(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 744 379**

(51) Int. CI.:

G06F 9/44 (2008.01)
G06F 17/00 (2009.01)
G06F 9/455 (2008.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **14.05.2007 PCT/US2007/011545**

(87) Fecha y número de publicación internacional: **22.11.2007 WO07133741**

(96) Fecha de presentación y número de la solicitud europea: **14.05.2007 E 07777032 (9)**

(97) Fecha y número de publicación de la concesión europea: **26.06.2019 EP 2024826**

(54) Título: **Lanzamiento del hipervisor bajo el sistema operativo en ejecución**

(30) Prioridad:

15.05.2006 US 383455

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
24.02.2020

(73) Titular/es:

MICROSOFT TECHNOLOGY LICENSING, LLC (100.0%)
One Microsoft Way
Redmond, WA 98052, US

(72) Inventor/es:

GANGULY, SHUVABRATA;
THORNTON, ANDREW, J.;
WIEDERHIRN, JOHN, F. y
RAY, KENNETH, D.

(74) Agente/Representante:

CARPINTERO LÓPEZ, Mario

DESCRIPCIÓN

Lanzamiento del hipervisor bajo el sistema operativo en ejecución

Antecedentes

Una de las funciones principales de un sistema operativo es interactuar con los recursos físicos en un sistema informático. Un sistema operativo típico podría evitar el acceso a los recursos físicos de una manera inapropiada. Por ejemplo, cuando una aplicación usa un segmento particular de memoria, el sistema operativo puede proteger ese segmento de memoria para que no sea alterado por otra aplicación administrada por ese mismo sistema operativo. En caso contrario, las aplicaciones pueden no funcionar como se espera. Tales protecciones de acceso a menudo se basan en el supuesto de que el sistema operativo es el único sistema operativo que se ejecuta en el sistema informático.

Sin embargo, a veces puede ser ventajoso ejecutar múltiples sistemas operativos en el mismo sistema informático. En ese caso, las protecciones implícitas en cada sistema operativo para garantizar una operación segura con recursos pueden ya no ser suficientes. Es posible que un sistema operativo no pueda controlar el acceso a los mismos recursos físicos por parte de otro sistema operativo y es posible que ni siquiera tenga un mecanismo para conocer la existencia de ese otro sistema operativo en ejecución.

Un hipervisor es una capa de software que está configurada para interponerse entre uno o más sistemas operativos en ejecución y recursos físicos protegidos (como procesadores, puertos de E / S, memoria, interrupciones, etc.). El hipervisor funcionalmente multiplexa los recursos físicos protegidos para los sistemas operativos y manifiesta los recursos para cada sistema operativo de manera virtualizada. Por ejemplo, como un simple ejemplo, supongamos que hay dos sistemas operativos que se ejecutan en un sistema informático que tiene un procesador y 1 Gigabyte (GB) de memoria de acceso aleatorio (RAM). El hipervisor puede asignar la mitad de los ciclos del procesador a cada sistema operativo y la mitad de la memoria (512 Megabytes (MB) de RAM) a cada sistema operativo. Asimismo, el hipervisor puede proporcionar un rango de RAM virtualizado dirigido a cada sistema operativo, de modo que a ambos sistemas operativos les parece que solo hay 512 MB de RAM disponibles.

Cuando el sistema operativo intenta comunicarse con un recurso físico y viceversa, el hipervisor realiza el almacenamiento intermedio y las transformaciones adecuadas para permitir que cada sistema operativo experimente su entorno como si fuera el único sistema operativo que se ejecuta en el sistema informático. Asimismo, el hipervisor hace esto de una manera en la que los recursos físicos en el sistema informático pueden ser compartidos por múltiples instancias del sistema operativo mientras están protegidos.

Tradicionalmente, los hipervisores se inician antes de ejecutar un sistema operativo. Esto permite que el hipervisor inicie el sistema operativo en una máquina virtual presentando una vista virtualizada de los recursos físicos. Para iniciar inmediatamente el sistema operativo en la máquina virtual, el hipervisor incluye un extenso código para descubrir los recursos físicos y sus características esenciales. Dado que el descubrimiento de recursos físicos se realiza antes de que haya sistemas operativos en ejecución, no se puede confiar en el sistema operativo en este procedimiento de descubrimiento. Por consiguiente, el código para descubrir recursos físicos en un hipervisor puede ser bastante complejo.

En King S. T. y col.: "Soporte de sistema operativo para máquinas virtuales", Actas de la Conferencia Técnica Anual de USENIX, 4 de junio de 2003, páginas 71-84; los autores examinaron la sobrecarga de rendimiento asociada con la ejecución de un monitor de máquina virtual (VMM) en un sistema operativo principal en comparación con la ejecución fuera de una máquina virtual (un sistema independiente) o con VMM que son construidas directamente en el hardware.

