



**ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ**

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: 2017102351, 26.06.2014

(24) Дата начала отсчета срока действия патента:
26.06.2014

Дата регистрации:
15.12.2017

Приоритет(ы):

(22) Дата подачи заявки: 26.06.2014

(45) Опубликовано: 15.12.2017 Бюл. № 35

(85) Дата начала рассмотрения заявки РСТ на национальной фазе: 26.01.2017

(86) Заявка РСТ:
CN 2014/080801 (26.06.2014)

(87) Публикация заявки РСТ:
WO 2015/196405 (30.12.2015)

Адрес для переписки:
129090, Москва, ул. Б.Спасская, 25, строение 3,
ООО "Юридическая фирма Городиский и
Партнеры"

(72) Автор(ы):

**МЭН Бипин (CN),
ИСТХЭМ Майкл (US),
СЮЙ Хой (US),
ЧЖОУ Сяобо (CN)**

(73) Патентообладатель(и):
ГУГЛ ИНК. (US)

(56) Список документов, цитированных в отчете о поиске: US 2011/0055314 A1, 03.03.2011. US 8346755 B1, 01.01.2013. US 8738686 B2, 27.05.2014. US 2012/0036264 A1, 09.02.2012. RU 2500026 C2, 27.11.2013.

(54) ОПТИМИЗИРОВАННЫЙ ПРОЦЕСС ВОСПРОИЗВЕДЕНИЯ БРАУЗЕРА

(57) Реферат:

Изобретение относится к области воспроизведения веб-страниц в браузере. Техническим результатом является идентификация необязательных встраиваемых ресурсов веб-страниц и обеспечение в подсистеме воспроизведения веб-страниц возможности пропускать такие ресурсы, что сокращает время воспроизведения веб-страниц. Раскрытый способ включает в себя идентификацию URL-шаблона, общего для множества URL, хранящихся в записях выборки, и выбор выборочной совокупности URL из этого множества. Этот способ также включает в себя, для каждого URL в выборочной совокупности, определение того, является ли URL необязательным, посредством

генерации первого результата воспроизведения, с использованием контента для URL, и второго результата воспроизведения, без использования контента для URL, и вычисления оценки подобия для URL посредством сравнения первого результата воспроизведения и второго результата воспроизведения, причем URL является необязательным, когда оценка подобия является большей, чем порог подобия. Этот способ может также включать в себя запоминание URL-шаблона в хранилище данных шаблонов необязательных ресурсов, когда большинство URL в выборочной совокупности являются необязательными. 3 н. и 13 з.п. ф-лы, 7 ил.



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(12) **ABSTRACT OF INVENTION**

(21)(22) Application: **2017102351, 26.06.2014**

(24) Effective date for property rights:
26.06.2014

Registration date:
15.12.2017

Priority:

(22) Date of filing: **26.06.2014**

(45) Date of publication: **15.12.2017** Bull. № 35

(85) Commencement of national phase: **26.01.2017**

(86) PCT application:
CN 2014/080801 (26.06.2014)

(87) PCT publication:
WO 2015/196405 (30.12.2015)

Mail address:
**129090, Moskva, ul. B.Spasskaya, 25, stroenie 3,
OOO "Yuridicheskaya firma Gorodisskij i Partnery"**

(72) Inventor(s):
**MEN Bipin (CN),
ISTKHEM Majkl (US),
SYUJ Khoj (US),
CHZHOU Syaobo (CN)**

(73) Proprietor(s):
GUGL INK. (US)

(54) **OPTIMIZED BROWSER REPRODUCTION PROCESS**

(57) Abstract:

FIELD: information technology.

SUBSTANCE: disclosed method includes identifying a URL pattern common to the plurality of URLs stored in the sampling records, and selecting a selectable plurality of URLs from this set. The method includes, for each URL in the sample set, determining whether the URL is optional, by generating the first playback result, using the content for the URL, and the second playback result, without using the content for

the URL, and calculating the similarity estimate for the URL by comparing the first playback result and the second playback result. URL is optional when the similarity score is greater than the similarity threshold. This method can also include storing the URL pattern in the data store of optional resource templates when most URLs in the sample set are optional.

EFFECT: reducing the playback time of web pages.
16 cl, 7 dwg

C 1
6
2
6
3
8
7
2
6
R U

R U
2
6
3
8
7
2
6
C 1

УРОВЕНЬ ТЕХНИКИ

[0001] Многие веб-страницы создаются динамически, что означает, что конечная страница, просматриваемая пользователем веб-браузера, генерируется по меньшей мере частично процессом воспроизведения браузера. Встраиваемые объекты, такие как сценарий, изображения и таблицы стилей, часто встроены в веб-страницу и могут влиять на контент и внешний вид воспроизводимой веб-страницы или могут быть включены для других целей, не связанных с генерацией динамического контента. Например, код Google Analytics JavaScript может быть встраиваемым объектом, который помогает отслеживать трафик для веб-сайта, но не влияет на контент. Встраиваемые объекты сами по себе могут включать в себя дополнительные встраиваемые объекты. Подобным образом, некоторые встраиваемые объекты могут быть скрытыми и невидимыми для пользователя. Веб-страница может иметь сотни встраиваемых объектов, и подсистема воспроизведения браузера тратит много времени на воспроизведение веб-страницы для пользователя, который тратит время на ожидание выборки браузером этих встраиваемых ресурсов.

СУЩНОСТЬ ИЗОБРЕТЕНИЯ

[0002] Реализации идентифицируют необязательные встраиваемые ресурсы веб-страниц и обеспечивают подсистеме воспроизведения возможность пропускать такие ресурсы, что улучшает время воспроизведения веб-страниц. Необязательные ресурсы представляют собой встраиваемые объекты, которые могут быть пропущены без значительного или существенного воздействия на контент воспроизводимой веб-страницы. Браузер может пропустить такие необязательные встраиваемые ресурсы во время процесса воспроизведения, что приводит к меньшему времени загрузки.

[0003] В одном аспекте, система включает в себя по меньшей мере один процессор и память, запоминающую команды, которые, при исполнении по меньшей мере одним процессором, обеспечивают выполнение системой операций. Операции могут включать в себя идентификацию URL-шаблона, общего для множества URL, хранящихся в записях выборки, и выбор выборочной совокупности URL из множества URL. Операции могут также включать в себя определение, для каждого URL в выборочной совокупности, является ли URL необязательным, посредством воспроизведения первого результата воспроизведения для объекта встраивания URL, с использованием контента для URL, воспроизведения второго результата воспроизведения для объекта встраивания, без использования контента для URL, и вычисления оценки подобия для URL посредством сравнения первого результата воспроизведения и второго результата воспроизведения. Этот способ также включает в себя определение того, что URL является необязательным, когда оценка подобия является большей, чем порог подобия, и, когда заданное количество URL в выборочной совокупности являются необязательными, запоминание URL-шаблона в хранилище данных шаблонов необязательных ресурсов.

[0004] Одна или несколько реализаций изобретения, описанного здесь, может включать в себя один или несколько из следующих признаков. Например, операции могут также включать в себя прием запроса на контент запрашиваемого встраиваемого ресурса, определение того, соответствует ли запрашиваемый встраиваемый ресурс шаблону в хранилище данных, и возвращение указания на то, что запрашиваемый встраиваемый ресурс является необязательным, если запрашиваемый встраиваемый ресурс соответствует шаблону. В некоторых реализациях, указание может быть ошибкой «URL не найден». В качестве другого примера, идентификация URL-шаблона, общего для множества URL, хранящихся в записях выборки, может включать в себя генерацию группового URL для соответствующих URL, хранящихся в записях выборки, посредством

удаления по меньшей мере участка строки запроса из URL, и кластеризацию URL посредством группового URL. В качестве другого примера, идентификация URL-шаблона, общего для множества URL, хранящихся в записях выборки, может включать в себя генерацию группового URL для соответствующих URL, хранящихся в записях
5 выборки, посредством удаления по меньшей мере участка строки запроса из URL, кластеризацию URL посредством группового URL, и выбор группового URL кластера с наибольшим количеством элементов в качестве URL-шаблона. В качестве другого примера, вычисление оценки подобия может включать в себя определение самой длинной общей последовательности для дерева DOM первого результата воспроизведения и
10 дерева DOM второго результата воспроизведения, и использование самой длинной общей последовательности для определения оценки подобия.

[0005] В другом аспекте, реализуемый на компьютере способ включает в себя идентификацию URL-шаблона, общего для множества URL, хранящихся в записях выборки, и выбор, с использованием по меньшей мере одного процессора, выборочной
15 совокупности URL из множества URL. Способ может также включать в себя определение, для каждого URL в выборочной совокупности, является ли URL необязательным, посредством генерации первого результата воспроизведения для объекта встраивания URL с использованием контента для URL, генерации второго результата воспроизведения для объекта встраивания без использования контента для
20 URL, и вычисления оценки подобия для URL посредством сравнения первого результата воспроизведения и второго результата воспроизведения, причем URL является необязательным, когда оценка подобия является большей, чем порог подобия. Способ может также включать в себя запоминание URL-шаблона в хранилище данных шаблонов необязательных ресурсов, когда большинство URL в выборочной совокупности
25 являются необязательными.

[0006] Одна или несколько реализаций изобретения, описанного здесь, может включать в себя один или несколько из следующих признаков. Например, способ может также включать в себя определение того, что количество, представленное множеством URL, превышает размер порога, перед определением того, являются ли URL в
30 выборочной совокупности необязательными. В качестве другого примера, способ может включать в себя запоминание URL-шаблона в хранилище данных шаблонов необязательных ресурсов, когда все URL в выборочной совокупности являются необязательными. В качестве другого примера, идентификация URL-шаблона, общего для множества URL в записях выборки, может включать в себя, для соответствующих
35 URL в записях выборки, генерацию группового URL для URL в записи выборки, посредством удаления по меньшей мере участка строки запроса из URL, кластеризации посредством группового URL, и выбора группового URL кластера, когда количество элементов в кластере соответствует порогу.

[0007] В другом аспекте, способ включает в себя прием запроса на воспроизведение
40 веб-страницы, идентификацию по меньшей мере одного встраиваемого ресурса в веб-странице, которому необходима выборка, определение того, что встраиваемый ресурс является необязательным ресурсом, и воспроизведение веб-страницы, как если бы встраиваемый ресурс был недоступным, без выборки контента для встраиваемого ресурса.

[0008] Одна или несколько реализаций предмета изобретения, описанного здесь, может включать в себя один или несколько из следующих признаков. Например, определение того, что встраиваемый ресурс является необязательным ресурсом, может включать в себя определение того, что URL для встраиваемого ресурса соответствует

шаблону в хранилище данных необязательных ресурсов. В качестве другого примера, способ может также включать в себя прием хранилища данных шаблонов для необязательных ресурсов от услуги, причем определение того, что встраиваемый ресурс является необязательным ресурсом, включает в себя определение того, что URL для встраиваемого ресурса соответствует шаблону в хранилище данных. В качестве другого примера, определение того, что встраиваемый ресурс является необязательным ресурсом, может включать в себя перезапись URL для встраиваемого ресурса, посредством удаления участка строки запроса из URL, и установление соответствия перезаписанного URL шаблону в хранилище данных необязательных ресурсов, причем встраиваемый ресурс является необязательным, когда перезаписанный URL соответствует этому шаблону. В качестве другого примера, определение того, что встраиваемый ресурс является необязательным ресурсом, может включать в себя перезапись URL для встраиваемого ресурса, посредством удаления по меньшей мере участка строки запроса из URL, и установление соответствия перезаписанного URL шаблону в хранилище данных необязательных ресурсов, причем встраиваемый ресурс является необязательным, когда перезаписанный URL соответствует этому шаблону.

