



**ФЕДЕРАЛЬНАЯ СЛУЖБА
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ**

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ

(21)(22) Заявка: 2011144841/08, 16.04.2010

(24) Дата начала отсчета срока действия патента:
16.04.2010

Приоритет(ы):

(30) Конвенционный приоритет:
05.05.2009 US 12/435,737

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

(45) Опубликовано: 27.06.2015 Бюл. № 18

(56) Список документов, цитированных в отчете о поиске: EP 1630668 A1, 01.03.2006. US 2007/0124573 A1, 31.05.2007. US 2005/0132346 A1, 16.06.2005. US 5812859 A, 22.09.1998. RU 2348067 C2, 27.02.2009

(85) Дата начала рассмотрения заявки РСТ на национальной фазе: 03.11.2011

(86) Заявка РСТ:
US 2010/031457 (16.04.2010)

(87) Публикация заявки РСТ:
WO 2010/129159 (11.11.2010)

Адрес для переписки:

129090, Москва, ул. Б. Спасская, 25, строение 5,
ООО "Юридическая фирма Городисский и
Партнеры"

(72) Автор(ы):

**АБЗАРИАН Дэвид (US),
КАРПЕНТЕР Тодд Л. (US),
КУЛКАРНИ Хариш С. (US)**

(73) Патентообладатель(и):

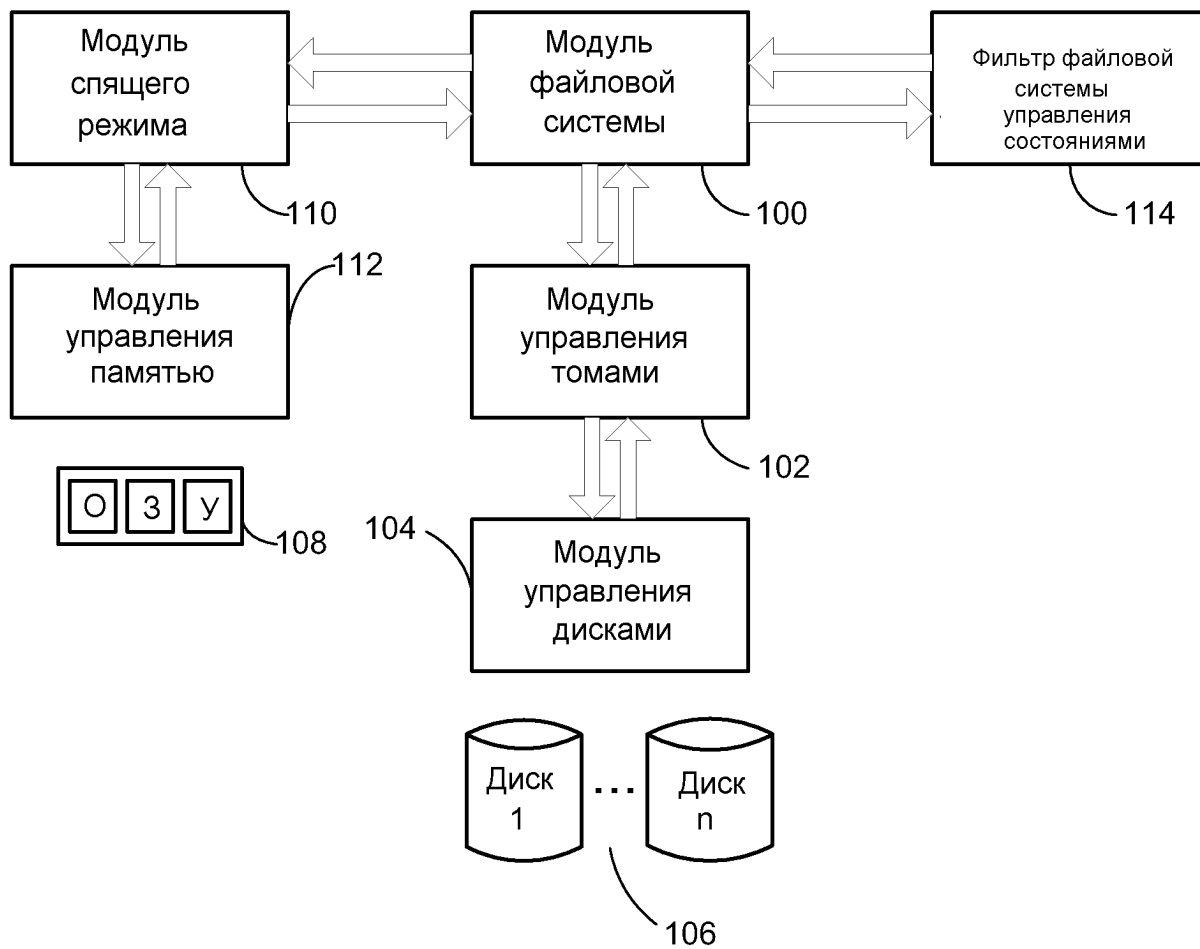
**МАЙКРОСОФТ ТЕКНОЛОДЖИ
ЛАЙСЕНСИНГ, ЭлЭлСи (US)**

(54) ЗАХВАТ И ЗАГРУЗКА СОСТОЯНИЙ ОПЕРАЦИОННОЙ СИСТЕМЫ

(57) Реферат:

Изобретение относится к области захвата и загрузки состояний операционной системы. Техническим результатом является повышение эффективности восстановления операционной системы к базовому состоянию. В одном варианте воплощения выполняется сохранение состояний памяти операционной системы компьютера, которые желают восстановить в будущем. Не разрешаются изменения в постоянном хранилище, связанном с компьютером. Вместо этого изменения, которые были бы произведены в

постоянном хранилище во время последующего компьютерного сеанса, если бы они не были предотвращены, сохраняются в отдельном файле компьютерного сеанса. Всякий раз, когда желают вернуть операционную систему к своему базовому состоянию, сохраненные базовые системные состояния памяти загружаются в память операционной системы вместо текущих состояний памяти операционной системы. 3 н. и 16 з.п. ф-лы, 7 ил.



ФИГ.1



FEDERAL SERVICE
FOR INTELLECTUAL PROPERTY

(51) Int. Cl.
G06F 9/22 (2006.01)
G06F 3/06 (2006.01)

(12) **ABSTRACT OF INVENTION**

(21)(22) Application: **2011144841/08, 16.04.2010**
 (24) Effective date for property rights:
16.04.2010
 Priority:
 (30) Convention priority:
05.05.2009 US 12/435,737
 (43) Application published: **10.05.2013 Bull. № 13**
 (45) Date of publication: **27.06.2015 Bull. № 18**
 (85) Commencement of national phase: **03.11.2011**
 (86) PCT application:
US 2010/031457 (16.04.2010)
 (87) PCT publication:
WO 2010/129159 (11.11.2010)
 Mail address:
**129090, Moskva, ul. B. Spasskaja, 25, stroenie 3,
OOO "Juridicheskaja firma Gorodisskij i Partnery"**

(72) Inventor(s):
**ABZARIAN Dehvid (US),
KARPENTER Todd L. (US),
KULKARNI Kharish S. (US)**
 (73) Proprietor(s):
**MAJKROSOFT TEKNOLODZHI
LAJSENSING, EhlEhlSi (US)**

(54) **CAPTURING AND LOADING OPERATING SYSTEM STATES**

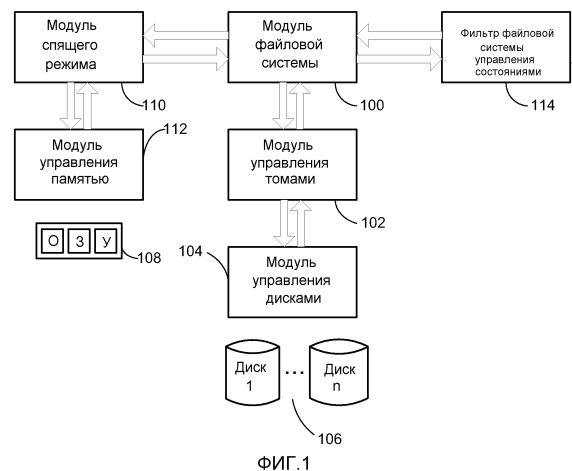
(57) Abstract:

FIELD: physics, computer engineering.

SUBSTANCE: invention relates to capturing and loading operating system states. On embodiment includes storing the states of computer operating system memory to be restored at a future time. No changes are permitted to the permanent storage associated with the computer. Instead, changes that would have been made to the permanent storage during an ensuing computing session, had they not been prevented, are stored in a separate computing session file. Whenever it is desired to return the operating system to its basic condition, the stored basic system memory states are loaded into the operating system memory, instead of the current states of the operating system memory.

EFFECT: high efficiency of restoring an operating system to the basic state.

19 cl, 7 dwg



RU 2 554 851 C 2

RU 2 554 851 C 2

УРОВЕНЬ ТЕХНИКИ

Интернет-кафе и другие совместно используемые узлы доступа нуждаются в восстановлении базового состояния системы на совместно используемых компьютерах после каждого компьютерного сеанса, чтобы гарантировать устойчивое и надежное взаимодействие с пользователем. Как правило, это требует перезагрузки системы после каждого сеанса для восстановления базового состояния.

Спящий режим дает возможность сохранять содержание системной памяти и регистров (например, состояния аппаратных регистров устройства, состояния файла виртуальной памяти, прикладных программ и драйверов и т.д.) в постоянном хранилище, таком как жесткий диск, в целях более быстрого восстановления состояний, связанных с последним компьютерным сеансом. Эта функция используется прежде всего для предоставления возможности операционной системе компьютера завершить работу таким образом, чтобы при следующем запуске операционная система была восстановлена из файла спящего режима и возобновила работу с того момента, когда ее работа была завершена.

Эта функциональная возможность спящего режима обычно использует один файл (например, hiberfil.sys), который записывается после инициирования этой функциональной возможности. Затем файл спящего режима используется для восстановления операционной системы к ее предыдущему состоянию как часть процесса загрузки.