Sumario

La invención es lo que se especifica en las reivindicaciones independientes.

Se especifican las reivindicaciones preferentes en las reivindicaciones dependientes.

Aunque los principios de la presente invención no están limitados a las realizaciones señaladas en este breve sumario, algunas realizaciones descritas en el presente documento se relacionan con el lanzamiento de un hipervisor después de que ya existe un sistema operativo en ejecución. Aunque no es necesario, se puede usar el sistema operativo en ejecución en lugar del hipervisor para descubrir los recursos físicos que se ejecutan en el sistema informático. De esta manera, si se desea, el hipervisor puede confiar en el sistema operativo para descubrir los recursos, en lugar de tener que tener código que realice la misma funcionalidad.

Este sumario se proporciona para introducir una selección de conceptos de una forma simplificada, que se describirán de manera más pormenorizada, más adelante, en la descripción detallada. Este sumario no tiene la intención de identificar características clave o características esenciales del objeto reclamado, ni está destinado a ser utilizado como una ayuda para determinar el ámbito del objeto reivindicado.

Dibujos

Para aclarar aún más las ventajas y características anteriores y otras de la presente invención, una descripción más particular de la invención se mostrará por referencia a las realizaciones específicas de la misma que se ilustran en los dibujos adjuntos. Se aprecia que estos dibujos representan solo realizaciones típicas de la invención que, por lo tanto, no deben considerarse como limitación de su ámbito. La invención se describirá y explicará con especificidad adicional y detalle a través del uso de los dibujos adjuntos en los que:

La figura 1 ilustra un sistema informático adecuado en el que los principios de la presente invención pueden emplearse;
La figura 2 ilustra un diagrama de flujo de un procedimiento para iniciar un hipervisor utilizando un sistema operativo en ejecución;
La figura 3 ilustra un diagrama de flujo de un procedimiento para lanzar instancias adicionales del sistema operativo; La figura 4A ilustra una configuración en la que un sistema operativo raíz, en ausencia de un hipervisor, está en comunicación directa con los recursos físicos del sistema informático;
La figura 4B ilustra una configuración en la que el sistema operativo ha lanzado un hipervisor para que actúe como intermediario entre el sistema operativo y los recursos físicos;
La figura 4C ilustra una configuración en la que el hipervisor ha lanzado y soportado sistemas operativos adicionales; y
La figura 5 ilustra un diagrama de flujo de un procedimiento para transportar el sistema operativo al entorno del hipervisor.

Descripción detallada

De acuerdo con realizaciones de la presente invención, se puede iniciar un hipervisor después de que ya haya un sistema operativo en ejecución. El sistema operativo en ejecución se puede utilizar en lugar del hipervisor para descubrir los recursos físicos que se ejecutan en el sistema informático. Se pueden iniciar otras instancias del sistema operativo después de que el hipervisor esté operativo. Un entorno informático general en el que se pueden poner en práctica los principios de la presente invención se describirá primero con respecto a la figura 1. A continuación, detalles adicionales con respecto a las realizaciones de la presente invención se describirán con respecto a las figuras posteriores.

Los sistemas informáticos ahora están tomando cada vez más una amplia variedad de formas. Los sistemas informáticos pueden, por ejemplo, ser dispositivos de mano, accesorios, ordenadores portátiles, ordenadores de escritorio, Ordenadores centrales, sistemas informáticos distribuidos, o incluso dispositivos que no se han considerado convencionalmente como un sistema informático. En esta descripción y en las reivindicaciones, el término "sistema informático" se define ampliamente como que incluye cualquier dispositivo o sistema (o una combinación de ellos) que incluya al menos un procesador, y una memoria capaz de tener en ella instrucciones ejecutables por ordenador que puedan ser ejecutadas por el procesador. La memoria puede tomar cualquier forma y puede depender de la naturaleza y la forma del sistema informático. Un sistema informático puede estar distribuido en un entorno de red y puede incluir sistemas informáticos de múltiples componentes.

En referencia a la figura 1, en su configuración más básica, un sistema 100 informático típicamente incluye al menos una unidad 102 de procesamiento y una memoria 104. La memoria 104 puede ser memoria física del sistema, que puede ser volátil, no volátil, o alguna combinación de las dos. Un ejemplo de memoria volátil incluye memoria de acceso aleatorio (RAM). Los ejemplos de memoria no volátil incluyen memoria de solo lectura (ROM), memoria flash o similares. El término "memoria" también se puede usar en el presente documento para referirse al almacenamiento masivo no volátil, como los medios de almacenamiento físico. Dicho almacenamiento puede ser removible o no removible, y puede incluir (pero no está limitado a) tarjetas PCMCIA, discos magnéticos y ópticos, cinta magnética, y similares.

