



**ФЕДЕРАЛЬНАЯ СЛУЖБА  
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ**

**(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ**

(21)(22) Заявка: 2013152619/08, 09.10.2011

(24) Дата начала отсчета срока действия патента:  
09.10.2011

Приоритет(ы):

(30) Конвенционный приоритет:  
27.05.2011 US 13/118,204

(43) Дата публикации заявки: 10.06.2015 Бюл. № 16

(45) Опубликовано: 10.06.2016 Бюл. № 16

(56) Список документов, цитированных в отчете о  
поиске: US 2010/0146384 A1, 10.06.2010. US  
2009/0293013 A1, 26.11.2009. EP 1939718 A1,  
02.07.2008. US 2009/0305732 A1, 10.12.2009.(85) Дата начала рассмотрения заявки РСТ на  
национальной фазе: 26.11.2013(86) Заявка РСТ:  
US 2011/055520 (09.10.2011)(87) Публикация заявки РСТ:  
WO 2012/166179 (06.12.2012)

Адрес для переписки:

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

(72) Автор(ы):

**ДОНАХЬЮ Тайлер Дж. (US),  
МАКНИЛ Брайан К. (US),  
УОЛИ Кевин Майкл (US),  
ЭЙЕРС Мэттью Р. (US),  
АНАНД Гаурав С. (US),  
РАВАТ Аншул (US),  
ИВАНОВИЧ Релджа (US)**

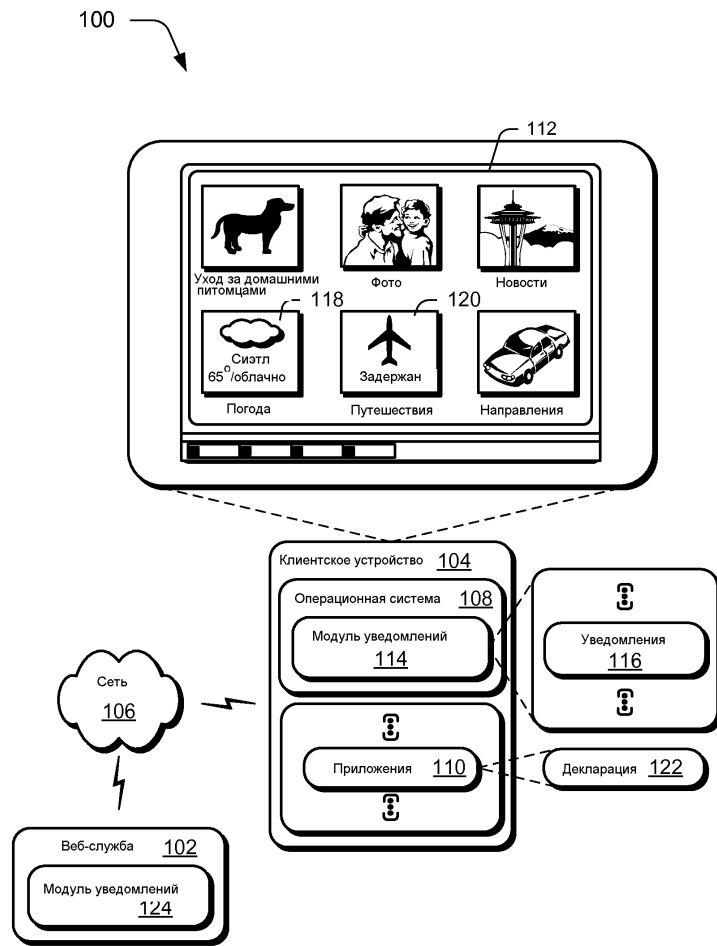
(73) Патентообладатель(и):

**МАЙКРОСОФТ ТЕКНОЛОДЖИ  
ЛАЙСЕНСИНГ, ЭлЭлСи (US)****(54) УВЕДОМЛЕНИЯ О ПРИЛОЖЕНИЯХ**

(57) Реферат:

Изобретение относится к методам уведомления о приложениях. Технический результат - эффективное управление уведомлениями, связанными с приложениями. Способ управления уведомлениями, связанными с приложениями, реализуемый одним или более вычислительными устройствами, при этом способ содержит этапы, на которых: отображают представление приложения в пользовательском интерфейсе; принимают одно или более уведомлений, которые относятся к приложению, для их отображения в качестве части представления приложения в

пользовательском интерфейсе; анализируют манифест приложения без исполнения приложения, чтобы найти критерии, задающие, как и когда должны отображаться упомянутые уведомления в качестве части представления приложения в пользовательском интерфейсе; и отображают представление приложения в пользовательском интерфейсе включающим в себя упомянутые одно или более уведомлений в пользовательском интерфейсе в соответствии с упомянутыми задающими критериями. 3 н. и 17 з.п. ф-лы, 12 ил.



ФИГ.1



FEDERAL SERVICE  
FOR INTELLECTUAL PROPERTY

(51) Int. Cl.  
*G06F 9/44* (2006.01)  
*G06F 3/14* (2006.01)

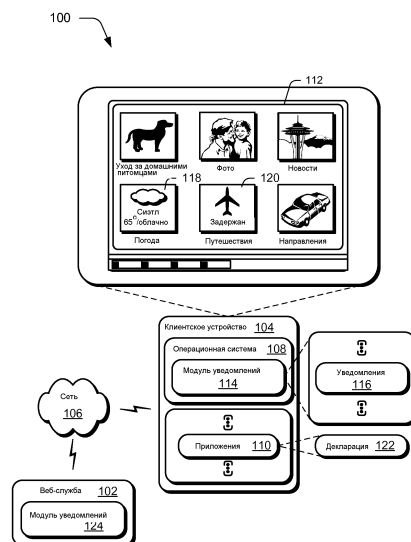
(12) **ABSTRACT OF INVENTION**

(21)(22) Application: 2013152619/08, 09.10.2011  
 (24) Effective date for property rights: 09.10.2011  
 Priority:  
 (30) Convention priority: 27.05.2011 US 13/118,204  
 (43) Application published: 10.06.2015 Bull. № 16  
 (45) Date of publication: 10.06.2016 Bull. № 16  
 (85) Commencement of national phase: 26.11.2013  
 (86) PCT application: US 2011/055520 (09.10.2011)  
 (87) PCT publication: WO 2012/166179 (06.12.2012)  
 Mail address: 129090, Moskva, ul. B. Spasskaja, 25, stroenie 3, OOO "Juridicheskaja firma Gorodisskij i Partnery"

(72) Inventor(s): DONAKHYU Tajler Dzh. (US), MAKNIL Brajan K. (US), UOLI Kevin Majkl (US), EJERS Mettyu R. (US), ANAND Gaurav S. (US), RAVAT Anshul (US), IVANOVICH Reldzha (US)  
 (73) Proprietor(s): MAJKROSOFT TEKNOLODZHI LAJSENSING, EIEISi (US)

(54) **APPLICATION NOTIFICATIONS**

(57) Abstract:  
 FIELD: information technology.  
 SUBSTANCE: method for control of notifications associated with applications, implemented by one or more computing devices, said method comprising steps of: displaying an application user interface representation; receiving one or more notifications, which relates to an application, for their display as a part of presentation of an application user interface; analysing application manifest without executing an application to be setting criteria as and when said notification should be displayed as a part of presentation of an application user interface; and displaying an application user interface representation including said one or more notifications in a user interface in accordance with said define criteria.  
 EFFECT: efficient control of notifications associated with applications.  
 20 cl, 12 dwg



ФИГ. 1

RU 2 586 869 C 2

RU 2 586 869 C 2

[0001] Распространенность вычислительных устройств постоянно растет. Например, пользователи могут взаимодействовать с традиционным настольным компьютером, планшетным компьютером, мобильным телефоном и т.д., чтобы получить доступ к различным функциональным возможностям для работы и личного использования.

5 Кроме того, различные функциональные возможности, доступные для пользователей этих устройств, также продолжают возрастать.

[0002] Это особенно верно в случае приложений. Доступность приложений была традиционно ограничена, так как пользователи обычно приобретали приложения в "bricks and mortar" (традиционных, не связанных с Интернетом) магазинах. С появлением  
10 доступности приложений через Интернет количество приложений, которые становятся доступными для пользователя, выросло вместе с легкостью, с которой пользователь может получить доступ к этим приложениям. Соответственно пользователи могут выбирать и устанавливать большое количество приложений на пользовательском вычислительном устройстве.

15 [0003] Однако взаимодействие пользователя с приложениями может быть затруднено количеством приложений, которые установлены на вычислительном устройстве. Это может включать в себя поиск интересующего приложения, а также доступ к функциональности приложения. Например, пользователь может выбрать приложение «погода», чтобы получить информацию о погоде, затем взаимодействовать с  
20 приложением «новости», чтобы просмотреть последние новости, и так далее через ряд других приложений. Следовательно, навигация по этим приложениям, чтобы найти нужную информацию, может занять значительное количество времени, что может быть дополнительно осложнено количеством приложений, которые доступны на вычислительном устройстве.

## 25 СУЩНОСТЬ ИЗОБРЕТЕНИЯ

[0004] Описаны методики уведомления для приложений. Описаны реализации, в которых представление приложения может включать в себя уведомления, которые относятся к приложению. Далее описаны методики, которые могут быть использованы для управления уведомлениями, включая замену уведомлений, использование очередей,  
30 переопределения, выбор уведомлений на основе состояния исполнения приложения, циклирование отображения множества уведомлений (например, пользователем или автоматически с помощью вычислительного устройства), циклирование отображений различных подмножеств уведомлений, рассмотрение манифеста приложения, чтобы определять критерии, которые должны использоваться для отображения уведомлений,  
35 приоритет отображения уведомлений и так далее.

[0005] Это краткое изложение сущности изобретения предоставлено, чтобы ввести подборку концепций в упрощенной форме, которые дополнительно описаны ниже в подробном описании. Это краткое изложение сущности изобретения не предназначено ни для идентификации ключевых признаков или существенных признаков заявленного  
40 изобретения ни для использования в качестве помощи в определении объема заявленного изобретения.

## КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ

[0006] Подробное описание приведено со ссылками на прилагаемые чертежи. На чертежах самая(ые) левая(ые) цифра(ы) ссылочной позиции определяет(ют) фигуру, на  
45 которой ссылочная позиция появляется впервые. Использование одних и тех же ссылочных позиций в различных случаях в описании и на чертежах может указывать подобные или идентичные элементы.

[0007] Фиг. 1 является иллюстрацией среды в примерной реализации, которая

выполнена с возможностью применять методики уведомления для приложений.

[0008] Фиг. 2 изображает систему в примерной реализации, в которой множество уведомлений отображается последовательно как часть представления приложения.

5 [0009] Фиг. 3 показывает систему, в которой используется тег для указания того, что соответствующее уведомление должно быть использовано для замены другого уведомления.

[0010] Фиг. 4 показывает блок-схему, изображающую процедуру примерной реализации, в которой используется очередь для управления уведомлениями для вывода в качестве части представления приложения.

10 [0011] Фиг. 5 показывает блок-схему, изображающую процедуру примерной реализации, в которой используются теги для управления заменой уведомлений для отображения в качестве части представления приложения.

[0012] Фиг. 6 показывает блок-схему, изображающую процедуру примерной реализации, в которой используется очередь для управления уведомлениями для отображения в качестве части представление приложения.

15 [0013] Фиг. 7 показывает блок-схему, изображающую процедуру примерной реализации, в которой состояние выполнения приложения используется в качестве основы для определения того, какие уведомления должны быть отображены в качестве части представления приложения.

20 [0014] Фиг. 8 показывает блок-схему, изображающую процедуру примерной реализации, в которой уведомления отображаются в представлениях приложений, которые могут быть выбраны, чтобы запустить приложение, причем отображение циклируется для последовательного отображения уведомлений.

25 [0015] Фиг. 9 показывает блок-схему, изображающую процедуру примерной реализации, в которой подмножества уведомлений циклируются для отображения в качестве части соответствующих представлений приложений.

[0016] Фиг. 10 показывает блок-схему, изображающую процедуру примерной реализации, в которой манифест приложения используется для указания критериев, которые должны использоваться для управления отображением уведомлений в качестве части представления приложения.

[0017] Фиг. 11 показывает блок-схему, изображающую процедуру примерной реализации, в которой веб-служба используется для управления предоставлением уведомлений на клиентском устройстве для отображения.

35 [0018] Фиг. 12 показывает блок-схему, изображающую процедуру примерной реализации, в которой уведомления приоритизируются для отображения в качестве части представления приложения.

**ПОДРОБНОЕ ОПИСАНИЕ ОБЗОР**[0019] Уведомления для приложений могут использоваться приложениями для отображения информации в представлении приложения (например, иконка, мозаика и т.д.) без того, что пользователь специально

40 запускает приложение. Например, представление приложения «погода» может включать уведомление, которое описывает текущие погодные условия. Однако обычные методы, которые использовались для отображения уведомления, были часто статическими и поэтому не позволяли разработчикам управлять тем, каким образом уведомления отображались в качестве части представления.

45 [0020] Однако здесь описаны методики, которые могут быть использованы для управления уведомлениями. Кроме того, в одной или более реализаций эти методики могут быть выполнены без исполнения соответствующего приложения, сохраняя тем самым срок службы аккумулятора и повышая производительность вычислительных

устройств, которые выполняют эти методики. В следующих разделах описаны примеры множества различных методик, которые относятся к уведомлениям для приложений, например замена уведомлений, использование очередей, переопределения, выбор уведомления на основе состояния выполнения приложения, циклирование отображения множества уведомлений, циклирование отображения различных подмножеств уведомлений, рассмотрение манифеста приложения, чтобы определить критерии, которые будут использоваться для отображения уведомлений, отображение приоритета уведомлений и так далее. Дальнейшее обсуждение этих методов и другого может быть найдено в следующих разделах.

[0021] В последующем обсуждении сначала описывается иллюстративная среда, которая может использовать методики, описанные в данном документе. Затем описываются процедуры, которые могут быть выполнены в иллюстративной среде, а также в других средах. Следовательно, выполнение иллюстративных процедур не ограничивается иллюстративной средой, и иллюстративная среда не ограничивается выполнением иллюстративных процедур.

### ИЛЛЮСТРАТИВНАЯ СРЕДА

[0022] Фиг. 1 является иллюстрацией среды 100 в иллюстративной реализации, которая выполнена с возможностью использования методов, описанных в данном документе. Иллюстративная среда 100 включает в себя веб-службу 102 и клиентское устройство 104, которые коммуникативно связаны через сеть 106. Веб-служба 102 и клиентское устройство 104 могут быть реализованы посредством широкого диапазона вычислительных устройств.

[0023] Например, вычислительное устройство может быть сконфигурировано как компьютер, который способен осуществлять связь по сети 106, такой как настольный компьютер, мобильная станция, развлекательное устройство, телеприставка, коммуникативно-соединенная с устройством отображения, мобильное устройство связи (например, беспроводной телефон, как показано для клиентского устройства 104), игровая консоль и так далее. Таким образом, вычислительное устройство может варьироваться от полноресурсного устройства с существенными ресурсами памяти и процессора (например, персональные компьютеры, игровые консоли) до малоресурсного устройства с ограниченной памятью и/или ресурсами обработки (например, традиционные телевизионные приставки, портативные игровые консоли). Кроме того, вычислительное устройство может быть представителем множества различных устройств, таких как несколько серверов, используемых в бизнесе для выполнения операций, комбинация пульта дистанционного управления и приставки, устройство захвата изображения и игровая консоль, конфигурированные для захвата (съемки) жестов и так далее.

[0024] Хотя сеть 106 проиллюстрирована как Интернет, сеть может принимать широкое разнообразие конфигураций. Например, сеть 106 может включать в себя глобальную сеть (WAN), локальную сеть (LAN), беспроводную сеть, телефонную сеть общего пользования, интранет и так далее. Кроме того, хотя показана одиночная сеть 106, сеть 106 может быть сконфигурирована, чтобы включать в себя несколько сетей.

[0025] Клиентское устройство 104 дополнительно проиллюстрировано как включающее в себя операционную систему 108. Операционная система 108 сконфигурирована, чтобы абстрагировать базовую функциональность клиента 104 от приложений 110, которые выполняются на клиентском устройстве 104. Например, операционная система 108 может абстрагировать обработку, память, сеть и/или функциональные возможности отображения клиентского устройства 104, так что

приложения 110 могут быть записаны, не зная, "как" эта базовая функциональность реализована. Приложение 110, например, может предоставить операционной системе 108 данные, подлежащие воспроизведению и отображению устройством 112 отображения, без понимания того, как это воспроизведение будет выполняться.

5 [0026] Операционная система 108 может также представлять многообразие другой функциональности, такой как управление файловой системой и пользовательский интерфейс, который обеспечивает выполнение поиска и просмотра пользователем клиентского устройства 104. Пример этого показан в виде запускающего модуля приложения (например, рабочего стола), который отображается на устройстве 112  
10 отображения клиентского устройства 104. Рабочий стол включает представления множества приложений 110, такие как иконки, мозаика, текстовые описания и так далее. Рабочий стол может рассматриваться как корневой уровень иерархической файловой структуры. Представления, показанные в приведенном примере, могут выбираться для запуска соответствующего одного из приложений 110 для выполнения на клиентском  
15 устройстве 104. Таким образом, пользователь может легко перемещаться по файловой структуре и инициировать выполнение приложения, представляющего интерес.

[0027] Операционная система 108 также показана как включающая в себя модуль 114 уведомлений. Модуль 114 уведомлений представляет функциональность для  
20 управления уведомлениями 116, которые должны быть отображены как часть представлений приложений. Например, представление 118 приложения «погода» иллюстрируется как включающее в себя уведомление, которое указывает название и текущие погодные условия, например "Сиэтл 65°/облачно". Кроме того, представление 120 для приложения «путешествия» иллюстрируется как включающее в себя уведомление, которое указывает, что текущий статус полета - "задержан". Таким образом,  
25 пользователь может легко просмотреть информацию, относящуюся к приложениям 110, без запуска и навигации по каждому из приложений. Хотя показаны представления конкретных приложений, другие представления приложений также рассматриваются, такие как представление приложения, которое ссылается на причастность пользователя к службе, например друг в службе социальной сети.

30 [0028] В одной или нескольких реализациях уведомлениями 116 можно управлять без выполнения соответствующего приложения 110. Например, модуль 114 уведомлений может получать уведомления 116 из множества различных источников, например из программного обеспечения (например, других приложений, выполняемых клиентским устройством 104), из веб-службы 102 через сеть 106 и так далее. Модуль 114 уведомлений  
35 может тогда управлять тем, каким образом уведомления 116 отображаются в качестве части представлений без исполнения приложений 110. Это может быть использовано для улучшения срока службы аккумулятора и производительности клиентского устройства 104 за счет невыполнения каждого из приложений 110 для вывода уведомления 116.

40 [0029] Кроме того, модуль 114 уведомления может поддерживать методы, чтобы позволять разработчикам соответствующих приложений 110 указывать, каким образом уведомления 116 должны быть отображены как часть представления приложений 110. Например, модуль 114 уведомлений может рассмотреть манифест 122 приложения 110, например манифест установки, используемый для установки приложения 110, во время  
45 исполнения, используя системный вызов, и так далее. Манифест 110 может описывать, как и когда уведомления 116 должны отображаться, а также то, каким образом уведомления 116 подлежат регулированию, чтобы определить, какие уведомления 116 должны отображаться.

