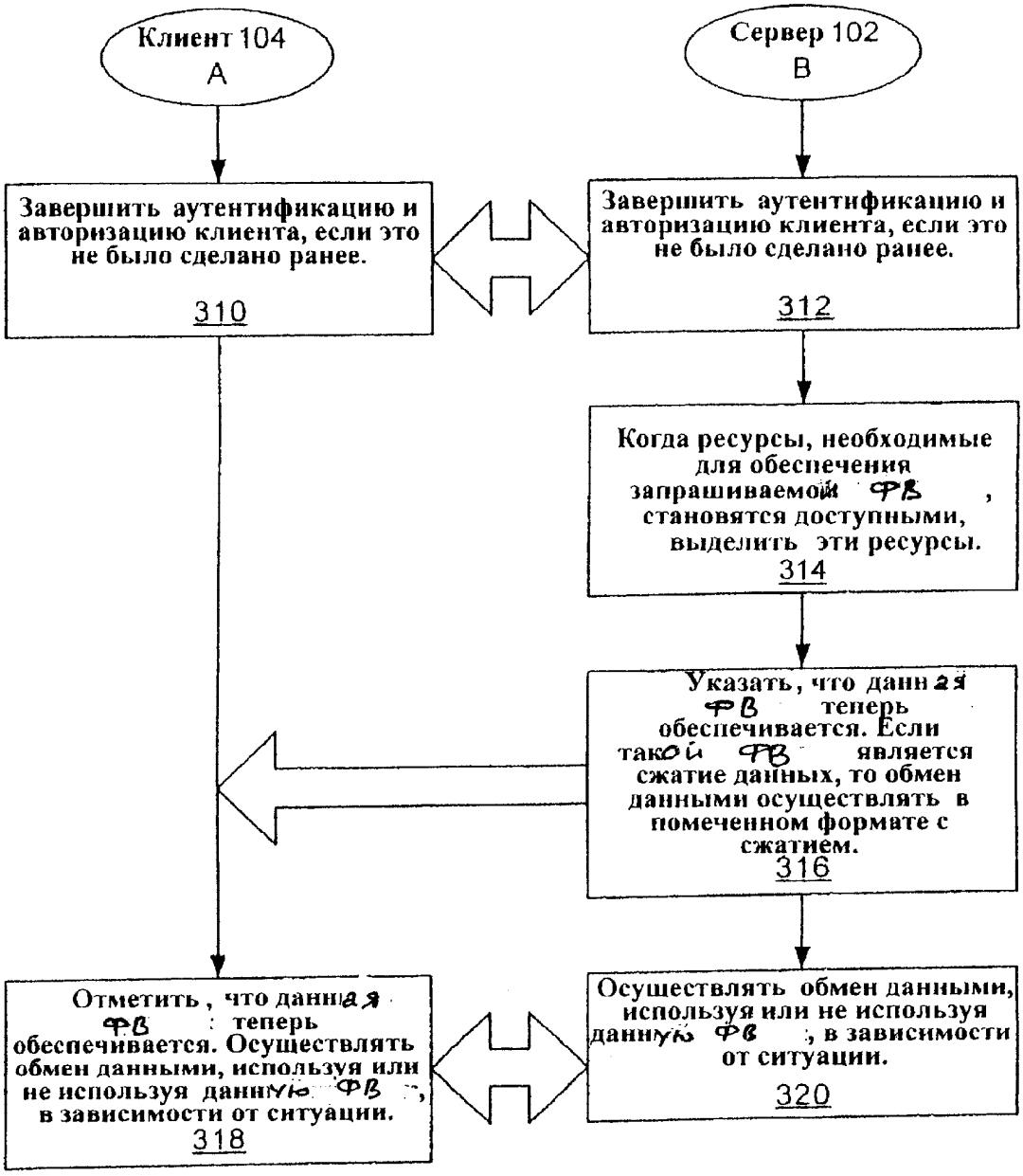
и клиент использует запрашиваемую функциональную возможность.