СУЩНОСТЬ ИЗОБРЕТЕНИЯ

Это описание сущности изобретения дано для того, чтобы в упрощенной форме представить выбор концепций, которые далее описаны в подробном описании. Это описание сущности изобретения не предназначено для выявления ключевых признаков или основных признаков заявленного предмета, а также не предназначено для использоваться в качестве средства определения контекста заявленного предмета.

Описанные здесь варианты воплощения методик захвата и загрузки состояний операционной системы обычно включают в себя по меньшей мере захват и загрузку базовых состояний системы вне процесса загрузки. Это выполняется в одном варианте воплощения посредством сохранения состояний памяти операционной системы компьютера, которые желают восстановить как базовое состояние в будущем. Как только базовые системные состояния памяти сохранены, не разрешаются изменения в постоянном хранилище, связанном с компьютером. Вместо этого изменения, которые были бы произведены в постоянном хранилище во время последующего компьютерного сеанса, если бы они не были предотвращены, сохраняются в отдельном файле компьютерного сеанса. Данные изменений, сохраненные в файле компьютерного сеанса, в комбинации с данными, сохраненными в постоянном хранилище, используются во время компьютерного сеанса.

Всякий раз когда требуется вернуть операционную систему к своему базовому состоянию, например, при завершении компьютерного сеанса, сохраненные базовые системные состояния памяти загружаются в память операционной системы вместо текущих состояний памяти операционной системы. Поскольку изменения в постоянном хранилище, связанном с компьютером, не разрешались, операционная система теперь восстанавливается к своему базовому состоянию.

Если единственная цель захвата состояний операционной системы состоит в том, чтобы вернуть операционную систему к базовому состоянию системы, то в одном варианте воплощения файл компьютерного сеанса отбрасывается. Однако в другом варианте воплощения файл компьютерного сеанса может быть сохранен и загружен в

будущем для возобновления соответствующего компьютерного сеанса в момент, когда он был закончен.

Этот последний вариант воплощения обычно влечет за собой захват и восстановление базовых состояний, как описано выше, в конце каждого компьютерного сеанса. Однако всякий раз когда определено что текущий компьютерный сеанс был завершен, также имеют место дополнительные действия. А именно сохраняется файл состояния сеанса, представляющий состояние памяти операционной системы компьютера, существующее в момент времени, когда компьютерный сеанс был завершен. Захваченные изменения, которые были бы произведены в постоянном хранилище, также сохраняются в файле компьютерного сеанса. Чтобы возобновить предыдущий компьютерный сеанс, загружаются сохраненные состояния компьютерного сеанса. Это сначала влечет за собой доступ к сохраненному файлу состояния сеанса, который соответствует компьютерному сеансу, который желают возобновить в новом сеансе. Состояния сеанса, к которым осуществляют доступ, затем загружаются в память операционной системы компьютера вместо текущих (то есть базовых) состояний памяти операционной системы. Кроме того, осуществляют доступ к файлу компьютерного сеанса, соответствующему компьютерному сеансу, который желают возобновить в новом сеансе. Далее, изменения, которые были бы произведены в постоянном хранилище во время текущего компьютерного сеанса, снова захватываются. Захваченные изменения сохраняются в файле компьютерного сеанса. Как и прежде, захваченные изменения, а также другие данные из постоянного хранилища используются во время текущего компьютерного сеанса.

ОПИСАНИЕ ЧЕРТЕЖЕЙ

Конкретные функциональные возможности, аспекты и преимущества описания станут лучше понятны со ссылкой на последующее описание, приложенную формулу изобретения и сопроводительные чертежи.

Фиг. 1 - упрощенная схема архитектуры управления состояниями операционной системы, в которой могут быть реализованы описанные здесь части вариантов воплощения методик захвата и загрузки состояний операционной системы.

Фиг. 2 - блок-схема последовательности операций, изображающая в общих чертах один вариант воплощения процесса для захвата и загрузки базовых состояний системы.

Фиг. 3 - блок-схема последовательности операций, изображающая в общих чертах один вариант воплощения процесса для захвата состояний системы компьютерного сеанса.

Фиг. 4 - блок-схема последовательности операций, изображающая в общих чертах один вариант воплощения процесса для загрузки захваченных состояний компьютерного сеанса для возобновления предыдущего компьютерного сеанса, когда он был закончен, в новом компьютерном сеансе.

Фиг. 5А-5В изображает блок-схему последовательности операций, изображающую в общих чертах один вариант воплощения процесса для инкрементной загрузки состояний компьютерного сеанса с целью отладки.

Фиг. 6 - схема, изображающая вычислительное устройство, составляющее иллюстративную систему для реализации вариантов воплощения описанных здесь методик захвата и загрузки состояний операционной системы.

ПОДРОБНОЕ ОПИСАНИЕ

В последующем описании вариантов воплощения методик захвата и загрузки состояний операционной системы делается ссылка на сопроводительные чертежи, которые являются частью этого описания и на которых посредством иллюстрации

показаны конкретные варианты воплощения, в которых может быть осуществлена методика. Подразумевается, что могут быть использованы другие варианты воплощения и могут быть сделаны структурные изменения без отступления от контекста методики.

1.0. Архитектура управления состояниями операционной системы

5 Перед описанием вариантов воплощения методик захвата и загрузки состояний операционной системы будет дано общее описание подходящей архитектуры управления состояниями операционной системы, в которой могут быть реализованы их составляющие части. В общем случае варианты воплощения методики могут функционировать в многочисленных средах или конфигурациях вычислительных систем
10 общего или специального назначения, что будет описано подробнее в более позднем разделе этого описания.

Эти компьютеры общего или специального назначения используют операционную систему. Одна из целей операционной системы состоит в том, чтобы управлять памятью операционной системы и постоянным хранилищем, связанным с компьютером. Основу
15 этой схемы управления представляет собой модуль 100 файловой системы, как показано на фиг. 1. Модуль 100 файловой системы находится во взаимодействии и работает через модуль 102 управления томами для управления постоянным хранилищем, связанным с компьютером. Постоянное хранилище может представлять собой любое устройство или устройства энергонезависимой памяти, такие как жесткие диски 106, показанные
20 на фиг. 1. Как правило, постоянным хранилищем управляют через модуль управления устройствами. В примере на фиг. 1 он принимает вид модуля 104 управления дисками, который находится во взаимодействии с модулем 102 управления томами, и в конечном счете им управляет модуль 100 файловой системы.

Текущие состояния операционной системы компьютера хранятся в памяти
25 операционной системы и изменяются по ходу компьютерного сеанса, проводимого на компьютере. Как правило, память операционной системы принимает вид устройства энергозависимой памяти. Например, в иллюстративной архитектуре на фиг. 1 память операционной системы представляет собой оперативное запоминающее устройство 108 (ОЗУ; RAM). Текущие состояния памяти операционной системы могут быть
30 захвачены в файл, называемый файлом-образом для спящего режима. Этот механизм спящего режима позволяет сохранять состояния системной памяти и регистров в постоянное хранилище с целью быстрого восстановления. Это прежде всего используется для того, чтобы дать возможность операционной системе быть выключенной на компьютере и затем после перезапуска операционной системы с использованием файла-
35 образа для спящего режима восстановить предыдущие состояния. Таким образом, операционная система может возобновить операции с того момента, когда ее работа была завершена. В текущих операционных системах файл-образ для спящего режима создается во время завершения работы на основе команды пользователя. Упомянутый выше процесс восстановления состояния обычно происходит как часть
40 последовательности начальной загрузки, когда операционная система перезапускается. В контексте иллюстративной архитектуры на фиг. 1 модуль 100 управления файлами находится во взаимодействии с модулем 110 спящего режима. Когда модулю 110 спящего режима дается команда, он формирует файл-образ для спящего режима и использует помощь модуля 100 файловой системы для сохранения файла в постоянном хранилище.
45 Модуль 112 управления памятью операционной системы используется для захвата текущего состояния памяти операционной системы (например, оперативного запоминающего устройства 108 (ОЗУ; RAM), как показано на фиг. 1) и его доставки модулю 110 спящего режима.

Хотя описанные здесь варианты воплощения методик захвата и загрузки не являются частью традиционной архитектуры управления операционной системой, они используют фильтр 114 файловой системы управления состояниями, чтобы "отфильтровать" и захватить изменения, которые обычно производились с постоянным хранилищем. В иллюстративной архитектуре на фиг. 1 фильтр 114 файловой системы управления состояниями находится во взаимодействии с файловой системой 100 и использует ее помощь при сохранении захваченных изменений в постоянном хранилище в файле, называемом здесь файлом компьютерного сеанса. Работа фильтра 114 и роль файла компьютерного сеанса будут описаны более подробно в следующих разделах.

Следует отметить, что предыдущая архитектура управления состояниями операционной системой представляет собой только один пример подходящей архитектуры и не предназначена для какого-либо ограничения относительно объема использования или функциональности описанных здесь вариантов воплощения методик захвата и загрузки состояний операционной системы. Архитектура управления состояниями операционной системы не должна быть интерпретирована как имеющая какую-либо зависимость или требование, относящиеся к какому-либо компоненту или комбинации компонентов, проиллюстрированных на фиг. 1.

2.0. Захват и загрузка состояний операционной системы

Типичный компьютерный сеанс операционной системы включает в себя изменения в упомянутой выше системной памяти и в постоянном хранилище. В общем случае описанные здесь варианты воплощения методик захвата и загрузки состояний операционной системы дают возможность захвата состояний как постоянного хранилища, так и системной памяти в момент времени и последующей загрузки захваченных состояний системы. Это выполняется в одной реализации с использованием файла-образа для спящего режима для захвата состояния системной памяти. Следует отметить, что даже хотя используется файл-образ для спящего режима, это не означает, что операционная система должна быть восстановлена с использованием процедуры перезагрузки, как было бы в типичном случае. Вместо этого, как будет описано более подробно в последующих разделах, описанные здесь варианты воплощения методик захвата и загрузки состояний операционной системы работают вне стандартного завершения работы для спящего режима и процедур перезапуска последовательности начальной загрузки.

