



ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(52) СПК

G06F 16/2477 (2019.02); G06F 16/248 (2019.02)

(21)(22) Заявка: 2017108205, 14.09.2015

(24) Дата начала отсчета срока действия патента:
14.09.2015

Дата регистрации:
14.06.2019

Приоритет(ы):

(30) Конвенционный приоритет:
15.09.2014 US 14/486,638

(43) Дата публикации заявки: 14.09.2018 Бюл. № 26

(45) Опубликовано: 14.06.2019 Бюл. № 17

(85) Дата начала рассмотрения заявки РСТ на
национальной фазе: 14.03.2017

(86) Заявка РСТ:
US 2015/049876 (14.09.2015)

(87) Публикация заявки РСТ:
WO 2016/044114 (24.03.2016)

Адрес для переписки:
129090, Москва, ул. Большая Спасская, д. 25,
строение 3, ООО "Юридическая фирма
Городисский и Партнеры"

(72) Автор(ы):

НОВИК Лев (US),
НАНО Оливье (US),
ДОС САНТОС Иво Хосе Гарсиа (US),
БУССАРД Лоран (US),
РЭЙЗМАН Алекс (US),
БАНИК Дипанджан (US)

(73) Патентообладатель(и):

МАЙКРОСОФТ ТЕКНОЛОДЖИ
ЛАЙСЕНСИНГ, ЭлЭлСи (US)

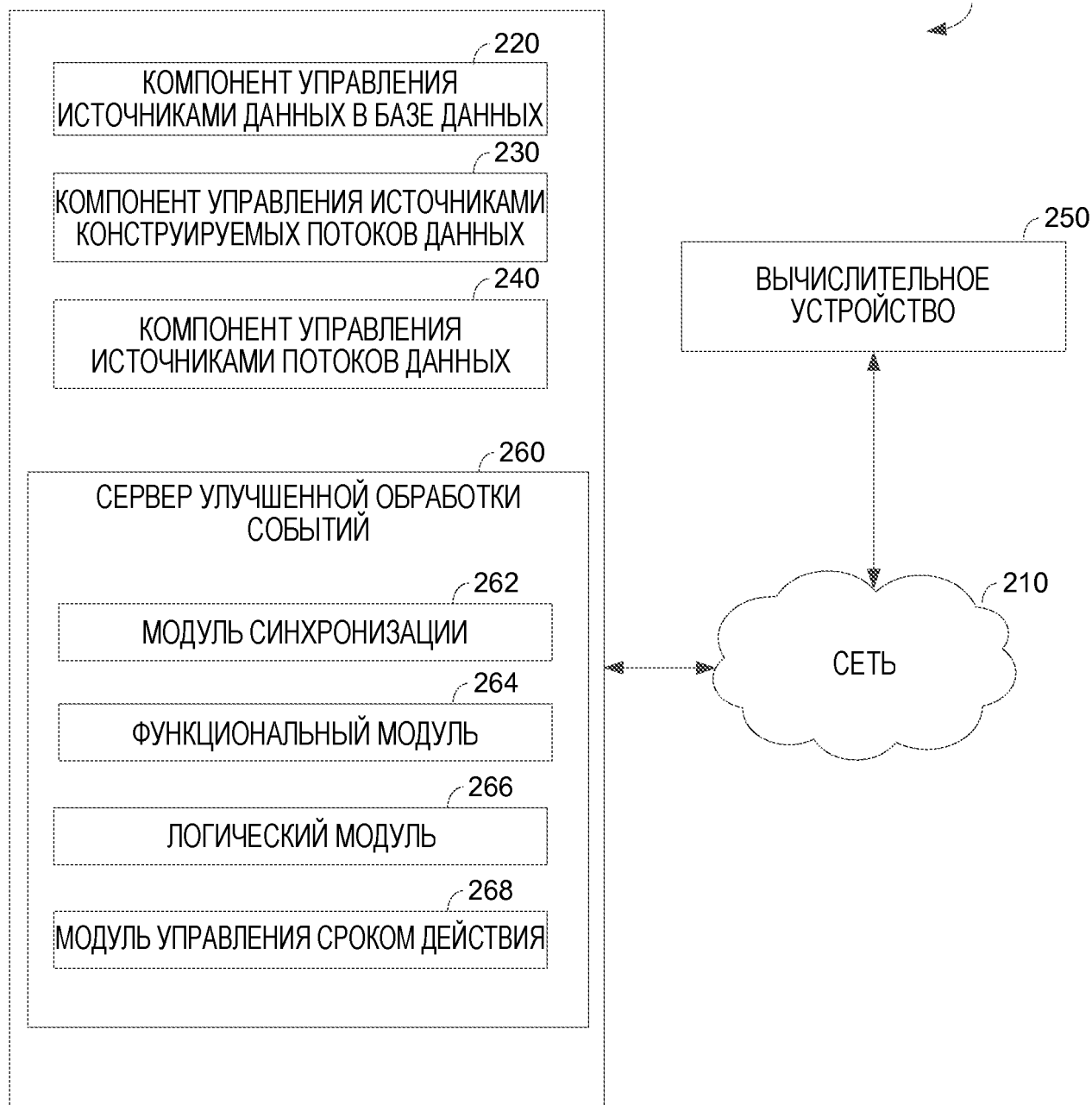
(56) Список документов, цитированных в отчете
о поиске: US 2014/0095462 A1, 03.04.2014. US
8122008 B2, 21.02.2012. US 8296316 B2,
23.10.2012. RU 2333530 C2, 10.09.2008.

(54) КОНСТРУИРУЕМЫЙ ПОТОК ДАННЫХ ДЛЯ УЛУЧШЕННОЙ ОБРАБОТКИ СОБЫТИЙ

(57) Реферат:

Изобретение относится к обработке событий. Технический результат – обеспечение улучшенной обработки событий. Способ для улучшенной обработки событий содержит этапы, на которых принимают запрос на улучшенную обработку событий, причем запрос на улучшенную обработку содержит индикатор входных данных из источника данных в базе данных, осуществляют доступ с использованием конструируемого потока данных к входным

данным, указываемым в запросе, при этом конструируемый поток данных предоставляет доступ к совокупности моментальных снимков во времени данных в базе данных из источника данных в базе данных, формируют вывод для запроса на улучшенную обработку событий на основе входных данных, к которым осуществляется доступ с использованием данных конструируемого потока данных. 3 н. и 7 з.п. ф-лы, 5 ил.



ФИГ. 2



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(12) **ABSTRACT OF INVENTION**

(52) CPC

G06F 16/2477 (2019.02); *G06F 16/248* (2019.02)(21)(22) Application: **2017108205, 14.09.2015**(24) Effective date for property rights:
14.09.2015Registration date:
14.06.2019

Priority:

(30) Convention priority:
15.09.2014 US 14/486,638(43) Application published: **14.09.2018 Bull. № 26**(45) Date of publication: **14.06.2019 Bull. № 17**(85) Commencement of national phase: **14.03.2017**(86) PCT application:
US 2015/049876 (14.09.2015)(87) PCT publication:
WO 2016/044114 (24.03.2016)

Mail address:

**129090, Moskva, ul. Bolshaya Spasskaya, d. 25,
stroenie 3, OOO "Yuridicheskaya firma
Gorodisskij i Partnery"**

(72) Inventor(s):

**NOVIK Lev (US),
NANO Olive (US),
DOS SANTOS Ivo Khose Garsia (US),
BUSSARD Loran (US),
REJZMAN Aleks (US),
BANIK Dipandzhan (US)**

(73) Proprietor(s):

**MAJKROSOFT TEKNOLODZHI
LAJSENSING, EIEISI (US)**(54) **CONSTRUCTED DATA STREAM FOR IMPROVED EVENT PROCESSING**

(57) Abstract:

FIELD: calculating; counting.

SUBSTANCE: invention relates to event processing. Method for improved event processing comprises steps of receiving a request for improved event processing, wherein a request for improved processing comprises an input data indicator from a data source in a database, performing access using constructed data stream to input data specified in request, wherein the constructed

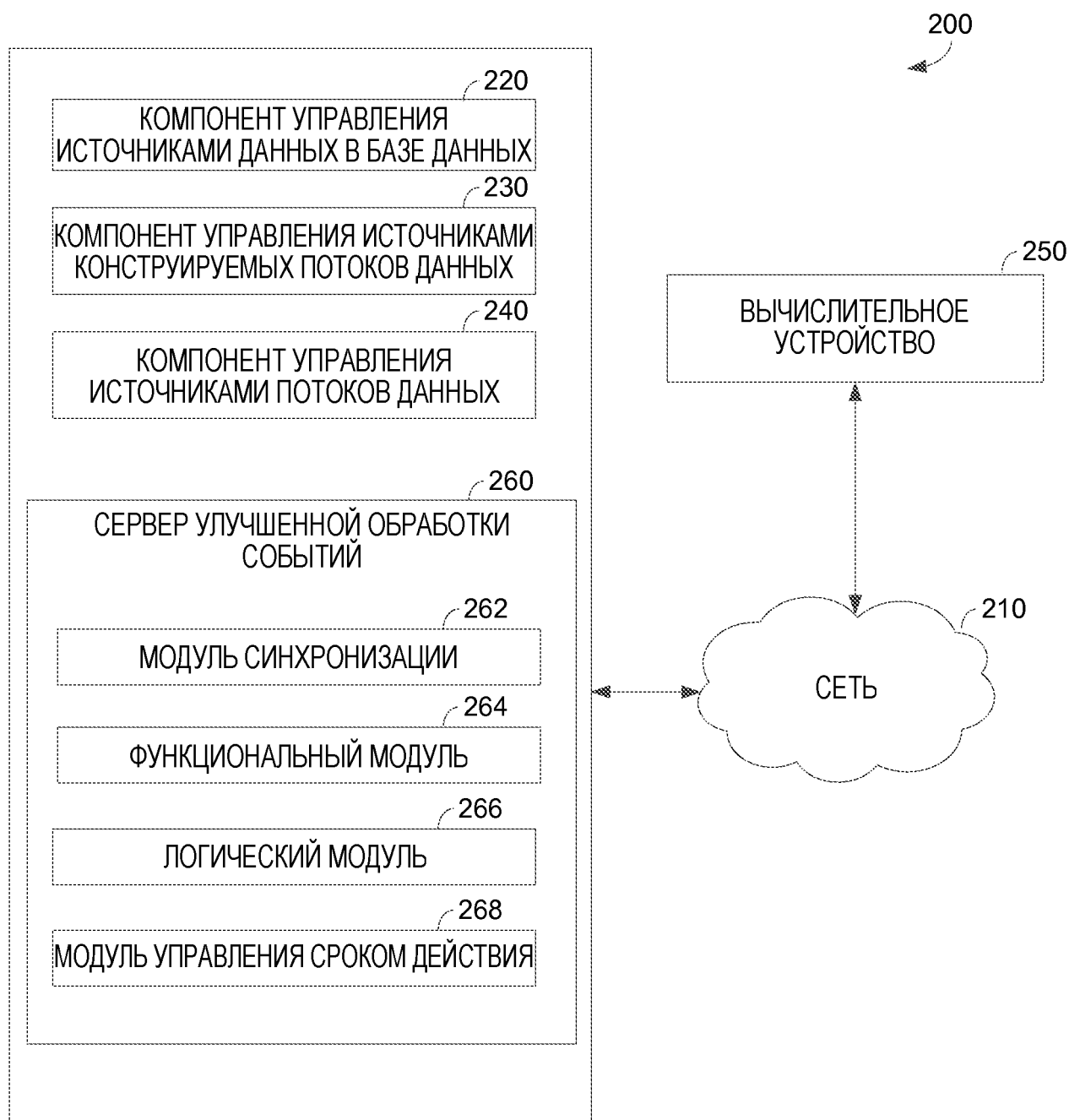
data stream provides access to a plurality of snapshots in time of data in the database from a data source in a database, generating an output for a request for improved processing of events based on input data accessed using data of the constructed data stream.

EFFECT: technical result is improved processing of events.

10 cl, 5 dwg

R U 2 6 9 1 5 9 5 C 2

R U 2 6 9 1 5 9 5 C 2



ФИГ. 2

Уровень техники

[0001] Обработка событий означает отслеживание и анализ потоков информации, чтобы извлекать из них заключение. Потоки информации, в частности, могут представлять события. Обработка событий также включает в себя обработку событий, которая комбинирует данные из нескольких источников данных для того, чтобы логически выводить события или шаблоны, которые предлагают более сложные обстоятельства. В связи с этим, обработка событий позволяет упрощать идентификацию значимых событий и реагировать на них максимально быстро.