[0030] Модуль 114 уведомлений также может использовать различные другие методы, чтобы дать возможность разработчику или другому объекту определить, каким образом уведомления 116 должны отображаться и управляться. Например, уведомления 116 могут быть переданы вместе с соответствующими тегами, которые определяют, каким образом уведомления 116 должны отображаться и/или управляться. Таким образом, инициатор (отправитель, источник) тегов может указать, каким образом соответствующие уведомления 116 должны отображаться или управляться. Это может быть выполнено, чтобы запускать многообразие различных функциональностей, как дополнительно описано в следующих разделах.

[0031] Хотя в этом обсуждении описывалось включение модуля 114 уведомлений на клиенте, функциональность модуля 114 уведомлений может быть реализована в различных формах. Например, функциональность модуля 124 уведомлений может быть включена в веб-службу 102 полностью или частично. Модуль 124 уведомлений, например, может обрабатывать уведомления, полученные от других веб-служб, и управлять уведомлениями для распределения на клиентское устройство 104 через сеть 106.

[0032] Многообразие различных методов может быть использовано модулем 124 уведомлений, например, для обработки уведомлений в соответствии с методами, которые будут использоваться самим клиентским устройством 104, например, для рассмотрения манифеста, обработки тегов в ответ на сообщение, полученное от клиентского устройства 104, которое задает критерии для управления уведомлениями 116, и так далее. Кроме того, эта обработка может быть выполнена в случаях, когда клиентское устройство 104 недоступно, например, из-за отсутствия соединения сети 106. Таким образом, веб-служба 102 может эффективно использовать пространство хранения и пропускную способность сети 106. Другие примеры также рассматриваются, такие как распределение функциональных возможностей модуля 114 уведомлений между клиентским устройством 104 и веб-службой 102, включение службы третьей стороны и так далее.

[0033] Как правило, любая из функций, описанных здесь, может быть реализована с использованием программного обеспечения, микропрограммного обеспечения, аппаратных средств (например, фиксированных логических схем) или комбинации этих реализаций. Термины "модуль", "функциональность" и "логика", используемые здесь, как правило, представляют программное обеспечение, микропрограммное обеспечение, аппаратные средства или их сочетание. В случае программной реализации модуль, функциональность или логика представляет программный код, который выполняет определенные задачи при исполнении на процессоре (например, центральном процессоре (CPU) или центральных процессорах). Программный код может храниться в одном или более машиночитаемых запоминающих устройств. Особенности методов, описанных ниже, являются независимыми от платформы, а это означает, что способы могут быть реализованы на различных коммерческих вычислительных платформах, имеющих различные процессоры.

[0034] Например, вычислительное устройство может также включать в себя объект (например, программное обеспечение), который вызывает выполнение операций аппаратными средствами вычислительного устройства, например процессорами, функциональными блоками и так далее. Например, вычислительное устройство может включать в себя машиночитаемый носитель, который может быть выполнен с возможностью поддержания инструкций, которые побуждают вычислительное устройство, более конкретно аппаратные средства вычислительного устройства к



выполнению операций. Таким образом, инструкции функционируют, чтобы конфигурировать аппаратные средства выполнять операции, и, таким образом, приводят в результате к трансформации аппаратных средств для выполнения функций. Инструкции могут быть предоставлены машиночитаемым носителем вычислительному устройству

5 через множество различных конфигураций.

[0035] Одной такой конфигурацией считываемого компьютером носителя является среда, несущая сигнал и, следовательно, конфигурированная для передачи инструкций (например, в виде несущей волны) к аппаратным средствам вычислительного устройства, например через сеть. Считываемый компьютером носитель также может быть

10 сконфигурирован как считываемый компьютером запоминающий носитель и, таким образом, не среда, несущая сигнал. Примеры считываемого компьютером запоминающего носителя включают в себя оперативное запоминающее устройство (RAM), постоянное запоминающее устройство (ROM), оптический диск, флэш-память, память на жестком диске и другие устройства памяти, которые могут использовать

15 магнитные, оптические и другие методы для хранения инструкций и других данных.

[0036] На фиг. 2 изображена система 200 в иллюстративной реализации, в которой множество уведомлений последовательно отображаются как часть представления приложения. Система 200 проиллюстрирована посредством использования первого, второго и третьего этапов 202, 204, 206. Уведомления 116 могут принимать различные

20 формы и упорядочиваться различными способами для отображения в пределах представления 118.

[0037] Как иллюстрируется на фиг. 2, например, показано уведомление 208, имеющее множество фреймов 210, 212. На первом этапе 202 первый 210 из фреймов уведомления 208 используется для отображения графического изображения облака и текста "Сиэтл"

25 для представления приложения погода.

[0038] На втором этапе 204 второй 212 из фреймов уведомления 208 используется для отображения текста "Сиэтл" и текста "65°/облачно", чтобы описать дополнительные текущие погодные условия в Сиэтле. Таким образом, уведомление 208 может быть

30 сконфигурировано, чтобы включать в себя фреймы 210, 212, которые должны отображаться последовательно как часть представления 118.

[0039] Дополнительные уведомления также могут быть отображены в последовательности как часть представления 118. Как показано на третьем этапе 206, например, уведомление 214 используется для отображения графики и текста из солнца и текущих погодных условий для "Спокан, 52°". Таким образом, в этом примере системы

35 200 модуль 114 уведомлений может быть использован для циклического переключения между уведомлениями 208, 214 для отображения в качестве части представления 118 приложения 110. Как отмечалось ранее, управление уведомлениями 116 может быть выполнено с использованием различных методов, пример которых с использованием тегов описан со ссылками на следующий чертеж.

[0040] На фиг. 3 изображена система 300, в которой тег используется для указания того, что соответствующее уведомление должно быть использовано для замены другого уведомления. Веб-служба 102 показана как коммуникативно связанная с клиентским устройством 104 через сеть 106, как описано выше со ссылкой на фиг. 1. В этом примере веб-служба 102 передает уведомление 302 и соответствующий тег 304 на клиентское

45 устройство 104.

[0041] При получении уведомления 302 модуль 114 уведомлений может определить, что имеется соответствующий тег 304. В соответствии с этим модуль 114 уведомлений может проверить память, которая является локальной для клиентского устройства 104,

чтобы определить, имеет ли другое уведомление 306 совпадающий тег.

[0042] В показанном примере уведомление 306 включает в себя тег 304, который совпадает с тегом 304 принятого уведомления 302. Соответственно модуль 114 уведомления может заменить уведомление 306 последним принятым уведомлением 302.

5 Таким образом, инициатор (отправитель) тега 302 может легко заменить одно уведомление на другое через использование тегов.

[0043] Тег 304 может быть задан в различных формах. Например, тег 304 может быть сконфигурирован таким образом, что значение тега 304 может быть назначено инициатором тега из любого значения. Таким образом, в этом примере модуль 114  
10 уведомления может управлять заменой уведомлений без "знания" того, что соответствующие теги "означают". Множество других примеров также могут учитываться, например, путем указания конкретного сегмента в очереди, пример которого описан со ссылкой на следующий чертеж. Хотя эта функциональность была описана со ссылкой на модуль 114 уведомления клиентского устройства 104, как  
15 указывалось ранее, эта функциональность может быть реализована с использованием разнообразия различных устройств, пример которых показан модулем 124 уведомлений веб-сайта 102.

[0044] На фиг. 4 изображена система 400 в примере реализации, в котором очередь используется для управления уведомлениями 116 для вывода в качестве части  
20 представления приложения. Как и прежде, клиентское устройство 104 коммуникативно связано с веб-службой 102 через сеть 106. Клиентское устройство 104 включает в себя модуль 114 уведомлений, который в этом случае иллюстрируется как часть операционной системы 108.

[0045] Клиентское устройство 104 также показано как включающее в себя очередь  
25 402, которая сконфигурирована для хранения уведомлений 116 для соответствующих приложений 404. Очередь 402, например, может быть сконфигурирована для хранения максимального количества уведомлений 116 для конкретных приложений и, таким образом, иллюстрируется на чертеже как "внутри" приложения 404 в очереди 402. Однако должно быть очевидно, что очередь 402 может быть сконфигурирована в  
30 различных формах.

[0046] Модуль 114 уведомления может управлять уведомлениями 116 с использованием очереди 402 различными способами. Например, уведомление может использовать метод "первым вошел/первым обслужен" (FIFO) для управления уведомлениями 116, метод "последним вошел/первым обслужен" и так далее. Например,  
35 модуль 114 уведомления может получить уведомление 406, которое указывает конкретный сегмент в очереди 402. Сегменты, например, могут описывать приоритет, с которым уведомления 116 должны отображаться как часть представления. Дополнительные методы управления могут использовать эту спецификацию, например, для перемещения предыдущих уведомлений 116 "вниз" по приоритету, так что  
40 уведомление, имеющее самый низкий приоритет, удаляется, чтобы заменить уведомление 408 в сегменте принятым уведомлением 406 и сохранить порядок в остальном, и так далее. Другие методы также возможны, например использование команды переопределения, как дополнительно описано в иллюстративных процедурах.

#### ИЛЛЮСТРАТИВНЫЕ ПРОЦЕДУРЫ

45 [0047] Следующее обсуждение описывает методы уведомления, которые могут быть реализованы с использованием ранее описанных систем и устройств. Аспекты каждой из процедур могут быть реализованы в аппаратных средствах, встроенном программном обеспечении или программном обеспечении или их комбинации. Процедуры

представлены в виде набора блоков, которые определяют операции, выполняемые одним или несколькими устройствами, и не обязательно ограничиваются порядком, показанным для выполнения операций соответствующими блоками. В частях последующего описания будут делаться ссылки на иллюстративную среду.

5 [0048] На фиг. 5 изображена процедура 500 в иллюстративной реализации, в которой теги используются для управления заменой уведомлений для отображения как части представления приложения. Принимается уведомление, которое должно быть  
10 отображено как часть представления приложения в пользовательском интерфейсе, причем уведомление связано с тегом (блок 502). Например, уведомление может быть принято в модуле 114 уведомлений клиентского устройства 104 из приложения, выполняемого на клиентском устройстве 104, из веб-службы и так далее. В другом примере модуль 124 уведомления веб-службы 102 может принимать уведомление из другой веб-службы и управлять хранением и распределением уведомления на клиентское устройство 104. Также возможно использование множества других примеров.