[0009] В другом аспекте, долговременный машиночитаемый носитель (среда) может включать в себя инструкции, исполняемые по меньшей мере одним процессором, образованном на подложке, которые обеспечивают выполнение вычислительной системой одного или нескольких способов, описанных выше.

[0010] Одна или несколько реализаций изобретения, описанного здесь, может быть реализована таким образом, чтобы она обеспечивала одно или несколько из следующих преимуществ. В качестве одного примера, пропуск необязательных встраиваемых ресурсов обеспечивает подсистеме воспроизведения возможность более быстрого воспроизведения веб-страницы без потери контента. Браузер может не только пропускать выборку необязательного встраиваемого ресурса, он может также пропускать выборку объектов, встраиваемых в пропущенный необязательный встраиваемый ресурс. Таким образом, описанные реализации сохраняют пропускную способность сети и уменьшают потребление аккумуляторной батареи. Поскольку веб-страницы могут включать в себя сотни встраиваемых объектов, такая оптимизация может быть существенной.

КРАТКОЕ ОПИСАНИЕ ЧЕРТЕЖЕЙ

[0011] Фиг. 1 показывает иллюстративную систему, согласно раскрытому изобретению.

[0012] Фиг. 2 является блок-схемой результата воспроизведения для воспроизводимой веб-страницы, согласно одной реализации.

[0013] Фиг. 3 является блок-схемой последовательности операций, показывающей иллюстративный процесс для идентификации шаблонов необязательных ресурсов, согласно одной реализации.

[0014] Фиг. 4 является блок-схемой последовательности операций, показывающей иллюстративный процесс для определения того, являются ли подобными два результата воспроизведения, согласно одной реализации.

[0015] Фиг. 5 является блок-схемой последовательности операций, показывающей иллюстративный процесс для обеспечения информации для оптимизированного воспроизведения, согласно одной реализации.

[0016] Фиг. 6 показывает пример вычислительного устройства, которое может быть использовано для реализации описанных технологий.

[0017] Фиг. 7 показывает пример распределенного вычислительного устройства,

которое может быть использовано для реализации описанных технологий.

ПОДРОБНОЕ ОПИСАНИЕ

[0018] Раскрытые реализации идентифицируют необязательные встраиваемые объекты, или встраиваемые объекты, которые незначительно воздействуют на контент воспроизводимой веб-страницы, что обеспечивает подсистеме воспроизведения браузера возможность воспроизведения таких веб-страниц без выборки необязательных встраиваемых объектов. Это уменьшает время воспроизведения и нагрузку на систему выборки, что является важным для вычислительных устройств с ограниченными ресурсами, таких как смартфоны и планшеты. Встраиваемый ресурс может быть подходящим для оптимизированного воспроизведения, когда результат воспроизведения веб-страницы как объекта встраивания, воспроизводимый с необязательным встраиваемым объектом, подобен результату воспроизведения веб-страницы как объекта встраивания, воспроизводимой без необязательного встраиваемого объекта. Подобие может быть определено по подобию маркеров на странице, по подобию внешних связей, по подобию компоновки страницы и т.д. В некоторых реализациях, система может использовать измерение самой длинной общей последовательности для определения того, являются ли подобными результаты воспроизведения. В некоторых реализациях, когда основные компоненты моментальных снимков имеют оценку подобия, которая соответствует порогу подобия, эти моментальные снимки могут считаться подобными. Если система определяет, что встраиваемый ресурс является необязательным, то система может запомнить встраиваемый ресурс (например, его унифицированный указатель информационного ресурса (URL)) в списке необязательных встраиваемых ресурсов. В некоторых реализациях, система может агрегировать URL необязательных встраиваемых ресурсов для идентификации шаблона, который может быть использован для идентификации других необязательных встраиваемых ресурсов.

[0019] Когда браузер начинает воспроизведение веб-страницы, он может спросить у некоторой услуги, является ли какой-либо из встраиваемых ресурсов необязательным. В некоторых реализациях, это может быть выполнено в виде части запроса выборки. Услуга может установить соответствие URL запрашиваемых встраиваемых ресурсов в хранилище данных необязательных встраиваемых объектов, либо посредством установления точного соответствия, либо посредством установления соответствия шаблону в хранилище данных. Если услуга находит соответствие, то услуга может указать браузеру, какие встраиваемые ресурсы являются необязательными, и браузер может воспроизвести веб-страницу без выборки этих встраиваемых ресурсов. Такое оптимизированное воспроизведение браузера делает процесс воспроизведения браузера гораздо более эффективным с точки зрения ресурсов. В некоторых реализациях, хранилище данных необязательных встраиваемых ресурсов может различать типы браузеров, например, мобильные браузеры и браузеры для настольных компьютеров.

[0020] Фиг. 1 является блок-схемой системы согласно иллюстративной реализации. Система 100 может быть использована для идентификации необязательных встраиваемых ресурсов и шаблонов необязательных ресурсов и использования шаблонов для оптимизации процесса воспроизведения браузера. Система 100 может включать в себя вычислительные устройства, которые принимают форму некоторого количества разных устройств. Например, система 100 может включать в себя вычислительные устройства, которые являются примерами вычислительного устройства 600, показанного на фиг. 6, и вычислительного устройства 700, показанного на фиг. 7.

[0021] Система 100 может включать в себя систему 110 пакетного воспроизведения. Система 110 пакетного воспроизведения может быть стандартным сервером, группой

таких серверов, системой клиент-сервер, или системой стоечных серверов.

Дополнительно, система 110 пакетного воспроизведения может быть реализована в персональном компьютере. Система 110 пакетного воспроизведения может включать в себя один или несколько процессоров, выполненных с возможностью исполнения одной или нескольких исполняемых машинами команд или фрагментов программного обеспечения, аппаратно-программного обеспечения, или их комбинации, хранящихся в памяти. Например, система 110 пакетного воспроизведения может включать в себя подсистемы 120 воспроизведения, услугу 122 выборки, подсистема 124 автоматического веб-сканирования, и подсистема 126 идентификации необязательных ресурсов. Система 110 пакетного воспроизведения может создавать воспроизводимую веб-страницу в пакетном режиме, например, в виде части процесса индексирования, и, в этом процессе, может создавать записи 132 выборки, которые система 110 пакетного воспроизведения может использовать для генерации шаблонов 130 необязательных ресурсов.

[0022] Нижестоящий процесс может запросить, чтобы система 110 пакетного воспроизведения воспроизвела веб-страницу в пакетном режиме. В некоторых реализациях, система 110 пакетного воспроизведения может включать в себя нижестоящий процесс, такой как подсистема 126 идентификации необязательных ресурсов или подсистема индексирования (не показан). В некоторых реализациях, нижестоящий процесс может быть выполнен на вычислительном устройстве, которое отличается от системы 110 пакетного воспроизведения. Например, нижестоящий процесс может быть подсистемой индексирования или рекламной платформой, которая отправляет запросы к системе пакетного воспроизведения посредством вызова удаленной процедуры. Нижестоящий процесс может запросить, чтобы один или несколько подсистем 120 пакетного воспроизведения сгенерировали результат воспроизведения конкретной веб-страницы. Каждый подсистема 120 пакетного воспроизведения может быть выполнен с возможностью эмуляции рендерера для персонального веб-браузера, но с оптимизациями для пакетного воспроизведения. Система 110 пакетного воспроизведения может включать в себя тысячи подсистем 120 пакетного воспроизведения и может использовать балансировку нагрузки для выбора одного из подсистем 120 пакетного воспроизведения в ответ на конкретный запрос. Запрашиваемая веб-страница может включать в себя встраиваемые объекты, такие как таблицы стилей, объекты JavaScript, изображения и т.д. Подсистемы 120 пакетного воспроизведения могут запрашивать контент для встраиваемых объектов с использованием услуги 122 выборки.

[0023] Услуга 122 выборки может определить, какие встраиваемые ресурсы должны быть выбраны с хост-сервера 190, какие встраиваемые ресурсы могут быть возвращены из кэш-памяти, и какие ресурсы не нужно возвращать. Хост-серверы 190 могут быть вычислительными устройствами любого типа, доступными через интернет, которые служат в качестве хостов для одной или нескольких веб-страниц или ресурсов, встраиваемых в одну или несколько веб-страниц. Если встраиваемый ресурс должен быть выбран, услуга 122 выборки может запросить контент для встраиваемого объекта через подсистема 124 автоматического веб-сканирования с использованием общепринятых технологий. Нижестоящий процесс, такой как подсистема индексирования, может также запрашивать контент из серверов 190 через подсистема 124 автоматического веб-сканирования. В результате запросов выборки через подсистема 124 автоматического веб-сканирования, система 110 пакетного воспроизведения может сгенерировать записи 132 выборки. Записи 132 выборки могут включать в себя информацию о том, какие веб-страницы и встраиваемые объекты

запрашивались и извлекались из хост-серверов 190. Записи 132 выборки могут также включать в себя дополнительную информацию, например, время запроса, веб-страницу как объект встраивания, при наличии и т.д.

5 [0024] Подсистема 124 автоматического веб-сканирования, подсистемы 120 пакетного воспроизведения, и услуга 122 выборки работают совместно для эффективного воспроизведения большого количества веб-страниц, таких как веб-страницы, которые могут быть найдены во всемирной паутине. Воспроизведение веб-страницы является результатом воспроизведения, который включает в себя различные информационные элементы, доступные и, иначе, недоступные для нижестоящего процесса запрашивания.

10 Подсистема 126 идентификации необязательных ресурсов может использовать подсистемы 120 пакетного воспроизведения для генерации шаблонов 130 необязательных ресурсов. Подсистема 126 идентификации необязательных ресурсов может запускаться периодически (например, ежедневно, два раза в неделю и т.д.), с использованием записей 132 выборки в качестве входных данных. Подсистема 126 идентификации

15 необязательных ресурсов может анализировать записи 132 выборки и генерировать шаблоны для URL, выбранных в предшествующем периоде. Например, подсистема 126 идентификации необязательных ресурсов может вырезать строку запроса из URL в записях выборки, при генерации группового URL для каждого соответствующего URL. Строка запроса может быть любыми символами после вопросительного знака (?) в

20 URL. В некоторых реализациях, только части строки запроса могут быть вырезаны для генерации группового URL для URL. Подсистема 126 идентификации необязательных ресурсов может, затем, отсортировать или кластеризовать групповые URL и определить, какие групповые URL имели наибольшее количество запросов выборки за данный период времени.