Сущность изобретения

[0002] Эта сущность изобретения предоставлена для того, чтобы представлять в упрощенной форме выбор концепций, которые дополнительно описаны ниже в подробном описании. Эта сущность не имеет намерение идентифицировать ключевые или важнейшие признаки заявляемого предмета изобретения, а также не имеет намерение изолированно использоваться в качестве помощи при определении объема заявленного предмета изобретения.

Варианты осуществления, описанные в данном документе, направлены на способы, системы и компьютерные носители хранения данных для улучшенной обработки событий. Запрос на улучшенную обработку событий может приниматься, причем запрос на улучшенную обработку содержит индикатор относительно входных данных из источника данных в базе данных, которые должны использоваться для того, чтобы извлекать выходные данные. К входным данным, указываемым в запросе на улучшенную обработку событий, осуществляется доступ с использованием источника конструируемых потоков данных, а не источника данных в базе данных. Источник данных конструируемого потока данных предоставляет доступ к совокупности моментальных снимков во времени данных в базе данных из источника данных в базе данных. К данным из источника конструируемых потоков данных может осуществляться доступ в качестве функции от выбранного значения и выбранного времени. Данные конструируемого потока данных содержат временной элемент. Данные конструируемого потока данных включают в себя временной элемент, который недоступен в данных в базе данных, временной элемент упрощает выполнение улучшенной обработки событий для данных в базе данных с использованием данных конструируемого потока данных. Вывод для запроса на улучшенную обработку событий формируется на основе входных данных, к которым осуществляется доступ с использованием данных конструируемого потока данных.

Улучшенная обработка событий дополнительно может включать в себя комбинирование источника потоков данных и источника конструируемых потоков данных, причем источник конструируемых потоков данных выступает в качестве прокси для источника данных в базе данных. Запрос на улучшенную обработку событий может включать в себя как первый индикатор относительно входных данных из источника данных в базе данных, так и второй индикатор относительно ввода из источника потоков данных. В этом отношении, к входным данным, указываемым в первом индикаторе относительно входных данных, может осуществляться доступ через источник конструируемых потоков данных, и к входным данным, указываемым во втором индикаторе, может осуществляться доступ через источник потоков данных. Входные данные комбинируются и коррелируются с использованием первого временного элемента конструируемого потока данных и второго временного элемента потока данных для того, чтобы формировать извлеченный вывод для запроса на улучшенную обработку событий. В вариантах осуществления, механизм управления сроком действия

предоставляет алгоритмы и процедуры для указания неопровергнутых выводов для запросов на улучшенную обработку событий. В частности, временной элемент срока действия указывает время, когда вывод на основе одного или более источников данных точно отражает эти данные в источнике данных в это время.

Краткое описание чертежей

[0003] Ниже подробно описывается настоящее изобретение со ссылкой на прилагаемые чертежи, на которых:

[0004] Фиг. 1 является блок-схемой примерного вычислительного окружения, подходящего для использования в реализации вариантов осуществления, описанной в данном документе;

[0005] Фиг. 2 является блок-схемой примерного операционного окружения, в котором могут использоваться варианты осуществления, описанные в данном документе;

[0006] Фиг. 3 является блок-схемой последовательности операций, показывающей примерный способ для выполнения улучшенной обработки событий, в соответствии с вариантами осуществления, описанными в данном документе;

[0007] Фиг. 4 является блок-схемой последовательности операций, показывающей примерный способ для выполнения улучшенной обработки событий, в соответствии с вариантами осуществления, описанными в данном документе; и

[0008] Фиг. 5 является блок-схемой последовательности операций, показывающей примерный способ для выполнения улучшенной обработки событий, в соответствии с вариантами осуществления, описанными в данном документе.

Подробное описание изобретения

[0009] Предмет вариантов осуществления изобретения описывается с конкретикой в данном документе, чтобы удовлетворять обязательным требованиям. Тем не менее, само описание не предназначено ограничивать объем защиты этого патента. Наоборот, изобретатели предполагают, что заявленный предмет изобретения также может быть осуществлен другими способами, так что он включает в себя другие этапы или комбинации этапов, аналогичные этапам, описанным в этом документе, в связи с другими современными или будущими технологиями. Более того, хотя термины "этап" и/или "блок" могут использоваться в данном документе для того, чтобы означать различные элементы используемых способов, термины не должны быть интерпретированы как подразумевающие какой-либо отдельный порядок для/между различными этапами, раскрытыми в данном документе, за исключением и кроме случаев, когда порядок отдельных этапов описывается явно.

[0010] Для целей этого раскрытия сущности, слово "включающий в себя" имеет широкий смысл, идентичный смыслу слова "содержащий". Помимо этого, такие слова, как "а" и "an", если не указано иное, включают в себя множественное число, а также единственное число. Таким образом, например, требование "признака" удовлетворяется, если присутствуют один или более признаков. Кроме того, термин "или" включает в себя конъюнктивное суждение, дизъюнктивное суждение, а также и то, и другое ("а или b" в силу этого включает в себя "или а, или b", а также "а и b").

[0011] Для целей нижеприведенного подробного пояснения, варианты осуществления описываются со ссылкой на проведение инвентаризации стоечного оборудования в центре обработки и хранения данных и в качестве примера, на проведение инвентаризации оборудования перед самоинициализацией. Тем не менее, поток обработки по проведению может использоваться для проведения инвентаризации нового стоечного оборудования для множества различных типов аппаратных услуг, и самоинициализация центра обработки и хранения данных является просто примерной.

Предполагается, что описанные технологии могут быть расширены на другие контексты реализации.

[0012] Обработка событий означает отслеживание и анализ потоков информации, чтобы извлекать из них заключение. Потоки информации, в частности, могут представлять события. Обработка событий также включает в себя обработку событий, которая комбинирует данные из нескольких источников данных для того, чтобы логически выводить события или шаблоны, которые предлагают более сложные обстоятельства. В связи с этим, обработка событий позволяет упрощать идентификацию значимых событий (например, потенциальных возможностей или рисков) и реагировать на них максимально быстро. События, в общем, могут включать в себя целеуказания на потенциальных покупателей, вызов в службу заказов или обслуживания клиентов, новые товары, текстовые сообщения, сообщения в средствах социального общения, новостные ленты фондового рынка, отчеты по трафику, прогнозы погоды, данные Интернета вещей или другие виды данных.

[0013] Источник данных обработки событий может включать в себя оцифрованные данные в потоках данных и базах данных. В примерном сценарии, обработка потоков данных может включать в себя отфильтровывание или агрегирование событий на основе идентифицированных критериев. Другие типы обработки могут выполняться на основе различных событий. Обработка событий также может быть основана на анализе нескольких входящих наборов данных из различных источников потоков данных.

[0014] Источник данных обработки событий также может представлять собой базу данных. База данных является организованной совокупностью данных. Данные типично организуются для того, чтобы моделировать аспекты реальности таким способом, который поддерживает требуемые процессы. Например, моделирование клиентских транзакций таким способом, который поддерживает знание того, кем является клиент и где живет клиент (например, имя клиента, адрес клиента). Система управления базами данных поддерживает таблицы базы данных (например, таблицы реляционной базы данных) в приложениях, которые взаимодействуют с пользователями и другими приложениями, чтобы захватывать и анализировать данные.

[0015] Данные в базе данных, в частности, могут включать в себя опорные данные, которые включают в себя набор связанных или классифицированных глобальных данных. Опорные данные могут быть данными, ассоциированными с событием, которое сохраняется в таблице базы данных. Опорные данные могут быть доступными в доверенных открытых источниках или от поставщиков премиального коммерческого контента. Опорные данные также могут включать в себя набор допустимых значений, которые должны использоваться посредством других полей данных.

[0016] Взаимодействие с таблицами базы данных может выполняться с использованием декларативной инфраструктуры, которая использует логику, внутренне ассоциированную с таблицами базы данных. Декларативная инфраструктура может минимизировать описание последовательности операций управления (т.е. порядка, в котором выполняются отдельные операторы, инструкции или вызовы функций). В этом отношении, взаимодействие с таблицами базы данных может упрощаться посредством простого указания того, какие данные необходимы из таблиц базы данных, а не того, как добиться получения данных из таблиц. В качестве примера, SQL (язык структурированных запросов) может использоваться для управления данными, хранимыми в реляционных базах данных.

[0017] Данные в базе данных, включающие в себя опорные данные, не моделируют

события потока. Например, данные в базе данных не сконфигурированы в качестве временных данных (например, временного элемента, такого как время), в силу чего операции потока данных, которые основываются на временных элементах событий потока данных, не могут выполняться для данных в базе данных. В связи с этим, операции обработки событий на основе данных в базе данных или комбинации данных в базе данных и данных потока данных могут преимущественно использовать различную инфраструктуру для улучшения запросов на обработку событий, ассоциированных только с источником данных в базе данных или в комбинации с источником потоков данных. Без инфраструктуры, которая задает улучшенную обработку событий данных в базе данных, потоков данных и комбинаций вышеозначенного, операции, которые могут выполняться для потоков данных, не могут выполняться для данных в базе данных, за счет этого ограничивая возможности для логического вывода событий и шаблонов из этих двух источников данных.

[0018] Альтернативные решения могут включать в себя активное отслеживание временных меток для записей данных в базе данных. Например, начальное время и конечное время для каждой записи данных в базе данных. Тем не менее, такие реализации обычно становятся очень затратными вследствие дополнительного объема данных, который должен сохраняться, и в свою очередь, снижают эффективность реляционных баз данных. Дополнительно, альтернативные решения не являются простыми готовыми решениями и не имеют обратной совместимости, поскольку логика для обработки событий с использованием данных в базе данных не сконфигурирована с возможностью работать с данными в базе данных в состоянии, в котором данные захвачены в таблицах базы данных. Иногда, таблицы базы данных, возможно, должны быть заполнены, чтобы реализовывать обработку событий с данными в базе данных.

[0019] Соответственно, варианты осуществления настоящего изобретения предоставляют простые и эффективные способы и системы для улучшенной обработки событий на основе источника конструируемых потоков данных. Запрос на улучшенную обработку событий может формироваться и передаваться, чтобы извлекать выходные данные. Запрос на улучшенную обработку событий указывает входные данные, к которым должен осуществляться доступ из источника данных в базе данных. Улучшенная обработка событий предоставляет осуществление доступа к входным данным с использованием источника конструируемых потоков данных. Источник конструируемых потоков данных предоставляет доступ к совокупности моментальных снимков данных в базе данных, извлекаемых во времени таким образом, что к данным из источника конструируемых потоков данных может осуществляться доступ в качестве функции от выбранного значения и выбранного времени.

[0020] К входным данным осуществляется доступ в качестве конструируемого потока данных, имеющего временной элемент. Данные конструируемого потока данных включают в себя временной элемент, который недоступен в источнике данных в базе данных, временной элемент упрощает выполнение улучшенной обработки событий для данных в базе данных с использованием конструируемого потока данных. В связи с этим, операции потока данных, которые не могут выполняться с использованием данных в базе данных, вместо этого могут выполняться с использованием данных конструируемого потока данных. Вывод для запроса на улучшенную обработку событий формируется на основе данных конструируемого потока данных. Конструируемый поток данных может использоваться для того, чтобы выполнять запрос на поиск на основе выбранного значения и выбранного времени. В этом отношении, конструируемый поток данных может использоваться для того, чтобы извлекать данные в базе данных

на основе запросов.

[0021] Улучшенная обработка событий дополнительно может включать в себя комбинирование, по меньшей мере, одного источника потоков данных и, по меньшей мере, одного источника конструируемых потоков данных, причем, по меньшей мере, один источник конструируемых потоков данных выступает в качестве прокси для источника данных в базе данных. В этом отношении, запрос на улучшенную обработку событий может включать в себя как первый индикатор относительно входных данных (например, опорных данных) из источника данных в базе данных, так и второй индикатор относительно входных данных (например, данных событий) из источника потоков данных. К входным данным, указываемым в первом индикаторе относительно входных данных, может осуществляться доступ через источник конструируемых потоков данных, и к входным данным, указываемым во втором индикаторе, может осуществляться доступ через источник потоков данных. Входные данные комбинируются и коррелируются с использованием первого временного элемента данных конструируемого потока данных и второго временного элемента данных потока данных для того, чтобы формировать извлеченный вывод для запроса на улучшенную обработку событий. Поток данных и конструируемый поток данных могут проще коррелироваться при выполнении операций обработки событий.