15 [0049] Тег уведомления сравнивается с одним или несколькими другими тегами, ассоциированными с одним или несколькими другими уведомлениями (блок 504). Модуль 114 уведомлений, например, может сравнить тег для уведомления для приложения к тегам других уведомлений для этого приложения, чтобы найти соответствие.

20 [0050] В ответ на определение того, что принятый тег соответствует по меньшей мере одному из других тегов, другое уведомление, ассоциированное с по меньшей мере одним из других тегов, заменяется принятым уведомлением для отображения как часть представления приложения в пользовательском интерфейсе (блок 506). Например,  
25 модуль 114 уведомления может сравнивать текстовые строки тегов в соответствующих уведомлениях, чтобы определить, совпадают ли они. Таким образом, в этом примере текстовые строки не должны быть "известными" в отношении того, "что они делают или представляют", модулю 114 уведомлений, чтобы содействовать функциональности замены. Таким образом, модуль 114 уведомлений может поддерживать разнообразные определения и идентификаторы тегов и при этом поддерживать функциональность  
30 замены, описанную в данном документе.

[0051] Принятое уведомление может затем отображаться в пользовательском интерфейсе клиентского устройства как часть представления приложения (этап 508). Отображение может быть выполнено в различных формах, таких как часть циклического отображения, как описано со ссылкой на фиг. 2, статическое отображение, отображение  
35 представление, которое может выбираться для запуска представляемого приложения (например, элемент мозаики в запускающем модуле приложения, выводимый посредством операционной системы 108) и так далее.

[0052] Кроме того, уведомление может быть использовано для запуска соответствующего приложения в контексте уведомления. Например, если пользователь  
40 устанавливает приложение «новости», выбор представления приложения будет запускать приложение в «домашнее» или «по умолчанию» состояние приложения. В примере приложения «новости» - на переднюю полосу газеты. Однако если приложение «новости» получает уведомление о землетрясении и пользователь выбирает это уведомление, то приложение «новости» может затем запускаться в контексте этого уведомления.  
45 Например, приложение «новости» может быть запущено непосредственно со статьи о землетрясении, а передняя полоса «пропускается». Также возможно множество других примеров.

[0053] На фиг. 6 изображена процедура 600 в примере реализации, в котором

используется очередь для управления уведомлениями для отображения как части представления приложения. Принимается множество уведомлений, которые должны быть отображены как часть представления приложений в пользовательском интерфейсе (блок 602). Как и прежде, уведомление может приниматься в модуле 114 уведомлений клиентского устройства 104 из приложения, выполняемого на клиентском устройстве 104, из веб-службы и так далее. В другом примере модуль 124 уведомлений веб-службы 102 может принимать уведомления из другой веб-службы и управлять хранением и распределением уведомления на клиентское устройство 104. Также возможно множество других примеров.

[0054] Одно или более уведомлений управляются с использованием очереди, которая конфигурирована, чтобы хранить до заданного количества уведомлений, которые должны быть сделаны доступными для отображения как часть представления приложения (этап 604). Уведомления, сохраненные в очереди, затем побуждаются для отображения как часть представления приложения в пользовательском интерфейсе (блок 606). Установленное количество, например, может устанавливаться разработчиком приложения 110, может устанавливаться инициатором уведомления 116, может предварительно определяться самим модулем 114 и так далее.

[0055] Модуль 114 уведомлений может использовать очередь 402, которая указывает объект, который используется для сбора уведомлений, в различных формах, например с использованием метода "первым вошел/первым обслужен", с использованием тегов, которые задают конкретные сегменты в очереди 402, и так далее. В дополнительных примерах очередь 402 может использовать методы установления приоритетов, например указание конкретного порядка, в котором уведомления 116 должны циклироваться на дисплее, и так далее, как описано со ссылкой на фиг. 12. Таким образом, очередь 402 может использоваться для управления количеством уведомлений, которые должны быть использованы для отображения как часть представления приложения, а также порядком, в котором отображаются уведомления.

[0056] Как отмечалось ранее, хотя эта функциональность описана как реализованная на клиенте, эта функциональность также может быть реализована с помощью различных других объектов, таких как модуль 124 уведомлений веб-службы 102, распределенных между модулями 114, 124 уведомлений, и так далее. Кроме того, эта функциональность может быть реализована модулем 114 уведомлений без выполнения представляемого приложения 110, тем самым сохраняя ресурсы обработки и батареи питания.

[0057] Фиг. 7 изображает процедуру 700 в иллюстративной реализации, в которой состояние выполнения приложения используется в качестве основы для определения того, какие уведомления должны быть отображены как часть представления приложения. Определяется состояние выполнения приложения, которое выполняется на клиентском устройстве (этап 702). Например, модуль 114 уведомлений может контролировать приложение 110, которое сконфигурировано для воспроизведения контента, например для приложения проигрывания музыки. Модуль 114 уведомлений может затем определить, что приложение 110 выполняется, чтобы воспроизводить конкретный элемент контента, например конкретную песню.

[0058] Выбирается уведомление для отображения как часть представления приложения в пользовательском интерфейсе клиентского устройства (блок 704). Уведомление отображается как часть представления приложения в пользовательском интерфейсе (блок 706). Продолжая предыдущий пример, модуль 114 уведомлений может определить название конкретного элемента контента, который выполняется. Это название затем может отображаться как часть представления приложения, например как представление

приложения, которые можно выбрать, чтобы запустить приложение.

[0059] Например, пользователь может выбрать список исполняемых произведений в приложении воспроизведения музыки клиентского устройства 104 и затем перейти из приложения, чтобы найти другие приложения, представляющие интерес. Приложение воспроизведения музыки может продолжать воспроизводить контент, например определенную песню, когда пользователь выполняет этот поиск. Чтобы обеспечить контекст для пользователя о состоянии приложения, уведомление может описывать это состояние выполнения, которое в данном случае может включать в себя воспроизводимый контент, например название, исполнитель, альбом и так далее. Также может использоваться множество других примеров, которые связаны с отображением уведомлений в течение представления приложений, которые описывают состояние выполнения приложения, и представление может выбираться для вывода пользовательского интерфейса этого приложения, например окна, которое включает в себя пользовательский интерфейс, переносимый «на передний план», так что пользователь может взаимодействовать с приложением.

[0060] Фиг. 8 изображает процедуру 800 в примере реализации, в которой уведомления отображаются в представлениях приложений, которые могут быть выбраны, чтобы запустить приложение, причем отображение циклируется, чтобы последовательно отображать уведомления. Принимается множество уведомлений, которые должны отображаться как часть представления приложения в пользовательском интерфейсе (блок 802). Как и прежде, уведомление может приниматься в модуле 114 уведомлений клиентского устройства 104 из приложения, выполняемого на клиентском устройстве 104, из веб-службы и так далее. В другом примере модуль 124 уведомлений веб-службы 102 может принимать уведомление из другой веб-службы и управлять хранением и распределением уведомления на клиентское устройство 104. Также возможно множество других примеров.

[0061] Отображение циклируется по множеству уведомлений как часть представления приложения в пользовательском интерфейсе, так что уведомления отображаются в последовательности (блок 804). Как описано выше со ссылкой на фиг. 2, например изображение 118 может отображаться, чтобы включать в себя уведомления, которые последовательно циклируются, одно за другим, для отображения разнообразной информации.

[0062] Таким образом, уведомления могут расширить объем информации, который может быть отображен без навигации "внутри" выполнения приложения. Как описано выше, это также может выполняться модулем 114 уведомлений без запуска приложения 110, тем самым сохраняя ресурсы батареи питания и/или обработки клиентского устройства 104, серверов веб-службы 102, вычислительных устройств третьих сторон и т.д.

[0063] В одной или более реализаций циклирование отображения множества уведомлений как части представления приложения в пользовательском интерфейсе приостанавливается в ответ на обнаружение ввода, который указывает взаимодействие с представлением приложения (блок 806), например, с самим приложением и/или другими приложениями, которые отображаются "вблизи". Например, отображение уведомлений может быть сконфигурировано для автоматического циклирования модулем 114 уведомлений без вмешательства пользователя.

[0064] Однако если обнаружен ввод, что предполагает вероятность взаимодействия с представлением, такой как жест, выбор с использованием устройства управления курсором и так далее, циклирование может быть приостановлено. Также возможны

различные другие реализации, например циклирование в ответ на обнаружение ввода (например, "зависание" курсора, жест и так далее), который указывает вероятное желание вызвать циклирование, которое может затем приостанавливается, когда обнаружено прекращение ввода. В другом примере обнаружение ввода (например, жест) может быть использовано для отображения каждого из уведомлений одновременно в пользовательском интерфейсе.

[0065] Фиг. 9 изображает процедуру 900 в иллюстративной реализации, в которой подмножества уведомлений циклируются для отображения как часть соответствующих представлений приложений. Принимается множество уведомлений, которые должны отображаться как часть соответствующих представлений из множества приложений в пользовательском интерфейсе (блок 902). Как и прежде, этот прием может быть выполнен различными объектами, и уведомления могут приниматься из различных источников, например приниматься модулем 114 уведомлений клиентского устройства 104.

[0066] Множество уведомлений отображается как часть представлений приложений в пользовательском интерфейсе, так что различные подмножества уведомлений для приложений циклируются в различные моменты времени (блок 904). Обращаясь к фиг. 1, например, шесть различных представлений приложений отображаются на устройстве отображения 112 клиентского устройства 104. В случае, в котором уведомления принимаются для отображения для каждого из представлений, может быть желательно циклировать различные подмножества уведомлений в различные моменты времени с тем, чтобы уменьшить "помехи" на дисплее. Например, уведомления о погоде и новостях могут циклироваться в первый момент времени, уведомления для ухода за домашними животными и предписания - во второй последующий момент времени и так далее. Таким образом, подмножества могут быть использованы для уменьшения отвлечений, которые могут быть вызваны, если большое количество уведомлений "перемещались" в любой момент времени.