25 [0025] Подсистема 126 идентификации необязательных ресурсов может выбрать в качестве потенциальных шаблонов групповые URL с наибольшим количеством запросов выборки. Такие шаблоны представляют встраиваемые ресурсы, которые имеют наилучший потенциал для оптимизации процесса просмотра, поскольку они выбираются чаще всего. Подсистема идентификации необязательных ресурсов может определить,

30 являются ли необязательными URL, которые соответствуют потенциальному шаблону, или выборочная совокупность URL, которая соответствует потенциальному шаблону. Например, подсистема 126 идентификации необязательных ресурсов может идентифицировать встраиваемый ресурс, который соответствует потенциальному шаблону, из записей 132 выборки, и может запросить, чтобы подсистема 120 пакетного

35 воспроизведения воспроизвел первый результат воспроизведения веб-страницы как объекта встраивания для встраиваемого ресурса. Веб-страница как объект встраивания может быть идентифицирована, например, из записи выборки для встраиваемого ресурса. Подсистема 126 идентификации необязательных ресурсов может, затем, запросить, чтобы подсистема 120 пакетного воспроизведения воспроизвел второй

40 результат воспроизведения веб-страницы как объекта встраивания без выборки встраиваемого ресурса.

[0026] Для определения того, подвергся ли воздействию воспроизводимый контент веб-страницы как объекта встраивания при пропуске конкретного встраиваемого ресурса, подсистема 126 идентификации необязательных ресурсов может сравнить

45 первый результат воспроизведения со вторым результатом воспроизведения. В некоторых реализациях, если контент подвергся значительному воздействию, то конкретный встраиваемый ресурс может быть добавлен в список необходимых ресурсов. Если контент подвергся незначительному воздействию (например, контент является

подобным), то подсистема 126 идентификации необязательных ресурсов может идентифицировать встраиваемый ресурс в качестве необязательного ресурса. В некоторых реализациях, встраиваемый ресурс (например, его URL) может храниться в хранилище данных, таких как шаблоны 130 необязательных ресурсов. В некоторых реализациях, встраиваемый ресурс может храниться во временном списке необязательных ресурсов, используемых позже для определения шаблонов для необязательных ресурсов. Подсистема 126 идентификации необязательных ресурсов может выполнить этот тест (сравнение результатов воспроизведения) с каждым встраиваемым ресурсом, который соответствует потенциальному шаблону в записях выборки, или для выборочной совокупности встраиваемых ресурсов, которые соответствуют потенциальному шаблону. В некоторых реализациях, если какие-либо из встраиваемых ресурсов, которые соответствуют потенциальному шаблону, являются релевантными, то потенциальный шаблон не является шаблоном необязательных ресурсов. В некоторых реализациях, если только небольшое процентное отношение встраиваемых ресурсов, которые соответствуют потенциальному шаблону (например, 1% или меньше), являются необходимыми ресурсами, то потенциальный шаблон считается шаблоном необязательных ресурсов, и подсистема 126 идентификации необязательных ресурсов может включить этот потенциальный шаблон в качестве шаблона в шаблоны 130 необязательных ресурсов. Конечно, если все протестированные встраиваемые ресурсы, которые соответствуют потенциальному шаблону, являются необязательными, то подсистема 126 идентификации необязательных ресурсов может добавить этот потенциальный шаблон к шаблонам 130 необязательных ресурсов.

[0027] В некоторых реализациях, подсистема 126 идентификации необязательных ресурсов может также проверить то, что шаблоны в шаблонах 130 необязательных ресурсов все еще являются необязательными. Например, подсистема 126 идентификации необязательных ресурсов может выбрать выборочную совокупность URL, соответствующих шаблону, попросить подсистема воспроизведения воспроизвести результат воспроизведения со встраиваемым ресурсом, который соответствует шаблону, и без этого встраиваемого ресурса, и сравнить два результата воспроизведения. Если URL, соответствующий шаблону в шаблонах 130 необязательных ресурсов, больше не является необязательным, то шаблон может быть удален.

[0028] В некоторых реализациях, подсистема 126 идентификации необязательных ресурсов может определить то, является ли шаблон необязательным, для мобильных браузеров отдельно от полных браузеров. Например, некоторые веб-сайты загружают разный контент для мобильной версии веб-страницы и полной версии, и встраиваемый ресурс может быть необязательным для мобильного браузера, но необходимым для полного браузера. Таким образом, подсистема 126 идентификации необязательных ресурсов может воспроизвести четыре результата воспроизведения для встраиваемого ресурса. Первые два результата воспроизведения могут быть воспроизведены подсистемой воспроизведения, который эмулирует полный браузер, в то время как последние два результата воспроизведения могут быть воспроизведены подсистемой воспроизведения, который эмулирует мобильный браузер на мобильном устройстве, таком как смартфон или планшет. Шаблоны 130 необязательных ресурсов могут, таким образом, включать в себя данные, указывающие на то, является ли шаблон необязательным, согласно типу браузера, например, для мобильного браузера или для полного браузера.

[0029] Хотя это и не показано на фиг. 1 для краткости, в некоторых реализациях, система 110 пакетного воспроизведения может быть распределена по двум или более

отдельным вычислительным устройствам. Дополнительно, один или несколько из подсистем 120 пакетного воспроизведения, услуги 122 выборки, подсистемы 124 автоматического веб-сканирования, и подсистемы 126 идентификации необязательных ресурсов могут быть распределены по одному или нескольким вычислительным устройствам. В некоторых реализациях, один или несколько из подсистем 120 пакетного воспроизведения, услуги 122 выборки, подсистемы 124 автоматического веб-сканирования, и подсистемы 126 идентификации необязательных ресурсов могут совместно использовать ресурсы, такие как память или аппаратные процессоры, с другими компонентами системы 110 пакетного воспроизведения. Подобным образом, записи 132 выборки и шаблоны 130 необязательных ресурсов могут также храниться в памяти, распределенной по множественным вычислительным устройствам. В некоторых реализациях, различные компоненты системы 110 пакетного воспроизведения могут совместно использовать аппаратные компоненты вычислительного устройства, или могут быть логическими частями одного и того же вычислительного устройства.

[0030] Система 110 пакетного воспроизведения может быть связана с клиентом (клиентами) 180 и серверами 190 через сеть 160. Сеть 160 может быть, например, интернетом, или сеть 160 может быть проводной или беспроводной локальной сетью (LAN), глобальной сетью (WAN), их комбинацией и т.д., реализованной с использованием, например, шлюзовых устройств, мостов, коммутаторов и/или т.п. Через сеть 160, система 110 пакетного воспроизведения может устанавливать связь с клиентами 180 и/или хост-серверами 190 и передавать данные к клиентам 180 и/или хост-серверам 190 или принимать данные от них. Например, система 110 пакетного воспроизведения может обеспечить шаблоны 130 необязательных ресурсов для клиента 180, или может принять запрос от клиента 180, спрашивающий, является ли конкретный встраиваемый ресурс необязательным, и может обеспечить ответ на основе шаблонов 130 необязательных ресурсов.

[0031] Клиент 180 может быть персональным вычислительным устройством, таким как вычислительное устройство 600, показанное на фиг. 6, на котором установлен персональный веб-браузер 140. Примеры персональных веб-браузеров включают в себя CHROME, SAFARI, INTERNET EXPLORER, FIREFOX и т.д., либо для мобильного устройства, такого как смартфон или планшет, либо для персонального компьютера, такого как компактный портативный компьютер или настольный компьютер. Браузер 140 может быть выполнен с возможностью использования услуги для определения того, следует ли выбрать конкретный встраиваемый ресурс при воспроизведении веб-страницы. В некоторых реализациях, система может переслать копию шаблонов 130 необязательных ресурсов к клиенту 180 для использования браузером 140. В этом смысле, услуга может быть локальной услугой. В некоторых реализациях, браузер 140 может быть выполнен с возможностью запрашивания услуги на базе сервера об определении того, соответствует ли конкретный встраиваемый ресурс одному из шаблонов в шаблонах 130 необязательных ресурсов. Браузер 140 выполняет оптимизированное воспроизведение с использованием услуги, для пропуска необязательных встраиваемых ресурсов при воспроизведении веб-страницы. Таким образом, браузер 140 может быть выполнен с возможностью селективного пропуска встраиваемых ресурсов для улучшения времени воспроизведения и сохранения ресурсов при отсутствии воздействия на контент.

[0032] Фиг. 2 является блок-схемой результата 200 воспроизведения, воспроизводимого подсистемой 120 пакетного воспроизведения, согласно одной реализации. Результат 200 воспроизведения может включать в себя множество

компонентов, не все из которых показаны на фиг. 2. Например, результат 200 воспроизведения может включать в себя изображение 205 воспроизводимой страницы. Изображение 205 может быть изображением, которое может быть отображено через устройство отображения для пользователя через веб-браузер. Изображение 205 может быть использовано, например, для отображения уменьшенного изображения воспроизводимой страницы для пользователя, а также для определения того, где на устройстве отображения находятся элементы веб-страницы (например, на основе их координат x и y). Результат 200 воспроизведения может также включать в себя дерево 210 объектной модели документа (DOM). Дерево 210 DOM представляет HTML-структуру веб-страницы. Результат 200 воспроизведения может также включать в себя компоновку 215. Компоновка 215 обычно включает в себя блок для элементов веб-страницы, причем этот блок задает координаты x и y элемента в изображении 205. Таким образом, компоновка 215 обеспечивает указание на то, где на веб-странице находится элемент, сколько пространства он занимает на веб-странице и т.д. Таким образом, компоновка 215 обеспечивает информацию о том, какую часть веб-страницы занимают рекламные объявления, насколько выступает абзац (например, выше сгиба или ниже сгиба), является ли видимым элемент и т.д. Другими словами, компоновка 215 обеспечивает геометрическую информацию об элементах воспроизводимой веб-страницы. Результат 200 воспроизведения может также включать в себя ошибки 220. Ошибки 220 включают в себя ошибки, происходящие в результате запуска объектов сценариев, например, объектов JavaScript. Результат 200 воспроизведения может также включать в себя список встраиваемых ресурсов 225, выбранных во время воспроизведения, и может включать в себя другие элементы (не показаны), генерируемые в качестве части процесса воспроизведения. Таким образом, результат 200 воспроизведения обеспечивает не только информацию, доступную посредством выборки контента с хост-сервера. Нижестоящие процессы запрашивания, такие как подсистема идентификации необязательных ресурсов, могут использовать информацию о результате воспроизведения в различных целях, например, для определения того, значительно ли изменяется контент воспроизводимой веб-страницы при запуске объектов сценариев. Например, контент в рекламном участке веб-страницы может считаться неважным, так что URL может быть все же необязательным, даже при различиях в рекламном участке. В некоторых реализациях, только различия в основных компонентах результата воспроизведения могут считаться важными. Основные компоненты могут быть наибольшими блоками в компоновке 215 (например, блоками с наибольшей высотой и шириной).

[0033] Фиг. 3 является блок-схемой последовательности операций, показывающей иллюстративный процесс 300 для идентификации шаблонов необязательных ресурсов, согласно одной реализации. Процесс 300 может быть выполнен системой, такой как система 110 по фиг. 1. Система может использовать процесс 300 для идентификации шаблонов для необязательных ресурсов на основании записей выборки. Система может использовать шаблоны для необязательных ресурсов на основе сравнения результата воспроизведения веб-страницы как объекта встраивания с ресурсом и без ресурса. Шаблоны могут храниться в хранилище данных и могут использоваться в качестве части услуги, которая обеспечивает браузеру, либо полному браузеру, либо мобильному браузеру, либо подсистеме пакетного воспроизведения, возможность оптимизировать воспроизведение посредством пропуска (например, невыборки) необязательных ресурсов.