[0022] Обработка событий с конструируемым потоком данных сконфигурирована с возможностью гарантировать повторяемость. Повторяемость или повторяющееся вычисление означает повторное выполнение анализа обработки событий впоследствии с идентичным источником данных и формирование идентичных результатов. Например, запрос, выполняемый для передаваемого вживую потока данных в первое время, приводит к идентичному результату при выполнении во второе время после первого времени. Для конструируемого потока данных и потока данных, запрос, комбинирующий входные данные из обоих потоков, приводит к идентичным результатам, даже если данные в базе данных, из которых формируется конструируемый поток данных, изменены. При работе, временной элемент (например, время приложения) идентифицируется с возможностью соответствовать данным в базе данных для того, чтобы формировать конструируемый поток данных. Конструируемый поток данных содержит дискретизированные моментальные снимки данных в базе данных, так что к конструируемому потоку данных, имеющему временной элемент, может осуществляться доступ, чтобы выполнять введенный запрос. В силу повторяемости, обработка событий может включать в себя проведение сравнения между запросами, которые выполняются к идентичным входным данным. Запросы могут сравниваться, чтобы оценивать и оптимизировать характер запроса, конструировать альтернативные версии запроса, чтобы формировать улучшенный запрос, сравнивать результаты между варьированиями запроса или выполнять альтернативную версию запроса, чтобы получать дополнительную информацию.

[0023] Инфраструктура улучшенной обработки событий ("улучшенная инфраструктура") дополнительно включает в себя конструируемый поток данных и дополнительные модули/компоненты и механизмы (например, алгоритмы и процедуры), которые предоставляют систему поддержки для реализации обработки событий на основе потока данных и данных в базе данных в качестве конструируемого потока данных. В вариантах осуществления, механизм управления сроком действия предоставляет алгоритмы и процедуры для указания неопровергнутых выводов для запросов на улучшенную обработку событий. В частности, временной элемент срока действия указывает время, когда вывод на основе одного или более источников данных

точно отражает эти данные в источнике данных в это время. В этом отношении, варианты осуществления, описанные в данном документе, могут преимущественно предоставлять возможность статистических запросов при поддержании срока действия и производительности временных запросов состояния.

5 [0024] Различные термины используются в этом описании. Хотя дополнительные сведения относительно различных терминов предоставляются в этом описании, общие определения некоторых терминов включены ниже для того, чтобы предоставлять более ясное понимание идей, раскрытых в данном документе:

[0025] Запрос на улучшенную обработку событий означает запрос, имеющий один
10 или более вводов, которые включают в себя, по меньшей мере, данные в базе данных, которые не ассоциированы с временным элементом, а вместо этого доступны с использованием конструируемого потока данных, имеющего временной элемент. Улучшенная обработка событий также может включать в себя вводы как для данных в базе данных, так и для потока данных, которые комбинируются для того, чтобы
15 формировать вывод.

[0026] Данные в базе данных означают организованную совокупность данных. Данные в базе данных, в частности, могут включать в себя опорные данные, которые включают в себя набор связанных или классифицированных глобальных данных. Данные в базе данных, возможно, не имеют временной элемент, ассоциированный с
20 данными в базе данных. Временной элемент при использовании в данном документе может означать конкретное время, когда возникает событие, ассоциированное с записью в таблице базы данных.

[0027] Поток данных означает оцифрованные данные в потоках данных. Обработка событий может быть основана на анализе нескольких входящих наборов данных из
25 различных источников потоков данных. В связи с этим, обработка событий позволяет упрощать идентификацию значимых событий (например, потенциальных возможностей или рисков) и реагировать на них максимально быстро. События, в общем, могут включать в себя целеуказания на потенциальных покупателей, вызов в службу заказов или обслуживания клиентов, новые товары, текстовые сообщения, сообщения в
30 средствах социального общения, новостные ленты фондового рынка, отчеты по трафику, прогнозы погоды, данные Интернета вещей или другие виды данных.

[0028] Конструируемый поток данных означает данные, которые формируются из данных в базе данных в качестве совокупности моментальных снимков во времени. Конструируемый поток данных ассоциирован с временным элементом, поскольку
35 доступ к такому источнику конструируемых потоков данных может осуществляться в качестве функции от выбранного значения и выбранного времени. Доступ к конструируемому потоку данных может осуществляться в качестве потока данных с временными элементами, а также осуществляться с использованием механизма поиска, который извлекает данные по запросу из конструируемого потока данных на основе
40 запроса.

[0029] Временной элемент означает конкретное время, когда возникает событие. Временной элемент может быть ассоциирован с событием, которое вводится в базу данных, в которой временной элемент не сохраняется. Временной элемент может быть ассоциирован с событием потока данных. Временной элемент также может быть
45 ассоциирован с событием конструируемого потока данных, которое извлекается из момента, когда событие возникает и сохраняется в качестве части моментального снимка данных в базе данных. Конструируемый поток данных также может быть ассоциирован с несколькими временными элементами, так что конструируемый поток

данных поддерживает бивременные присоединения, как подробнее описано ниже.

[0030] Соответственно, в первом варианте осуществления, описанном в данном документе, предусмотрен реализуемый компьютером способ для улучшенной обработки событий. Способ содержит прием запроса на улучшенную обработку событий. Запрос на улучшенную обработку событий содержит индикатор относительно входных данных из источника данных в базе данных. Способ дополнительно включает в себя осуществление доступа к входным данным, указываемым в индикаторе относительно входных данных с использованием источника конструируемых потоков данных, при этом источник конструируемых потоков данных предоставляет доступ к совокупности моментальных снимков во времени. К входным данным осуществляется доступ с использованием источника конструируемых потоков данных, который включает в себя временной элемент. Способ также включает в себя формирование вывода для запроса на улучшенную обработку событий на основе входных данных, к которым осуществляется доступ с использованием конструируемого потока данных.

[0031] Во втором варианте осуществления, описанном в данном документе, предусмотрены один или более компьютерных носителей хранения данных, сохраняющих машиноприменимые инструкции, которые при использовании посредством одного или более вычислительных устройств инструктируют одному или более вычислительных устройств осуществлять способ для улучшенной обработки событий. Способ включает в себя идентификацию источника данных в базе данных. Способ дополнительно включает в себя дискретизацию, по меньшей мере, части данных в базе данных из источника данных в базе данных для того, чтобы формировать один или более моментальных снимков данных в базе данных. Способ также включает в себя формирование конструируемого потока данных на основе одного или более моментальных снимков данных в базе данных. Конструируемый поток данных ассоциирован с временным элементом. Способ дополнительно включает в себя предоставление доступа к конструируемому потоку данных на основе запроса на улучшенную обработку событий; при этом запрос на улучшенную обработку событий ассоциирован с индикатором относительно входных данных в базе данных источника данных в базе данных, доступного с использованием конструируемого потока данных.

[0032] В третьем варианте осуществления, описанном в данном документе, предусмотрена компьютерная система для выполнения улучшенной обработки событий. Система содержит компонент управления источниками данных в базе данных, сконфигурированный с возможностью предоставления доступа к данным в базе данных. Система включает в себя компонент управления источниками конструируемых потоков данных, сконфигурированный с возможностью: идентификации источника данных в базе данных; дискретизации, по меньшей мере, части данных в базе данных из источника данных в базе данных для того, чтобы формировать один или более моментальных снимков данных в базе данных; формирования конструируемого потока данных на основе одного или более моментальных снимков данных в базе данных, при этом конструируемый поток данных ассоциирован с временным элементом; и предоставления доступа к конструируемому потоку данных на основе запроса на улучшенную обработку событий. Запрос на улучшенную обработку событий ассоциирован, по меньшей мере, с индикатором относительно входных данных в базе данных источника данных в базе данных, доступного с использованием конструируемого потока данных. Система также включает в себя источник потоков данных, сконфигурированный с возможностью: предоставления доступа к потоку данных.

[0033] Система включает в себя сервер обработки событий, сконфигурированный с

возможностью: приема запроса на улучшенную обработку событий, запрос на улучшенную обработку событий содержит индикатор относительно входных данных из источника данных в базе данных и второй индикатор относительно входных данных из потока данных; осуществления доступа к входным данным в индикаторе
 5 относительно входных данных с использованием источника конструируемых потоков данных, источник конструируемых потоков данных предоставляет доступ к совокупности моментальных снимков данных в базе данных из источника данных в базе данных во времени, доступ к входным данным осуществляется с использованием конструируемого потока данных, который содержит первый временной элемент;
 10 осуществления доступа к входным данным во втором индикаторе относительно входных данных с использованием источника потоков данных; корреляции первого временного элемента конструируемого потока данных со вторым временным элементом потока данных; и формирования вывода для запроса на улучшенную обработку событий на основе входных данных конструируемого потока данных и входных данных потока
 15 данных.

[0034] После краткого описания общего представления вариантов осуществления настоящего изобретения, ниже описывается примерное операционное окружение, в котором могут реализовываться варианты осуществления настоящего изобретения, чтобы предоставлять общий контекст для различных аспектов настоящего изобретения.

20 Ссылаясь сначала на фиг. 1, в частности, примерное операционное окружение для реализации вариантов осуществления настоящего изобретения показано и обозначено, в общем, как вычислительное устройство 100. Вычислительное устройство 100 является всего лишь одним примером подходящего вычислительного окружения и не имеет намерение предлагать какое-либо ограничение относительно объема использования
 25 или функциональности изобретения. Вычислительное устройство 100 ни при каких условиях не должно интерпретироваться как имеющее какую-либо зависимость или требование, связанное с каким-либо одним или комбинацией иллюстрированных компонентов.

[0035] Изобретение может быть описано в общем контексте компьютерного кода или исполняемые компьютером инструкции, включающих в себя исполняемые компьютером инструкции, такие как программные модули, выполняемые посредством компьютера или другой машины, такой как персональное цифровое устройство или другое карманное устройство. В общем, программные модули включают в себя алгоритмы, программы, объекты, компоненты, структуры данных и т.д., называемые
 35 "кодом", которые выполняют отдельные задачи или реализуют отдельные абстрактные типы данных. Изобретение может осуществляться на практике во множестве конфигураций систем, включающих в себя карманные устройства, бытовую электронную аппаратуру, компьютеры общего назначения, более специализированные вычислительные устройства и т.д. Изобретение также может осуществляться на практике
 40 в распределенных вычислительных окружениях, в которых задачи выполняются посредством удаленных устройств обработки, которые связываются через сеть связи.

[0036] Со ссылкой на фиг. 1, вычислительное устройство 100 включает в себя шину 110, которая непосредственно или косвенно соединяет следующие устройства: запоминающее устройство 112, один или более процессоров 114, один или более
 45 компонентов 116 представления, порты 118 ввода/вывода, компоненты 120 ввода/вывода и иллюстративный источник 122 питания. Шина 110 представляет то, что может представлять собой одну или более шин (таких как адресная шина, шина данных либо комбинация вышеозначенного). Хотя различные блоки на фиг. 1 показаны с помощью

линий для понятности, фактически, разделение различных компонентов не является настолько очевидным, и метафорически более точно, линии должны быть серыми и нечеткими. Например, можно считать, что компонент представления, такой как устройство отображения, должен представлять собой компонент ввода/вывода. Так же, процессоры имеют запоминающее устройство. Следует признавать, что таков характер области техники, и вновь повторять, что схема на фиг. 1 является просто иллюстрирующей примерное вычислительное устройство, которое может использоваться в связи с одним или более вариантов осуществления настоящего изобретения. Различие между такими категориями, как "рабочая станция", "сервер", "дорожный компьютер", "карманное устройство" и т.д., не проводится, поскольку все они рассматриваются в пределах объема по фиг. 1 и ссылаются на "вычислительное устройство".

[0037] Вычислительное устройство 100 типично включает в себя множество машиночитаемых носителей. Машиночитаемые носители могут представлять собой любые носители, к которым может осуществляться доступ посредством вычислительного устройства 100, и включают в себя энергозависимые и энергонезависимые носители, съемные и стационарные носители. В качестве примера, а не ограничения, машиночитаемые носители могут содержать компьютерные носители хранения данных и среды связи.