102, 104
204
202
Системная память
Энергозависимая память
Энергонезависимая память
Процессор
200
Съемное запоминающее устройство
206
Несъемное запоминающее устройство
208
Каналы связи
210
Устройства ввода данных
212
Устройства вывода данных
214
Блок питания
216
Сеть
100

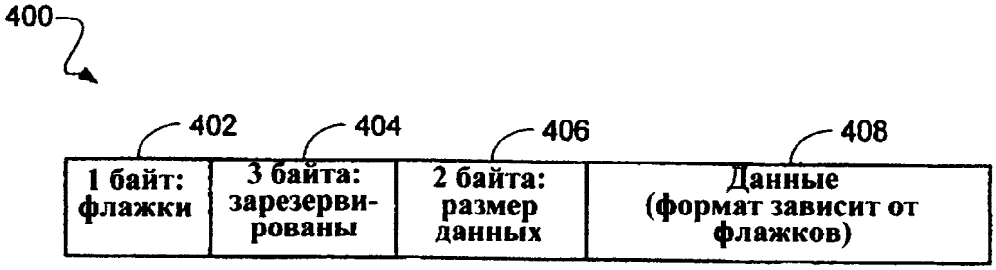
Фиг. 2


Клиент 104
Сервер 102
Послать запрос на РВ передачи данных.
300
Предварительно принять данный запрос. Т.е. ответить сообщением о приеме, которое указывает, что данная РВ еще не обеспечивается. (Следует обратить внимание, что ресурсы еще не выделяются.)
302
Принять сообщение о приеме, но отметить, что данная РВ еще не обеспечивается.
304
Осуществлять обмен данными, не используя запрашиваемую РВ. Если данная РВ является сжатием данных, то осуществлять обмен данными в помеченом формате, но без сжатия.
306
Осуществлять обмен данными, не используя запрашиваемую РВ. Если данная РВ является сжатием данных, то осуществлять обмен данными в помеченом формате, но без сжатия.
308
A
B