Что касается захвата изменений, упомянутый выше фильтр файловой системы управления состояниями используется в одном варианте воплощения для захвата изменений, которые обычно производятся в постоянном хранилище. Таким образом, никакие изменения не разрешены для постоянного хранилища. Загрузка захваченных состояний системы возвращает операционную систему в состояние, в котором она была когда состояния системы были захвачены. Это справедливо даже если состояние системной памяти было изменено во время компьютерного сеанса. Также следует отметить, что поскольку изменение постоянного хранилища не допускается, оно сохраняется в состоянии, в котором оно было, когда состояния системы были захвачены.

В одном варианте воплощения захваченное состояние системы представляет базовое состояние, в которое можно возвратиться в окончании компьютерного сеанса. Это дает возможность заново начинать каждый компьютерный сеанс в известном базовом состоянии. Таким образом, в случае интернет-кафе или другой ситуации совместно используемого узла доступа базовое состояние системы восстанавливается после каждого компьютерного сеанса, чтобы гарантировать устойчивое и надежное взаимодействие с пользователем.

В другом варианте воплощения захватываются более одного состояний системы. Базовые состояния захватываются как прежде, но в дополнение захватываются состояния системы компьютерного сеанса. Это позволяет пользователю загружать захваченные состояния системы компьютерного сеанса в будущем и продолжать с того момента, когда пользователь закончил предыдущий компьютерный сеанс. Операционная система по-прежнему возвращается к своему базовому состоянию после каждого компьютерного сеанса. Таким образом, не имеет значения, загружает ли пользователь захваченное состояние системы компьютерного сеанса для непосредственно следующего компьютерного сеанса или для компьютерного сеанса, который происходит до того как имели место один или более промежуточных сеансов. Это также имеет преимущества в случае интернет-кафе или другой ситуации совместно используемого узла доступа. Не только базовое состояние системы восстанавливается после каждого компьютерного сеанса (что обычно включает в себя нескольких пользователей), но и отдельный пользователь может захватывать свой компьютерный сеанс и затем возвращаться в будущем к состояниям системы, связанным с этим компьютерным сеансом.

В еще одном варианте воплощения методики захвата и загрузки состояний операционной системы дают возможность быстрой отладки с помощью инкрементной загрузки состояний сеанса в известное хорошее базовое состояние. Например, в случае аварийного сбоя операционной системы файл выгрузки сбоя может использоваться в качестве файла состояния сеанса и инкрементно загружаться, пока проблема не будет найдена.

Предыдущие, а также другие варианты воплощения будут описаны более подробно в следующих разделах.

2.1. Захват и загрузка базовых состояний системы

Варианты воплощения методик захвата и загрузки состояний операционной системы, включающие в себя захват базовых состояний системы, могут быть выполнены в одной реализации, как изображено на фиг. 2. Сначала сохраняются состояния памяти операционной системы компьютера, которые желательно загрузить в будущем (этап 200). Сохраненные состояния могут называться базовыми состояниями системы. В одной версии состояния памяти операционной системы сохраняются как файл-образ для спящего режима. Однако, по желанию, также могут использоваться другие типы файлов.

С этого момента не разрешаются никакие изменения в постоянном хранилище, связанном с компьютером (этап 202). Вместо этого изменения, которые были бы произведены в постоянном хранилище во время последующего компьютерного сеанса, если бы они не были предотвращены, сохраняются в отдельном файле компьютерного сеанса (этап 204). В одной версии это выполняется с использованием упомянутого выше фильтра файловой системы управления состояниями, который перенаправляет изменения постоянного хранилища в файл компьютерного сеанса. Файл любого типа может использоваться для файла компьютерного сеанса. Например, подходящим выбором был бы виртуальный жесткий диск (VHD). Данные изменений, сохраненные в файле компьютерного сеанса, в комбинации с данными, сохраненными в постоянном хранилище, используются во время компьютерного сеанса нормальным образом, точно так же, как если бы изменения фактически были произведены в постоянном хранилище.

Оставление постоянного хранилища, связанного с компьютером, не тронутым во время компьютерного сеанса, позволяет восстановить базовые состояния системы просто посредством загрузки сохраненных базовых состояний системы в системную память. Никакое реконфигурирование постоянного хранилища не требуется. Другими

словами, со ссылкой на фиг. 2, определяется, желательно ли вернуть операционную систему компьютера к ее состоянию, когда были сохранены состояния памяти операционной системы (этап 206). Например, инициализацией для этого определения может являться команда пользователя восстановить операционную систему к базовому состоянию или некоторая другая причина для завершения текущего компьютерного сеанса. Если определено, что операционная система еще не была восстановлена, действие этапа 206 повторяется. Однако если определено, чтобы операционная система была восстановлена к предыдущему сохраненному состоянию, то сохраненные состояния загружаются в память операционной системы вместо текущих состояний памяти операционной системы (этап 208).

Следует отметить, что если единственная цель захвата состояний операционной системы состоит в том, чтобы вернуть операционную систему к базовому состоянию системы, то файл компьютерного сеанса просто отбрасывается в конце компьютерного сеанса. Как будет описано в дальнейших разделах, другие варианты воплощения будут использовать файлы компьютерного сеанса. В этих вариантах воплощения файлы сеанса сохраняются, а не отбрасываются.

Также следует отметить, что в любое время после захвата базовых состояний системы они могут быть заменены на текущие состояния системы для формирования новых базовых состояний. Для выполнения этой задачи фильтр файловой системы управления состояниями временно деактивируется, и изменения, которые были захвачены в файле компьютерного сеанса, реализуются в постоянном хранилище, связанном с компьютером. Это синхронизирует постоянное хранилище с текущими состояниями памяти операционной системы. Новые базовые состояния системы затем могут быть захвачены, как описано ранее, и новые состояния памяти операционной системы сохраняются вместо ранее сохраненных состояний.

2.2. Захват и загрузка состояний компьютерного сеанса

Как описано ранее, также могут быть захвачены состояния компьютерного сеанса пользователя. Это позволяет пользователю загружать захваченные состояния системы компьютерного сеанса в будущем и продолжать с того момента, когда пользователь закончил предыдущий компьютерный сеанс. Базовое состояние по-прежнему захватывается, как описано выше, но в дополнение захватывается отдельное состояние системы компьютерного сеанса.

В одной реализации описанное выше выполнено, как проиллюстрировано на фиг. 3. Сначала состояния памяти операционной системы компьютера, существующие в момент времени, заданный пользователем компьютера, сохраняются и назначаются в качестве базовых состояний (этап 300). Постоянное хранилище компьютера затем защищается таким образом, чтобы изменения, обычно происходящие во время компьютерного сеанса, были предотвращены (этап 302). Таким образом, как описано ранее, постоянное хранилище компьютера остается в том же самом состоянии, как при захвате базовых состояний памяти операционной системы. Вместо этого изменения, которые были бы произведены в постоянном хранилище во время последующего компьютерного сеанса, если бы они не были предотвращены, сохраняются в ранее описанном файле компьютерного сеанса (этап 304). Данные изменений, сохраненные в файле компьютерного сеанса, в комбинации с данными, сохраненными в постоянном хранилище, используются во время компьютерного сеанса нормальным образом, как описано ранее.

Затем определяется, завершен ли текущий компьютерный сеанс (этап 306). Если определено, что сеанс еще не был завершен, действие этапа 306 повторяется. Однако

если определено, что компьютерный сеанс был завершен, имеют место несколько действий. Сначала сохраняется файл состояния сеанса, представляющий состояния памяти операционной системы компьютера, существующие в тот момент, когда компьютерный сеанс был завершен (этап 308). Как и в случае с базовым файлом состояний, в одной версии файл сохраняется как файл-образ для спящего режима. Однако, при желании, также могут использоваться файлы других типов. Кроме того, захваченные изменения, которые были бы произведены в постоянном хранилище, сохраняются в файле компьютерного сеанса (этап 310). Здесь опять для файла компьютерного сеанса может использоваться файл любого типа, такой как файл VHD. Следует отметить, что файл состояния сеанса и файл компьютерного сеанса связаны с указателями, идентифицирующими пользователя и завершенный компьютерный сеанс. Захват изменений, которые были бы произведены в постоянном хранилище во время компьютерного сеанса, также прерывается в этот момент (этап 312). Операционная система компьютера затем возвращается к своему базовому состоянию. Это влечет за собой загрузку сохраненных базовых состояний в памяти операционной системы вместо текущих состояний памяти операционной системы (этап 314).

Результат описанной выше процедуры состоит в том, что операционная система компьютера находится в своем базовом условии, и состояния компьютерного сеанса пользователя были захвачены в виде файла состояния сеанса и файла компьютерного сеанса. С учетом этого теперь будет описано, как состояния компьютерного сеанса пользователя загружаются таким образом, чтобы пользователь мог продолжать с того момента, когда он закончил последний сеанс. Следует отметить, что не имеет значения, загружает ли пользователь сохраненный сеанс как следующий имеющий место сеанс на компьютере, или после завершения нескольких других сеансов. Промежуточные сеансы не должны даже быть проведены этим пользователем. Это справедливо потому, что операционная система компьютера возвращается к одному и тому же базовому состоянию после каждого сеанса независимо от того, кто был пользователем или в каком порядке произошли сеансы.