[0038] Компьютерный носитель хранения включает в себя энергозависимые и энергонезависимые, съемные и стационарные носители, реализованные любым способом или технологией для хранения информации, такой как машиночитаемые инструкции, структуры данных, программные модули или другие данные. Компьютерные запоминающие носители включают в себя, но не только, RAM, ROM, EEPROM, флэш-память или другую технологию запоминающих устройств, CD-ROM, универсальные цифровые диски (DVD) или другие устройство хранения данных на оптических дисках, магнитные кассеты, магнитную ленту, устройство хранения данных на магнитных дисках или другие магнитные устройства хранения данных, либо любой другой носитель, который может использоваться для хранения требуемой информации и к которому может осуществляться доступ посредством вычислительного устройства 100.

Компьютерные носители хранения данных исключают сигналы как таковые.

[0039] Среда связи типично осуществляет машиночитаемые инструкции, структуры данных, программные модули или другие данные в модулированном сигнале данных, таком как несущая или другой транспортный механизм, и включают в себя любые среды доставки информации. Термин "модулированный сигнал данных" означает сигнал, который имеет одну или более характеристик, заданных или измененных таким образом, чтобы кодировать информацию в сигнале. В качестве примера, а не ограничения, среды связи включают в себя проводные среды, такие как проводная сеть или прямое проводное соединение, и беспроводные среды, такие как акустические, радиочастотные (RF), инфракрасные и другие беспроводные среды. Комбинации любых из вышеприведенных элементов также должны быть включены в объем машиночитаемых носителей.

[0040] Запоминающее устройство 112 включает в себя компьютерные носители хранения в форме энергозависимого и/или энергонезависимого запоминающего устройства. Запоминающее устройство может быть съемным, несъемным или комбинацией означенного. Примерные аппаратные устройства включают в себя полупроводниковое запоминающее устройство, накопители на жестких дисках, накопители на оптических дисках и т.д. Вычислительное устройство 100 включает в себя один или более процессоров, которые считывают данные из различных объектов,

таких как запоминающее устройство 112 или компоненты 120 ввода-вывода. Компонент (ы) 116 представления представляют индикаторы данных пользователю или в другое устройство. Примерные компоненты представления включают в себя устройство отображения, динамик, печатающий компонент, вибрирующий компонент и т.д.

5 [0041] Порты 118 ввода-вывода обеспечивают возможность логического соединения вычислительного устройства 100 с другими устройствами, включающими в себя компоненты 120 ввода-вывода, некоторые из которых могут быть встроенными. Иллюстративные компоненты включают в себя микрофон, джойстик, игровой планшет, спутниковую антенну, сканер, принтер, беспроводное устройство и т.д.

10 [0042] Обращаясь теперь к фиг. 2, блок-схема иллюстрирует примерную систему 200 улучшенной обработки событий, подходящую для использования в реализации вариантов осуществления, описанных в данном документе. Обычно, система 200 иллюстрирует операционное окружение, которое может использоваться, в числе прочего, для выполнения улучшенной обработки событий. Улучшенная обработка событий
15 может быть ассоциирована с запросами на обработку событий, указывающими входные данные, по меньшей мере, из источника данных в базе данных.

[0043] Из других компонентов (не показаны), система 200, в общем, может включать в себя компонент 220 управления источниками данных в базе данных, компонент 230
20 управления источниками конструируемых потоков данных, компонент 240 управления источниками потоков данных, вычислительное устройство 250 и сервер 260 улучшенной обработки событий, все из которых поддерживают связь между собой через сеть 270. Сеть 210 может включать в себя, без ограничения, одну или более локальных вычислительных сетей (LAN) и/или глобальных вычислительных сетей (WAN). Такие сетевые окружения широко распространены в офисах, корпоративных компьютерных
25 сетях, сетях intranet и в Интернете.

[0044] В некоторых вариантах осуществления, один или более проиллюстрированных компонентов и/или модулей могут реализовываться как автономные приложения. В
дополнительных вариантах осуществления, один или более проиллюстрированных компонентов и/или модулей могут реализовываться через вычислительное устройство,
30 как Интернет-услуга и/или как модуль в сервере 260 улучшенной обработки событий. Специалисты в данной области техники должны понимать, что компоненты и модули, проиллюстрированные на фиг. 2 являются примерными по своему характеру и числу и не должны истолковываться в качестве ограничения. Любое число компонентов и/или модулей может использоваться для того, чтобы достигать функциональности,
35 описанной в данном документе. Например, любое число вычислительных устройств, компонентов управления источниками данных в базе данных, компонентов управления источниками конструируемых потоков данных, компонентов управления источниками потоков данных, вычислительных устройств, серверов улучшенной обработки событий и сетей может использоваться в системе 200 в пределах объема вариантов осуществления
40 настоящего документа. Каждое из них может содержать одно устройство/интерфейс или несколько устройств/интерфейсов, взаимодействующих в операционном окружении. Например, несколько устройств и/или модулей, размещаемых в распределенном окружении, могут совместно предоставлять функциональность сервера 260 улучшенной обработки событий, описанную в данном документе. Фраза "приложение" или "услуга"
45 при использовании в данном документе может широко означать любое программное обеспечение или части программного обеспечения, которая запускается поверх или осуществляет доступ к ячейке запоминающего устройства в вычислительном устройстве и/или в нескольких вычислительных устройствах, к примеру, в нескольких

вычислительных устройствах в центре обработки и хранения данных.

[0045] Следует понимать, что эта и другие компоновки, описанные в данном документе, изложены только в качестве примеров. Другие компоновки и элементы (например, машины, интерфейсы, функции, порядки и/или группировки функций) могут использоваться в дополнение или вместо показанных компоновок и элементов, и некоторые элементы могут вообще опускаться. Дополнительно, многие элементы, описанные в данном документе, представляют собой функциональные объекты, которые могут реализовываться как дискретные или распределенные компоненты либо в сочетании с другими компонентами и в любой подходящей комбинации и местоположении. Различные функции, описанные в данном документе как выполняемые посредством одного или более объектов, могут выполняться посредством аппаратных средств, микропрограммного обеспечения и/или программного обеспечения. Например, различные функции, включающие в себя функции, описанные ниже относительно системы 200, могут выполняться посредством процессора, выполняющего инструкции, сохраненные в запоминающем устройстве.

[0046] Обращаясь теперь к различным компонентам, которые могут быть включены в систему 200, компонент 220 управления источниками данных в базе данных отвечает за сохранение и предоставление доступа к данным в базе данных. Компонент управления источниками данных в базе данных может сохранять данные в базе данных, имеющей таблицы реляционной базы данных. Данные в базе данных, возможно, не имеют временные элементы, ассоциированные с данными в базе данных. Временной элемент при использовании в данном документе может означать конкретное время, когда возникает событие, ассоциированное с записью в таблице базы данных. Например, для клиента, первая транзакция может захватывать первый адрес клиента. Клиент может выполнять вторую транзакцию во второе время и затем обновлять первый адрес на новый второй адрес клиента. В связи с этим, база данных обновляет адрес клиента без учета временного элемента. Таблицы базы данных включают в себя записи, которые изменяются во времени на основе новой информации, которая записывается без учета времени, ассоциированного с предыдущей записью или с новой записью. В этом отношении, запросы данных (например, запросы) в компонент обработки данных в базе данных не являются повторяющимися, поскольку данные изменяются во времени. Повторяющееся вычисление означает анализ обработки событий повторного выполнения впоследствии с идентичным источником данных и формирование идентичных результатов.

[0047] Дополнительно, компонент обработки данных в базе данных может поддерживать операции, которые не включают в себя временные элементы. Запрос на информацию адреса клиента предоставляет информацию адреса клиента для клиента в то время, когда выполняется запрос. Операция для того, чтобы узнавать информацию адреса клиента, в частности, в ходе первой транзакции или в ходе второй транзакции, не поддерживается, поскольку таблица базы данных просто записывает наиболее актуальную информацию адреса клиента, ассоциированную с транзакцией.

[0048] В примерном варианте осуществления, компонент 220 управления источниками данных в базе данных может предоставлять доступ к базе данных, которая поддерживает административный временной элемент. Административный временной элемент означает временной элемент (например, время), используемый для рабочей функциональности (например, аудита, технического обслуживания и подготовки к работе) данных в базе данных. В базах данных, имеющих административный временной элемент, административные временные элементы не имеют намерение записываться в таблицах

базы данных. Тем не менее, административный временной элемент может использоваться в качестве части механизма уведомлений, который информирует один или более внешним компонентам в отношении изменения данных в базе данных. Механизм уведомлений дополнительно может реализовывать признак пакетной обработки, причем признак

5 пакетной обработки может определять предварительно определенное число изменений или определять предварительно определенный период времени перед инициированием уведомления на то, чтобы выполнять дискретизацию данных в базе данных. В связи с этим, даже если намеченный характер административного временного элемента может

10 быть таким, что административный временной элемент является скоротечным и не записывается, временной элемент может использоваться при формировании конструируемого потока данных. Предполагается, что компонент управления источниками конструируемых потоков данных может иметь механизм управления подписками для приема уведомлений в отношении изменений данных в базе данных и

формирования конструируемого потока данных с использованием дискретизации, которая создает дифференциальные моментальные снимки, как подробнее описано

15 ниже.

[0049] Дополнительно предполагается, что компонент 220 управления источниками данных в базе данных, возможно, разработан до идентификации логики улучшенной обработки событий для обработки данных в компоненте обработки данных в базе

20 данных при использовании обработки событий автономно или интегрированно с источником потоков данных. В этом отношении, компонент 220 управления источниками данных в базе данных может не включать в себя признаки, конкретно ассоциированные с выбранной логикой для улучшенной обработки событий. Другими

словами, логика улучшенной обработки событий в вариантах осуществления, описанных

25 в данном документе, сконфигурирована с возможностью функционировать с унаследованными системами или быть обратно совместимой с базой данных, которая разрабатывается независимо от логики улучшенной обработки событий, как подробнее описано ниже.

[0050] Компонент 230 управления источниками конструируемых потоков данных

30 отвечает за управление и предоставление доступа к конструируемому потоку данных. Конструируемый поток данных формируется таким образом, что доступ к нему может осуществляться на основе значения и времени, ассоциированного с данными в базе данных. В этом отношении, конструируемый поток данных является функцией от

значения и временного элемента. Конструируемый поток данных ассоциирован с

35 данными в базе данных на основе значения и временного элемента данных. Предполагается, что ключевой атрибут или набор атрибутов может использоваться для того, чтобы гарантировать, что запись существует для данных в базе данных, которые должны конструироваться в поток данных.

[0051] Конструируемый поток данных сохраняет моментальные снимки в течение

40 данной длительности, чтобы обеспечивать повторяемость. При присоединении потока данных к данным в базе данных, моментальный снимок конструируемого потока данных может задаваться посредством временного элемента (например, временной метки времени приложения) потока данных (например, передаваемого вживую потока

данных). Данные в базе данных управляют не реляционными таблицами с

45 представлением времени, а административным временным элементом (например, временами фиксации системы баз данных) в качестве времени, ассоциированного с данными в базе данных, которые дискретизируются, чтобы формировать моментальный снимок. В этом отношении, источник данных в базе данных для того, чтобы

поддерживать создание моментальных снимков со временем фиксации в качестве времени синхронизации, данные в базе данных могут быть доступными в качестве источника потоковых данных в системе обработки события.

[0052] В вариантах осуществления, источник данных в базе данных может реализовываться с политикой временных приращений, которая автоматически обеспечивает ход вперед времени на стороне системы улучшенной обработки событий даже в отсутствие модификаций моментальных снимков. Политика также дополнительно может использоваться для того, чтобы управлять обновлением на основе извлечения по запросу из источника данных в базе данных. Источник данных в базе данных может обеспечивать возможность режима инициализации (аналогичного процедуре завершения), когда может предоставляться необязательная временная метка. Временные метки не обязательно могут располагаться в каком-то конкретном порядке в этом режиме. Эта операция может только выполняться один раз перед тем, как данные в базе данных вообще используются.