[0034] Процесс 300 может начаться с системной кластеризации выбранных

встраиваемых ресурсов с помощью потенциальных шаблонов (305). Система может выбрать все встраиваемые ресурсы, выбранные в течение некоторого предшествующего периода времени, например, в течение одного дня, двух дней, одной недели, до момента последнего запуска процесса 300 и т.д. Потенциальные шаблоны могут быть сгенерированы посредством вырезания всей или части строки запроса из URL, найденного в записях выборки. Записи выборки могли быть сгенерированы подсистемой автоматического веб-сканирования или услугой выборки. В некоторых реализациях, записи выборки могут быть сгенерированы в качестве части процесса индексирования, например, процесса индексирования для документов, доступных через интернет.

Встраиваемый ресурс может быть идентифицирован в записи выборки посредством идентификатора, такого как URL. Таким образом, на веб-страницу или встраиваемый ресурс можно также ссылаться по URL, и следует понимать, что ссылка на встраиваемый ресурс (или веб-страницу) и URL являются, в общем, синонимами, при использовании здесь. Потенциальный шаблон может быть сгенерирован для URL посредством вырезания всей или части строки запроса, причем строка запроса является любыми символами, которые следуют за вопросительным знаком (?) в URL для встраиваемого ресурса. Потенциальный шаблон может быть также сгенерирован для встраиваемого ресурса посредством вырезания или игнорирования поддоменного имени, игнорирования компонентов пути (например, компонентов между первой косой чертой («/») и строкой запроса («?»), или игнорирования некоторых параметров в строке запроса. В некоторых реализациях, встраиваемый ресурс из записей выборки может быть связан более чем с одним потенциальным шаблоном, сгенерированным из его URL. Потенциальные URL могут быть кластеризованы с помощью сортировки или с помощью других известных технологий кластеризации.

[0035] Система может выбрать кластер, представленный одним из потенциальных шаблонов (310), и определить, является ли количество элементов кластера большим или равным порогу (315). Количество элементов кластера представляет собой количество раз, которое встраиваемые ресурсы, которые соответствуют потенциальному шаблону, были выбраны в течение предшествующего периода времени. Таким образом, один конкретный встраиваемый ресурс может быть представлен много раз в этом количестве, если он был выбран более одного раза. Если это количество для кластера меньше порога (315, Нет), то система может перейти к следующему кластеру (355). Если потенциальный шаблон действительно имеет достаточное количество элементов (315, Да), то система может выбрать выборочную совокупность встраиваемых ресурсов, которые соответствуют потенциальному шаблону (320). Другими словами, система может отобразить уникальные элементы кластера. Конечно, в некоторых реализациях, система может выбрать все уникальные встраиваемые ресурсы в кластере в качестве выборочной совокупности (например, размер выборочной совокупности равен 100%). Затем, система может выбрать один из встраиваемых ресурсов в выборочной совокупности (325) и сгенерировать первый результат воспроизведения веб-страницы, в которую встраивается ресурс (например, веб-страницы как объекта встраивания), с выбранным встраиваемым ресурсом (330). Веб-страница как объект встраивания может быть определена на основании записей выборки. Система может также сгенерировать второй результат воспроизведения веб-страницы как объекта встраивания без выборки выбранного встраиваемого ресурса (330). Система может сгенерировать два результата воспроизведения с использованием, например, подсистемы пакетного воспроизведения. Затем, система может сравнить первый результат воспроизведения и второй результат воспроизведения для определения того, являются ли подобными (335) результаты

воспроизведения.

[0036] Система может использовать множество тестов для определения того, являются ли подобными результаты воспроизведения, что описано более подробно в отношении фиг. 4. Если результаты воспроизведения не являются подобными (335, Нет), то система может перейти к следующему кластеру (335). Если результаты воспроизведения являются подобными (335, Да), то система может определить, существуют ли другие встраиваемые ресурсы в выборочной совокупности кластера для проверки (340). Если существуют другие встраиваемые ресурсы (340, Нет), то система может выбрать следующий встраиваемый ресурс для выборочной совокупности (345) и повторить этап 330 и 335, с генерацией двух результатов воспроизведения и сравнением этих результатов. Если все встраиваемые ресурсы в выборочной совокупности были проверены (340, Да), то система может добавить потенциальный шаблон для кластера в хранилище данных шаблонов необязательных ресурсов (350). Это хранилище данных может быть использовано услугой для идентификации встраиваемых ресурсов, которые не должны быть выбраны, как объяснено более подробно в отношении фиг. 5. Затем, система может определить, существуют ли другие кластеры для проверки (355). Если существуют другие оставшиеся кластеры для анализа (355, Нет), то система может выбрать следующий кластер, с повторением этапов 315-355 для встраиваемых ресурсов, которые соответствуют потенциальному шаблону следующего кластера. Если больше не существует кластеров для анализа (355, Да), то процесс 300 завершается.

[0037] Иллюстративный процесс 300 показывает реализацию, в которой один встраиваемый ресурс, который не является необязательным, обеспечивает то, что потенциальный шаблон не будет включен в хранилище данных шаблонов необязательных ресурсов (например, 335, Нет). Однако в некоторых реализациях система может хранить список встраиваемых ресурсов для потенциального шаблона, которые являются необязательными, и список встраиваемых ресурсов для потенциального шаблона, которые являются необходимыми (например, два результата воспроизведения не считаются подобными). В таких реализациях система может вычислить процентное отношение встраиваемых ресурсов, которые соответствуют потенциальному шаблону, которые являются необходимыми. Когда это процентное отношение является очень низким (например, 1% или меньше), система может все же добавить потенциальный шаблон в хранилище данных шаблонов необязательных ресурсов (350). Таким образом, реализации не ограничены точным процессом 300, показанным на фиг. 3.

[0038] В некоторых реализациях, система может повторить некоторые или все этапы процесса 300 с результатами воспроизведения, воспроизведенными браузером другого типа. Например, тип браузера может быть полным браузером или мобильным браузером. Если первый и второй результат воспроизведения воспроизводились полным браузером, то система может повторить этапы 330-350 с использованием мобильного браузера. Таким образом, список шаблонов необязательных ресурсов может включать в себя указание на тип браузера, и шаблоны в хранилище данных могут также включать в себя указание на тип браузера. Это обеспечивает системе возможность указания на то, что встраиваемый ресурс является необязательным для одного типа браузера (например, мобильного браузера) и является необходимым для другого типа браузера (например, полного браузера).

[0039] Фиг. 4 является блок-схемой последовательности операций, показывающей иллюстративный процесс 400 для определения того, являются ли подобными два результата воспроизведения, согласно одной реализации. Процесс 400 может быть

выполнен, например, в виде части этапа 335 фиг. 3. В то время как процесс 400 показывает три теста для определения подобия результатов воспроизведения, следует понимать, что реализации могут включать в себя один, два, или все показанные тесты, и могут включать в себя дополнительные тесты подобия, которые не показаны.

5 [0040] Процесс 400 может включать в себя сравнение маркеров в двух результатах воспроизведения (405). Маркеры включают в себя текст документа, например, слова, видимые для пользователя. Маркеры могут быть созданы посредством обработки дерева DOM, такого как дерево 210 DOM по фиг. 2. В некоторых реализациях, игнорируемые слова и числа могут быть удалены перед выполнением сравнения. Если
10 маркеры для первого результата воспроизведения отличаются от маркеров для второго результата воспроизведения (410, Да), то система может необязательно определить, является ли количество уникальных маркеров, добавленных при воспроизведении с включенными сценариями, меньшим, чем порог маркеров (415). Другими словами, если воспроизведение с включенными сценариями обеспечивает только малое количество
15 уникальных маркеров, то система может посчитать, что различия являются незначительными. В некоторых реализациях, порог маркеров может быть равен 5. Если количество добавленных уникальных маркеров соответствует порогу маркеров (415, Нет), то система может посчитать, что результаты воспроизведения не являются подобными (430). Когда результаты воспроизведения не являются подобными, встраиваемый ресурс, используемый для генерации результатов воспроизведения, не является необязательным. В некоторых реализациях, система может пропустить тест
20 415, и любые различия в маркерах могут считаться значительными, а результаты воспроизведения - неподобными. Если количество не соответствует порогу маркеров (415, Да), то система может посчитать, что различие в маркерах является
25 незначительным. Таким образом, если уникальные маркеры для первого результата воспроизведения являются такими же, как маркеры во втором результате воспроизведения (410, Нет), или количество уникальных маркеров не соответствует порогу маркеров (415, Да), то результаты воспроизведения могут считаться подобными, и система может продолжить выполнение других тестов. В реализациях, которые
30 включают в себя только тест маркеров, система может приступить прямо к этапу 445, с определением того, что результаты воспроизведения являются подобными.

[0041] В некоторых реализациях, если маркеры являются одинаковыми (410, Нет) или (415, Да), то система может сравнить внешние связи (420). Внешние связи представляют собой связи воспроизводимой страницы с другими веб-страницами или
35 другими документами. Внешние связи могут быть также извлечены из узлов тегов привязки (<a> тегов) дерева DOM результата воспроизведения. Если система найдет различие во внешних связях (425, Да), то система посчитать, что различие является значительным, и результаты воспроизведения не являются подобными (430). Таким образом, встраиваемый ресурс не считается необязательным. Если внешние связи в
40 двух результатах воспроизведения являются одинаковыми (425, Нет), то система может посчитать, что результаты воспроизведения являются подобными (например, посредством перехода к этапу 450), или может перейти к выполнению других тестов подобия. В некоторых реализациях, система может выполнить этапы 420 и 425 перед этапами 405 и 415 и независимо от них.

45 [0042] В другом тесте подобия, который может быть выполнен дополнительно к другим тестам или вместо них, система может определить подобие по изображению (например, моментальному снимку), дереву DOM, или компоновке. Например, система может вычислить самую длинную общую последовательность (LCS) для дерева DOM,

компоновки, изображения, или другого компонента результата воспроизведения (435). Система может использовать LCS для вычисления оценки подобия, которая представляет собой процентное отношение компонента, который является таким же (440). Когда это процентное отношение соответствует порогу (445, Да), система может определить, что результаты воспроизведения являются подобными (450). Когда процентное отношение не соответствует порогу (445, Нет), система может определить, что результаты воспроизведения не являются подобными (430). После того, как система определяет, что результаты воспроизведения являются подобными (450) или не являются подобными (430), процесс 400 завершается.

10 [0043] Конечно, система может выполнить другие тесты в качестве части процесса 400. Например, в другом тесте подобия (не показан на фиг. 4), система может определить основные компоненты компоновки. Компоновка результата воспроизведения выполнена из блоков - причем каждый блок определен экранными координатами. Обычно, каждый блок соответствует соответствующему элементу веб-страницы. Например, компоновка 15 может включать в себя блочные представления DOM-узлов в дереве DOM (хотя не все элементы DOM могут иметь соответствующий блок воспроизведения). Блоки могут быть организованы в структуру дерева, также известную как дерево воспроизведения. Таким образом, например, таблица может быть представлена блоком в компоновке, а абзац может быть представлен другим блоком в компоновке. Основные компоненты 20 веб-страницы являются элементами в компоновке с наибольшими блоками, определенными экранными координатами. В некоторых реализациях, система может включать в себя заданное количество основных компонентов в данном наборе, например, таким образом, чтобы этот набор включал в себя наибольший из основных компонентов. В некоторых реализациях, система может включать в себя основные 25 компоненты, которые определяют процентное отношение экрана в этом наборе. В такой реализации, если наибольший основной компонент занимает больше, чем данное процентное отношение, то наибольший основной компонент может быть единственным элементом набора. В некоторых реализациях, любые компоненты с размером блока, большим порогового размера, могут быть включены в набор основных компонентов. 30 Компонент, который не входит в набор основных компонентов, может считаться незначительным компонентом. В некоторых реализациях, система может использовать технологию постепенного углубления в проблему для нахождения основного компонента. Например, начиная с корневого блока дерева воспроизведения, система может, затем, выполнить поиск в ширину и идентифицировать наибольший дочерний 35 блок корневого блока. Затем, система может выбрать наибольшего потомка и пройти глубже, чтобы найти наибольший дочерний блок текущего блока (например, наибольший дочерний блок корневого блока). Система может остановить проход глубже, если никакие из дочерних блоков не будут доминировать, например, не будут занимать больше половины площади родительского блока. Когда система останавливает 40 проход глубже, основным компонентом является блок, не имеющий доминирующего потомка.