Tal como se usa en el presente documento, el término "módulo" o "componente" puede referirse a objetos de software o rutinas que se ejecutan en el sistema informático. Los diferentes componentes, módulos, motores y servicios descritos en el presente documento pueden implementarse como objetos o procesos que se ejecutan en el sistema informático (por ejemplo, como hilos separados). Si bien el sistema y los procedimientos descritos en el presente documento pueden implementarse en software, Las implementaciones en hardware, y en combinaciones de software y hardware también son posibles y contempladas.

En la descripción que sigue, las realizaciones de la invención se describen con referencia a actos que son realizados por uno o más sistemas informáticos. Si tales actos se implementan en software, uno o más procesadores del sistema informático asociado que realiza la acción dirigen el funcionamiento del sistema informático en respuesta a la ejecución de instrucciones ejecutables por ordenador. Un ejemplo de tal operación involucra la manipulación de datos. Las instrucciones ejecutables por ordenador (y los datos manipulados) pueden almacenarse en la memoria 104 del sistema 100 informático.

El sistema 100 informático también puede contener canales 108 de comunicación que permiten que el sistema 100 informático se comunique con otros sistemas informáticos a través de, por ejemplo, una red 110. Los canales 108 de comunicación son ejemplos de medios de comunicación. Los medios de comunicación típicamente incluyen instrucciones legibles por ordenador, estructuras de datos, módulos de programa u otros datos en una señal de datos

modulada, tal como una onda portadora u otro mecanismo de transporte e incluyen cualquier medio de entrega de información. A modo de ejemplo, y sin limitación, los medios de comunicación incluyen medios por cable, tales como redes cableadas y conexiones directas por cable, y medios inalámbricos como acústicos, radio, infrarrojos, y otros medios inalámbricos. El término medios legibles por ordenador como se usa en el presente documento incluye tanto los medios de almacenamiento como los medios de comunicación.

Las realizaciones dentro del ámbito de la presente invención incluyen medios legibles por ordenador para llevar o tener instrucciones ejecutables por ordenador o estructuras de datos almacenadas en los mismos. Tales medios legibles por ordenador pueden ser cualquier medio disponible al que se puede acceder por un ordenador de objetivo general o de objetivo especial. A modo de ejemplo, y sin limitación, tales medios legibles por ordenador pueden comprender almacenamiento físico y / o medios de memoria tales como RAM, ROM, EEPROM, CD-ROM u otro almacenamiento en disco óptico, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético o cualquier otro medio que se puede usar para llevar o almacenar medios de código de programas deseados en forma de instrucciones ejecutables por ordenador o estructuras de datos y a las que se pueden acceder por un ordenador de objetivo general o de objetivo especial. Cuando la información se transfiere o proporciona sobre una red u otra conexión de comunicaciones (ya sea cableada, inalámbrica o una combinación de cableada o inalámbrica) a un ordenador, el ordenador ve correctamente la conexión como un medio legible por ordenador. De esta manera, cualquiera de tales conexiones se denomina correctamente un medio legible por ordenador. Se deberían incluir también combinaciones de lo anterior dentro del ámbito de los medios legibles por ordenador.

Las instrucciones ejecutables por ordenador comprenden, por ejemplo, instrucciones y datos que provocan que el ordenador de objetivo general, el ordenador de objetivo especial o el dispositivo de procesamiento de objetivo especial lleve a cabo una cierta función o grupo de funciones. Aunque el objeto ha sido descrito en un lenguaje específico para características estructurales y / o actos metodológicos, debe entenderse que el objeto definido en las reivindicaciones adjuntas no está necesariamente limitado a las características o actos específicos descritos en el presente documento. En su lugar, las características y actos específicos descritos en el presente documento se han desvelado como ejemplos de formas de implementar la invención reivindicada.

La figura 2 ilustra un diagrama de flujo de un procedimiento 200 para que un sistema operativo en ejecución inicie un hipervisor. En esta descripción y en las reivindicaciones, un "hipervisor" es una capa de software que está configurada para ser dispuesta entre uno o más sistemas operativos en ejecución y recursos físicos protegidos. El hipervisor funcionalmente multiplexa los recursos físicos protegidos para los sistemas operativos y manifiesta los recursos para cada sistema operativo de manera virtualizada. De esta manera, el hipervisor actúa como un tipo especial de capa de abstracción entre el sistema operativo y los recursos físicos del sistema informático.