[0053] Конструируемый поток данных, указываемый в нижеприведенном коде в качестве опорных данных, или refData, может поддерживать интерфейс прикладного программирования (API), который предоставляет возможность выполнения операций с использованием конструируемого потока данных. API может представлять собой интерфейс CRUD (создания, считывания, обновления и удаления), в котором пользователь может выполнять действия. Примерный сценарий в нижеприведенном коде может быть ассоциирован с прикладной услугой, которая может быть сконфигурирована с возможностью собирать регистрируемую информацию об ошибках из различных устройств, чтобы выполнять анализ практически в реальном времени. Каждое устройство может идентифицироваться посредством идентификатора устройства (например, Machine_ID). Предполагается, что идентификатор устройства может преобразовываться в соответствующее имя и сведения по устройству (например, MachineName и MachineDetails). Преобразование из идентификатора устройства в соответствующее имя устройства может сохраняться в качестве метаданных в таблице базы данных, преобразующей множество идентификаторов устройств в имена устройств. API, в частности, может поддерживать табличную схему хранилища ключей/значений, которая реализует временные элементы (например, время транзакции в качестве системного времени). Посредством нижеприведенного сравнения, API с явным временем может использоваться для инициализации конструируемого потока данных.

```
var refData = new ReferenceData<int, MachineDetails>(new Uri("http://..."));

refData.Add(1, new MachineDetails(machineId:1, machineName:"x", domain:"A"));
refData.Add(2, new MachineDetails(2, "y", "A"));
refData.Update(1, new MachineDetails(1, "x2", "A"));
refData.Delete(2);
```

```
var refData2 = new ReferenceData<int, MachineDetails, long>(new Uri("http://myRefData"));

refData2.Add(1, new MachineDetails(machineId: 1, machineName: "x", domain: "A", 1000);
refData2.Add(2, new MachineDetails(2, "y", "A", 5000);
refData2.Update(1, new MachineDetails(1, "x2", "A", 3000);
refData2.Delete(2, 5000);
refData2.Add(2, new MachineDetails(2, "yz", "ABC", 4000);
```

```
refData2.Add(2, new MachineDetails(2, "y", "A"));
```

[0054] Информация времени не извлекается в каком-то конкретном порядке, и вызовы для связанной со временем информации для источника данных в базе данных могут смешиваться с нормальными API-вызовами, которые должны задавать время как текущее системное время.

[0055] API управления конструируемыми потоками данных может реализовывать ступенчатые функции в том, что когда изменение не существует между t1 и t2, значение во время t2 является значением в t1. API может использоваться для того, чтобы считывать значение конкретного ключа в данное время:

```
refData2.MoveTimeTo(4500);

MachineDetails value = refData2.ReadAt(key: 2, time: 3500);           // (2, "y", "A")
value = refData2.ReadAt(key: 2, time: 4200);                         // null
```

[0056] Предполагается, что считывание конструируемого потока данных после последнего времени приложения в конструируемом потоке данных может запрещаться, поскольку оно приводит к невоспроизводимому результату. В случае времени приложения, механизм проверки пунктуации может использоваться для того, чтобы командовать ход вперед времени, продвигать время приложения или закрывать фазу инициализации и переключаться на системное время. В случае системного времени, варианты осуществления в данном документе могут запрещать считывание после текущего времени, как подробнее описано ниже.

[0057] API также может использоваться для того, чтобы считывать все значения в данное время:

```
Dictionary<int, MachineDetails> values = refData2.ReadAt(2500);       // {{1, {1, "x", "A"}},
                                                                    // {2, {2, "y", "A"}}}
```

[0058] Результат представляет собой все значения, которые локально известны. В зависимости от фрагментации/сегментации, он может представлять собой поднабор опорных данных в конструируемом потоке данных.

[0059] Механизм потоковой передачи может реализовываться для того, чтобы передавать в потоковом режиме конструируемый поток данных. Базовая функция считывания (например, ReadAT) API не может надлежащим образом поддерживать временные элементы конструируемого потока данных, поскольку необходимо знать, когда происходят изменения. Механизм потоковой передачи использует конструируемый поток данных для того, чтобы создавать соответствующие граничные и интервальные события. Механизм потоковой передачи включает в себя возможность создавать граничные события для каждого изменения: каждое изменение приводит к конечному граничному событию с предыдущим значением и к начальному граничному событию с новым значением. Механизм потоковой передачи также включает в себя возможность знать начальное состояние, когда конструируемый поток данных должен передаваться в потоковом режиме с конкретного времени, т.е. не с начала времени. В качестве примера:

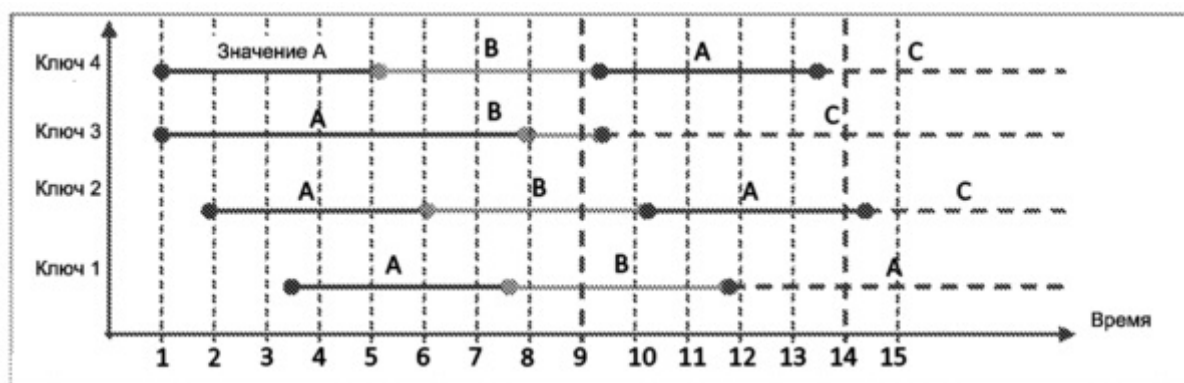


Таблица. Конструируемые потоковые данные

[0060] Потокковая передача изменений, соответствующих этим конструируемым потоковым данным от 9 до 14, должна приводить к следующему:

1) Начальное "состояние" во время 9

- Временная метка: 5,2, ключ: 4, значение: В
- Временная метка: 6,1, ключ: 2, значение: В
- Временная метка: 7,5, ключ: 1, значение: В
- Временная метка: 7,9, ключ: 3, значение: В

2) Изменения между 9 и 14

- Временная метка: 9,2, ключ: 4, значение: А
- Временная метка: 9,3, ключ: 3, значение: С
- Временная метка: 10,1, ключ: 2, значение: А
- Временная метка: 11,8, ключ: 1, значение: А
- Временная метка: 13,5, ключ: 4, значение: С

[0061] Тем не менее, создание потока граничных/интервальных событий из этого потока изменений требует отслеживания каждого ключа (последней временной метки и последнего значения), с тем чтобы создавать конечные граничные события, соответствующие предыдущему значению. Сохранение и потокковая передача изменений в качестве пары значений (старого и нового) должно это упрощать:

1) Начальное "состояние" во время 9 (может использоваться для того, чтобы создавать начальные граничные события в расчете на событие изменения)

- Временная метка: 5,2, ключ: 4, значение: В, старое: нуль
- Временная метка: 6,1, ключ: 2, значение: В, старое: нуль
- Временная метка: 7,5, ключ: 1, значение: В, старое: нуль
- Временная метка: 7,9, ключ: 3, значение: В, старое: нуль

2) Изменения между 9 и 14 (могут использоваться для того, чтобы создавать одно конечное граничное событие и одно начальное граничное событие в расчете на событие изменения)

- Временная метка: 9,2, ключ: 4, значение: А, старое: {значение: В, с момента: 5,2}
- Временная метка: 9,3, ключ: 3, значение: С, старое: {значение: В, с момента: 7,9}
- Временная метка: 10,1, ключ: 2, значение: А, старое: {значение: В, с момента: 6,1}
- Временная метка: 11,8, ключ: 1, значение: А, старое: {значение: В, с момента: 7,5}
- Временная метка: 13,5, ключ: 4, значение: С, старое: {значение: А, с момента: 9,2}

[0062] Этот поток является достаточным для того, чтобы создавать граничные события, соответствующие конструируемому потоку данных. Это используется внутренне для того, чтобы представлять конструируемый поток данных в качестве источника, который может использоваться посредством системы улучшенной обработки событий.

```

IObservable<Change<int, MachineDetails, long>> observable = refData.StreamChanges(fromTime:
0, toTime: 5000);

observable.ForEachAsync(change => Console.WriteLine(string.Format(
5   "time: {0}, key: {1}, oldValue: {2}, newValue: {3}",
    change.Time, change.Key,
    change.OldValue!=null ? change.OldValue.Domain + "\\\" + change.OldValue.MachineName :
    "null",
    change.NewValue!=null ? change.NewValue.Domain + "\\\" + change.NewValue.MachineName :
10   "null")
));

```

[0063] Предполагается, что при потоковой передаче опорных данных от t1 до t2 и затем от t2 до t3, может игнорироваться начальное состояние в t2. При этом либо имеется способ не запрашивать означенное, либо необходимо дедуплицировать

результатирующие начальные граничные события.

[0064] Конструируемые потоковые данные дополнительно могут поддерживать оператор для того, чтобы присоединять конструируемый поток данных к потоку точечных событий (например, к потокам данных). В качестве примера, ниже приведен код:

```

20   var queryDef = ReferenceData.From(refData)
    .TemporalJoin(EventStream.From(stream),
        (ref, str) => ref.MachineId == ref.Id,
        (ref, str) => new { Id = str.Id, State = str.State, Name = ref.MachineName } )
    .To(sink);

25   var query = ctx.StartQuery("q1", queryDef);

```

[0065] Параметры присоединения следующие: 1) поток, 2) предикат присоединения, 3) переключатель результатов (т.е. проекция). Предикат присоединения может игнорироваться или заменяться посредством "(ref, str)=>true", когда поток и опорные

данные уже перегруппированы посредством идентификатора машины.

[0066] Предполагается, что бивременное присоединение может выполняться с использованием конструируемого потока данных. Например, когда опорные данные в базе данных используются в качестве конструируемого потока данных с использованием механизма потоковой передачи, временной элемент (например, временная метка) событий может использоваться во временном присоединении, а когда опорные данные в базе данных используются в качестве конструируемого потока данных с использованием механизма поиска, другая временная шкала (например, другое время в рабочих данных события) может использоваться в качестве условия присоединения.

[0067] Компонент 230 управления конструируемыми потоками данных сконфигурирован с возможностью формировать конструируемый поток данных с использованием идентифицированного источника данных в базе данных, имеющего данные в базе данных, которые используются в улучшенной обработке событий. Дополнительно предполагается, что ожидаемый запрос на улучшенную обработку событий может определять и указывать заранее то, какие данные в базе данных используются при формировании конструируемого потока данных. Данные в базе данных, используемые при конструировании конструируемого потока данных, могут быть фактором ожидаемых запросов на улучшенную обработку событий, которые

должны осуществляться к данным в базе данных.

[0068] Запрос на улучшенную обработку событий включает в себя запрос на данные в компоненте 220 обработки данных в базе данных, к которому фактически осуществляется доступ с использованием конструируемого потока данных компонента

230 управления источниками конструируемых потоков данных. Конструируемый поток данных может выступать в качестве прокси для компонента обработки данных в базе данных при моделировании признаков и функциональности потока данных.

Предполагается, что конструируемый поток данных может задаваться как тип данных в системе улучшенной обработки событий. Тип данных может задаваться как схема, которая определяет то, как организован конструируемый поток данных.

[0069] При работе, конструируемый поток данных может формироваться на основе механизма дискретизации, который поддерживает дискретизацию дискретного экземпляра данных в базе данных для того, чтобы создавать моментальный снимок данных в базе данных. Дискретизация может выполняться для всей базы данных или только для идентифицированной части данных в базе данных. Дискретизация также может выполняться, в частности, на основе предполагаемых запросов на улучшенную обработку событий, так что значение, ассоциированное с запросом на улучшенную обработку событий, сохраняется в моментальном снимке базы данных.

[0070] В одном варианте осуществления, компонент 230 управления источниками конструируемых потоков данных может реализовывать механизм периодической дискретизации. Механизм периодической дискретизации поддерживает выполнение дискретизации в предварительно определенном или обозначенном временном интервале.