[0067] Фиг. 10 изображает процедуру 100 в иллюстративной реализации, в которой манифест приложения используется для задания критериев, которые должны использоваться для управления отображением уведомлений как часть представления приложения. Манифест приложения рассматривается без выполнения приложения, чтобы найти критерии, заданные для отображения уведомлений как части представления приложения в пользовательском интерфейсе (блок 1002). Например, модуль 114 уведомлений может анализировать манифест, который должен использоваться для определения различных критерии и может делать это без запуска приложения. Примеры критериев включают в себя задание установленного количества, которое должно использоваться для очереди, должны ли поддерживаться теги (например, теги замены или переопределения), период времени, который должен использоваться для циклирования уведомлений, должны ли уведомления быть связаны с другими уведомлениями других приложений, чтобы сформировать предопределенные подмножества, источник уведомлений, должно ли циклирование выполняться автоматически и без вмешательства пользователя или в ответ на пользовательский ввод и так далее. Таким образом, модуль 114 уведомлений может позволить разработчикам приложения указывать, каким образом уведомления должны быть отображены как часть представлений приложения. Существует множество других примеров, которые также возможны, например использование вызова функции во время выполнения.

[0068] Представление приложения отображается в пользовательском интерфейсе в

соответствии с заданными критериями (блок 1004). Например, представление может отображаться как часть запускающего модуля приложения (например, рабочий стол), используемого операционной системой 108 в виде корневого уровня файловой системы. Возможно также множество других случаев, например, в различных точках в иерархии.

5 [0069] Фиг. 11 изображает процедуру 1100 в иллюстративной реализации, в котором веб-служба используется для управления предоставлением уведомлений клиентскому устройству для отображения. Принимается множество уведомлений на одном или более вычислительных устройствах веб-службы, которые должны быть отображены как часть представлений приложений в пользовательском интерфейсе клиентского устройства  
10 (блок 1102). Как описано выше, веб-служба 102 может принимать уведомления 116 для передачи на клиентское устройство 104 через сеть 104.

[0070] В ответ на определение того, что клиентское устройство не доступно для приема множества уведомлений, множеством уведомлений управляют с использованием очереди и критериев, которые в противном случае использовались бы клиентским  
15 устройством, чтобы определить, какие из множества уведомлений должны храниться в очереди (блок 1104). Например, веб-служба 102 может принимать сообщение, описывающее, какие приложения включены на клиентском устройстве 104, и как эти приложения управляются. В другом примере веб-служба 102 может основывать это управление на самих уведомлениях 116, например тегах, включенных в уведомления.  
20 Также возможно множество других примеров. Таким образом, модуль 124 уведомлений веб-службы 102 может эффективно управлять уведомлениями 116 для сбережения пространства памяти и снижения потребления пропускной способности сети 106.

[0071] Фиг. 12 изображает процедуру 1200 в иллюстративной реализации, в которой уведомлениям присвоен приоритет для отображения как части представления  
25 приложения. Как и прежде, принимается множество уведомлений, которые должны быть отображены как часть представления приложения в пользовательском интерфейсе, который позволяет сделать выбор для запуска приложения (блок 1202).

[0072] Определяется приоритет для упорядочения множества уведомлений (блок 1204). Модуль 114 уведомлений, например, может использовать метод "первым вошел/  
30 первым обслужен", чтобы управлять тем, какие уведомления должны храниться в очереди 402. В другом примере приоритеты могут быть назначены инициатором уведомления, например, как иерархическое значение, конкретный сегмент в очереди 402 и так далее. Приоритет может быть также определен модулем 114 уведомлений на основе разнообразия других критериев, например отображалось ли уже уведомление,  
35 или будет ли это в первый раз.

[0073] Множество уведомлений побуждается к отображению в соответствии с определенным приоритетом последовательно как часть представления приложения в пользовательском интерфейсе (блок 1206). Модуль 124 уведомлений, например, может передать уведомления 116 и/или приоритет на клиентское устройство 104. В другом  
40 случае модуль 114 уведомлений может определить приоритет и вызвать отображение уведомлений в рамках представления приложения 110. Также возможно разнообразие других примеров без отклонения от сущности и объема изобретения.

## ЗАКЛЮЧЕНИЕ

[0074] Хотя изобретение было описано в терминах, специфичных для структурных  
45 признаков и/или методологических действий, должно быть понятно, что изобретение, определенное в прилагаемой формуле изобретения, не обязательно ограничено конкретными признаками или описанными действиями. Напротив, конкретные признаки и действия раскрыты как примерные формы реализации заявленного изобретения.

## Формула изобретения

1. Способ управления уведомлениями, связанными с приложениями, реализуемый одним или более вычислительными устройствами, при этом способ содержит этапы, на которых:
- 5       отображают представление приложения в пользовательском интерфейсе; принимают одно или более уведомлений, которые относятся к приложению, для их отображения в качестве части представления приложения в пользовательском интерфейсе;
- 10       анализируют манифест приложения без исполнения приложения, чтобы найти критерии, задающие, как и когда должны отображаться упомянутые уведомления в качестве части представления приложения в пользовательском интерфейсе; и отображают представление приложения в пользовательском интерфейсе включающим в себя упомянутые одно или более уведомлений в пользовательском интерфейсе в соответствии с упомянутыми задающими критериями.
- 15       2. Способ по п. 1, в котором представление является выбираемым для запуска приложения.
3. Способ по п. 1, в котором уведомление является выбираемым как часть представления, чтобы запустить приложение в контексте уведомления.
- 20       4. Способ по п. 1, в котором отображение является выполняемым без исполнения приложения.
5. Способ по п. 1, в котором выполнение упомянутых анализа и отображения вызывается посредством исполнения операционной системы одним или более вычислительными устройствами.
- 25       6. Способ по п. 1, в котором упомянутые критерии задают, что должна быть использована очередь для управления уведомлениями, которые должны быть сделаны доступными для отображения в качестве части представления приложения.
7. Способ по п. 6, в котором упомянутые критерии задают максимальное число уведомлений, которые должны быть сохранены с использованием очереди.
- 30       8. Способ по п. 1, в котором упомянутые критерии задают, должно ли быть разрешено отображение уведомления в качестве части представления приложения.
9. Способ по п. 1, в котором упомянутые критерии задают, разрешено ли или запрещено циклирование отображения уведомлений в качестве части представления приложения в пользовательском интерфейсе, так что множество уведомлений
- 35       отображаются в последовательности.
10. Способ управления уведомлениями, связанными с приложениями, содержащий этапы, на которых:
- 40       принимают на одном или более вычислительных устройствах веб-службы множество уведомлений, которые должны отображаться как часть циклически переключаемого отображения в пределах соответственных представлений приложений в пользовательском интерфейсе клиентского устройства:
- 45       в качестве реакции на определение того, что клиентское устройство не является доступным для приема этого множества уведомлений, выполняют управление данным множеством уведомлений с использованием очереди и критериев, которые в противном случае использовались бы клиентским устройством, для определения того, какие из упомянутого множества уведомлений должны быть сохранены в очереди для последующей передачи на клиентское устройство, причем критерии, которые в противном случае использовались бы клиентским устройством, определяются путем



анализа манифеста, задающего, как и когда должны отображаться уведомления.

11. Способ по п. 10, в котором упомянутые критерии определяются на основе приложений, которые установлены на клиентском устройстве.

12. Способ по п. 10, в котором упомянутые критерии определяются путем анализа по меньшей мере одного манифеста установки приложения, которое установлено на клиентском устройстве, или использования вызова системной функции во время исполнения.

13. Способ по п. 10, в котором упомянутые критерии задаются через сообщение, принимаемое на веб-службе от клиентского приложения.

14. Способ управления уведомлениями, связанными с приложениями, реализуемый одним или более вычислительными устройствами, при этом способ содержит этапы, на которых:

принимают множество уведомлений, которые должны отображаться как часть представления приложения в пользовательском интерфейсе, каковое представление является выбираемым для запуска приложения, при этом в представлении приложения имеется множество фреймов для отображения графического изображения и текста;

анализируют манифест приложения для нахождения критериев, задающих, как и когда должно отображаться множество уведомлений в качестве части представления приложения в пользовательском интерфейсе;

определяют приоритет, в котором следует упорядочивать множество уведомлений; и

обеспечивают отображение множества уведомлений в соответствии с упомянутым определенным приоритетом последовательно, как часть циклирования отображения в пределах представления приложения в пользовательском интерфейсе.

15. Способ по п. 14, в котором приоритет основывается, по меньшей мере частично, на том, отображалось ли уже соответствующее уведомление.

16. Способ по п. 14, в котором приоритет основывается, по меньшей мере частично, на значениях, присваиваемых инициатором соответствующего уведомления.

17. Способ по п. 14, в котором приоритет основывается, по меньшей мере частично, на значениях, присваиваемых соответствующим уведомлениям, которые показывают конкретное место в очереди.