Por ejemplo, como un simple ejemplo, supongamos que hay dos sistemas operativos que se ejecutan en un sistema informático que tiene un procesador y 1 Gigabyte (GB) de memoria de acceso aleatorio (RAM). El hipervisor puede asignar la mitad de los ciclos del procesador a cada sistema operativo y la mitad de la memoria (512 Megabytes (MB) de RAM) a cada sistema operativo. Asimismo, el hipervisor puede proporcionar un rango de RAM virtualizado dirigido a cada sistema operativo, de modo que a ambos sistemas operativos les parece que solo hay 512 MB de RAM disponibles. Cuando el sistema operativo intenta comunicarse con un recurso físico y viceversa, el hipervisor realiza el almacenamiento intermedio y las transformaciones adecuadas para permitir que cada sistema operativo experimente su entorno como si fuera el único sistema operativo que se ejecuta en el sistema informático.

A diferencia de las configuraciones de hipervisor convencionales, las formas de realización de la presente invención permiten que el hipervisor se inicie después de que se haya lanzado el sistema operativo, incluso si el sistema operativo ya ha descubierto los recursos físicos del sistema informático. El sistema operativo que inicia el hipervisor pasa información que representa el estado de los recursos físicos descubiertos al hipervisor. De esta manera, el hipervisor no necesita tener un código separado para descubrir los recursos físicos del sistema informático.

Una vez que el sistema operativo inicia el hipervisor y pasa a la información del hipervisor con respecto a los recursos físicos descubiertos del sistema informático, el sistema operativo puede pausar la ejecución y pasar el control al hipervisor. El hipervisor podría entonces configurar una instancia de máquina virtual para manejar futuras solicitudes de acceso a cualquier recurso protegido de los recursos físicos del sistema informático. La instancia de la máquina virtual se inicializa con un estado que es consistente con el concepto del sistema operativo de los recursos físicos protegidos. Cuando esto se logra, el sistema operativo se reanuda en el entorno de la máquina virtual. En el entorno de máquina virtual, sin embargo, el sistema operativo interactúa con los recursos físicos indirectamente a través del hipervisor, en lugar de hacerlo directamente con los recursos físicos. El cambio es transparente para el sistema operativo, ya que la máquina virtual dedicada a la comunicación con el sistema operativo respeta la información que el sistema operativo había descubierto anteriormente sobre los recursos físicos.

En algunos casos y para algunos de los recursos físicos protegidos, puede que no sea posible que el sistema operativo descubra un recurso físico protegido sin el uso del hipervisor, y luego se conecte de manera indirecta con ese mismo recurso físico protegido a través del hipervisor de manera transparente. En ese caso, el sistema operativo puede configurarse para comprender que luego podría estar operando a través de un hipervisor y, por lo tanto, ignorar cualquiera de los recursos físicos que luego no se pueden virtualizar de manera transparente. Por ejemplo, el cargador del sistema operativo puede configurarse para informar a otros componentes del sistema operativo que no utilicen

recursos físicos protegidos particulares.

Haciendo referencia de nuevo a la figura 2, el procedimiento 200 comienza cuando se inicia el sistema operativo raíz (acción 201). Un sistema operativo "raíz" es el sistema operativo que inicia el hipervisor, en contraste con los sistemas operativos que pueden iniciarse después de que el hipervisor esté operativo. La figura 4A ilustra una configuración 400A en esta etapa de operación en la que el sistema 401 operativo se comunica directamente 404 con los recursos 402 físicos del sistema informático en ausencia de un hipervisor.

El sistema operativo raíz luego realiza varios actos que se ilustran en la columna izquierda de la figura 2 debajo del encabezado de la columna "SISTEMA OPERATIVO RAÍZ". Por ejemplo, el sistema operativo raíz descubre al menos un recurso físico (acción 211A). Los sistemas operativos suelen tener instrucciones que descubren los recursos físicos del sistema informático que se ejecutan poco después de iniciar el sistema operativo. En referencia a la figura 4A, por ejemplo, el sistema 401 operativo descubre el estado 403 de recursos físicos. Descubrir los recursos físicos de un sistema informático puede ser una tarea compleja.

Para reanudar el sistema operativo en el estado anterior al lanzamiento del hipervisor, el sistema operativo captura el estado de la máquina física antes de iniciar el hipervisor. En una forma de realización, el estado capturado incluye el estado de todos los procesadores físicos y los APIC físicos. El estado del procesador físico incluye:

1. Registros de fin general.
2. Punto flotante y registros XMM.
3. Registros de control (CR).
4. Registros de depuración (DR)
5. Puntero de instrucciones (RIP), Puntero de pila (RSP) y Registro de banderas (RFLAGS).
6. Estado del segmento para CS, DS, SS, ES, FS, segmentos GS y TR incluyendo las bases de segmentos, límites y derechos de acceso.
7. El Registro de la tabla de descriptor global (GDTR), Registro de tabla de descriptores de interrupción (IDTR) y registro de tabla de descriptor local (LDTR).
8. Ciertos registros específicos del modelo (MSR) que incluyen KernelGsBase, Star, Lstar, Cstar, Sfmask, SysenterCs, SysenterEip, SysenterEsp y Apic-Base MSRs.