В качестве примера, данные в базе данных могут дискретизироваться каждые пять минут. В другом примерном варианте осуществления, конструируемый поток данных может формироваться на основе механизма дифференциальной дискретизации, который поддерживает дискретизацию для того, чтобы формировать дифференциальные моментальные снимки данных в базе данных. Дифференциальный моментальный снимок указывает изменение данных в базе данных, а не весь моментальный снимок данных в базе данных. Предполагается, что дифференциальная дискретизация для того, чтобы формировать различные моментальные снимки, может реализовываться для того, чтобы уменьшать затраты на хранение всех моментальных снимков данных в базе данных. В качестве примера, медленно изменяющийся набор данных может быть идеальным для дифференциальной дискретизации, чтобы создавать дифференциальные моментальные снимки, поскольку изменения являются нечастыми, по сути,

дифференциальные моментальные снимки захватывают все изменения, которые могут возникать с этого момента. Моментальные снимки конструируемого потока данных поддерживают повторяемость в том, что запрос к конструируемому потоку данных, имеющему моментальные снимки, предоставляет идентичный ответ, тогда как запрос к данным в базе данных может предоставлять различный ответ по мере того, как данные в базе данных изменяются во времени.

[0071] Компонент 230 управления источниками конструируемых потоков данных может реализовывать механизм управления подписками, который принимает уведомления относительно изменений данных в базе данных. Уведомление относительно изменений может быть основано на механизме уведомлений, реализованном в компоненте 220 управления источниками данных в базе данных, который использует административный временной элемент, чтобы ассоциировать изменение данных в базе данных со временем изменений. Когда уведомление принимается, компонент данных конструируемого потока данных может дискретизировать данные в базе данных и

ассоциировать дискретизированные данные в базе данных со временем в административном временном элементе. Дискретизация может быть полновыборочной или дифференциальной выборочной, как пояснено выше. В этом отношении, дискретизация основана не на периодическом механизме, а вместо этого на уведомлении относительно

5 изменения данных в базе данных. Помимо этого, корректность вывода из запроса на улучшенную обработку событий повышается в том, что данные конструируемого потока данных точно отражают изменения данных в базе данных, в отличие от механизма периодической дискретизации.

[0072] Конструируемый поток данных преимущественно является гибким в том, что

10 он может поддерживать различные типы логики улучшенной обработки событий, ассоциированной с конструируемым потоком данных, и может добиваться этого обратно совместимым способом. Конструируемый поток данных может использовать дискретизацию для того, чтобы формировать моментальный снимок данных в базе данных. Дискретизация может быть основана на механизме периодической

15 дискретизации, механизме дифференциальной дискретизации. Конструируемый поток данных предоставляет улучшенную обработку событий с повторяемостью без необходимости изменять статистическую информацию или структуру данных в базе данных или конструируемого потока данных. Логика для формирования вывода для запросов на улучшенную обработку событий, как описано в данном документе, может

20 поддерживаться с различными способами дискретизации и даже с изменениями способов дискретизации. Запрос в конструируемый поток данных преимущественно функционирует в любом случае с конструируемым потоком данных моментального снимка данных в базе данных.

[0073] С дальнейшей ссылкой на фиг. 2, источник 240 потоков данных отвечает за

25 управление и предоставление доступа к потоку данных. Потоки данных или потоки информации, в частности, могут представлять события. Данные потока данных ассоциированы с временным элементом, который указывает то, когда возникает событие, ассоциированное с данными. Потоки данных могут активно прослушиваться через прослушивающее приложение. Потоки данных также могут временно сохраняться

30 в течение короткого периода времени (например, 5 минут) или длительного периода времени (например, от года до нескольких лет) и воспроизводиться, как если данные активно передаются в потоковом режиме. В качестве примера, запрос на обработку событий может использоваться для того, чтобы сравнивать новые потоковые данные с потоковыми данными с прошлого года, чтобы извлекать заключения из их сравнений.

35 В этом отношении, потоки данных могут комбинироваться, чтобы выполнять отслеживание и анализ для того, чтобы извлекать выводы. Дополнительно, поток данных может быть включен в запрос на улучшенную обработку событий с данными в базе данных, так что поток данных комбинируется с конструируемым потоком данных для того, чтобы формировать вывод для запроса на улучшенную обработку событий.

[0074] Вычислительное устройство 250 может включать в себя, например, любой тип вычислительного устройства 100, описанного выше со ссылкой на фиг. 1. Вычислительное устройство 250 может предоставлять доступ к различным компонентам, описанным в данном документе. В частности, вычислительное устройство 250 может осуществлять доступ к серверу 260 улучшенной обработки событий, чтобы выполнять

45 одну или более операций, упрощенных посредством сервера улучшенной обработки событий, как подробнее описано в данном документе.

[0075] Сервер 260 улучшенной обработки событий отвечает за управление улучшенной обработкой событий, как описано в данном документе. Сервер 260 улучшенной

обработки событий осуществляет доступ к конструируемому потоку данных в качестве совокупности моментальных снимков в качестве функции от выбранного значения и времени. Сервер 260 улучшенной обработки событий может принимать запрос на улучшенную обработку событий. Предполагается, что запрос на улучшенную обработку событий может приниматься через вычислительное устройство 250 или с использованием услуги запроса, программируемой с возможностью формировать запросы на систематическую обработку событий. Запрос на улучшенную обработку событий может обрабатываться на основе того, ассоциирован запрос на улучшенную обработку событий с одним или более источников данных в базе данных или ассоциирован как с источниками данных в базе данных, так и с источниками потоков данных. В этом отношении, сервер 260 улучшенной обработки событий может коррелировать поток данных и данные в базе данных с использованием конкретной логики, чтобы извлекать вывод для данных. Запрос на улучшенную обработку событий может запрашивать вывод на основе вводов отслеживаемых данных для того, чтобы анализировать данные, чтобы формировать логический вывод. Вывод может быть заключением, извлекаемым как из данных в потоке данных, так и из конструируемых потоковых данных.

[0076] С дальнейшей ссылкой на фиг. 2, сервер 260 улучшенной обработки событий может включать в себя несколько модулей, включающих в себя модуль 262 синхронизации, функциональный модуль 264, логический модуль 266 и модуль 268 управления сроком действия. Модуль синхронизации может реализовывать механизм синхронизации для событий обработки. В частности, механизм синхронизации конкретно идентифицирует временной элемент и элемент истечения срока действия, ассоциированные с событием, причем временной элемент указывает время, когда происходит событие, а элемент истечения срока действия указывает то, когда событие прекращает быть доступным. В этом отношении, модуль 262 синхронизации с использованием механизма синхронизации может поддерживать обработку событий на основе того, насколько быстро наступают события, чтобы незамедлительно формировать вывод на основе синхронизации, ассоциированной с принимаемыми вводами.

[0077] Функциональный модуль 264 отвечает за управление операциями, которые могут выполняться для потока данных, данных в базе данных, конструируемого потока данных и любой комбинации вышеозначенного. В частности, запрос на обработку событий может включать в себя индикатор относительно входных данных из данных в базе данных или может включать в себя комбинированный запрос, который включает в себя индикатор относительно входных данных для данных в базе данных и потока данных. Конструируемый поток данных может выступать в качестве прокси данных в базе данных и, по сути, обеспечивает возможность выполнения различных типов операций с использованием событий, ассоциированных с конструируемым потоком данных и данными в базе данных.

[0078] Запросы на улучшенную обработку событий могут включать в себя временные элементы, которые в противном случае должны быть невозможными через прямой доступ к данным в базе данных. В этом отношении, функциональный модуль 264 может поддерживать механизм запросов, который является декларативным, чтобы выдавать запросы данных улучшенной обработки событий. Декларативный механизм запросов минимизирует или исключает то, как должны работать операции, и поддерживает больше из того, чего должны достигать операции. Например, пользователь может вводить простую операцию присоединения, чтобы присоединять поток данных к опорным данным из таблицы базы данных с использованием конструируемого потока

данных, идентично объединению двух таблиц традиционной базы данных.

[0079] В качестве примера, предположим поток событий, соответствующий химическому анализу, проводимому посредством различных инструментов:

UseCase1_stream

Assay_time	Chem_id	Instrument_id
8:10	1	1
8:20	1	2
8:30	2	1
8:40	3	2

[0080] Чтобы знать, сколько тестов данного типа (например, H1AC) проведено, необходимо снабжать события именем (преобразование "идентификатор - имя" находится в опорных данных). В силу этого предположим, что следующий моментальный снимок является действительным с 8:00 до 9:00.

Chem_Id	Название	Описание
1	H1AC	Тест на гемоглобин Ale для диабета
2	Xyz	Тест xyz
3	Abc	Тест abc

[0081] Чтобы заменять chem_id на имя, запрос может быть следующим:

```

1  SELECT System.Timestamp AS Assay_time, stream.Chem_id, map.Name AS
   Chem_name,
2      stream.Instrument_id
3  FROM Timestamp(UseCase1_stream, Assay_time) stream
4  JOIN UseCase1_mapping map
5  ON stream.Chem_id = map.Chem_Id

```

[0082] Результирующий поток с именами должен быть следующим:

Assay_time	Chem_id	Chem_name	Instrument_id
8:10	1	H1 AC	1
8:20	1	H1 AC	2
8:30	2	Xyz	1
8:40	3	Abc	2

[0083] Преимущественно, формирование запросов на улучшенную обработку событий является согласованным с существующим декларативным механизмом запросов и поддерживает интуитивный подход для формирования выводов из идентифицированных вводов. В связи с этим, признаки улучшенной обработки событий являются прозрачными для пользователя, отправляющего запрос или операцию в функциональный компонент. Функциональный модуль 264 также может выполнять запрос на улучшенную обработку событий, по меньшей мере, частично на основе механизма управления сроком действия, как подробнее описано ниже.

[0084] Логический модуль 266 отвечает за управление конкретным способом улучшенной обработки событий на основе структуры конструируемого потока данных. Как пояснено выше, конструируемый поток данных может быть основан на дискретизации, которая формирует моментальные снимки данных в базе данных с использованием либо периодической дискретизации, либо дифференциальной

дискретизации. Способ дискретизации информирует логику в отношении логического модуля 266 для формирования выводов на основе вводов. Конкретный тип данных также затрагивает логику, которая может использоваться при формировании выводов.

[0085] В одном варианте осуществления, логический модуль 266 может реализовывать выбранную логику таким образом, что, операции, принятые через функциональный модуль 264, выполняются с использованием выбранной логики. Логика управления наиболее актуальными моментальными снимками использует совокупность моментальных снимков, включающих в себя наиболее актуальный моментальный снимок, чтобы выполнять операции, ассоциированные с запросом на улучшенную обработку событий. В этом отношении, данные в базе данных являются доступными вплоть до данного моментального снимка в конструируемом потоке данных, и операции выполняются с использованием совокупности моментальных снимков вплоть до наиболее актуального моментального снимка данных.

[0086] В другом примерном варианте осуществления, логический модуль 266 может реализовывать логику интерполяции, которая поддерживает аппроксимацию значения конструируемого потока данных с учетом значений из двух моментальных снимков конструируемого потока данных. Логика интерполяции может формировать промежуточное значение для конкретных типов для типов переменных данных, но не для остальных (например, для температуры, но не для адреса клиента). При работе, механизм дискретизации (например, периодической или дифференциальной) может включать в себя первое значение (например, 70°F) в первое время (например, 17:00) и второе значение (например, 73°F) во второе время (например, 17:05). Логика интерполяции сконфигурирована с возможностью аппроксимировать промежуточное значение (например, 70°F и 73°F) в качестве потенциального фактического значения, если запрос на улучшенную обработку событий выполнен для температуры в промежуточное время (например, 17:03). Предполагается, что механизм интерполяции может явно указывать то, что результирующий вывод основан на интерполяции, а не на фактических входных данных из конструируемого потока данных.

[0087] Преимущественно, логический модуль 266 может реализовывать изменение выбора логики (например, от логики управления наиболее актуальными моментальными снимками до логики интерполяции, или наоборот), при этом конструируемый поток данных является обратно совместимым с возможностью поддерживать любое изменение логики. Другими словами, характер конструируемого потока данных поддерживает изменение логики формирования выходных данных из конструируемого потока данных без восстановления непосредственно конструируемого потока данных. В качестве примера, к данным в базе данных, имеющим опорные данные, к которым осуществляется доступ либо в качестве потока данных, либо с использованием механизма поиска, может осуществляться доступ на основе различной логики в четко определенный момент времени. В частности, опорные данные могут быть показаниями температуры, снимаемыми каждую минуту (поминутно). Конструируемые потоковые данные являются функцией от снятого последнего значения показаний температуры (например, логика управления наиболее актуальными моментальными снимками). Соответственно, формируется ступенчатая функция. В случае если в другой момент времени (например, в указанное время изменения логики) логика изменяется на логику линейной интерполяции, запрос конструируемого потока данных может быть ассоциирован с логикой управления наиболее актуальными моментальными снимками вплоть до указанного времени изменения логики, и затем логика линейной интерполяции может использоваться для значений после указанного времени изменения логики. Такое

утверждение не оказывает влияние на опорные данные, накопленные в прошлом, в силу чего не требует повторных вычислений, и соответственно, конструируемый поток данных является обратно совместимым.