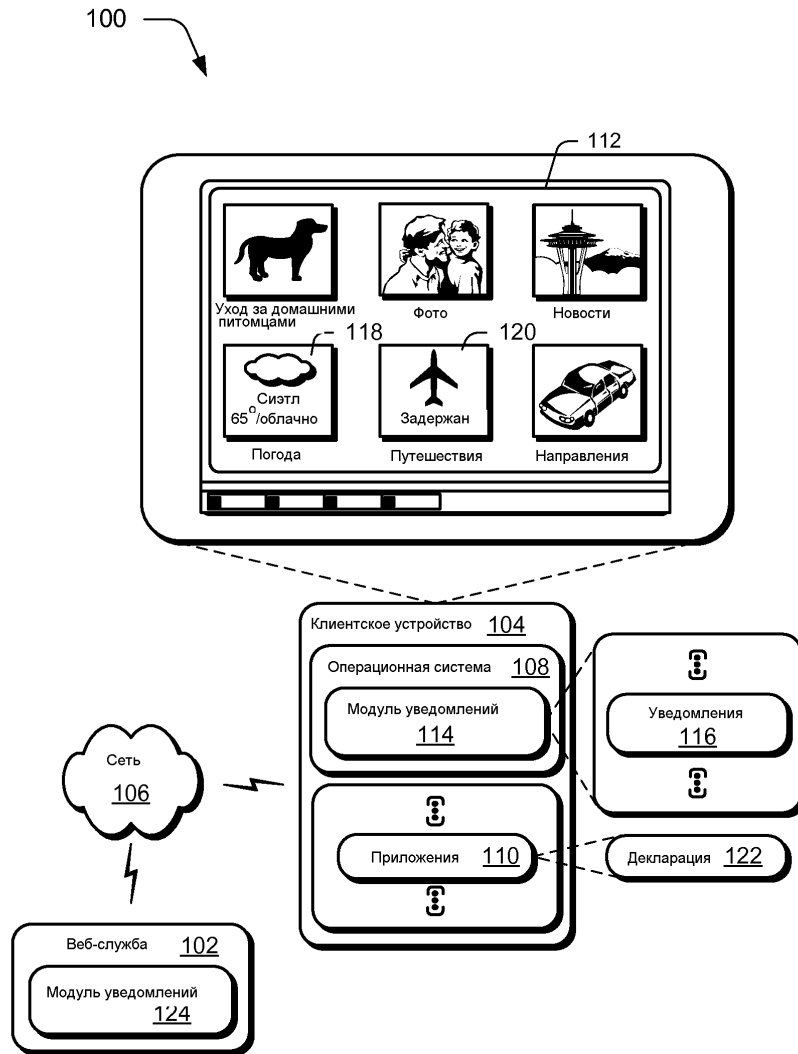
18. Способ по п. 14, в котором упомянутые этапы приема, анализа, определения и обеспечения выполняются одним или более вычислительными устройствами как часть веб-службы.

19. Способ по п. 14, в котором упомянутые этапы приема, анализа, определения и обеспечения выполняются операционной системой одного или более вычислительных устройств, сконфигурированных в качестве клиентского устройства.

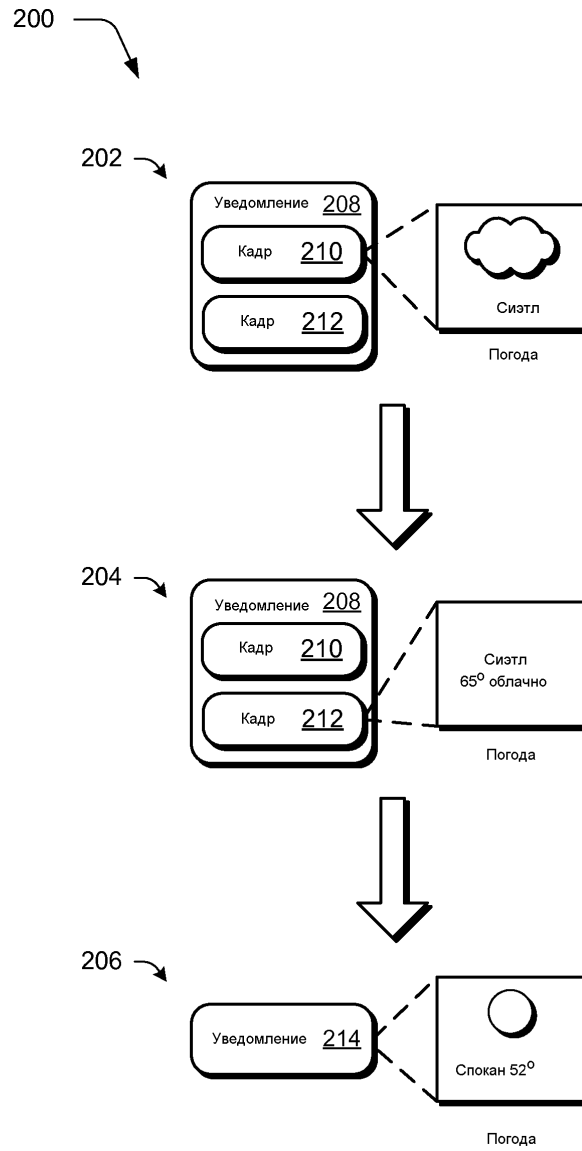
20. Способ по п. 14, в котором упомянутые этапы приема, анализа, определения и обеспечения выполняются без исполнения приложения.