Теперь со ссылкой на фиг. 4 сохраненные состояния компьютерного сеанса пользователя загружаются следующим образом. Сначала осуществляется доступ к сохраненному файлу состояния сеанса, соответствующему компьютерному сеансу, который желательно возобновить в новом сеансе (этап 400). Файл состояния сеанса, к которому произведен доступ, затем загружается в память операционной системы компьютера, вместо текущих (то есть базовых) состояний памяти операционной системы (этап 402). Кроме того, осуществляется доступ к файлу компьютерного сеанса, соответствующему компьютерному сеансу, который желательно возобновить в новом сеансе (этап 404). Способы доступа к описанным выше файлам будут описаны более подробно далее в этом описании. Затем фильтр файловой системы управления состояниями активируется, и изменения, которые были бы произведены в постоянном хранилище во время текущего компьютерного сеанса, захватываются, как описано ранее (этап 406). Захваченные изменения сохраняются в файле компьютерного сеанса. Как и прежде, захваченные изменения, а также другие данные из постоянного хранилища, используются во время текущего компьютерного сеанса.

Действия, описанные в отношении фиг. 3, имеющие отношение к сохранению файла состояния сеанса и файла компьютерного сеанса, связанные с компьютерным сеансом, после его завершения и к возврату операционной системы компьютера к своему базовому состоянию, будут вновь реализованы, когда текущий сеанс завершен. Таким образом, пользователь может захотеть возобновить только что завершенный сеанс в

будущем сеансе. Следует отметить, что файл компьютерных сеансов хранится под новым именем, с тем чтобы можно было отличить его от предыдущих файлов сеанса. Этот новый файл компьютерного сеанса представляет последний ранее сохраненный файл сеанса, обновленный посредством добавления изменений, захваченных во время
5 только что завершено компьютерного сеанса. Кроме того, поскольку файлы состояния сеанса сохраняются в конце каждого компьютерного сеанса, пользователь может выбирать из нескольких сохраненных сеансов, которые доступны для компьютера. С этой целью в одной реализации упомянутый выше доступ к файлу состояния сеанса и файлу компьютерного сеанса включает в себя предоставления списка
10 доступных файлов состояния сеанса и соответствующих файлов компьютерного сеанса пользователю, который начинает компьютерный сеанс. К файлу состояния сеанса и файлу компьютерного сеанса, связанным с предыдущим компьютерным сеансом, которые выбраны пользователем из списка, затем осуществляется доступ и выполняется их ввод. Поскольку файлы состояния сеанса и файлы компьютерного сеанса также
15 связаны с конкретным пользователем, сначала пользователь может быть идентифицирован посредством традиционных способов, и только затем эти файлы, связанные с этим пользователем, будут представлены в списке.

В предыдущей реализации компьютер сохраняет файлы состояния сеанса и файлы компьютерного сеанса в памяти, доступной для компьютера, например, на его жестком
20 диске. Однако в альтернативной реализации компьютер не сохраняет копии этих файлов. Например, в примере интернет-кафе или других совместно используемых узлов доступа может быть нежелательно сохранять файлы состояния сеанса и файлы компьютерного сеанса клиента на совместно используемом компьютере. В альтернативной реализации в конце компьютерного сеанса сохраненный файл состояния сеанса и файл
25 компьютерного сеанса для этого сеанса делаются доступными для загрузки пользователем. Тогда пользователь имеет вариант для сохранения этих файлов на переносимом запоминающем устройстве (например, на карте флэш-памяти и т.п.) или, возможно, для передачи файлов через сетевое соединение на другой компьютер (например, через вложение электронной почты и т.д.). Независимо от того, загружает
30 ли пользователь файлы или нет, они делаются недоступными для компьютера.

В реализациях, в которых файлы состояния сеанса и файлы компьютерного сеанса не сохраняются компьютером, должны быть предоставлены файлы, чтобы пользователь возобновил предыдущий компьютерный сеанс. С этой целью доступ к этим файлам приводит к тому, что компьютер запрашивает у пользователя ввод файла состояния
35 сеанса и файла компьютерного сеанса, соответствующих компьютерному сеансу, который пользователь хочет возобновить. Тогда пользователь вводит требуемые файлы через соответствующий способ в зависимости от того как они были сохранены.

Также следует отметить, что в реализациях, в которых файлы состояния сеанса и файлы компьютерного сеанса не сохраняются компьютером, возникает вопрос
40 относительно того, являются ли файлы совместимыми с компьютером, когда пользователь хочет возобновить предыдущий компьютерный сеанс. Например, если базовые состояния системы компьютера были изменены с тех пор, как файл состояния сеанса и вводимый файл компьютерного сеанса были захвачены, они могут быть больше не совместимыми. Один способ гарантировать совместимость состоит в том, чтобы
45 связать базовый файл состояний системы с указателем версии, который изменяется с каждым новым базовым состоянием. Таким образом, каждое базовое состояние однозначно определяется. Файл состояния сеанса и файл компьютерного сеанса, соответствующий завершено компьютерному сеансу, затем связываются с

информацией проверки допустимости во время их сохранения. Информация проверки допустимости включает в себя упомянутый выше указатель версии базового состояния, соответствующий версии базовых состояний системы, из которых были получены файлы. Этот указатель будет сравнен с указателем версии базовых состояний системы, в настоящий момент используемых компьютером. Если они соответствуют, введенные файлы являются совместимыми (поскольку компьютер был бы восстановлен к своему базовому состоянию в конце последнего компьютерного сеанса). Если они не соответствуют, пользователю сообщается, что компьютерный сеанс, связанный с введенными файлами, не может быть возобновлен.

Другая причина того, что файл состояния сеанса и файл компьютерного сеанса, введенные в компьютер пользователем, могут быть несовместимыми, состоит в том, что компьютер является не тем же самым компьютером, который использовался для формирования файлов. Обычно разные компьютеры имеют разные базовые состояния системы. Однако возможно сконфигурировать более одного компьютера, совместимого с одним и тем же набором файла состояния сеанса и файла компьютерного сеанса. Это может быть выполнено в одной реализации посредством создания общих базовых состояний системы с одним и тем же указателем версии базового состояния между всеми используемыми компьютерами. Таким образом, набор файла состояния сеанса и файла компьютерного сеанса, сформированный на одном компьютере, будет совместим с другим компьютером, поскольку базовые состояния системы, из которых были получены файлы, являются одинаковыми на обоих компьютерах.

В описанной выше управляемой с помощью версий схеме базовых состояний системы существует возможность уменьшить размер файла состояния сеанса. С этой целью в одной реализации идентифицируются различия между базовыми состояниями памяти операционной системы и состояниями памяти операционной системы, существующими во время завершения компьютерного сеанса. Эти различия затем сохраняются как файл состояния сеанса вместо текущих состояний всей памяти операционной системы. Этот вариант файла состояния сеанса может называться файлом разности состояний сеанса. Поскольку файл разности пропускает некоторые из состояний, которые необходимы для заполнения памяти операционной системы, он интегрируется в файл базовых состояний памяти операционной системы для воссоздания полной конфигурации состояний компьютерного сеанса, прежде чем загружается в память операционной системы для возобновления предыдущего компьютерного сеанса. С этой целью из входного файла разности состояний сеанса формируется восстановленный файл состояния сеанса. Это может быть сделано после успешной проверки совместимости входных файлов в реализациях, в которых проверяется допустимость. Восстановленный файл состояния сеанса формируется посредством применения различий в файле разности состояний сеанса к базовым состояниям памяти операционной системы. Другими словами, те состояния, которые найдены в файле разности, используются для замены соответствующих состояний в базовом файле состояний для получения восстановленного файла состояния сеанса. Как только файл состояния сеанса восстановлен, он загружается, как описано ранее, для возобновления связанного компьютерного сеанса.

2.3. Захват и инкрементная загрузка состояний компьютерного сеанса с целью отладки.

Как заявлено ранее, описанные здесь варианты воплощения методик захвата и загрузки состояний операционной системы также дают возможность быстрой отладки с помощью инкрементной загрузки состояний сеанса. В общем случае посредством введения инкрементных частей состояний памяти операционной системы в известное

хорошее базовое состояние прикладная или операционная система может быть отлажена. Пользователь может захотеть отлаживать прикладную программу или систему, или процесс отладки может быть инициирован аварийным сбоем системы вследствие ошибки. В этом последнем сценарии файл выгрузки сбоя, сформированный во время аварийного сбоя, используется в качестве файла состояния сеанса. На каждой итерации этого варианта воплощения используются традиционные методики отладки.

В одной реализации схема инкрементной отладки достигается следующим образом. Со ссылкой на фиг. 5А-5В процесс начинается с установления известного хорошего базового состояния. С этой целью сохраняются состояния памяти операционной системы компьютера, существующие в момент времени, заданный пользователем компьютера (этап 500). Эти состояния сохраняются в базовом файле состояний. Как и в других реализациях, предотвращаются изменения постоянного хранилища, связанного с компьютером, во время компьютерного сеанса (этап 502), и вместо этого изменения, которые были бы произведены в постоянном хранилище, захватываются (этап 504). Как и прежде, захваченные изменения, а также другие данные из постоянного хранилища используются во время компьютерного сеанса.

Затем определяется, был ли завершен текущий компьютерный сеанс (этап 506), например, когда пользователь завершает сеанс, чтобы инициировать отладку, или система испытывает аварийный сбой. Если определено, что сеанс еще не был завершен, этап 506 процесса повторяется. Однако когда компьютерный сеанс завершен, сохраняется файл состояния сеанса, представляющий состояния памяти операционной системы компьютера, существующие в тот момент времени, когда компьютерный сеанс завершен (этап 508). В сценарии аварийного сбоя системы этот файл состояния сеанса будет представлять собой файл выгрузки сбоя. Кроме того, захват изменений, которые были бы произведены в постоянном хранилище, прерывается и изменения сохраняются (этап 510). Затем операционная система компьютера возвращается к своему состоянию, когда базовые состояния памяти операционной системы были сохранены, посредством загрузки сохраненных базовых состояний в память операционной системы вместо текущих состояний памяти операционной системы (этап 512).