[0088] С дальнейшей ссылкой на фиг. 2, модуль 268 управления сроком действия отвечает за указание неопровергнутых выводов для запросов на улучшенную обработку событий. В частности, модуль 268 управления сроком действия реализует механизм управления сроком действия, который указывает временной элемент срока действия для выводов, ассоциированных с конструируемым потоком данных. Механизм управления сроком действия предоставляет алгоритмы и процедуры для указания неопровергнутых выводов для запросов на улучшенную обработку событий. В частности, временной элемент срока действия указывает время, когда вывод на основе одного или более источников данных точно отражает данные в источниках данных во время выполнения. Характер конструируемого потока данных является таким, что он может точно отражать данные в базе данных только вплоть до определенного времени до того, как один или более механизмов, поясненных в данном документе, используются для того, чтобы обновлять конструируемый поток данных. Возможность конструируемого потока данных отражать наиболее актуальные данные в базе данных является функцией дискретизации и, в частности, используемого механизма дискретизации. В связи с этим, запросы являются действительными только вплоть до времени (например, действительного временного элемента) последнего моментального снимка данных конструируемого потока данных. Другие ограничения (например, время задержки) могут делать задержку наиболее актуальной информацией, даже для данных потока данных.

[0089] Предполагается, что модуль 268 управления сроком действия может передавать срок действия для операции в функциональный модуль таким образом, что функциональный модуль может ожидать поступления всех данных событий (фактических потоков данных или конструируемых потоков данных) до санкционирования срока действия запроса. Дополнительно предполагается, что с учетом срока действия, указываемого с использованием модуля 268 управления сроком действия, функциональный модуль 264 может предотвращать выполнение запроса, имеющего время в запросе, вывод которого не должен быть ассоциирован с временным элементом срока действия. В вариантах осуществления, функциональный модуль 264 может выполнять запрос и предоставлять вывод, но ассоциировать вывод с последним временным элементом срока действия, так что известно, что запрос является действительным только вплоть до времени временного элемента срока действия.

[0090] В вариантах осуществления, срок действия может быть ускорен с использованием модуля 268 управления сроком действия. Обстоятельства (например, системные сбои или время задержки), окружающие систему обработки событий, могут задерживать доступ к данным в базе данных для обновления конструируемого потока данных. Чтобы продолжать обработку событий, модуль управления сроком действия может указывать новый временной элемент срока действия, ассоциированный с событием конструируемого потока данных, когда дискретизация для того, чтобы формировать моментальные снимки данных в базе данных, не выполнена, чтобы поддерживать временной элемент срока действия. В связи с этим, модуль 268 управления сроком действия может снижать точность конструируемого потока данных для того, чтобы совершенствовать запросы на улучшенную обработку событий.

[0091] Ссылаясь теперь на фиг. 3, показана блок-схема последовательности операций способа, которая иллюстрирует примерный способ 300 для выполнения улучшенной

обработки событий. Хотя термины "этап" и/или "блок" могут использоваться в данном документе для того, чтобы означать различные элементы используемых способов, термины не должны быть интерпретированы как подразумевающие какой-либо отдельный порядок для/между различными этапами, раскрытыми в данном документе,

за исключением и кроме случаев, когда порядок отдельных этапов описывается явно.

[0092] На этапе 310, принимается запрос на улучшенную обработку событий. Запрос на улучшенную обработку событий содержит индикатор относительно входных данных из источника данных в базе данных. Входные данные из источника данных в базе данных не ассоциированы с временным элементом для входных данных. На этапе 320, к входным данным в индикаторе относительно входных данных осуществляется доступ с использованием источника конструируемых потоков данных. Источник конструируемых потоков данных предоставляет доступ к совокупности моментальных снимков во времени. Моментальные снимки имеют данные в базе данных из источника данных в базе данных. К входным данным осуществляется доступ с использованием конструируемого потока данных, который содержит временной элемент. Временной элемент указывает время, когда возникает событие, ассоциированное с входными данными. Временной элемент конструируемого потока данных определяется из времени, когда событие конструируемого потока данных возникает и сохраняется в качестве части моментального снимка события в данных в базе данных. На этапе 330, вывод для запроса на улучшенную обработку событий формируется на основе входных данных, к которым осуществляется доступ с использованием конструируемого потока данных. Вывод ассоциирован с временным элементом срока действия, который указывает время, когда вывод на основе конструируемого потока данных точно отражает данные в данных в базе данных.

[0093] Обращаясь теперь к фиг. 4, блок-схема последовательности операций способа иллюстрирует примерный способ 400 для выполнения улучшенной обработки событий. На этапе 410, принимается запрос на улучшенную обработку событий. Запрос на улучшенную обработку событий содержит индикатор относительно входных данных из источника данных в базе данных и второй индикатор относительно входных данных из источника потоков данных. На этапе 420, к входным данным в индикаторе относительно входных данных осуществляется доступ с использованием источника конструируемых потоков данных. Источник конструируемых потоков данных предоставляет доступ к совокупности моментальных снимков данных в базе данных из источника данных в базе данных, во времени. К входным данным осуществляется доступ с использованием конструируемого потока данных, который содержит первый временной элемент.

[0094] На этапе 430, к входным данным во втором индикаторе относительно входных данных осуществляется доступ с использованием источника потоков данных. На этапе 440, первый временной элемент конструируемого потока данных коррелируется со вторым временным элементом потока данных. На этапе 450, формируется вывод для запроса на улучшенную обработку событий на основе входных данных конструируемого потока данных и входных данных потока данных.

[0095] Обращаясь теперь к фиг. 5, блок-схема последовательности операций способа иллюстрирует примерный способ 500 для выполнения улучшенной обработки событий. На этапе 510, идентифицируется источник данных в базе данных. Источник данных в базе данных ассоциирован с данными в базе данных. Данные в базе данных включают в себя опорные данные. На этапе 520, по меньшей мере, часть данных в базе данных из источника данных в базе данных дискретизируется, чтобы формировать один или

более моментальных снимков данных в базе данных. Дискретизация данных в базе данных основана на механизме периодической дискретизации, который формирует один или более моментальных снимков на основе предварительно определенного временного интервала, или на механизме дифференциальной дискретизации, который формирует один или более моментальных снимков, которые включают в себя только изменения данных в базе данных.

[0096] Дискретизация дополнительно может быть основана на механизме уведомлений, который указывает то, когда возникает изменение данных в базе данных, так что моментальный снимок формируется при индикаторе относительно изменения данных в базе данных. Механизм уведомлений может включать в себя признак пакетной обработки, признак пакетной обработки определяет предварительно определенное число изменений или определяет предварительно определенный период времени перед инициированием уведомления на то, чтобы выполнять дискретизацию. Система уведомлений может быть основана на административном временном элементе, который ассоциирован с рабочей функциональностью данных в базе данных, административный временной элемент не записывается в базе данных. Административный временной элемент указывает время, которое ассоциировано с моментальным снимком, который используется для того, чтобы формировать конструируемый поток данных с временным элементом.

[0097] На этапе 530, конструируемые потоки данных на основе одного или более моментальных снимков данных в базе данных, при этом конструируемый поток данных ассоциирован с временным элементом. На этапе 540, доступ к конструируемому потоку данных предоставляется на основе запроса на улучшенную обработку событий; запрос на улучшенную обработку событий ассоциирован с индикатором относительно входных данных в базе данных источника данных в базе данных, доступного с использованием конструируемого потока данных.

[0098] Варианты осуществления, представленные в настоящем документе, описаны относительно отдельных вариантов осуществления, которые во всех аспектах имеют намерение быть иллюстративными, а не ограничивающими. Альтернативные варианты осуществления должны становиться очевидными для специалистов в области техники, которой принадлежит настоящее изобретение, без отступления от его объема.

[0099] Из вышеописанного следует заметить, что это изобретение представляет собой изобретение, оптимально адаптированное с возможностью достигать всех результатов и целей, изложенных выше, вместе с другими преимуществами, которые являются очевидными и которые являются внутренне присущими структуре.

[00100] Следует понимать, что определенные признаки и субкомбинации являются полезными и могут быть применены без ссылки на другие признаки или субкомбинации.

Это предполагается и находится в пределах объема формулы изобретения.

(57) Формула изобретения

1. Реализуемый компьютером способ для улучшенной обработки событий, при этом способ содержит этапы, на которых:

- принимают запрос на улучшенную обработку событий, при этом запрос на улучшенную обработку событий содержит индикатор входных данных из источника данных в базе данных; при этом входные данные из источника данных в базе данных не ассоциированы с временным элементом для входных данных,

- осуществляют доступ к входным данным в индикаторе входных данных с использованием источника конструируемых потоков данных, при этом источник

конструируемых потоков данных предоставляет доступ к совокупности моментальных снимков данных в базе данных из источника данных в базе данных во времени, при этом к входным данным осуществляется доступ с использованием конструируемого потока данных, который содержит временной элемент; и

5 - формируют вывод для запроса на улучшенную обработку событий на основе входных данных, доступ к которым осуществляется с использованием конструируемого потока данных.

2. Способ по п. 1, в котором временной элемент указывает время, когда событие, ассоциированное с входными данными, возникает и сохраняется в качестве части
10 моментального снимка события в данных в базе данных.

3. Способ по п. 1, в котором конструируемый поток данных выступает в качестве прокси для данных в базе данных таким образом, что конструируемый поток данных предоставляет повторяющиеся вычисления с использованием данных в базе данных, которые изменяются во времени, при этом к конструируемому потоку данных
15 осуществляется доступ в качестве функции от выбранного значения и выбранного времени.

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

- коррелируют первый временной элемент конструируемого потока данных со вторым временным элементом потока данных; и

20 - формируют вывод для запроса на улучшенную обработку событий на основе входных данных конструируемого потока данных и входных данных потока данных.

5. Способ по п. 1, в котором вывод ассоциирован с временным элементом срока действия, который указывает время, когда вывод на основе конструируемого потока данных точно отражает данные в данных в базе данных.

25 6. Один или более компьютерных носителей хранения данных, имеющих осуществленные исполняемые компьютером инструкции, которые при выполнении посредством одного или более процессоров инструктируют одному или более процессоров осуществлять способ для улучшенной обработки событий, при этом способ содержит:

30 - идентификацию источника данных в базе данных;

- выборку, по меньшей мере, части данных в базе данных из источника данных в базе данных для того, чтобы формировать один или более моментальных снимков данных в базе данных,

35 - формирование конструируемого потока данных на основе одного или более моментальных снимков данных в базе данных, при этом конструируемый поток данных ассоциирован с временным элементом; и

- предоставление доступа к конструируемому потоку данных на основе запроса на улучшенную обработку событий; при этом запрос на улучшенную обработку событий ассоциирован с индикатором входных данных в базе данных источника данных в базе
40 данных, доступного с использованием конструируемого потока данных, при этом входные данные из источника данных в базе данных не ассоциированы с временным элементом для входных данных.

7. Носители по п. 6, в которых выборка данных в базе данных основана, по меньшей мере, на одном из следующего:

45 (1) механизм периодической выборки, который формирует один или более моментальных снимков на основе предварительно определенного временного интервала;

(2) механизм дифференциальной выборки, который формирует один или более моментальных снимков, которые включают в себя только изменения данных в базе

данных; и

(3) механизм уведомлений, который указывает то, когда возникает изменение данных в базе данных, так что моментальный снимок формируется при индикаторе изменения данных в базе данных.

5 8. Носители по п. 7, в которых механизм уведомлений дополнительно содержит, по меньшей мере, одно из следующего:

(1) признак пакетной обработки, при этом признак пакетной обработки определяет предварительно определенное число изменений или определяет предварительно определенный период времени перед инициированием уведомления на то, чтобы
10 выполнять выборку; и

(2) административный временной элемент, который ассоциирован с рабочей функциональностью данных в базе данных, при этом административный временной элемент не записывается в базе данных, при этом административный временной элемент указывает время, которое ассоциировано с моментальным снимком, который
15 используется для того, чтобы формировать конструируемый поток данных с временным элементом.