[0044] Система может вычислить оценку подобия по основным компонентам в наборе. Например, система может использовать оценку перекрытия блоков основных 45 компонентов. В одной оценке перекрытия, система может вычислить площадь перекрытия для основного компонента, которая представляет площадь основного компонента в первом результате воспроизведения, которая перекрывается с площадью соответствующего основного компонента во втором результате воспроизведения. Затем, система может вычислить оценку подобия, которая представляет собой

гармоническое среднее перекрытия относительно общей площади каждого основного компонента. Например, система может использовать формулу $2/((a1/oa)+(a2/oa))$, где oa является площадью перекрытия, $a1$ является общей площадью блока основного компонента в первом результате воспроизведения, и $a2$ является общей площадью блока основного компонента во втором результате воспроизведения. Конечно, система может использовать другие метрики подобия для вычисления оценки. Примеры таких метрик подобия включают в себя, но не ограничены этим, подобие Катца (Katz). Если оценка подобия соответствует порогу подобия, то результаты воспроизведения считаются подобными, и, таким образом, встраиваемый ресурс является необязательным.

В некоторых реализациях, порог подобия может быть высоким, например, 80% или более. Если оценка подобия не соответствует порогу, то система может посчитать, что результаты воспроизведения не являются подобными. В некоторых реализациях, если оценка подобия соответствует порогу подобия, то система может выполнить дополнительные тесты подобия на основе результатов воспроизведения, которые не показаны на фиг. 4. В некоторых реализациях, система может также вычислить оценку подобия для различий в незначительных компонентах, но оценить веса этих различий гораздо ниже, например, делая их гораздо меньшей частью конечной оценки подобия для результатов воспроизведения.

[0045] В некоторых реализациях, система может постараться минимизировать или устранить эффекты неопределенности в процессе 400. Неопределенность имеет место, когда результаты воспроизведения являются разными, даже если встраиваемые ресурсы являются идентичными. Неопределенность может быть причиной того, что встраиваемые ресурсы, которые являются необязательными, будут казаться необходимыми. Таким образом, учет неопределенности может значительно увеличить покрытие необязательных ресурсов. Для учета неопределенности, система может сгенерировать третий результат воспроизведения. Третий результат воспроизведения может быть воспроизведением с использованием ресурсов, возвращенных из генерации первого результата воспроизведения (например, результата воспроизведения, который включает в себя URL выборочной совокупности). Любые различия между третьим результатом воспроизведения и первым результатом воспроизведения происходят вследствие неопределенности. Система может улучшить подобие посредством удаления любых различий, которые происходят вследствие неопределенности (например, различий во внешних связях, маркерах, изображении, дереве воспроизведения и т.д., которые являются разными в первом и третьем результатах воспроизведения). Например, система может вычислить первое количество, представляющее количество пикселей, которые являются разными в изображении первого результата воспроизведения и изображении третьего результата воспроизведения, и второе количество, представляющее количество пикселей, которые являются разными в изображениях первого результата воспроизведения и второго результата воспроизведения. Система может вычесть первое количество из второго количества для получения некоторой разности. Чем ближе эта разность ко второму количеству, тем более вероятным является то, что любые различия между первым и вторым результатами воспроизведения не происходят вследствие неопределенности. Таким образом, система может настроить оценку подобия LCS на основе этой разности (например, не применять никакой настройки, если эта разность равна второму количеству, уменьшить вдвое, если эта разность равна половине второго количества, и т.д.). В качестве другого примера, система может определить DOM-узлы, которые являются разными в первом результате воспроизведения и третьем результате воспроизведения, и игнорировать эти узлы при сравнении первого результата

воспроизведения и второго результата воспроизведения (например, при вычислении LCS).

[0046] Фиг. 5 является блок-схемой последовательности операций, показывающей иллюстративный процесс 500 для обеспечения информации для оптимизированного воспроизведения, согласно одной реализации. Процесс 500 может быть запущен в качестве услуги для браузера. Браузер может быть клиентским браузером, либо мобильным, либо полным, таким как браузер 140 фиг. 1, или подсистемой пакетного воспроизведения, таким как подсистемы 120 пакетного воспроизведения фиг. 1. В некоторых реализациях, услуга может быть услугой на основе облака. Другими словами, браузер может отправить запрос через сеть к услуге на основе облака, которая может обеспечить ответ для браузера. В некоторых реализациях, услуга может быть запущена на сервере, таком как система 110 пакетного воспроизведения фиг. 1. В некоторых реализациях, услуга может быть локальной для вычислительного устройства, на котором браузер запускается. Например, хранилище данных шаблонов необязательных ресурсов может быть передано к вычислительному устройству, которое запускает браузер, и услуга может быть выполнена в вычислительном устройстве. В некоторых реализациях, услуга может быть запущена на клиенте, таком как клиент 180 фиг. 1.

[0047] Процесс 500 начинается с услуги, принимающей URL от браузера (505). Браузер может быть мобильным браузером, полным браузером, и может быть запущен на клиенте или в качестве подсистемы пакетного воспроизведения. Услуга может определить, соответствует ли URL из запроса шаблону необязательных ресурсов (510). В некоторых реализациях, шаблон может представлять полный URL, так что установление соответствия выполняется на основе всего запрашиваемого URL. В некоторых реализациях, шаблон может представлять частичный URL, например, с удаленной строкой запроса или с различными участками, замененными символами подстановки. Если запрашиваемый URL не соответствует по меньшей мере одному шаблону в хранилище данных необязательных ресурсов (510, Да), то услуга обеспечивает ответ, который указывает на то, что URL является необязательным, и веб-страница как объект встраивания может быть воспроизведена без выборки контента (515). В некоторых реализациях, это может быть ответом, подобным ответу «URL не найден», который браузер может принять, когда ресурс не может быть обнаружен. В ответ на прием этого указания, браузер может воспроизвести веб-страницу как объект встраивания с пропуском встраиваемого ресурса, что экономит ресурсы обработки и пропускной способности. Если запрашиваемый URL не соответствует шаблону в хранилище данных необязательных ресурсов (510, Нет), то услуга обеспечивает ответ, который указывает на то, что встраиваемый ресурс является необходимым (520). В некоторых реализациях, услуга может фактически выполнить выборку и обеспечить контент для встраиваемого ресурса. В некоторых реализациях, услуга может обеспечить ответ, который указывает браузеру на то, что следует приступить к запросу выборки для встраиваемого ресурса. Таким образом, браузер может воспроизвести веб-страницу как объект встраивания с контентом для встраиваемого ресурса. Затем, процесс 500 завершается, после обеспечения информации, которую браузер может использовать для оптимизации процесса воспроизведения. В некоторых реализациях, браузер может обеспечить тип браузера для услуги, и услуга может использовать тип браузера при установлении соответствия шаблонов в хранилище данных необязательных ресурсов.

[0048] Фиг. 6 показывает пример обобщенного вычислительного устройства 600, которое может функционировать в качестве системы 110 пакетного воспроизведения, и/или клиента 180 фиг. 1, которые могут быть использованы с технологиями,

описанными здесь. Вычислительное устройство 600 предназначено для представления различных иллюстративных форм вычислительных устройств, таких как компактные портативные компьютеры, настольные компьютеры, рабочие станции, персональные цифровые секретари, сотовые телефоны, смартфоны, планшеты, серверы, и другие вычислительные устройства, включая носимые устройства. Компоненты, показанные здесь, их соединения и связи, и их функции, как подразумевается, являются только примерами, и не подразумевается, что они ограничивают реализации изобретений, описанных и/или заявленных в этом документе.

[0049] Вычислительное устройство 600 включает в себя процессор 602, например, кремниевый аппаратный процессор, память 604, запоминающее устройство 606, и порты 610 расширений, соединенные через интерфейс 608. В некоторых реализациях, вычислительное устройство 600 может включать в себя приемопередатчик 646, коммуникационный интерфейс 644 и приемный модуль 648 глобальной системы позиционирования (GPS), помимо прочих компонентов, соединенных через интерфейс 608. Устройство 600 может передавать данные беспроводным способом через коммуникационный интерфейс 644, который может включать в себя схемы обработки цифровых сигналов, при необходимости. Каждый из компонентов 602, 604, 606, 608, 610, 640, 644, 646 и 648 может быть установлен на общей системной плате или другими подходящими способами.

[0050] Процессор 602 может обрабатывать команды для исполнения в пределах вычислительного устройства 600, включая команды, хранящиеся в памяти 604 или в запоминающем устройстве 606, для отображения графической информации для GUI на внешнем устройстве ввода/ вывода данных, таком как устройство 616 отображения. Устройство 616 отображения может быть монитором или плоским устройством отображения с сенсорным экраном. В некоторых реализациях, вместе с множественными устройствами памяти и типами памяти могут быть использованы, при необходимости, множественные процессоры и/или множественные шины. Также, множественные вычислительные устройства 600 могут быть соединены, причем каждое устройство обеспечивает часть необходимых операций (например, в виде банка серверов, группы ячеечных серверов, или многопроцессорной системы).

[0051] Память 604 хранит информацию внутри вычислительного устройства 600. В одной реализации, память 604 является блоком или блоками энергозависимой памяти. В другой реализации, память 604 является блоком или блоками энергонезависимой памяти. Память 604 может быть также другой формой машиночитаемой среды, такой как магнитный или оптический диск. В некоторых реализациях, память 604 может включать в себя расширенную память, обеспеченную через интерфейс расширения.

[0052] Запоминающее устройство 606 может обеспечить хранение больших объемов данных для вычислительного устройства 600. В одной реализации, запоминающее устройство 606 может быть машиночитаемой средой или может включать в себя машиночитаемую среду, такую как устройство на гибких дисках, устройство на жестких дисках, устройство на оптических дисках, или устройство на основе ленты, флэш-память или другое подобное твердотельное устройство памяти, или массив устройств, включающий в себя устройства в сети устройств для хранения данных, или другие конфигурации. Компьютерный программный продукт может быть материально реализован в такой машиночитаемой среде. Компьютерный программный продукт может также включать в себя команды, которые, при исполнении, выполняют один или несколько способов, таких как способы, описанные выше. Считываемая компьютерами или машиночитаемая среда является запоминающим устройством, таким

как память 604, запоминающее устройство 606, или память на процессоре 602.