Когда базовое состояние восстановлено, может начаться процесс инкрементной отладки. Это влечет за собой сначала доступ к предписанной части состояний из сохраненного файла состояния сеанса и ее загрузку в память операционной системы компьютера вместо соответствующих текущих состояний памяти операционной системы компьютера (этап 514). Затем повторно инициируется захват изменений, которые были бы произведены в постоянном хранилище (этап 516). Эти захваченные изменения, а также другие данные из постоянного хранилища и ранее сохраненные изменения затем используются во время текущей итерации отладки, чтобы отладить операционную систему компьютера или конкретную прикладную программу (этап 518). В конце процедуры отладки определяется, была ли отлажена операционная система компьютера (этап 520). Если да, то процесс заканчивается. Однако когда определено, что операционная система компьютера не была отлажена, завершается текущая итерация отладки (этап 522). Захват изменений, которые были бы произведены в постоянном хранилище во время текущей итерации отладки, прерывается, и захваченные изменения сохраняются (этап 524). Операционная система компьютера снова возвращается к своему базовому состоянию посредством загрузки сохраненных состояний от файла базовых состояний в память операционной системы вместо текущих состояний памяти операционной системы (этап 526). Затем осуществляется доступ к части состояний из сохраненного файла состояния сеанса, который увеличен посредством предписанного

инкремента от последней завершенной итерации отладки, и она загружается в память операционной системы компьютера вместо соответствующих текущих состояний памяти операционной системы компьютера (этап 528). Затем этапы с 516 по 528, при необходимости, повторяются, пока операционная система компьютера не будет

5 отлажена.

3.0. Вычислительная среда

Теперь будет дано краткое общее описание подходящей вычислительной среды, в которой могут быть реализованы части описанных здесь вариантов воплощения методик захвата и загрузки состояний операционной системы. Как заявлено ранее, варианты

10 воплощения метода могут функционировать с многочисленными средами или конфигурациями компьютерных систем общего или специального назначения. Примеры известных вычислительных систем, сред и/или конфигураций, которые могут являться подходящими, включают в себя, но без ограничения, персональные компьютеры, серверные компьютеры, карманные или переносные устройства, многопроцессорные

15 системы, системы на основе микропроцессоров, телеприставки, программируемую бытовую электронику, сетевые персональные компьютеры, мини-компьютеры, универсальные вычислительные системы, распределенные вычислительные среды, которые включают в себя любые из упомянутых выше систем или устройств, и т.п.

Фиг. 6 иллюстрирует пример подходящей вычислительной системной среды.

20 Вычислительная системная среда представляет собой только один пример подходящей вычислительной среды и не предназначена для наложения каких-либо ограничений относительно объема использования или функциональности описанных здесь вариантов воплощения методик захвата и загрузки состояний операционной системы. Также вычислительная среда не должна интерпретироваться как имеющая какие-либо

25 зависимости или требования по отношению к какому-либо одному или к комбинации компонентов, показанных в иллюстративной операционной среде. Со ссылкой на фиг. 6 иллюстративная система для реализации описанных здесь вариантов воплощения включает в себя вычислительное устройство, такое как вычислительное устройство 10.

В своей наиболее базовой конфигурации вычислительное устройство 10 обычно

30 включает в себя по меньшей мере один процессор 12 и память 14. Память 14 является комбинацией энергозависимой (такой как оперативное запоминающее устройство (ОЗУ; RAM)) и энергонезависимой (такой как постоянное запоминающее устройство (ПЗУ; ROM), флэш-память и т.д.) памяти. Эта наиболее базовая конфигурация проиллюстрирована на фиг. 6 пунктирной линией 16. Кроме того, устройство 10 также

35 может иметь дополнительные признаки/функциональность. Например, устройство 10 также может включать в себя дополнительную память (съемную и/или несъемную), в том числе, но без ограничения, магнитные или оптические диски или ленту. Такая дополнительная память проиллюстрирована на фиг. 6 посредством съемного устройства памяти и несъемного устройства 20 памяти. Компьютерные носители включают в себя

40 энергозависимые и энергонезависимые, съемные и несъемные носители, реализованные с помощью любого способа или технологии для хранения информации, таких как машиночитаемые команды, структуры данных, программные модули или другие данные. Память 14, съемное устройство 18 памяти и несъемное устройство 20 памяти являются примерами компьютерных носителей. Компьютерные носители включают в себя, но

45 без ограничения, оперативное запоминающее устройство (RAM; ОЗУ), постоянное запоминающее устройство (ROM; ПЗУ), электрически стираемое программируемое постоянное запоминающее устройство (EEPROM; ЭСППЗУ), флэш-память или другую технологию памяти, компакт-диск, предназначенный только для чтения (CD-ROM),

цифровые универсальные диски (DVD) или другую память на оптическом диске, магнитные кассеты, магнитную ленту, память на магнитном диске или другие магнитные запоминающие устройства или любой другой носитель, который может быть использован для хранения желаемой информации и к которому можно получить доступ с помощью компьютера. Любые такие компьютерные носители могут являться частью устройства 10.

Устройство 10 также может содержать соединение (соединения) 22 связи, которое позволяют устройству взаимодействовать с другими устройствами. Устройство 10 также может иметь устройство (устройства) 24 ввода данных, такое как клавиатура, мышь, перо, речевое устройство ввода данных, сенсорное устройство ввода данных, камера и т.д. Устройство (устройства) 26 вывода, такое как дисплей, динамики, принтер и т.д., также может быть включено в состав. Все эти устройства хорошо известны в области техники и не нуждаются в подробном обсуждении здесь.

Описанные здесь варианты воплощения методик захвата и загрузки состояний операционной системы могут быть также описаны в общем контексте исполнимых с помощью компьютера команд, таких как программные модули, исполняемые вычислительным устройством. Обычно программные модули включают в себя подпрограммы, программы, объекты, компоненты, структуры данных и т.д., которые выполняют конкретные задачи или реализуют конкретные абстрактные типы данных. Описанные здесь варианты воплощения также могут быть осуществлены в распределенных вычислительных средах, в которых задачи выполняются удаленными устройствами обработки, которые соединены через сеть связи. В распределенной вычислительной среде программные модули могут быть расположены как на локальных, так и на удаленных компьютерных носителях, включающих в себя устройства памяти.

3.0. Другие варианты воплощения

Следует отметить, что любые из упомянутых выше вариантов воплощения во всем описании могут использоваться в любой комбинации, желаемой для формирования дополнительных гибридных вариантов воплощения. Кроме того, хотя предмет был описан на языке, специальном для структурных признаков и/или действий способов, следует понимать, что предмет, определенный в приложенной формуле изобретения, не обязательно ограничивается признаками или действиям, описанными выше. Описанные выше признаки и действия раскрыты в качестве иллюстративных форм реализации пунктов формулы изобретения.

Формула изобретения

1. Компьютерно-реализуемый способ захвата и загрузки состояний операционной системы, в котором используют компьютер для выполнения следующих этапов, на которых:

сохраняют состояния памяти операционной системы компьютера, которые потребуется загрузить в будущем, в постоянном хранилище, связанном с компьютером; предотвращают изменения в постоянном хранилище, связанном с компьютером, во время компьютерного сеанса;

сохраняют изменения, которые были бы произведены в постоянном хранилище во время компьютерного сеанса, если бы они не были предотвращены, в отдельном файле сеанса и используют данные, сохраненные в файле сеанса и в постоянном хранилище, во время компьютерного сеанса;

сбрасывают файл сеанса в конце компьютерного сеанса; и

всякий раз, когда требуется вернуть операционную систему компьютера к ее

состоянию, когда были сохранены упомянутые состояния памяти операционной системы, загружают сохраненные состояния в память операционной системы вместо текущих состояний памяти операционной системы.

5 2. Способ по п. 1, в котором на этапе сохранения изменений в отдельном файле сеанса упомянутые изменения сохраняют в формате файла виртуального жесткого диска (VHD).

3. Способ по п. 1, в котором на этапе сохранения состояний памяти операционной системы упомянутые состояния сохраняют как файл-образ спящего режима.

10 4. Компьютерно-реализуемый способ захвата и загрузки состояний операционной системы, в котором используют компьютер для выполнения следующих этапов, на которых:

15 сохраняют состояния памяти операционной системы компьютера, существующие в момент времени, заданный пользователем компьютера, в постоянном хранилище, связанном с компьютером, причем упомянутые состояния назначаются в качестве базовых состояний памяти операционной системы;

предотвращают изменения в постоянном хранилище, связанном с компьютером, во время компьютерного сеанса;

20 захватывают изменения, которые были бы произведены в постоянном хранилище во время компьютерного сеанса, если бы они не были предотвращены, и используют захваченные изменения, а также другие данные из постоянного хранилища во время компьютерного сеанса; и

25 всякий раз, когда компьютерный сеанс завершается, сохраняют файл состояний сеанса, представляющий состояния памяти операционной системы компьютера, существующие в момент времени, когда компьютерный сеанс завершен, и связывают файл состояний сеанса с пользователем и завершаемым компьютерным сеансом,

30 сохраняют захваченные изменения в отдельном файле компьютерного сеанса и связывают файл компьютерного сеанса с пользователем и завершаемым компьютерным сеансом,

35 прерывают захват изменений, которые были бы произведены в постоянном хранилище во время компьютерного сеанса, и

возвращают операционную систему компьютера к ее состоянию, когда были сохранены базовые состояния памяти операционной системы, причем при упомянутом возврате загружают сохраненные базовые состояния в память операционной системы

5. Способ по п. 4, в котором на этапе сохранения изменений в отдельном файле сеанса упомянутые изменения сохраняют в формате файла виртуального жесткого диска (VHD).

40 6. Способ по п. 4, в котором на этапе сохранения состояний памяти операционной системы упомянутые состояния сохраняют как файл-образ спящего режима.