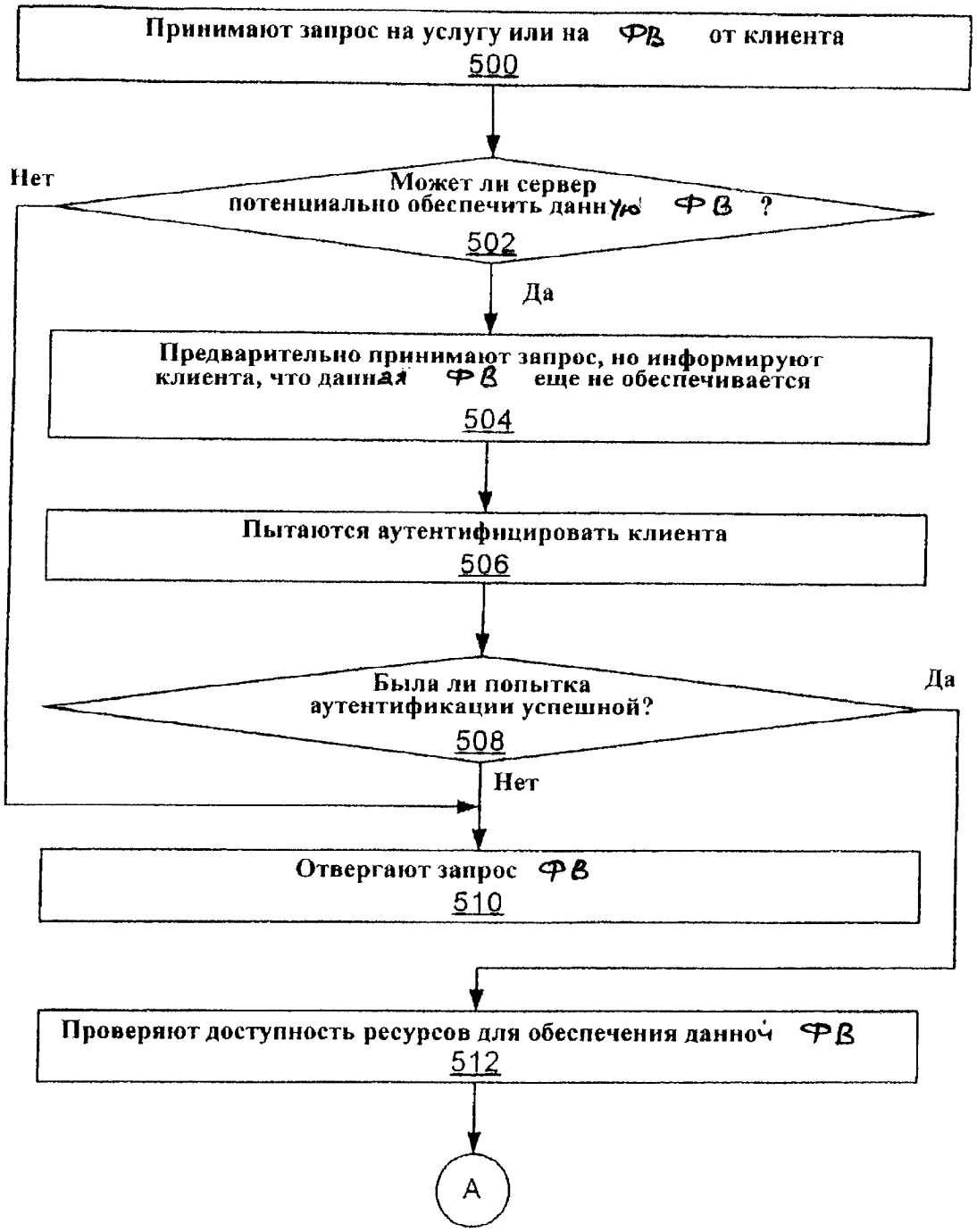
Фиг. 3а


Клиент 104
A
Сервер 102
B
Завершить аутентификацию и авторизацию клиента, если это не было сделано ранее.
310
Завершить аутентификацию и авторизацию клиента, если это не было сделано ранее.
312
Когда ресурсы, необходимые для обеспечения запрашиваемой ФВ, становятся доступными, выделить эти ресурсы.
314
Указать, что данная ФВ теперь обеспечивается. Если такой ФВ является сжатие данных, то обмен данными осуществлять в помеченном формате с сжатием.
316
Отметить, что данная ФВ теперь обеспечивается. Осуществлять обмен данными, используя или не используя данную ФВ, в зависимости от ситуации.
318
Осуществлять обмен данными, используя или не используя данную ФВ, в зависимости от ситуации.
320