El estado físico local de APIC puede incluir:

1. ID APIC local
2. Registro en solicitud (IRR)
3. Registro en servicio (ISR)
4. Registro de prioridad de tareas (TPR)

Además, pueden proporcionarse algunos aspectos del hardware al hipervisor como parámetros de inicio cuando se inicia el hipervisor. Estos podrían incluir:

1. Procesadores lógicos actuales y potenciales, incluyendo aquellos que pueden ser enchufados en caliente en tiempo de ejecución
2. Si el hipetratamiento está habilitado o deshabilitado en el BIOS
3. Rangos de RAM físicos actuales: rangos de direcciones físicas del sistema que se rellenan con RAM en el momento en que se inicia el hipervisor
4. Nodos físicos (incluidos aquellos que no tienen recursos asociados en el momento del arranque pero que pueden completarse en tiempo de ejecución)
5. Ratios de acceso a memoria entre nodos físicos
6. Direcciones de ciertas funciones de hardware a las que debe acceder el hipervisor (por ejemplo, el temporizador de administración de energía)

El sistema operativo también lanza el hipervisor (acción 211B) que debe interponerse entre el sistema operativo y los recursos físicos. Por ejemplo, con referencia a la figura 4B, el hipervisor 405 está interpuesto entre el sistema 401 operativo raíz y los recursos 402 físicos. Como parte de este lanzamiento, el sistema operativo proporciona información de estado para al menos los recursos físicos que deben protegerse en el hipervisor (acción 212). Esta información de estado incluye toda la información que un hipervisor necesitaría para descubrir el estado relevante de los recursos físicos protegidos que deben ser protegidos por el hipervisor. En una forma de realización, la información del estado puede incluir el estado de captura descrito anteriormente. Por lo menos, la información del estado incluye al menos una identificación de los recursos físicos correspondientes.

Asimismo, como parte del lanzamiento, el sistema operativo pasa el control al hipervisor. Luego, el hipervisor realiza los actos como se ilustra en la columna derecha de la figura 2 bajo el encabezado "HIPERVISOR". En particular, el hipervisor luego realiza las tareas necesarias para virtualizar los recursos físicos protegidos del sistema informático al sistema operativo raíz (acción 221).

Por ejemplo, el hipervisor puede crear una instancia de máquina virtual para el sistema operativo raíz (acción 231). En referencia a la figura 4B, el bloque 421 representa una instancia de máquina virtual para el sistema 401 operativo. Una

vez inicializado y operativo, la instancia de la máquina 421 virtual servirá como un proxy para los recursos 402 físicos para el sistema 401 operativo. La instancia de la máquina 421 virtual recibirá solicitudes de servicio del sistema 401 operativo para los recursos físicos, y realizará las transformaciones y el almacenamiento en memoria intermedia de las solicitudes correspondientes en función de la información de estado accesible a la instancia de la máquina 421 virtual. La instancia de la máquina 421 virtual hará que el hipervisor 405 solicite el servicio apropiado de los recursos 402 físicos. La instancia de la máquina 421 virtual también potencialmente reportará los resultados de la solicitud al sistema 401 operativo con las transformaciones y el almacenamiento en memoria intermedia apropiados según sea necesario.

La instancia de la máquina 421 virtual se comportará de manera diferente dependiendo de la información de estado accesible a la instancia de la máquina virtual. El hipervisor respeta la información de estado que el sistema 401 operativo ya ha descubierto con respecto a los recursos 402 físicos. Por consiguiente, el hipervisor 405 inicializa la instancia de la máquina 421 virtual con al menos parte de la información de estado proporcionada por el sistema operativo (acción 232). Por ejemplo, el hipervisor 405 puede inicializar la máquina virtual con el estado de captura proporcionado por el sistema operativo. En esta forma, la instancia de la máquina 421 virtual se inicializa con un estado coherente con la información que representa los recursos físicos detectados por el sistema operativo. El sistema operativo se reanuda luego en el entorno de la máquina virtual (acción 233 y acción 214). En este entorno, como se ve en la figura 4B, en lugar de que el sistema 401 operativo interactúe directamente con los recursos 402 físicos, los recursos 402 físicos se virtualizan para el sistema 401 operativo mediante el uso de la instancia de la máquina 421 virtual y el hipervisor 405. Dado que la información de estado utilizada por la máquina 421 virtual es consistente con la información de estado descubierta por el sistema 401 operativo, el cambio es transparente para el sistema 401 operativo en algunas realizaciones.