7. Способ по п. 4, дополнительно содержащий этапы, на которых всякий раз когда требуется вернуть компьютер к его состоянию, когда был завершен упомянутый компьютерный сеанс, и начать текущий компьютерный сеанс,

45 осуществляют доступ к и загружают сохраненный файл состояний сеанса, соответствующий компьютерному сеансу, в память операционной системы компьютера вместо текущих состояний памяти операционной системы,

осуществляют доступ к файлу компьютерного сеанса, соответствующему компьютерному сеансу,

повторно инициируют захват изменений, которые были бы произведены в постоянном хранилище во время компьютерного сеанса, и сохраняют захваченные изменения в файле компьютерного сеанса, и

используют захваченные изменения, а также другие данные из постоянного хранилища во время текущего компьютерного сеанса.

8. Способ по п. 7, дополнительно содержащий этапы, на которых всякий раз когда завершается текущий компьютерный сеанс,

сохраняют файл состояний сеанса, представляющий состояния памяти операционной системы компьютера, существующие в момент времени, когда завершен текущий компьютерный сеанс, и связывают файл состояний сеанса с пользователем и текущим завершаемым компьютерным сеансом,

формируют и сохраняют новый файл компьютерного сеанса, который обновляет последний ранее сохраненный файл сеанса посредством добавления предотвращенных изменений постоянного хранилища, захваченных во время текущего компьютерного сеанса, и связывают новый файл компьютерного сеанса с пользователем и текущим завершаемым компьютерным сеансом,

прерывают захват изменений, которые были бы произведены в постоянном хранилище во время текущего компьютерного сеанса, и

возвращают операционную систему компьютера к ее состоянию, когда были сохранены базовые состояния памяти операционной системы, причем при упомянутом возврате загружают сохраненные базовые состояния в память операционной системы вместо текущих состояний памяти операционной системы.

9. Способ по п. 8, в котором файлы состояний сеанса и файлы компьютерного сеанса, связанные с несколькими компьютерными сеансами, являются доступными для компьютера, причем этапы осуществления доступа к и загрузки сохраненного файла состояний сеанса, соответствующего компьютерному сеансу, в память операционной системы компьютера вместо текущих состояний памяти операционной системы и осуществления доступа к файлу компьютерного сеанса, соответствующему компьютерному сеансу, содержат этапы, на которых:

предоставляют список доступных файлов состояний сеанса и файлов компьютерного сеанса пользователю, который начинает текущий компьютерный сеанс;

вводят пользовательский выбор файла состояний сеанса и файла компьютерного сеанса из предоставленного списка, которые соответствуют предыдущему компьютерному сеансу, который пользователь хочет повторно открыть в качестве текущего сеанса; и

используют выбранные пользователем файл состояний сеанса и файл компьютерного сеанса, чтобы вернуть компьютер к его состоянию, когда был завершен предыдущий компьютерный сеанс, и начать текущий компьютерный сеанс.

10. Способ по п. 9, в котором этапы сохранения файлов состояний сеанса и файлов компьютерного сеанса содержат этап, на котором сохраняют эти файлы в энергонезависимой памяти, доступной для компьютера.

11. Способ по п. 8, в котором этап возврата операционной системы компьютера к ее состоянию, когда были сохранены базовые состояния памяти операционной системы, дополнительно содержит этапы, на которых:

делают сохраненный файл состояний сеанса и файл компьютерного сеанса, соответствующие последнему завершенному компьютерному сеансу, доступными пользователю для загрузки; и

удаляют сохраненный файл состояний сеанса и файл компьютерного сеанса,

соответствующие последнему завершеному компьютерному сеансу, из памяти, доступной для компьютера.

12. Способ по п. 11, в котором этапы доступа к файлу состояний сеанса и файлу компьютерного сеанса содержат этапы, на которых:

5 запрашивают пользовательский ввод файла состояний сеанса и файла компьютерного сеанса; и

вводят файл состояний сеанса и файл компьютерного сеанса, когда они приняты от пользователя.

13. Способ по п. 12, в котором этапы сохранения файла состояний сеанса и сохранения 10 файла компьютерного сеанса содержат этап, на котором связывают эти файлы с информацией проверки допустимости, которая содержит указатель базовой версии, идентифицирующий версию базовых состояний памяти операционной системы, из которых были получены упомянутые файлы.

14. Способ по п. 13, в котором этапы доступа к файлу состояний сеанса и файлу 15 компьютерного сеанса дополнительно содержат этапы, на которых:

определяют, были ли файл состояний сеанса и файл компьютерного сеанса получены из базовых состояний памяти операционной системы, используемой компьютером, на основе информации проверки допустимости, связанной с этими файлами;

20 осуществляют доступ к файлу состояний сеанса и файлу компьютерного сеанса всякий раз, когда определено, что файл состояний сеанса и файл компьютерного сеанса были получены из базовых состояний памяти операционной системы, используемой компьютером; и

25 сообщают пользователю, что к файлу состояний сеанса и файлу компьютерного сеанса нельзя осуществлять доступ, всякий раз когда определено, что файл состояний сеанса и файл компьютерного сеанса не были получены из базовых состояний памяти операционной системы, используемой компьютером.

15. Способ по п. 8, в котором этап сохранения файла состояний сеанса содержит этапы, на которых:

30 идентифицируют различия между базовыми состояниями памяти операционной системы и состояниями памяти операционной системы, существующими в момент времени, когда завершен последний проведенный компьютерный сеанс; и

сохраняют идентифицированные различия как файл разности состояний сеанса.

16. Способ по п. 15, в котором этап доступа к сохраненному файлу состояний сеанса, соответствующему компьютерному сеансу, содержит этапы, на которых:

35 формируют и сохраняют восстановленный файл состояний сеанса посредством применения различий в файле различий состояний сеанса к базовым состояниям памяти операционной системы; и

назначают восстановленный файл состояний сеанса в качестве сохраненного файла состояний сеанса, к которому должен быть осуществлен доступ.

40 17. Компьютерно-реализуемый способ захвата и загрузки состояний операционной системы, в котором используют компьютер для выполнения следующих этапов, на которых:

45 (a) сохраняют состояния памяти операционной системы компьютера, существующие в момент времени, заданный пользователем компьютера, причем упомянутые состояния сохраняются в файле базовых состояний в постоянном хранилище, связанном с компьютером;

(b) предотвращают изменения в постоянном хранилище, связанном с компьютером, во время компьютерного сеанса;

(с) захватывают изменения, которые были бы произведены в постоянном хранилище во время компьютерного сеанса, если бы они не были предотвращены;

(d) используют захваченные изменения, а также другие данные из постоянного хранилища во время текущего компьютерного сеанса,

5 (е) всякий раз, когда компьютерный сеанс завершается, сохраняют файл состояний сеанса, представляющий состояния памяти операционной системы компьютера, существующие в момент времени, когда компьютерный сеанс завершается,

10 прерывают захват изменений, которые были бы произведены в постоянном хранилище во время текущего компьютерного сеанса, и сохраняют захваченные изменения и

возвращают операционную систему компьютера к ее состоянию, когда были сохранены базовые состояния памяти операционной системы, причем при упомянутом возврате загружают сохраненные базовые состояния в память операционной системы

15 вместо текущих состояний памяти операционной системы; (f) осуществляют доступ к и загружают предписанную часть состояний из сохраненного файла состояний сеанса в память операционной системы компьютера вместо соответствующих текущих состояний памяти операционной системы компьютера;

(g) повторно иницируют захват изменений, которые были бы произведены в

20 постоянном хранилище; (h) используют захваченные изменения, а также другие данные из постоянного хранилища и сохраненные изменения во время текущей итерации отладки для отладки операционной системы компьютера;

(i) определяют, была ли отлажена операционная система компьютера;

25 (j) всякий раз, когда определено, что операционная система компьютера не была отлажена,

завершают текущую итерацию отладки,

прерывают захват изменений, которые были бы произведены в постоянном хранилище во время текущей итерации отладки, и сохраняют захваченные изменения,

30 возвращают операционную систему компьютера к ее состоянию, когда были сохранены базовые состояния памяти операционной системы, причем при упомянутом возврате загружают сохраненные состояния из файла базовых состояний в память операционной системы вместо текущих состояний памяти операционной системы,