[0053] Интерфейс 608 может быть высокоскоростным контроллером, который управляет операциями при интенсивном использовании полосы пропускания для вычислительного устройства 600, или низкоскоростным контроллером, который управляет операциями при менее интенсивном использовании полосы пропускания, или комбинацией таких контроллеров. Внешний интерфейс 640 может быть обеспечен для обеспечения возможности связи ближнего действия устройства 600 с другими устройствами. В некоторых реализациях, контроллер 608 может быть связан с запоминающим устройством 606 и портом 614 расширения. Порт расширения, который может включать в себя различные коммуникационные порты (например, USB, Bluetooth, Ethernet, беспроводной Ethernet), может быть связан с одним или несколькими устройствами ввода/вывода данных, такими как клавиатура, указывающее устройство, сканер, или сетевое устройство, такое как коммутатор или маршрутизатор, например, через сетевой адаптер.

[0054] Вычислительное устройство 600 может быть реализовано в нескольких разных формах, как показано на фигуре. Например, оно может быть реализовано в виде стандартного сервера 630, или много раз в группе таких серверов. Оно может быть также реализовано в виде части системы стоечных серверов. Дополнительно, оно может быть реализовано в персональном вычислительном устройстве, таком как компактный портативный компьютер 632, настольный компьютер 634, или планшет или смартфон 636. Полная система может быть создана из множественных вычислительных устройств 600, обменивающихся данными друг с другом. Возможны и другие конфигурации.

[0055] Фиг. 7 показывает пример обобщенного вычислительного устройства 700, которое может быть системой 110 фиг. 1, которое может быть использовано с технологиями, описанными здесь. Вычислительное устройство 700 предназначено для представления различных иллюстративных форм устройств обработки больших объемов данных, таких как серверы, ячеечные серверы, центры обработки данных, мэйнфреймы, и другие устройства обработки больших объемов данных. Вычислительное устройство 700 может быть распределенной системой, имеющей множественные процессоры, возможно, включающей в себя узлы хранения данных, подключенные к сети, которые связаны между собой одной или несколькими сетями связи. Компоненты, показанные здесь, их соединения и связи, и их функции, как подразумевается, являются только примерами, и не подразумевается, что они ограничивают реализации изобретений, описанных и/или заявленных в этом документе.

[0056] Распределенное вычислительное устройство 700 может включать в себя любое количество вычислительных устройств 780. Вычислительные устройства 780 могут включать в себя сервер или стоечные серверы, мэйнфреймы и т.д., обменивающиеся данными через локальную или глобальную сеть, специальные оптические каналы, модемы, мосты, маршрутизаторы, коммутаторы, проводные или беспроводные сети и т.д.

[0057] В некоторых реализациях, каждое вычислительное устройство может включать в себя множественные стойки. Например, вычислительное устройство 780a включает в себя множественные стойки 758a - 758n. Каждая стойка может включать в себя один или несколько процессоров, таких как процессоры 752a-752n и 762a-762n. Процессоры могут включать в себя процессоры данных, запоминающие устройства, подключенные к сети, и другие управляемые компьютерами устройства. В некоторых реализациях, один процессор может функционировать в качестве главного процессора и может управлять задачами планирования и распределения данных. Процессоры могут быть

взаимно соединены через один или несколько стоечных коммутаторов 758, и одна или несколько стоек могут быть соединены через коммутатор 778. Коммутатор 778 может управлять обменом данных между множественными соединенными вычислительными устройствами 700.

5 [0058] Каждая стойка может включать в себя память, такую как память 754 и память 764, и запоминающее устройство, такое как запоминающее устройство 756 и 766. Запоминающие устройства 756 и 766 могут обеспечить хранение больших объемов данных и могут включать в себя энергозависимые или энергонезависимые
10 запоминающие устройства, такие как диски, подключенные к сети, гибкие диски, жесткие диски, оптические диски, ленты, флэш-память или другие подобные твердотельные устройства памяти, или массив устройств, включающий в себя устройства в сети устройств для хранения данных, или другие конфигурации. Запоминающие устройства 756 или 766 могут совместно использоваться множественными процессорами,
15 множественными стойками, или множественными вычислительными устройствами, и могут включать в себя машиночитаемую среду, хранящую команды, исполняемые одним или несколькими процессорами. Память 754 и 764 может включать в себя, например, блок или блоки энергозависимой памяти, блок или блоки энергонезависимой памяти, и/или другие формы машиночитаемых сред, такие как магнитные или оптические
20 диски, флэш-память, кэш-память, память с произвольным доступом (RAM), постоянное запоминающее устройство (ROM), и их комбинации. Память, такая как память 754, может также совместно использоваться процессорами 752a-752n. Структуры данных, такие как индекс, могут храниться, например, в запоминающем устройстве 756 и памяти 754. Вычислительное устройство 700 может включать в себя другие компоненты, которые не показаны, такие как контроллеры, шины, устройства ввода/ вывода данных,
25 модули связи и т.д.

[0059] Полная система, такая как система 110, может быть создана из множественных вычислительных устройств 700, обменивающихся данными друг с другом. Например, устройство 780a может обмениваться данными с устройствами 780b, 780c, и 780d, и эти
30 устройства могут совместно представлять систему 100. В качестве другого примера, система 100 фиг. 1 может включать в себя одно или несколько вычислительных устройств 700. Некоторые из вычислительных устройств могут быть расположены географически близко друг к другу, а другие могут быть расположены географически далеко друг от друга. Компоновка вычислительного устройства 700 приведена только в качестве примера, и система может иметь другие компоновки или конфигурации.

35 [0060] Различные реализации могут включать в себя реализацию в одной или нескольких компьютерных программах, которые могут быть исполнены и/или интерпретированы на программируемой системе, включающей в себя по меньшей мере один программируемый процессор, образованный на подложке, который может быть процессором общего назначения или специального назначения, выполненным с
40 возможностью приема данных и команд от системы хранения данных, по меньшей мере одного устройства ввода данных, и по меньшей мере одного устройства вывода данных, или передачи данных и команд к ним.

[0061] Эти компьютерные программы (также известные как программы, программное обеспечение, программные приложения или код) включают в себя машинные команды
45 для программируемого процессора, и могут быть реализованы в высокоуровневом процедурном и/или объектно-ориентированном языке, и/или в языке ассемблера/ машинном языке. При использовании здесь, термины «машиночитаемая среда», «читаемая компьютерами среда» относятся к любому долговременному компьютерному

программному продукту, аппарату и/или устройству (например, магнитным дискам, оптическим дискам, памяти (включая постоянное запоминающее устройство), программируемым логическим устройствам (PLD)), используемым для обеспечения машинных команд и/или данных для программируемого процессора.

5 [0062] Системы и технологии, описанные здесь, могут быть реализованы в вычислительной системе, которая включает в себя серверный компонент базы данных (например, сервер базы данных), или которая включает в себя компонент программного обеспечения промежуточного слоя (например, сервер приложений), или которая
10 включает в себя интерфейсный компонент (например, клиентский компьютер, имеющий графический пользовательский интерфейс или веб-браузер, через которые пользователь может взаимодействовать с реализацией систем и технологий, описанных здесь), или любую комбинацию таких серверных компонентов базы данных, компонентов программного обеспечения промежуточного слоя и интерфейсных компонентов. Компоненты системы могут быть взаимно связаны через любую форму или среду
15 цифровой передачи данных (например, сеть связи). Примеры сетей связи включают в себя локальную сеть ("LAN"), глобальную сеть ("WAN") и интернет.

[0063] Вычислительная система может включать в себя клиенты и серверы. Клиент и сервер обычно удалены друг от друга и обычно взаимодействуют через сеть связи. Взаимосвязь клиента и сервера обеспечивается посредством компьютерных программ,
20 запускаемых на соответствующих компьютерах и имеющих взаимосвязь клиент-сервер друг с другом.

[0064] Таким образом, были описаны некоторые реализации. Тем не менее, различные модификации могут быть выполнены, не выходя за рамки сущности и объема настоящего изобретения. Дополнительно, логические последовательности операций, показанные
25 на фигурах, не требуют конкретного показанного порядка, или последовательного порядка, для достижения необходимых результатов. Дополнительно, могут быть обеспечены другие этапы, или этапы могут быть удалены из описанных последовательностей операций, а также другие компоненты могут быть добавлены к описанным системам или удалены из них. Таким образом, другие реализации находятся
30 в пределах объема нижеследующей формулы изобретения.

(57) Формула изобретения

1. Вычислительная система для идентификации необязательных встраиваемых ресурсов, содержащая по меньшей мере один процессор и память, хранящую инструкции,
35 которые при их исполнении по меньшей мере одним процессором предписывают системе: идентифицировать URL-шаблон, общий для множества URL, хранящихся в записях выборки;
выбирать выборочную совокупность URL из множества URL;
для каждого URL в выборочной совокупности, определять, является ли URL
40 необязательным, посредством:
генерации первого результата воспроизведения для объекта встраивания URL с использованием контента для URL,
генерации второго результата воспроизведения для объекта встраивания без использования контента для URL,
45 вычисления оценки подобия для URL посредством сравнения первого результата воспроизведения и второго результата воспроизведения и
определения того, что URL является необязательным, когда оценка подобия является большей, чем порог подобия, и когда заранее заданное количество URL в выборочной

совокупности является необязательным, сохранять URL-шаблон в хранилище данных шаблонов необязательных ресурсов, причем хранилище данных используется для воспроизведения веб-страницы без выборки контента для встраиваемого ресурса, для которого требуется выборка, когда URL для этого встраиваемого ресурса соответствует URL-шаблону.

2. Система по п. 1, в которой, как часть использования хранилища данных для воспроизведения веб-страницы, упомянутые инструкции дополнительно включают в себя инструкции, которые при их исполнении по меньшей мере одним процессором предписывают системе:

принимать запрос на контент запрашиваемого встраиваемого ресурса для веб-страницы,

определять, соответствует ли запрашиваемый встраиваемый ресурс упомянутому шаблону в хранилище данных, и,

когда запрашиваемый встраиваемый ресурс соответствует шаблону, возвращать указание того, что запрашиваемый встраиваемый ресурс является необязательным.

3. Система по п. 2, в которой упомянутое указание является ошибкой "URL не найден".

4. Система по п. 1, в которой упомянутое количество URL равно количеству URL в выборочной совокупности.

5. Система по п. 1, в которой упомянутая идентификация URL-шаблона, общего для множества URL, хранящихся в записях выборки, включает в себя:

генерацию группового URL для соответственных URL, хранящихся в записях выборки, посредством удаления по меньшей мере участка строки запроса из URL и кластеризацию URL посредством группового URL.

6. Система по п. 1, в которой упомянутая идентификация URL-шаблона, общего для множества URL, хранящихся в записях выборки, включает в себя:

генерацию группового URL для соответственных URL, хранящихся в записях выборки, посредством удаления по меньшей мере участка строки запроса из URL;

кластеризацию URL посредством группового URL и

выбор группового URL кластера с наибольшим количеством элементов в качестве URL-шаблона.

7. Система по п. 1, в которой упомянутое вычисление оценки подобия включает в себя:

определение самой длинной общей последовательности для дерева DOM первого результата воспроизведения и дерева DOM второго результата воспроизведения и

использование самой длинной общей последовательности для определения оценки подобия.