En una forma de realización, la virtualización se proporciona a través de una abstracción del procesador virtual que emula el comportamiento del procesador físico. Del mismo modo, proporciona un APIC virtual que emula el comportamiento del APIC físico. Esto se consigue de la siguiente manera:

1. El estado del procesador virtual se inicializa al estado capturado del procesador físico.
2. El estado de la APIC virtual se inicializa al estado capturado de la APIC local física.
3. El hipervisor instala las interceptaciones para evitar que el sistema operativo acceda a los recursos de hardware físicos privilegiados. Por ejemplo, la dirección física del huésped donde antes se encontraba el APIC local antes de iniciar el hipervisor se marca como no presente, de modo que todos los accesos al APIC local queden atrapados en el hipervisor.

Tras el lanzamiento del hipervisor, el hipervisor puede lanzar sistemas operativos adicionales, ya sea instancias del mismo sistema operativo, o instancias de diferentes sistemas operativos. Por ejemplo, con referencia a la figura 4C, los sistemas 412 y 413 operativos entre otros potencialmente representados por las elipses 414 también pueden ser lanzados. La figura 3 ilustra un diagrama de flujo de un procedimiento 300 para virtualizar recursos físicos a los sistemas operativos adicionales también. Cuando se va a lanzar un sistema operativo adicional, el hipervisor inicia primero una instancia de máquina virtual correspondiente (acción 301), a través del cual se lanza el sistema operativo (acción 302). El hipervisor utiliza la instancia de máquina virtual correspondiente para virtualizar los recursos físicos al sistema operativo adicional correspondiente (acción 303).

Cuando cada sistema operativo realiza el descubrimiento de los recursos físicos al iniciar el sistema operativo, las diversas solicitudes de información son interceptadas por la instancia de máquina virtual correspondiente. En lugar de averiguar la información de estado real asociada con los recursos físicos, la máquina virtual correspondiente proporciona información de estado virtualizada al sistema operativo.

Algunas veces, el sistema operativo que inicia el hipervisor puede estar en un tipo de entorno diferente. Por ejemplo, tal vez el sistema operativo está funcionando en modo de 32 bits, mientras que el hipervisor que se lanzará es operar en modo de 64 bits, o viceversa. De manera similar, el sistema operativo y el hipervisor pueden estar operando en diferentes modos de paginación. Algunas realizaciones de la presente invención permiten que el sistema operativo inicie el hipervisor incluso si el sistema operativo y el hipervisor están funcionando en diferentes entornos.

La figura 5 ilustra un diagrama de flujo de un procedimiento 500 para que el sistema operativo ingrese al entorno del hipervisor en preparación para lanzar el hipervisor. A partir de la acción 201 de la figura 5 en la que se lanza el sistema operativo, el sistema operativo primero hace inactiva cualquier interrupción no enmascarable (acción 501), y enmascara cualquier interrupción enmascarable. Las interrupciones que no se pueden enmascarar se pueden dejar inaccionables de diferentes maneras. Después de todo, cuando el sistema operativo está en transición al entorno operativo del hipervisor, se debe tener cuidado para garantizar que no se produzcan interrupciones o excepciones antes de que se cargue el estado inicial del hipervisor. Si se produce una interrupción o excepción después de abandonar el entorno del sistema operativo pero antes de ingresar al entorno del hipervisor, es probable que el procesador no pueda manejar la interrupción o excepción, ya que no hay una tabla de descriptores de interrupción o una pila. La mayoría de las excepciones pueden evitarse fácilmente ya que son iniciadas por software. Las interrupciones de hardware enmascarables pueden inhibirse durante este procedimiento al borrar el bit IF en el registro RFLAGS.

Las interrupciones no enmascarables (NMI) se pueden inhibir por cualquiera de los dos mecanismos:

1. Entregar automáticamente un NMI y no ejecutar una instrucción IRET: Esto se puede lograr modificando temporalmente la dirección del controlador NMI en la tabla del descriptor de interrupciones del sistema operativo para que apunte a un controlador diferente. Entonces se puede entregar un NMI al procesador actual. Esto hará que el procesador salte a la dirección proporcionada como el controlador NMI. En el controlador de NMI, podemos restaurar la dirección original del controlador NMI y continuar. Esto ocultará de manera efectiva a otros NMI, ya que en la arquitectura x86, los NMI se enmascaran después de recibir un NMI hasta que se ejecuta una instrucción IRET.
2. Ejecutar siempre con una tabla de descriptor de interrupción válida (IDT) y apile: Esto se puede lograr creando un IDT temporal y una pila. Las tablas de páginas temporales pueden asignar el IDT temporal, el controlador de NMI y la pila en sus direcciones virtuales originales y sus respectivas direcciones físicas. Esto garantiza que si llega un NMI cuando el procesador se está ejecutando con las tablas de páginas temporales, se entregará correctamente al controlador.