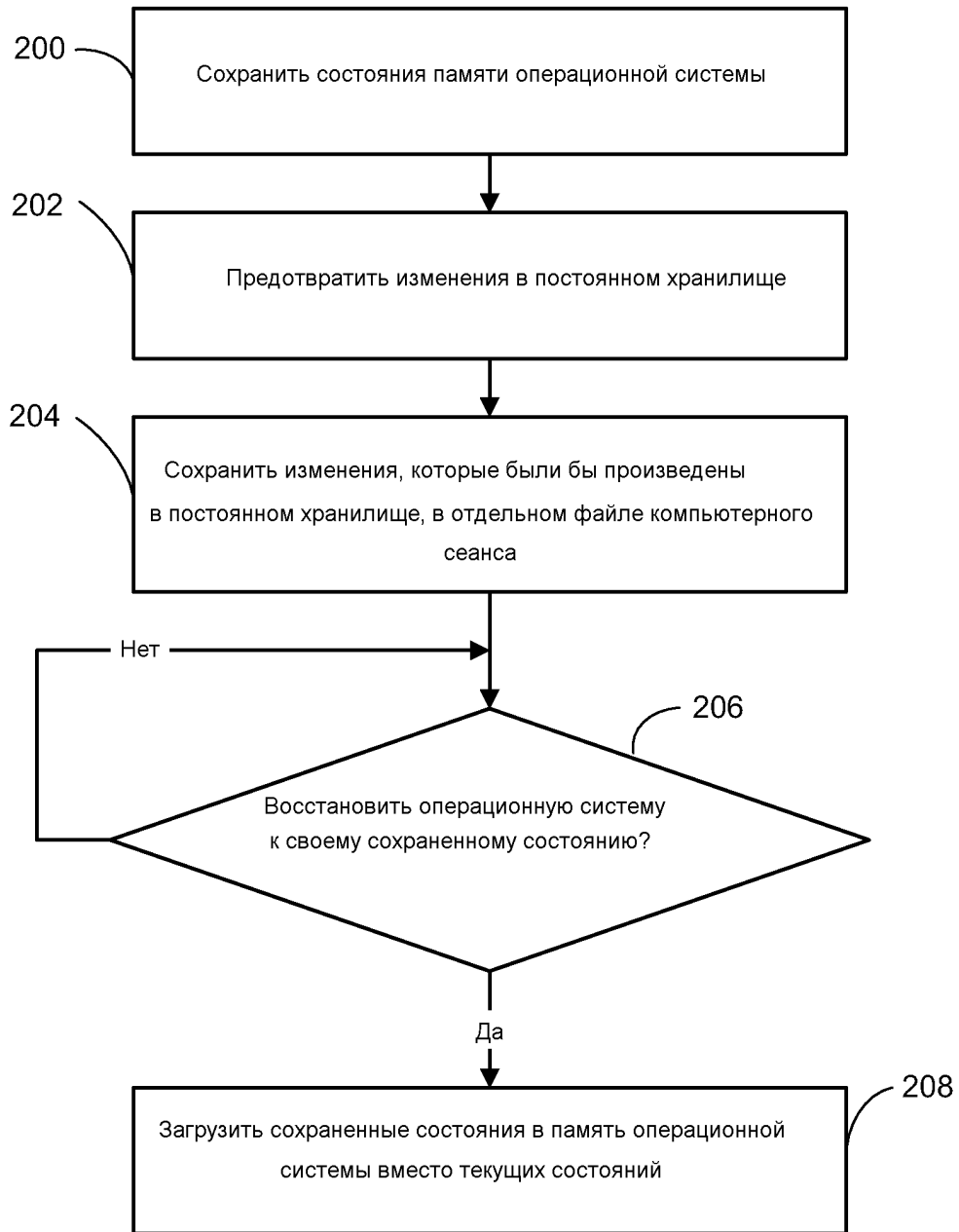
35 осуществляют доступ к и загружают увеличенную часть состояний из сохраненного файла состояний сеанса, соответствующую последней завершенной итерации отладки, в память операционной системы компьютера вместо соответствующих текущих состояний памяти операционной системы компьютера, причем эта увеличенная часть состояний из сохраненного файла состояний сеанса содержит состояния из сохраненного файла состояний сеанса, загруженные на последней завершенной итерации отладки,

40 плюс предписанную дополнительную часть состояний из сохраненного файла состояний сеанса, и

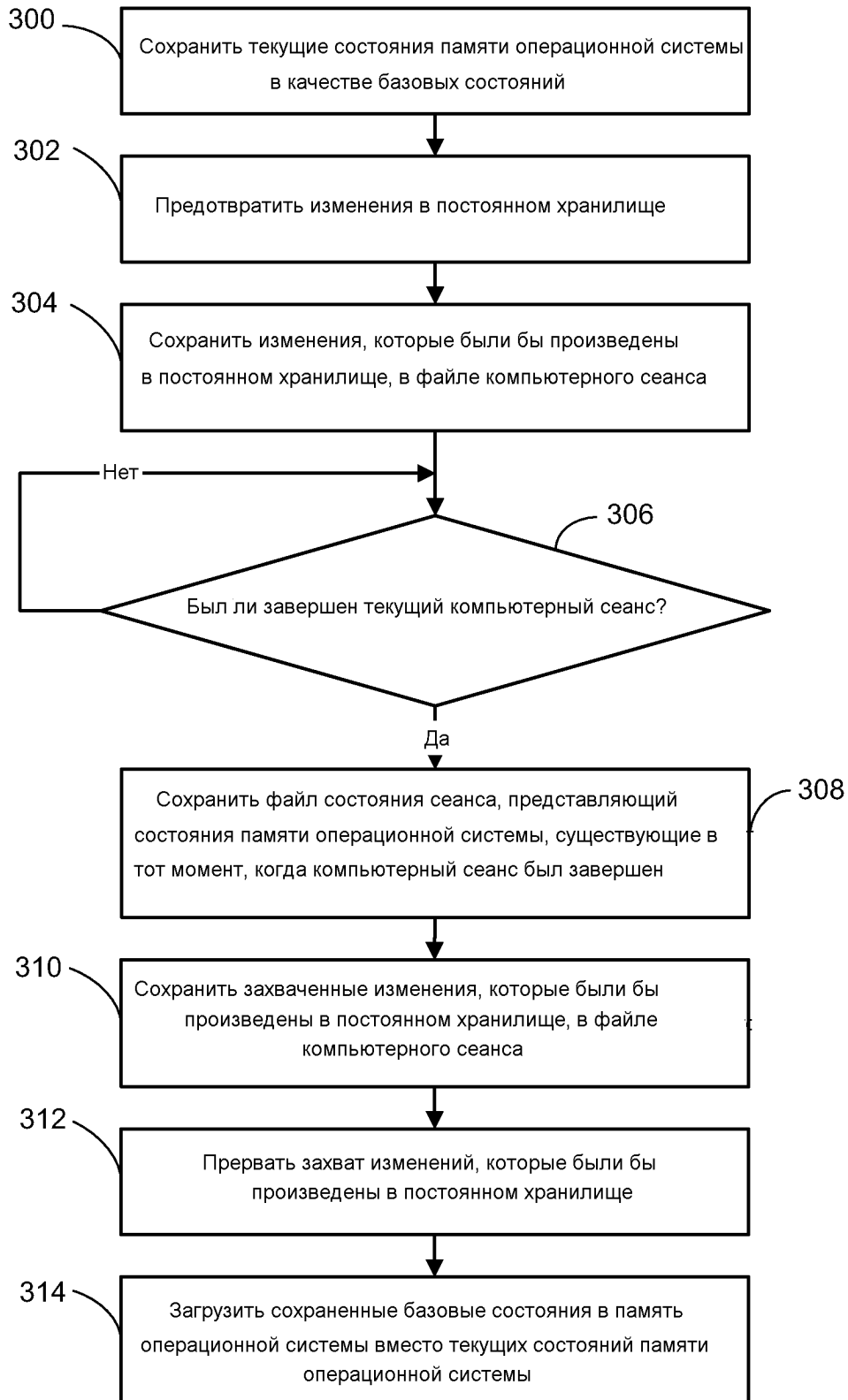
(k) повторяют этапы с (g) по (j) до тех пор, пока операционная системы компьютера не будет отлажена.

18. Способ по п. 17, в котором компьютерный сеанс автоматически завершается операционной системой компьютера вследствие ошибки, причем файл состояний сеанса представляет собой файл выгрузки сбоя.

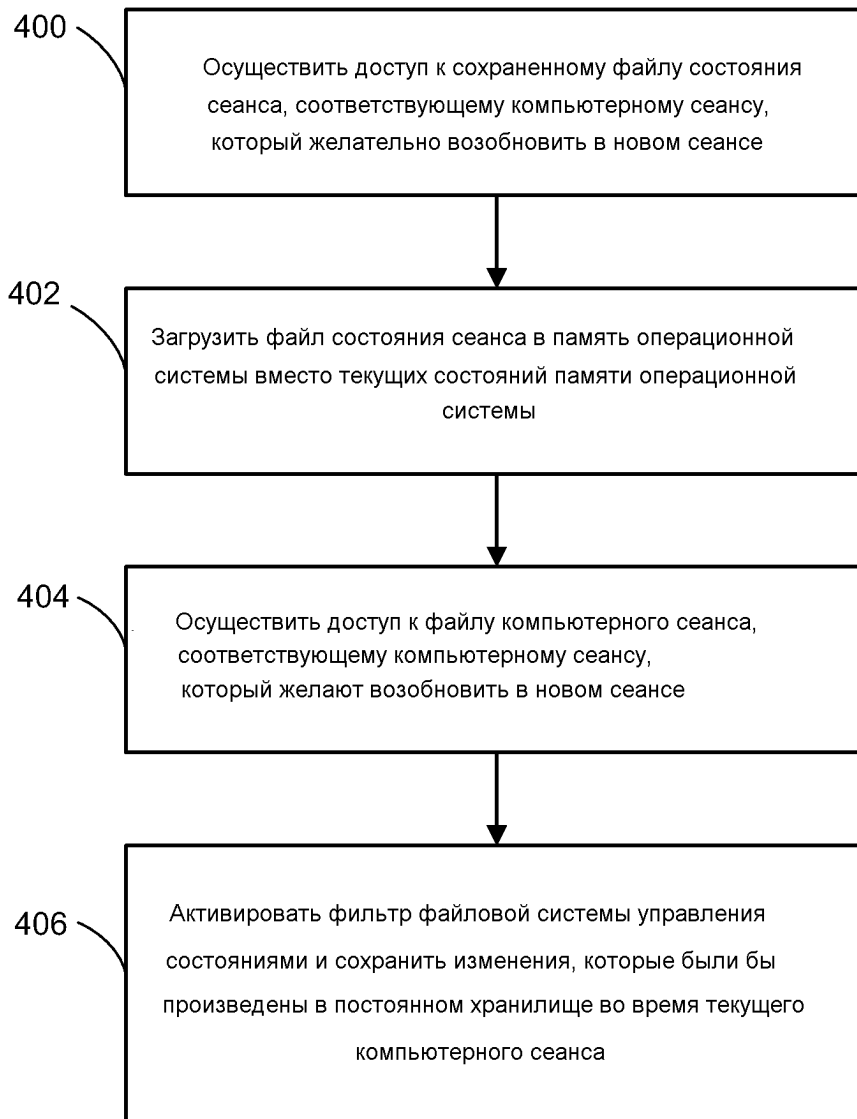
19. Способ по п. 17, в котором компьютерный сеанс завершается пользователем с целью отладки операционной системы компьютера.