8. Способ идентификации необязательных встраиваемых ресурсов, содержащий этапы, на которых:

идентифицируют, с использованием по меньшей мере одного процессора, URL-шаблон, общий для множества URL, хранящихся в записях выборки;

выбирают, с использованием по меньшей мере одного процессора, выборочную совокупность URL из множества URL;

для каждого URL в выборочной совокупности, определяют, является ли URL необязательным, посредством:

генерации первого результата воспроизведения для объекта встраивания URL с использованием контента для URL и второго результата воспроизведения для объекта встраивания без использования контента для URL и

вычисления оценки подобия для URL посредством сравнения первого результата

воспроизведения и второго результата воспроизведения, причем URL является необязательным, когда оценка подобия является большей, чем порог подобия, и когда большинство URL в выборочной совокупности являются необязательными, сохраняют URL-шаблон в хранилище данных шаблонов необязательных ресурсов, причем хранилище данных используют для воспроизведения веб-страницы без выборки контента для встраиваемого ресурса, для которого требуется выборка, когда URL для этого встраиваемого ресурса соответствует URL-шаблону.

9. Способ по п. 8, в котором использование хранилища данных для воспроизведения веб-страницы включает в себя этапы, на которых:

принимают запрос на контент запрашиваемого встраиваемого ресурса в веб-странице, определяют, соответствует ли запрашиваемый встраиваемый ресурс шаблону в хранилище данных, и,

когда запрашиваемый встраиваемый ресурс соответствует шаблону, возвращают указание того, что запрашиваемый встраиваемый ресурс является необязательным.

10. Способ по п. 8, дополнительно содержащий этап, на котором определяют, что количество, представленное множеством URL, превышает размер порога, перед определением того, являются ли URL в выборочной совокупности необязательными.

11. Способ по п. 8, дополнительно содержащий этап, на котором, когда все URL в выборочной совокупности являются необязательными, сохраняют URL-шаблон в хранилище данных шаблонов необязательных ресурсов.

12. Способ по п. 8, в котором упомянутая идентификация URL-шаблона, общего для множества URL в записях выборки, включает в себя этапы, на которых:

для соответственных URL в записях выборки, генерируют групповой URL для URL в записи выборки посредством удаления по меньшей мере участка строки запроса из URL и

выполняют кластеризацию посредством группового URL.

13. Способ по п. 8, в котором упомянутая идентификация URL-шаблона, общего для множества URL в записях выборки, включает в себя этапы, на которых:

для соответственных URL в записях выборки, генерируют групповой URL для URL в записи выборки посредством удаления по меньшей мере участка строки запроса из URL;

выполняют кластеризацию посредством группового URL и

выбирают групповой URL кластера, когда количество элементов в кластере соответствует порогу.

14. Способ воспроизведения веб-страницы, содержащий этапы, на которых:

принимают запрос на воспроизведение веб-страницы;

идентифицируют, с использованием по меньшей мере одного процессора, по меньшей мере один встраиваемый ресурс в веб-странице, для которого требуется выборка; определяют, что встраиваемый ресурс является необязательным ресурсом,

посредством определения того, что URL для встраиваемого ресурса соответствует шаблону в хранилище данных необязательных ресурсов, и

воспроизводят, с использованием по меньшей мере одного процессора, веб-страницу, как если бы встраиваемый ресурс был недоступным, без выборки контента для встраиваемого ресурса.

15. Способ по п. 14, дополнительно содержащий этап, на котором принимают хранилище данных шаблонов для необязательных ресурсов от услуги.

16. Способ по п. 14, в котором упомянутое определение того, что встраиваемый ресурс является необязательным ресурсом, включает в себя этапы, на которых:

перезаписывают URL для встраиваемого ресурса посредством удаления по меньшей мере участка строки запроса из URL и

сопоставляют перезаписанный URL с шаблоном в хранилище данных необязательных ресурсов, причем встраиваемый ресурс является необязательным, когда перезаписанный URL соответствует этому шаблону.