40

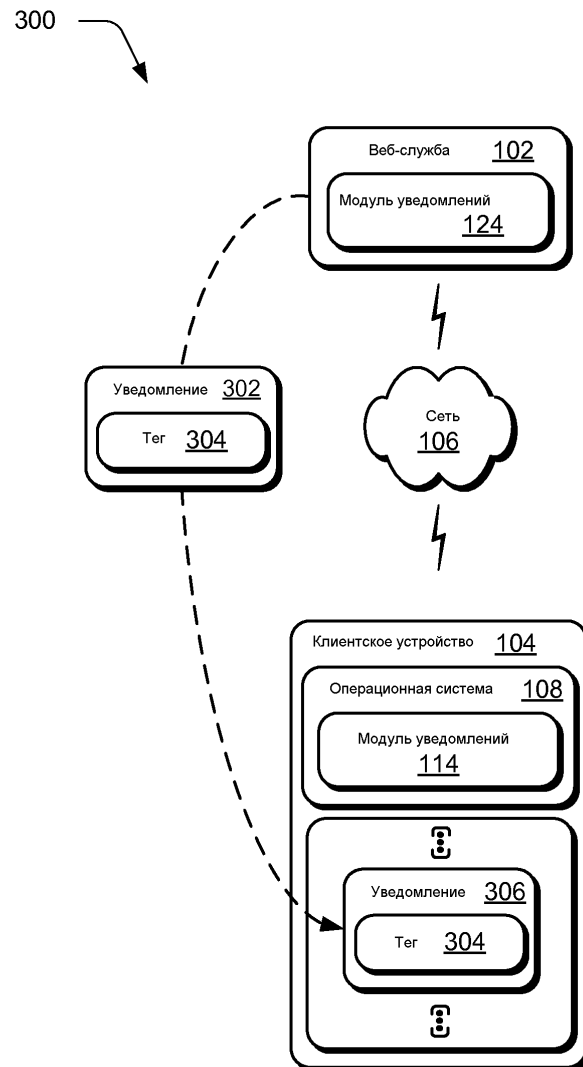
45



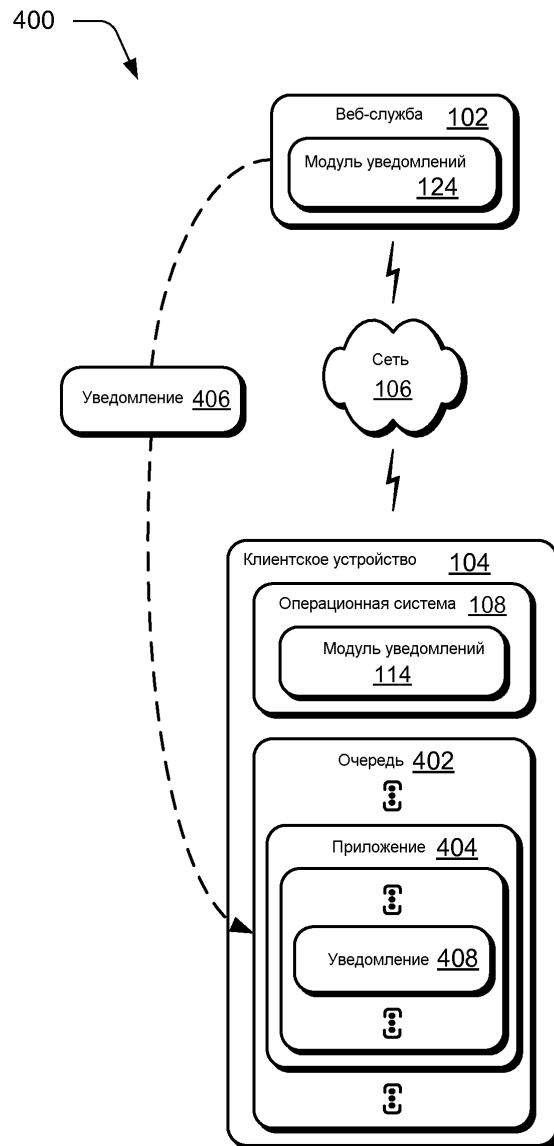
ФИГ. 1



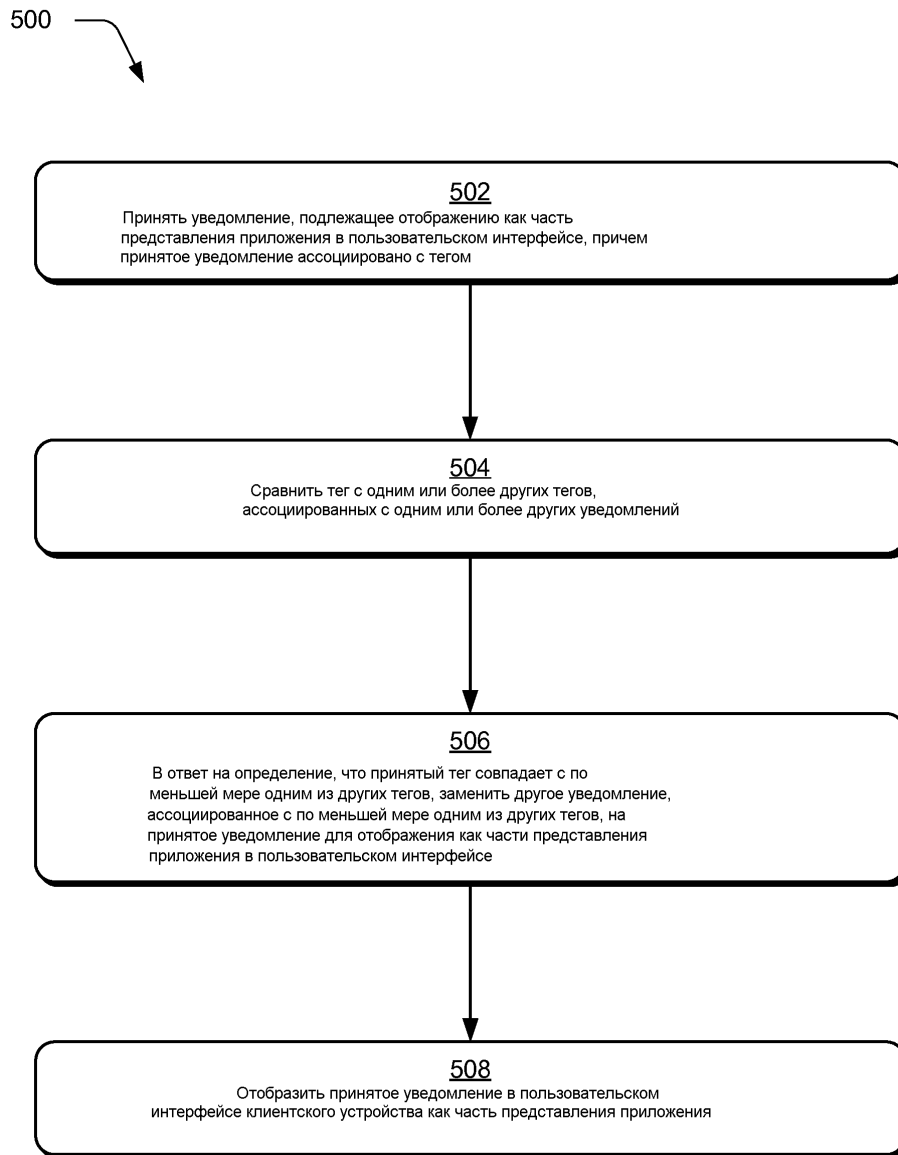
ФИГ.2



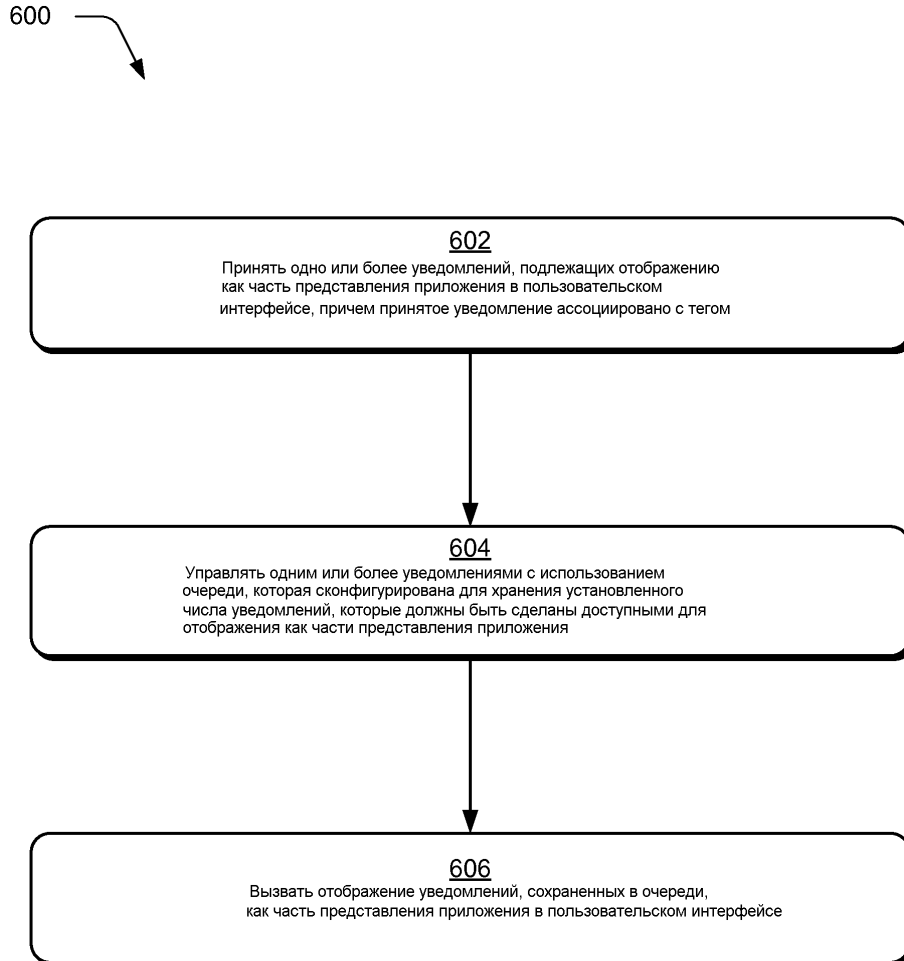
ФИГ.3



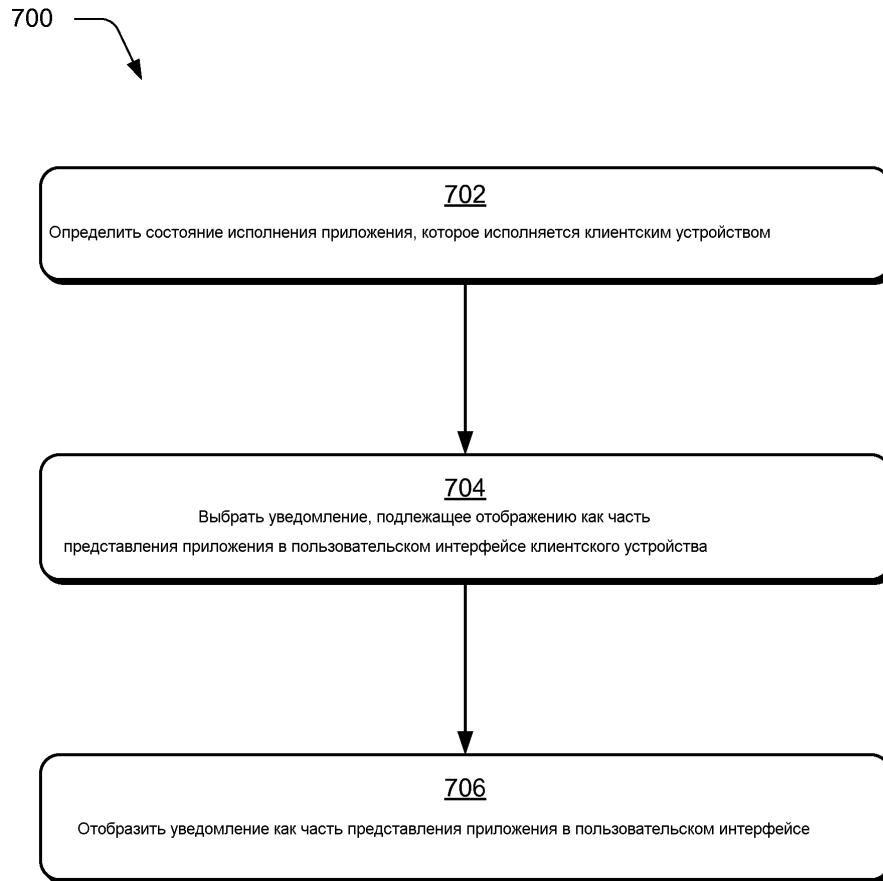
ФИГ. 4



ФИГ.5