9. Компьютерная система для выполнения улучшенной обработки событий, причем компьютерная система содержит:

- компонент управления источниками данных в базе данных, сконфигурированный
20 с возможностью:

- предоставления доступа к данным в базе данных;

- компонент управления источниками конструируемых потоков данных, сконфигурированный с возможностью:

- идентификации источника данных в базе данных;

25 - выборки, по меньшей мере, части данных в базе данных из источника данных в базе данных для того, чтобы формировать один или более моментальных снимков данных в базе данных;

- формирования конструируемого потока данных на основе одного или более моментальных снимков данных в базе данных, при этом конструируемый поток данных
30 ассоциирован с временным элементом; и

- предоставления доступа к конструируемому потоку данных на основе запроса на улучшенную обработку событий, при этом запрос на улучшенную обработку событий ассоциирован, по меньшей мере, с индикатором входных данных в базе данных
35 источника данных в базе данных, доступного с использованием конструируемого потока данных, при этом входные данные из источника данных в базе данных не ассоциированы с временным элементом для входных данных;

- источник потоков данных, сконфигурированный с возможностью:

- предоставления доступа к потоку данных;

- сервер улучшенной обработки событий, сконфигурированный с возможностью:

40 - приема запроса на улучшенную обработку событий, при этом запрос на улучшенную обработку событий содержит индикатор входных данных из источника данных в базе данных и второй индикатор входных данных из источника потоков данных;

- осуществления доступа к входным данным в индикаторе входных данных с использованием источника конструируемых потоков данных, при этом источник
45 конструируемых потоков данных предоставляет доступ к совокупности моментальных снимков данных в базе данных из источника данных в базе данных во времени, при этом к входным данным осуществляется доступ с использованием конструируемого потока данных, который содержит первый временной элемент;

- осуществления доступа к входным данным во втором индикаторе входных данных с использованием источника потоков данных; и

- корреляции первого временного элемента конструируемого потока данных со вторым временным элементом потока данных; и

5 - формирования вывода для запроса на улучшенную обработку событий на основе входных данных конструируемого потока данных и входных данных потока данных.

10. Система по п. 9, в которой сервер улучшенной обработки событий содержит логический модуль, при этом логический модуль сконфигурирован с возможностью:

10 - идентификации логики для формирования вывода на основе запроса на улучшенную обработку событий, при этом логика выбирается, по меньшей мере, из одного из следующего:

- логика управления наиболее актуальными моментальными снимками; и

- логика интерполяции;

15 - предоставления логики, в которой такой вывод формируется на основе логики, при этом логика является обратно совместимой с конструируемым потоком данных.

20

25

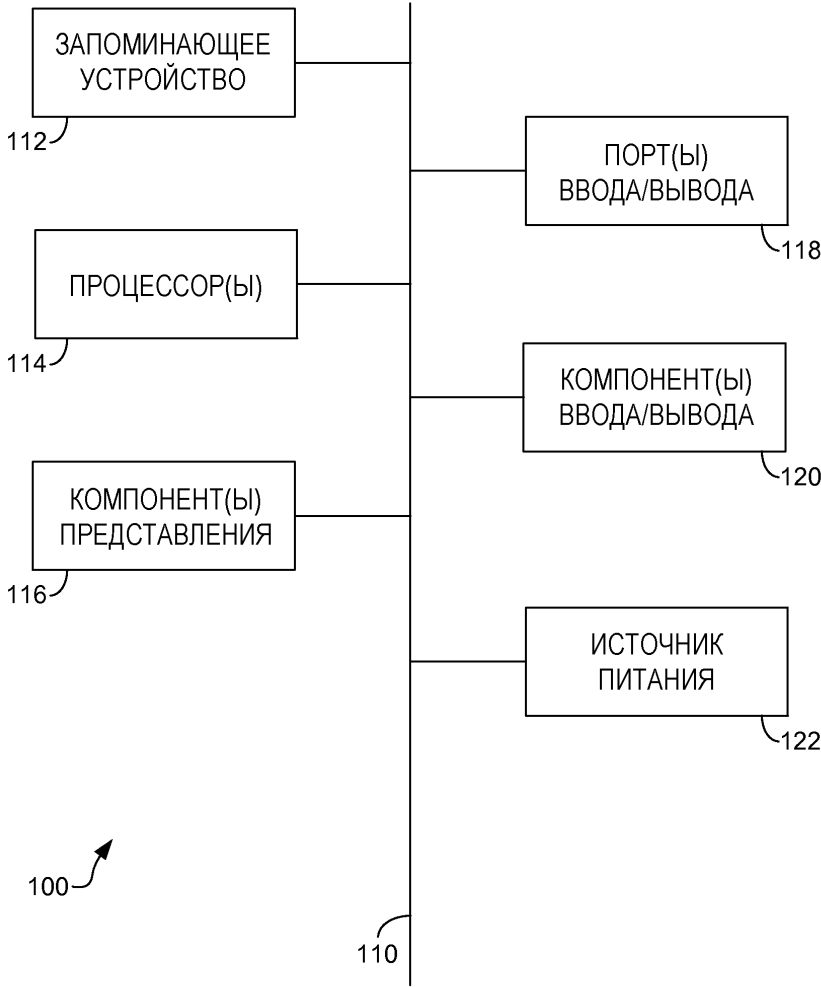
30

35

40

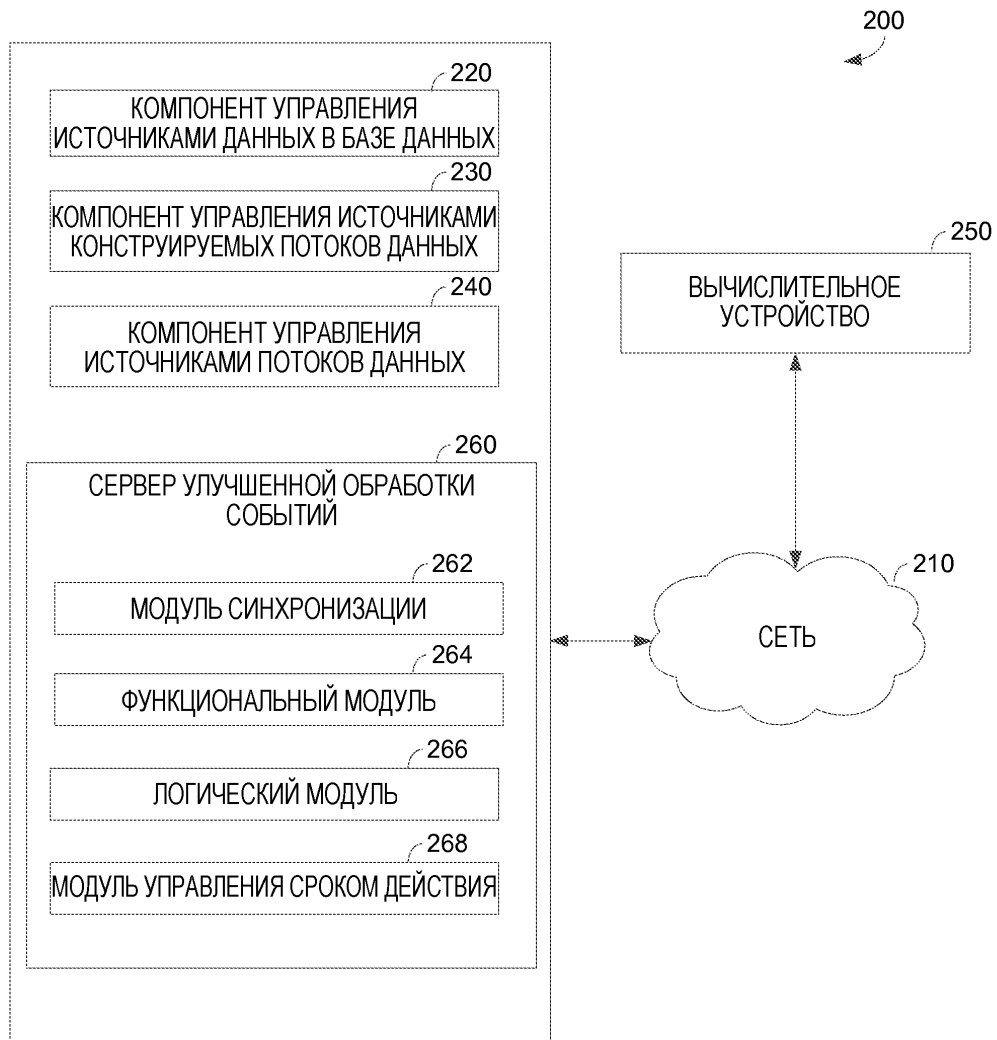
45

1/4

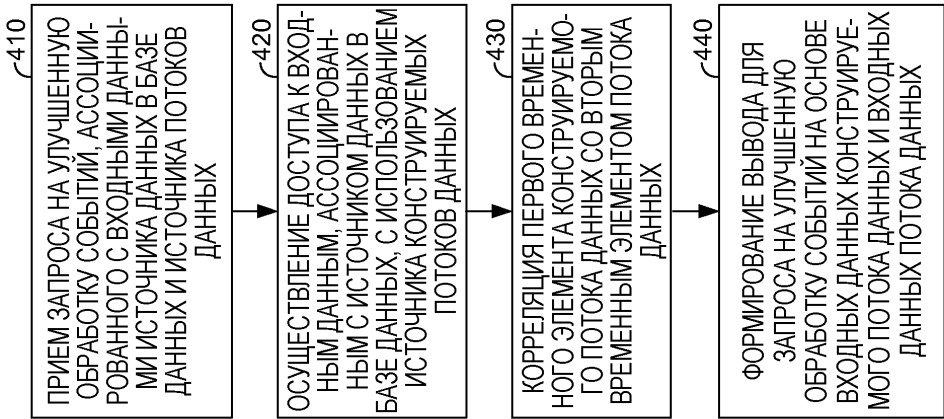


ФИГ. 1

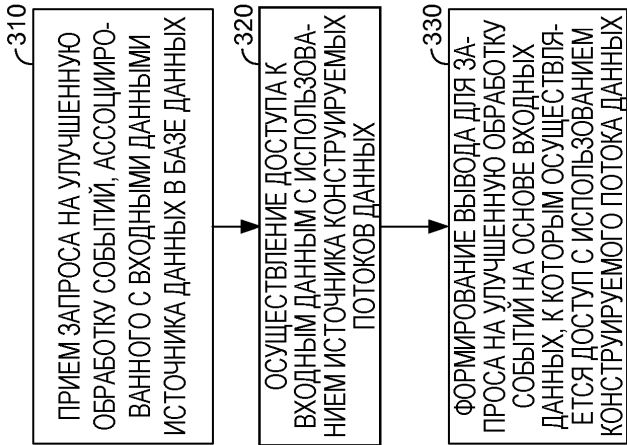
2/4



ФИГ. 2

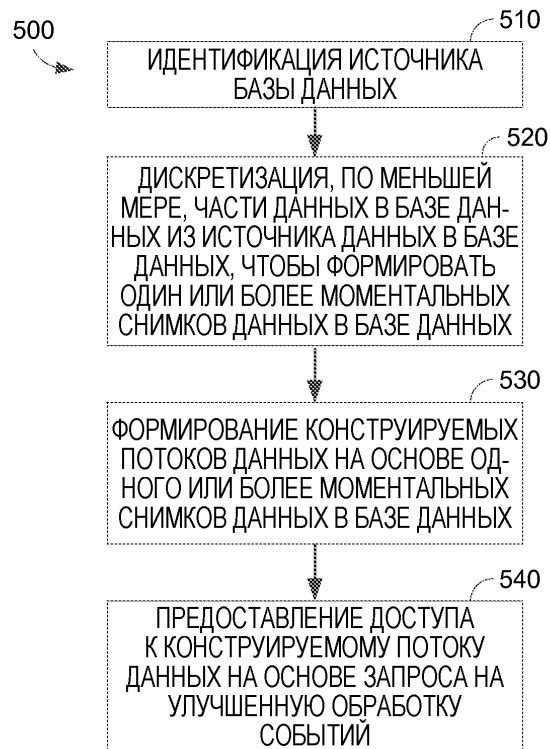


ФИГ. 4



ФИГ. 3

4/4



ФИГ. 5