10

15

20

25

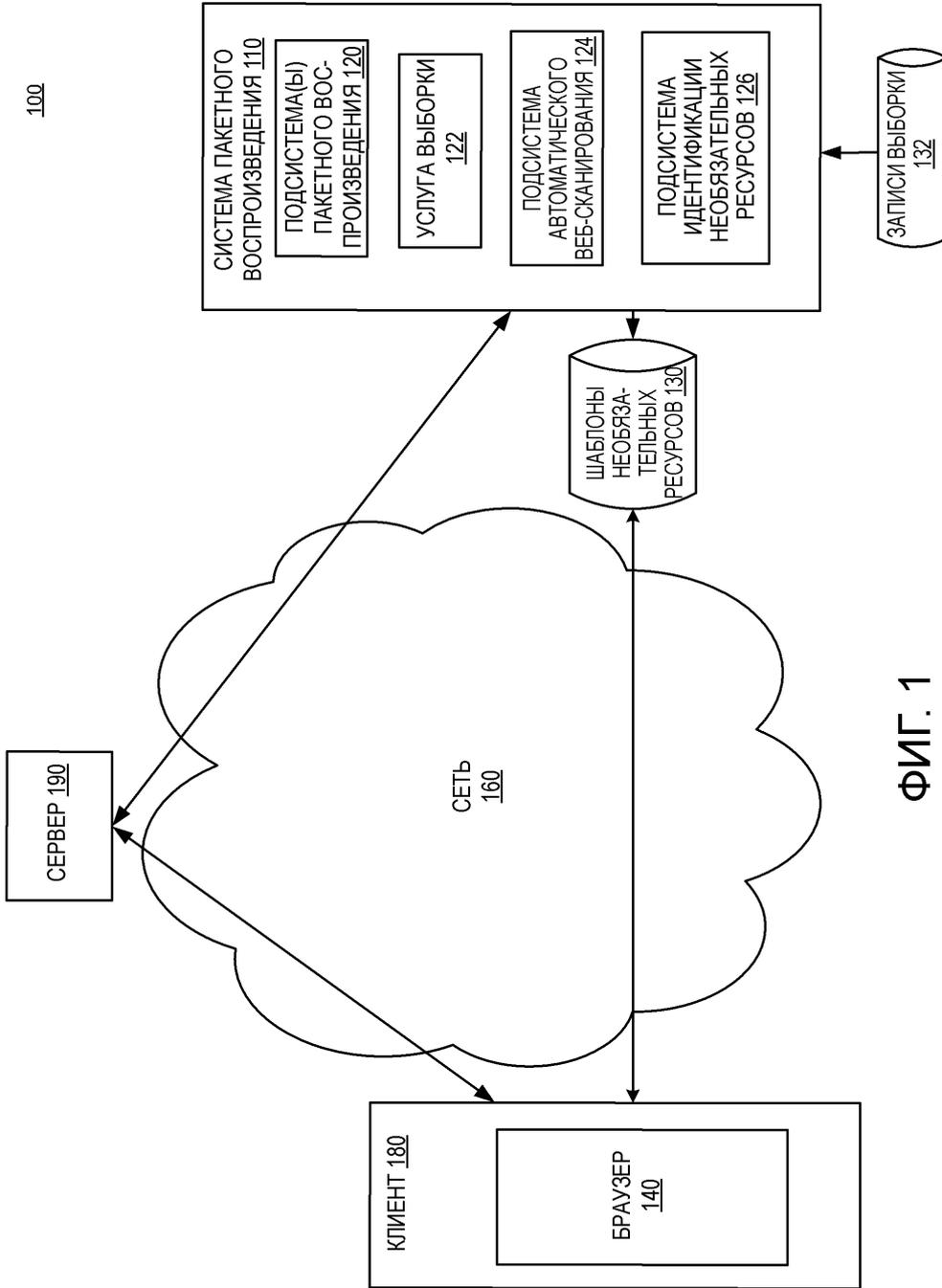
30

35

40

45

1/7

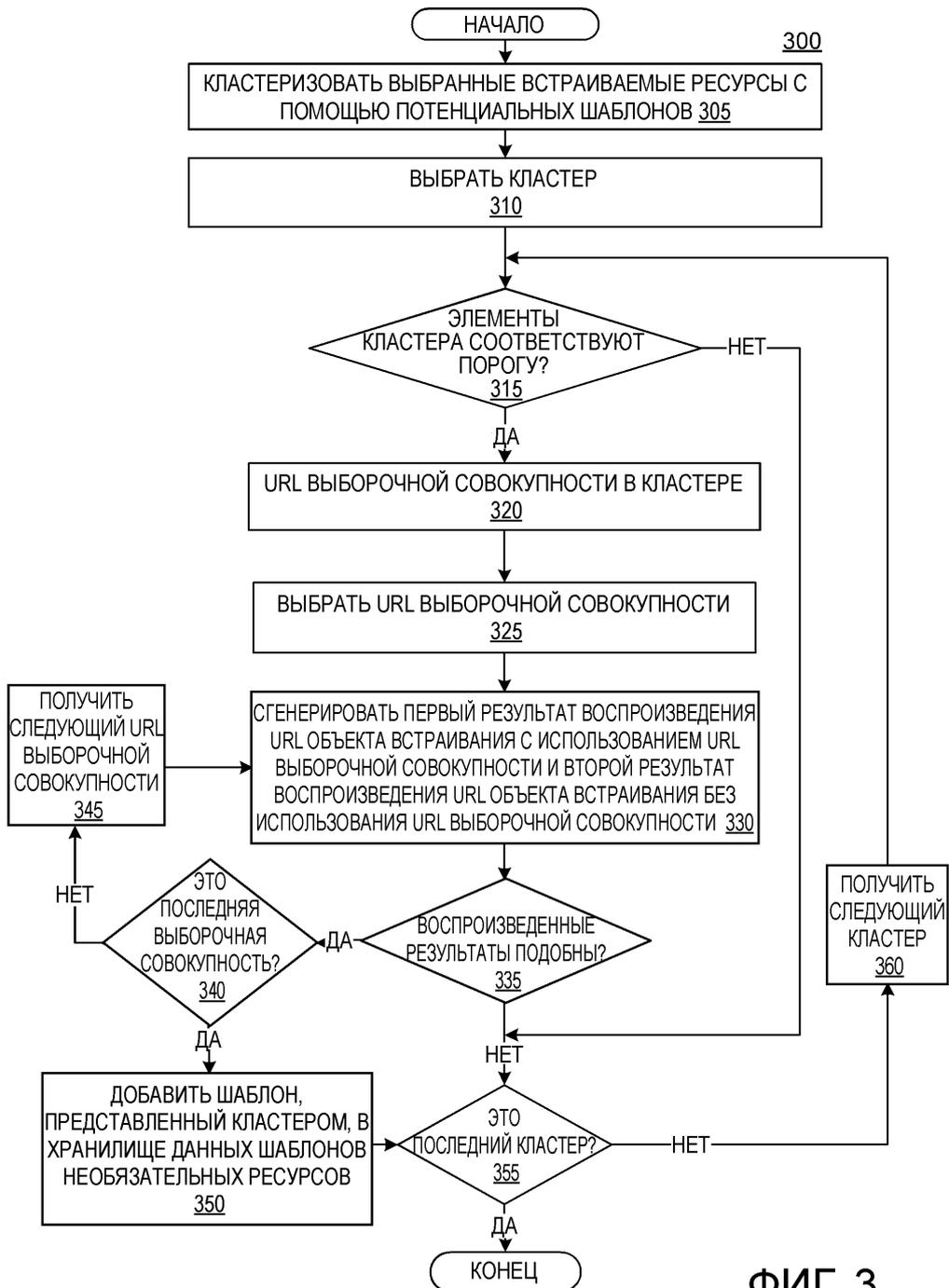


2/7



ФИГ. 2

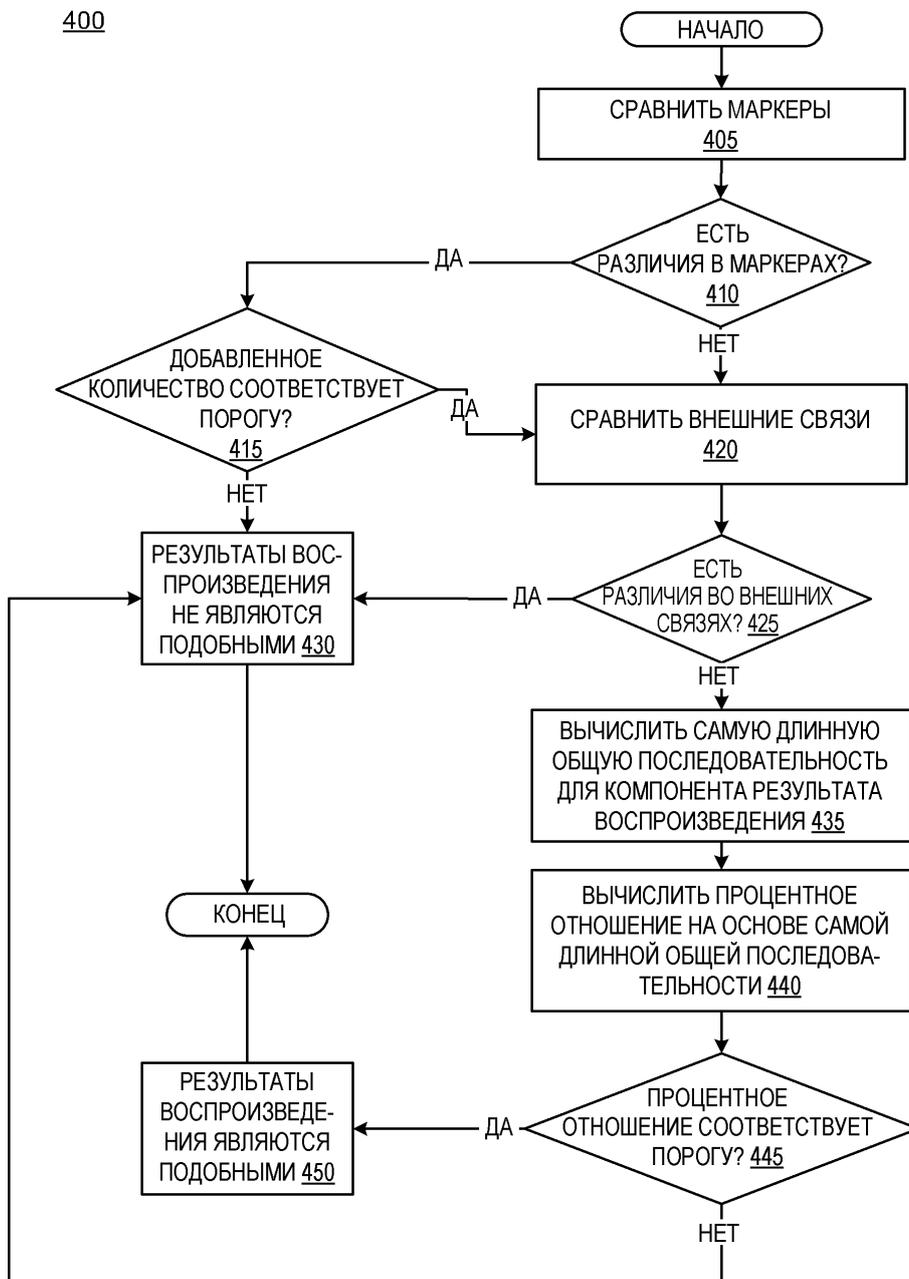
3/7



ФИГ. 3

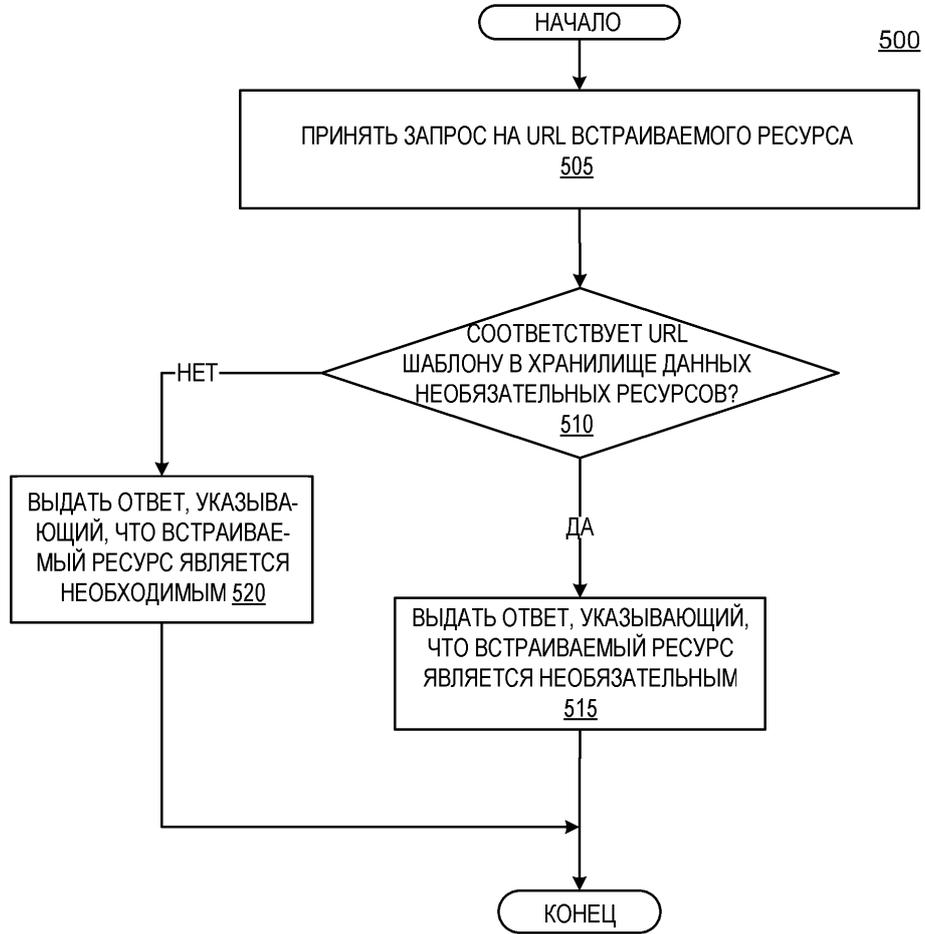
4/7

400



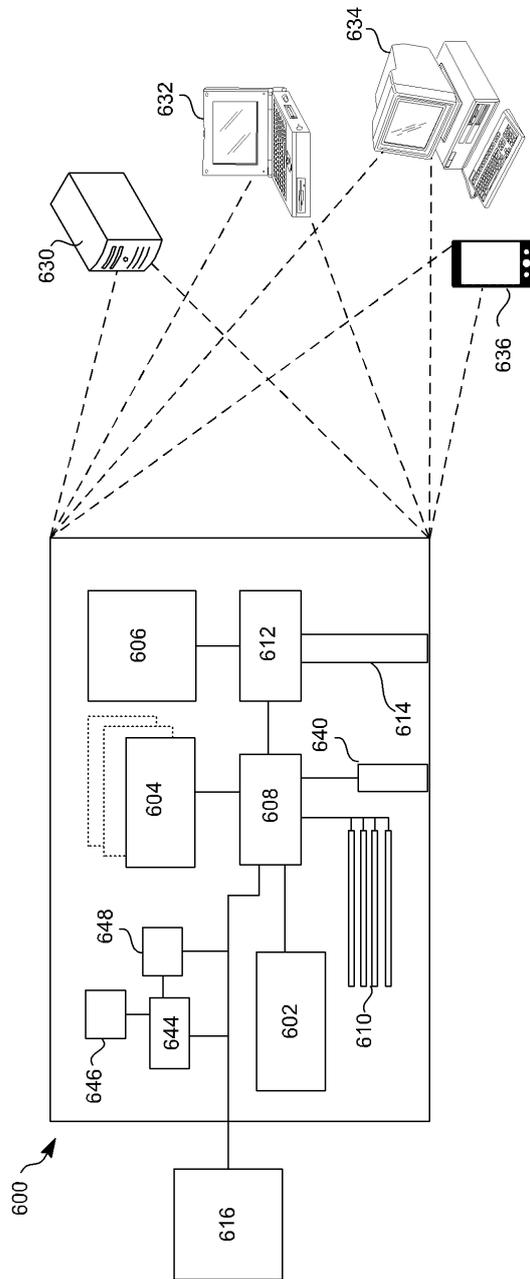
ФИГ. 4

5/7



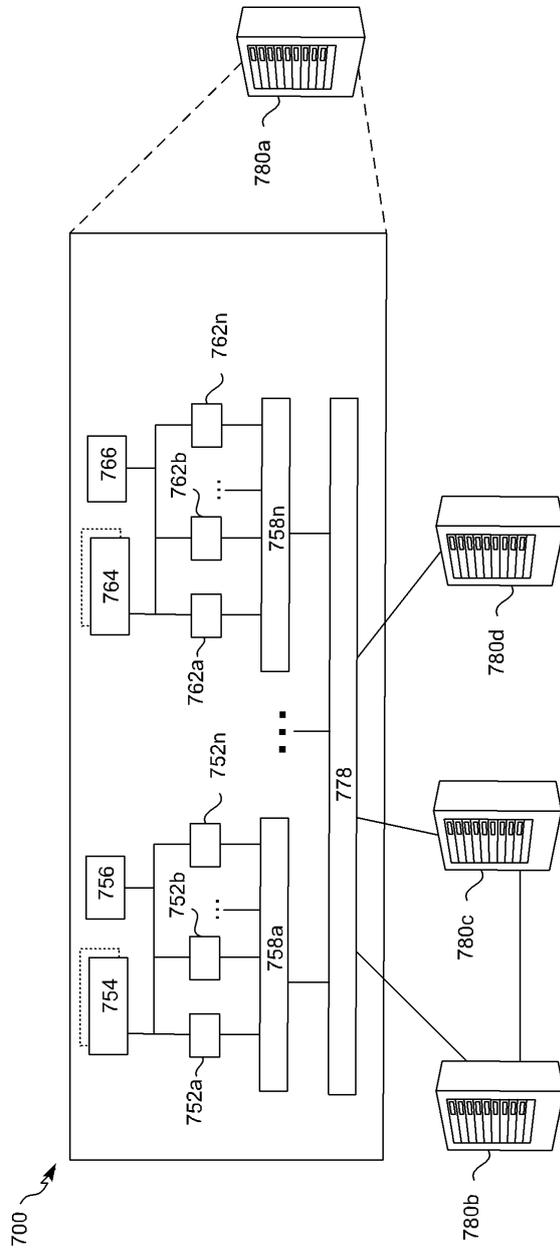
ФИГ. 5

6/7



ФИГ. 6

717



ФИГ. 7