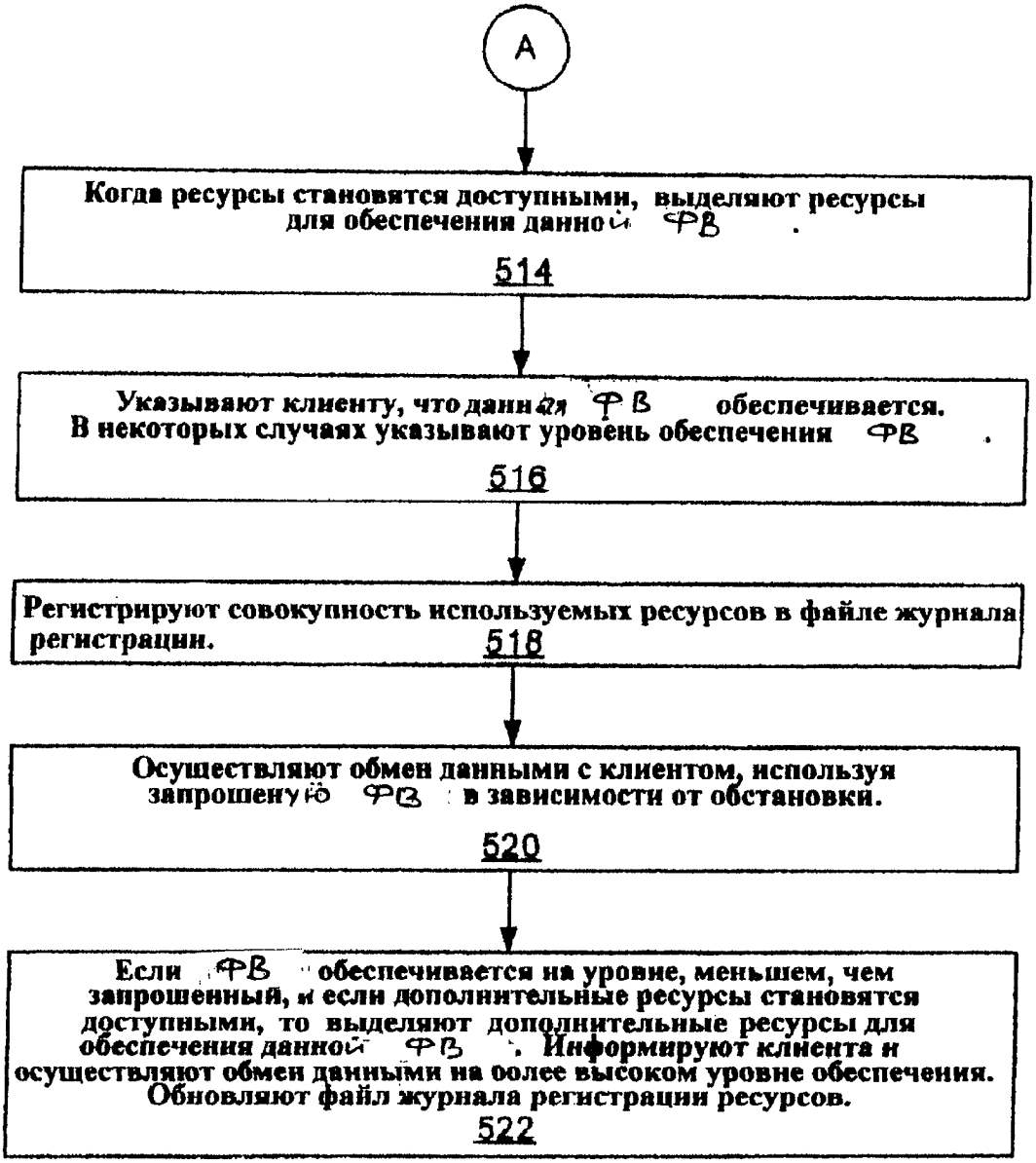
Фиг. 3b


400
402
404
406
408
1 байт: флажки
3 байта: зарезервированы
2 байта: размер данных
Данные (формат зависит от флажков)

Фиг. 4


Принимают запрос на услугу или на ФВ от клиента
500
Может ли сервер потенциально обеспечить данную ФВ ?
502
Нет
Да
Предварительно принимают запрос, но информируют клиента, что данная ФВ еще не обеспечивается
504
Пытаются аутентифицировать клиента
506
Была ли попытка аутентификации успешной?
508
Да
Нет
Отвергают запрос ФВ
510
Проверяют доступность ресурсов для обеспечения данной ФВ
512
A