Una vez que las interrupciones se enmascaran o de lo contrario se vuelven inacotables (acción 501), se crea una instancia de máquina virtual temporal (acción 502). El sistema operativo luego inicializa la instancia de la máquina virtual temporal con una instrucción que causa una intercepción (acción 503). Una intercepción es una transferencia de control del sistema operativo al hipervisor. Cuando la instancia de la máquina virtual temporal se reanuda (acción 504), la instancia de la máquina virtual ejecuta las instrucciones que causan la intercepción y, por lo tanto, se genera la intercepción (acción 505). En consecuencia, la instancia de la máquina virtual temporal comienza a ejecutarse utilizando el estado del hipervisor (acción 506), causando así que el sistema operativo continúe operando en modo hipervisor (acción 507). El sistema operativo puede lanzar el hipervisor. Opcionalmente, la instancia de la máquina virtual temporal puede ser destruida (acción 508), ya que solo era necesario poner el sistema operativo en el modo hipervisor necesario para iniciar el hipervisor. ,

Por consiguiente, Las realizaciones de la presente invención permiten que se inicie un hipervisor incluso después de que ya haya un sistema operativo en ejecución presente en el sistema informático. En algunas realizaciones, el sistema operativo puede iniciar el hipervisor incluso si el sistema operativo y el hipervisor están en entornos diferentes.

La presente invención puede materializarse en otras formas específicas sin apartarse de sus características esenciales. Las realizaciones descritas deben considerarse en todos los aspectos solo como ilustrativas y no restrictivas. El ámbito de la invención es, por lo tanto, indicado por las reivindicaciones adjuntas en lugar de por la descripción anterior. Todos los cambios que entren dentro del significado de las reclamaciones deben ser incluidos dentro de su ámbito.

REIVINDICACIONES

1. Un producto de programa de ordenador que comprende uno o más medios legibles por ordenador que tienen una o más instrucciones ejecutables por ordenador que, cuando es ejecutado por uno o más procesadores (102) de un sistema (100) informático, la una o más instrucciones ejecutables por ordenador hacen que el sistema informático realice un procedimiento (200) para usar un sistema operativo en ejecución para iniciar un hipervisor (405), comprendiendo el procedimiento:

una acción de lanzamiento de un sistema (401) operativo;
una acción del sistema operativo que descubre al menos un recurso (402) físico de un sistema informático;
una acción del sistema operativo que lanza un hipervisor (405);
una acción del sistema operativo que proporciona información (403) de estado para al menos un recurso físico al hipervisor, la información del estado que incluye al menos una identificación del recurso físico correspondiente;
una acción del sistema operativo que detiene la ejecución y pasa el control al hipervisor;
una acción del hipervisor que crea una instancia de máquina (421) virtual para el sistema operativo;
una acción de inicializar la instancia de la máquina virtual con información de estado proporcionada por el sistema operativo;
una acción del hipervisor que reanuda el sistema operativo en el entorno de la máquina virtual, después de inicializar la instancia de la máquina virtual; y
después de reanudar el sistema operativo, una acción del hipervisor que utiliza la instancia de la máquina virtual para virtualizar al menos un recurso físico al sistema operativo.

2. Un producto de programa informático de acuerdo con la reivindicación 1, en el que uno o más medios legibles por ordenador son la memoria del sistema físico y / o los medios de almacenamiento físico.

3. Un producto de programa informático de acuerdo con la reivindicación 1, en el que las instrucciones ejecutables por ordenador están estructuradas de manera tal que, cuando es ejecutado por uno o más procesadores del sistema informático, siendo el sistema informático obligado a realizar lo siguiente para cada sistema operativo adicional lanzado en el sistema informático:

una acción del hipervisor que lanza una instancia de máquina virtual correspondiente para cada sistema operativo adicional;
una acción de lanzar el sistema operativo adicional correspondiente después de que se lance la instancia de máquina virtual correspondiente; y
una acción del hipervisor que utiliza la instancia de máquina virtual correspondiente para virtualizar al menos un recurso físico al sistema operativo adicional correspondiente.

4. Un producto de programa informático de acuerdo con la reivindicación 1, en el que las instrucciones ejecutables por ordenador están estructuradas de manera tal que, cuando es ejecutado por uno o más procesadores del sistema informático, se hace que el sistema informático realice lo siguiente:

una acción del sistema operativo que crea una instancia de máquina virtual temporal;
una acción de inicializar la instancia de la máquina virtual temporal con una instrucción que genera una intercepción;
una acción de reanudar la instancia de máquina virtual temporal después de la acción de inicialización;
al detectar la intercepción resultante de la acción de reanudar la instancia de la máquina virtual temporal, una acción de iniciar la instancia de máquina virtual temporal para operar usando el estado de hipervisor.