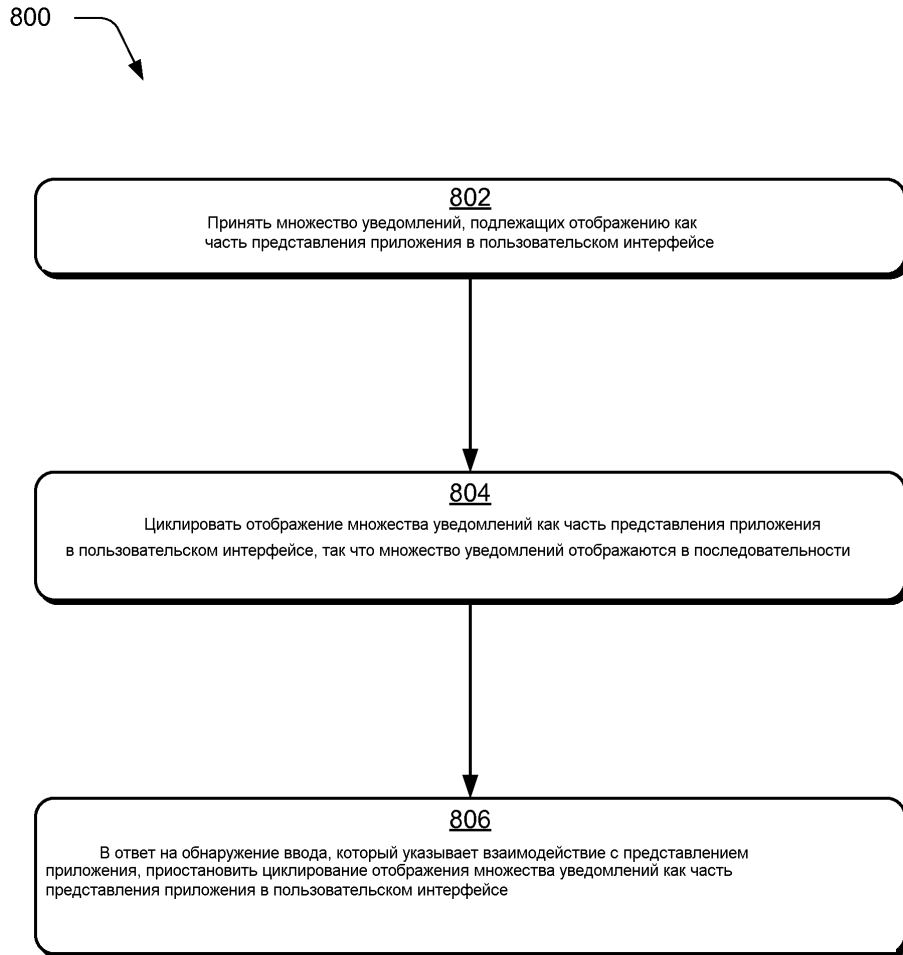


ФИГ.6

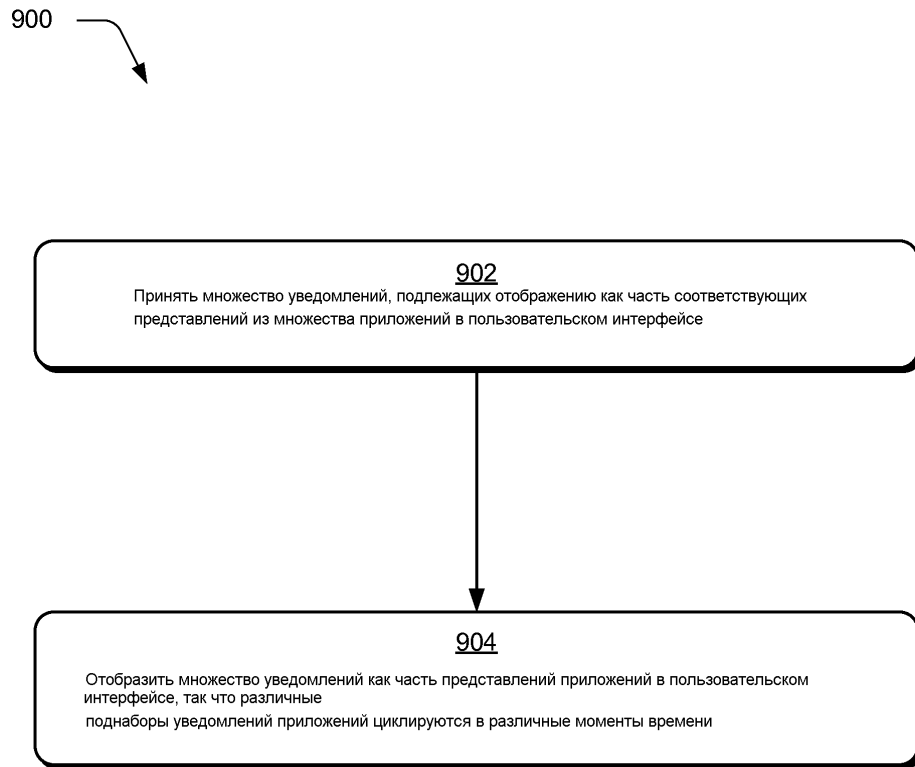


ФИГ.7

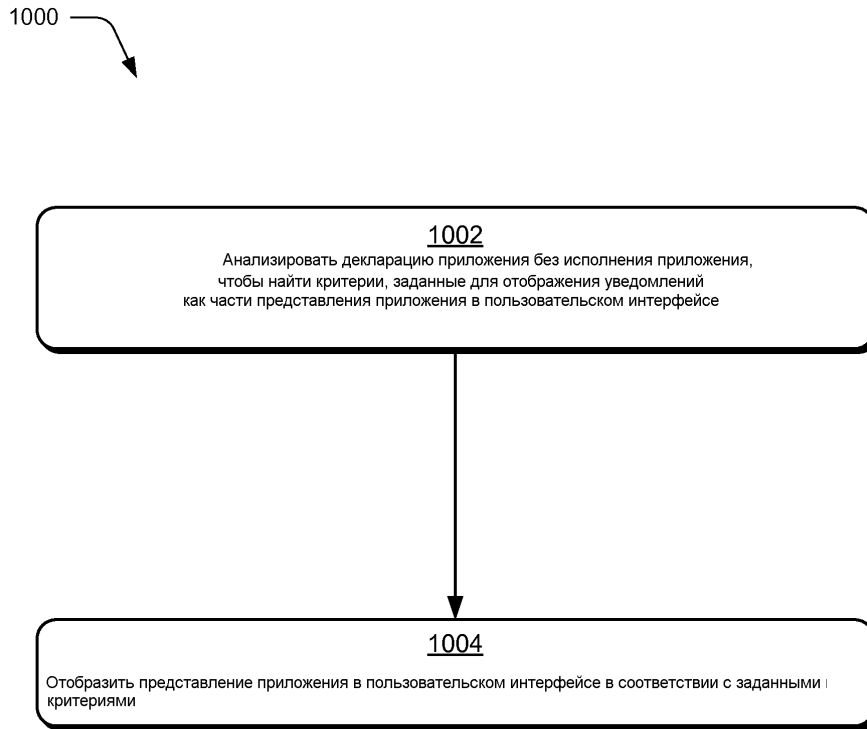




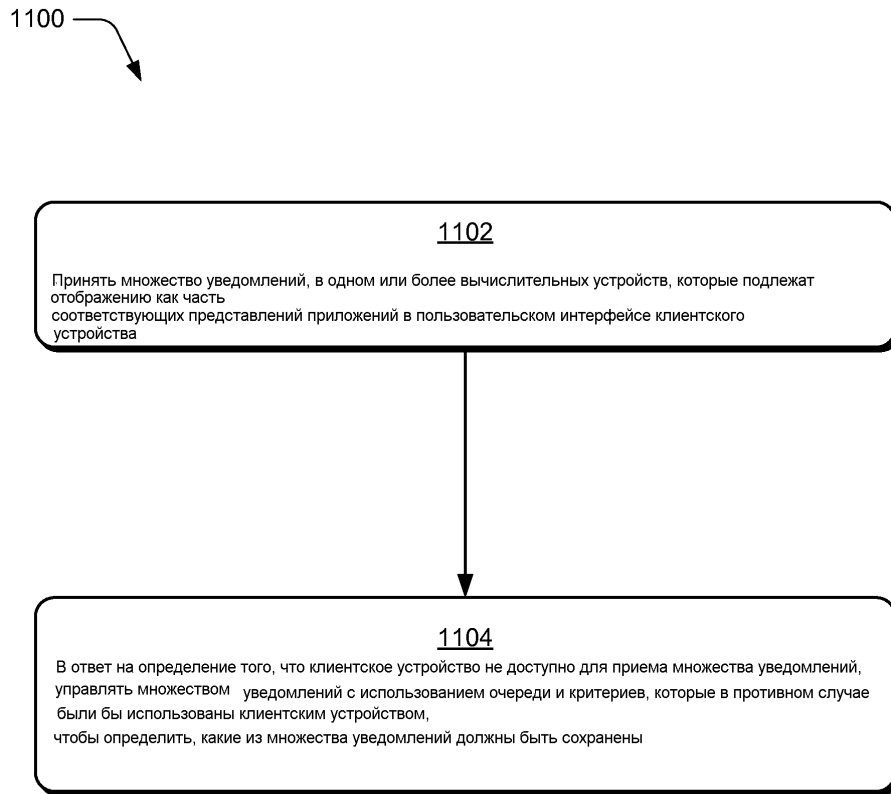
ФИГ. 8



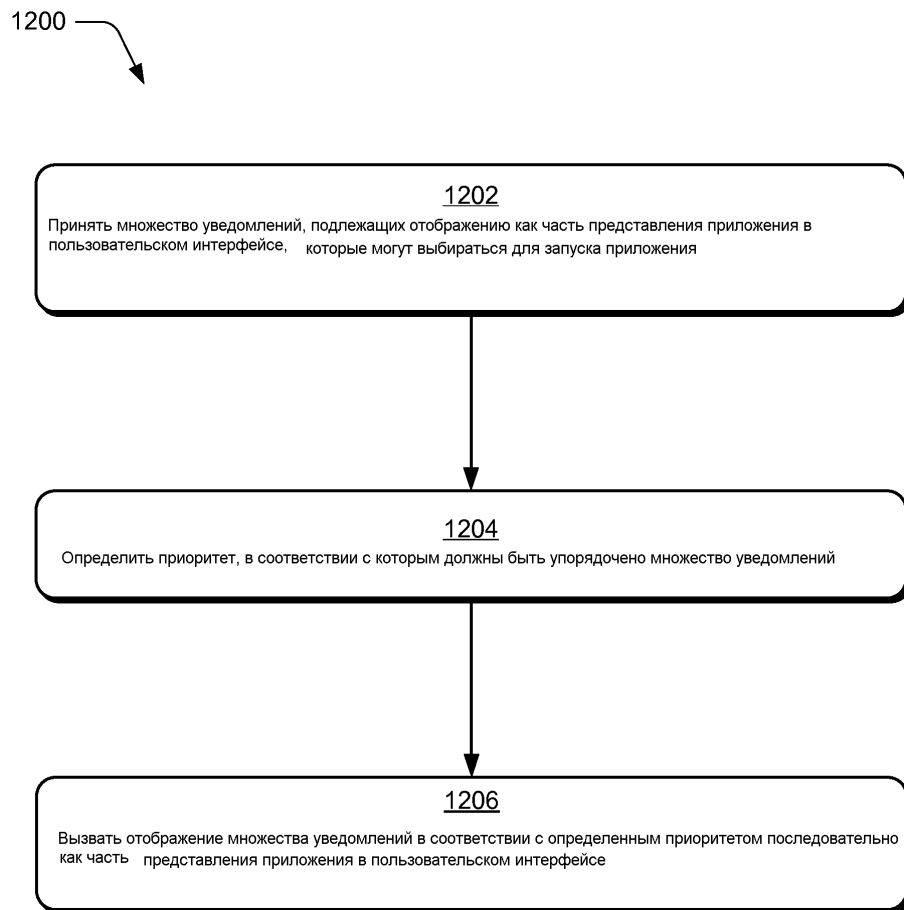
ФИГ.9



ФИГ. 10



ФИГ. 11



ФИГ. 12