Фиг. 5а



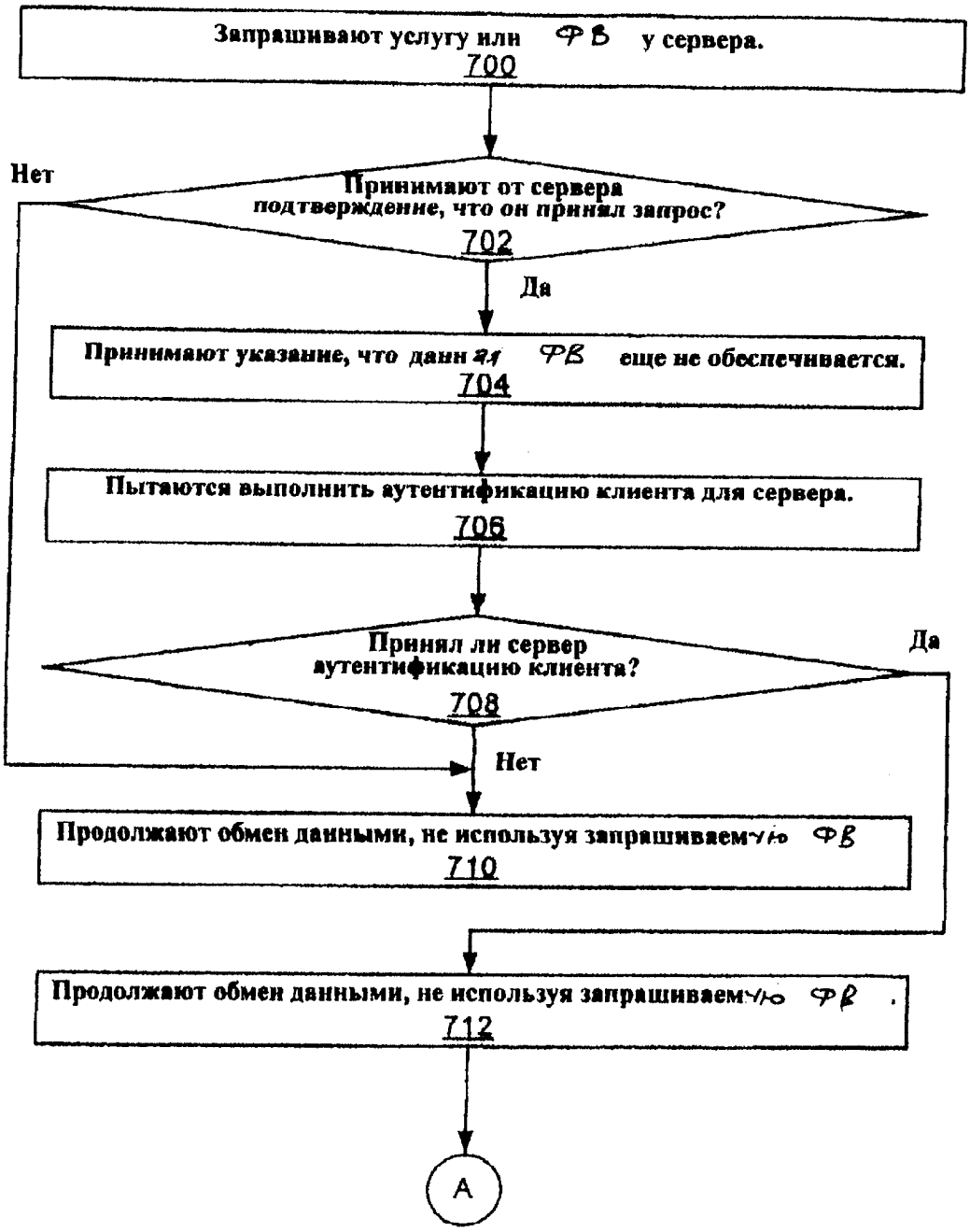
Фиг. 5b

Фиг. 6

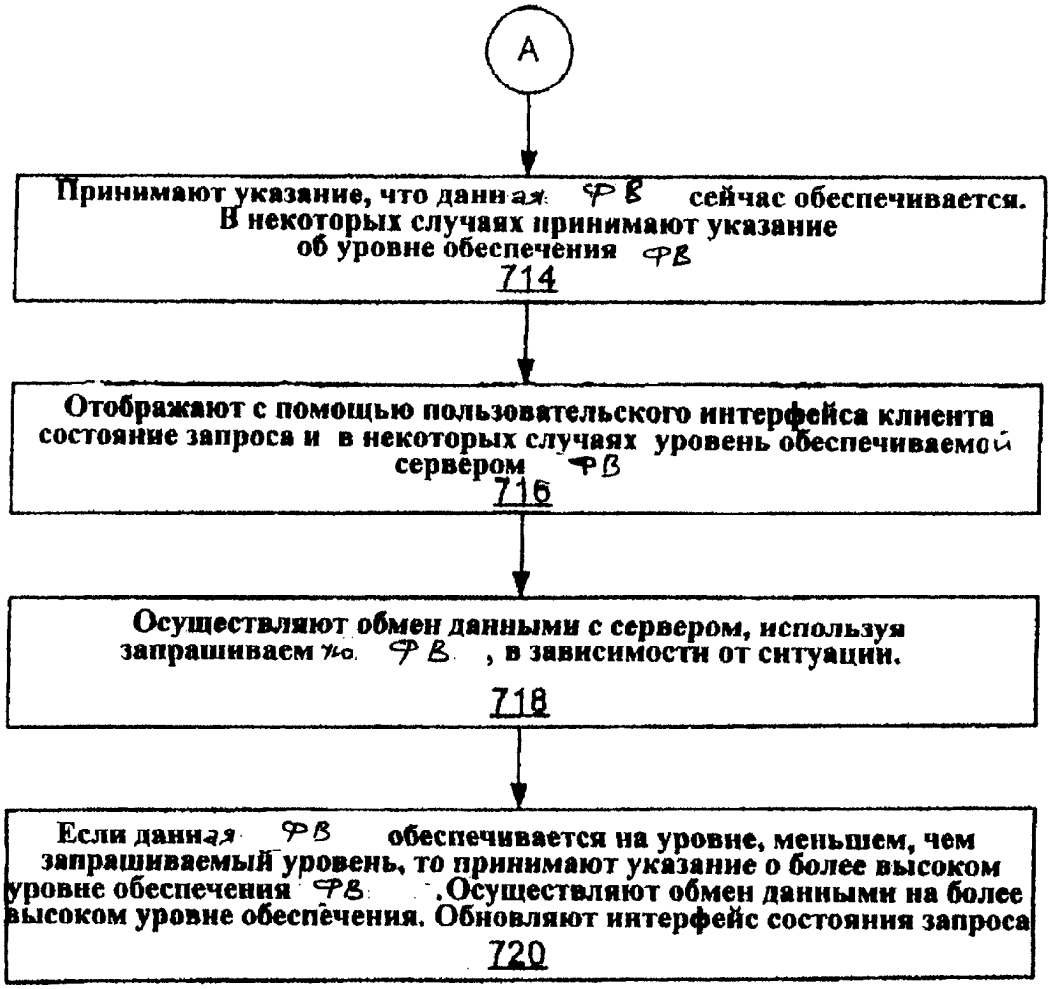
600

602 Клиент	604 Запрашиваемая ФВ	606 Состояние
104	Сжатие	Обеспечивается
106	Сжатие	Отвергнуто
108	Сжатие	Принято, но не обеспечивается
108	Гарантированная пропускная способность	Желательна 512 кбит/с; обеспечивается 128 кбит/с

Фиг. 6


Запрашивают услугу или ФВ у сервера.
700
Нет
Принимают от сервера подтверждение, что он принял запрос?
702
Да
Принимают указание, что данная ФВ еще не обеспечивается.
704
Пытаются выполнить аутентификацию клиента для сервера.
706
Принял ли сервер аутентификацию клиента?
708
Да
Нет
Продолжают обмен данными, не используя запрашиваемую ФВ
710
Продолжают обмен данными, не используя запрашиваемую ФВ
712
A

Фиг. 7а


A
Принимают указание, что данная ФВ сейчас обеспечивается. В некоторых случаях принимают указание об уровне обеспечения ФВ
714
Отображают с помощью пользовательского интерфейса клиента состояние запроса и в некоторых случаях уровень обеспечиваемой сервером ФВ
716
Осуществляют обмен данными с сервером, используя запрашиваемую ФВ, в зависимости от ситуации.
718
Если данная ФВ обеспечивается на уровне, меньшем, чем запрашиваемый уровень, то принимают указание о более высоком уровне обеспечения ФВ. Осуществляют обмен данными на более высоком уровне обеспечения. Обновляют интерфейс состояния запроса
720

Фиг. 7b