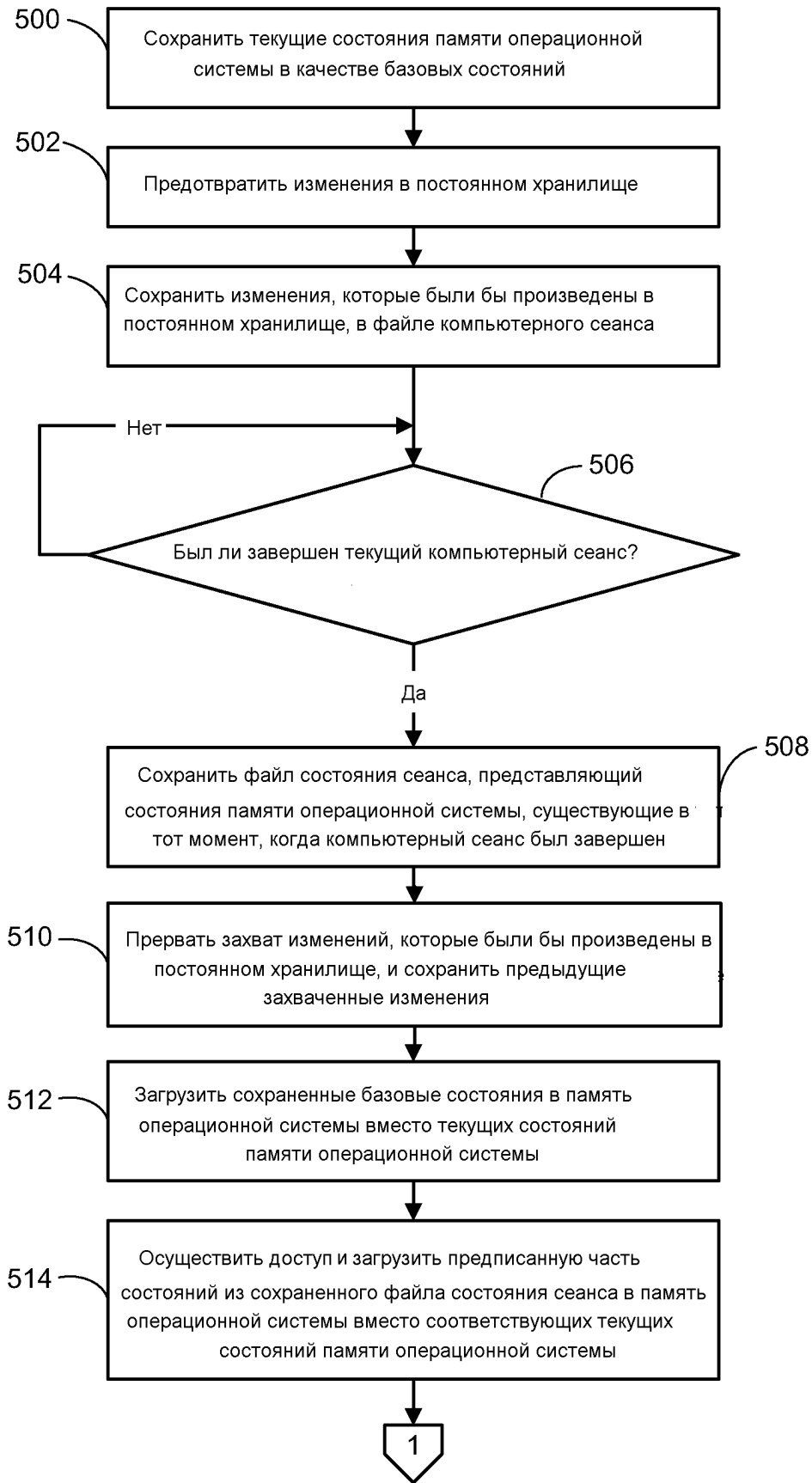
ФИГ.2



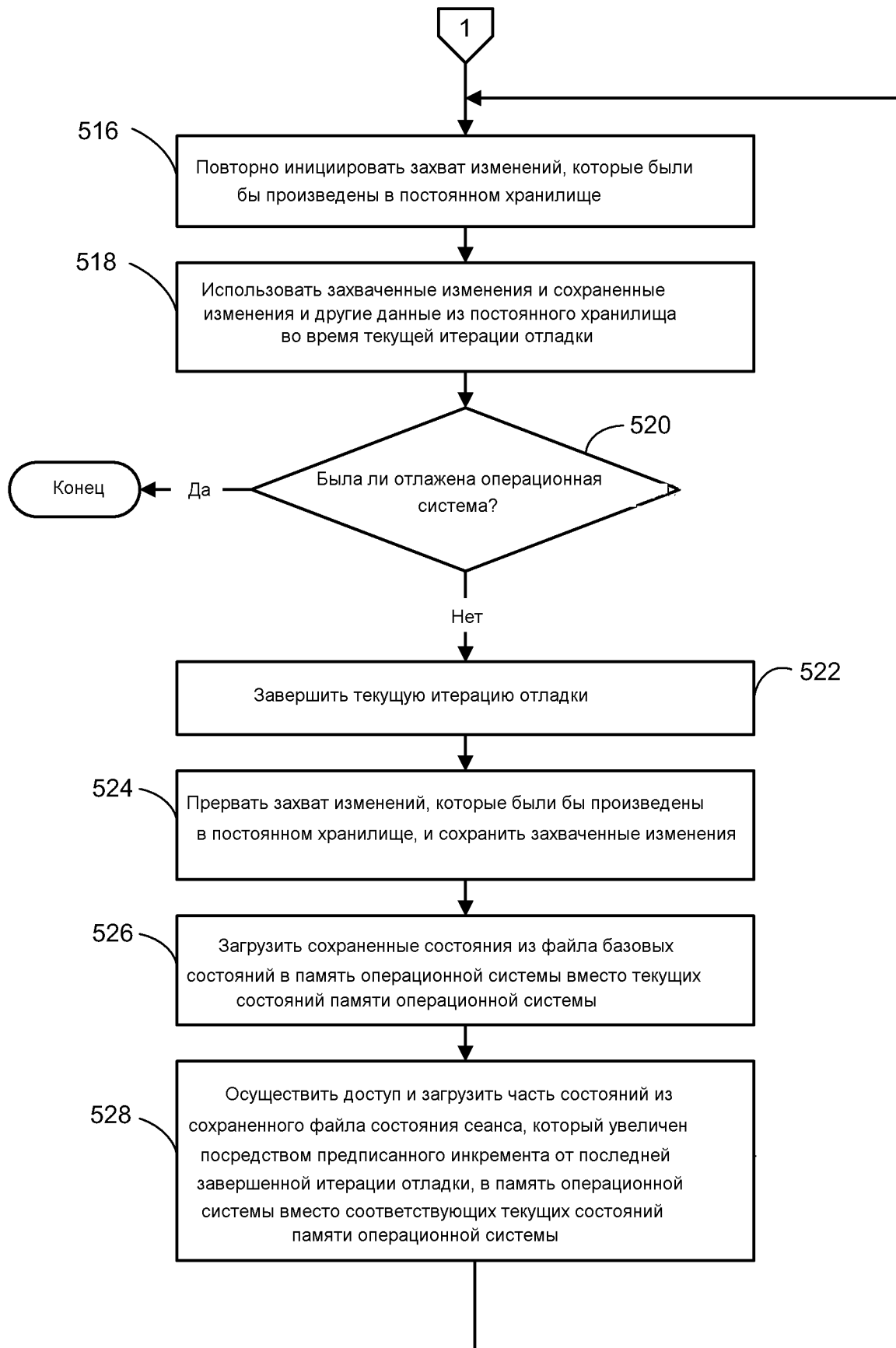
ФИГ.3



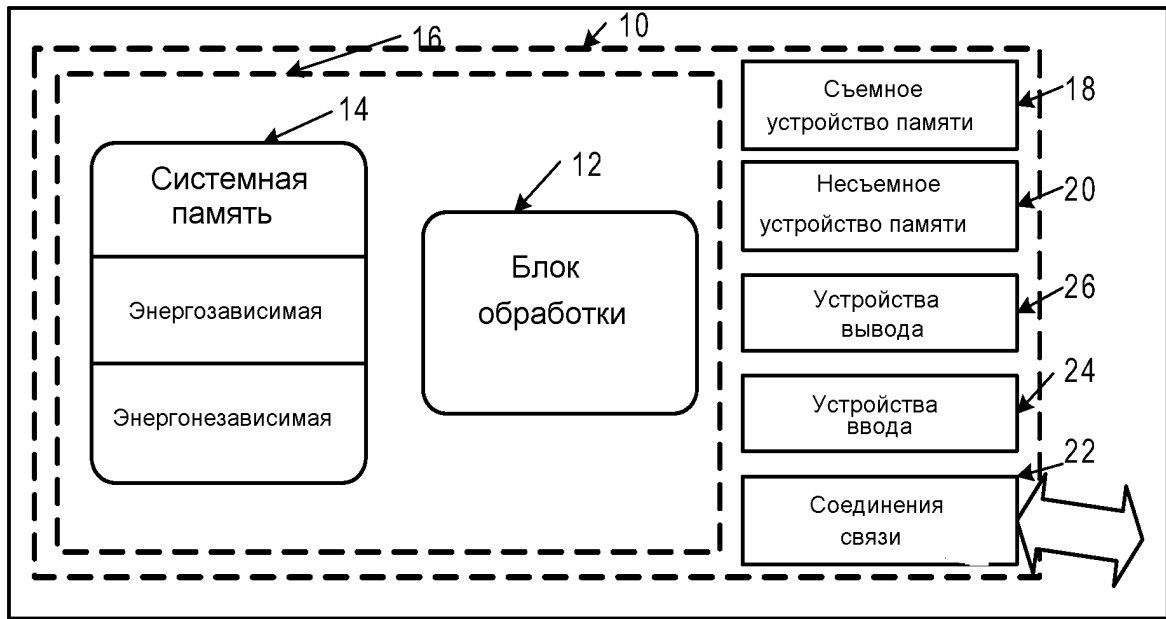
ФИГ.4



ФИГ.5А



ФИГ.5В



ФИГ.6