5. Un producto de programa informático de acuerdo con la reivindicación 4, en el que las instrucciones ejecutables por ordenador comprenden además instrucciones ejecutables por ordenador que, cuando son ejecutadas por uno o más procesadores del sistema informático, el sistema informático debe realizar lo siguiente antes de la creación de la instancia de la máquina virtual temporal: una acción de hacer no accionables las interrupciones no enmascarables.

6. Un producto de programa informático de acuerdo con la reivindicación 4, en el que las instrucciones ejecutables por ordenador comprenden además instrucciones ejecutables por ordenador que, cuando son ejecutadas por uno o más procesadores del sistema informático, se hace que el sistema informático realice lo siguiente después de que se inicie la instancia de la máquina virtual temporal utilizando el estado del hipervisor:

una acción de lanzamiento del hipervisor; y
una acción de destruir la instancia de la máquina virtual temporal.

7. Un producto de programa informático de acuerdo con la reivindicación 6, en el que uno de los sistemas operativos y el hipervisor opera en modo de 32 bits, mientras que el otro del sistema operativo y el hipervisor funciona en modo de 64 bits, o
en el que el sistema operativo y el hipervisor operan usando un mecanismo de paginación diferente.

8. Un procedimiento (200) para iniciar un hipervisor utilizando un sistema operativo en ejecución para iniciar un hipervisor, comprendiendo el procedimiento:

lanzar (201) un sistema operativo; descubrir (211A), mediante el sistema operativo, al menos un recurso físico de un sistema informático;
lanzar (211B), mediante el sistema operativo, un hipervisor;
proporcionar (212), mediante el sistema operativo, información de estado para el al menos un recurso físico al hipervisor, incluyendo la información del estado al menos una identificación del recurso físico correspondiente;
pausar la ejecución del sistema operativo y pasar el control al hipervisor;
crear (231), mediante el hipervisor, una instancia de máquina virtual para el sistema operativo;
inicializar (232), mediante el hipervisor, la instancia de máquina virtual con información de estado proporcionada por el sistema operativo;
reanudar (214 - 233), mediante el hipervisor, el sistema operativo en el entorno de máquina virtual, después de inicializar la instancia de la máquina virtual; y
después de reanudar el sistema operativo, utilizar, mediante el hipervisor, la instancia de la máquina virtual para virtualizar el al menos un recurso físico para el sistema operativo.

9. Un procedimiento de acuerdo con la reivindicación 8, que comprende además lo siguiente después de que se ejecute el hipervisor:

una acción de lanzar uno o más sistemas operativos adicionales, o
que comprende además lo siguiente para cada uno de los uno o más sistemas operativos adicionales:
una acción de iniciar una instancia de máquina virtual correspondiente para interactuar con el sistema operativo adicional correspondiente.

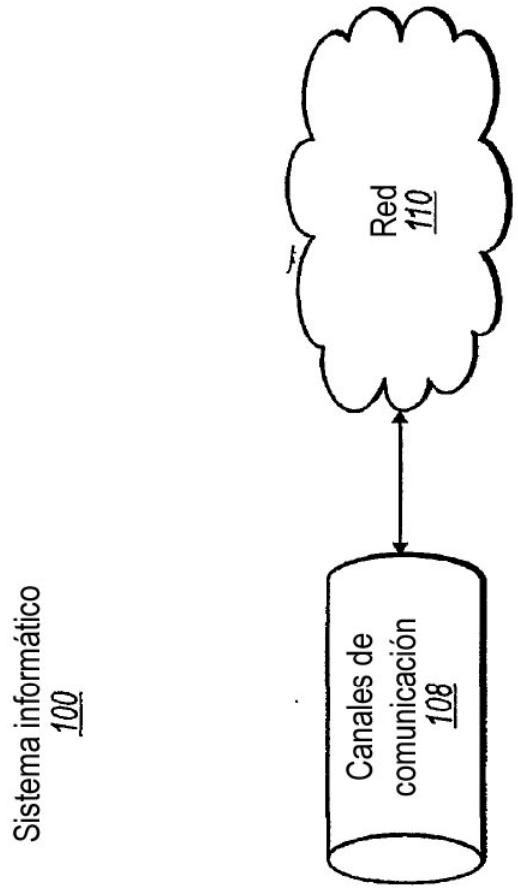
Sistema informático
100
Red
110
Canales de comunicación
108
Procesador(es)
102
Memoria
104
Volátil
No volátil

FIG. 1

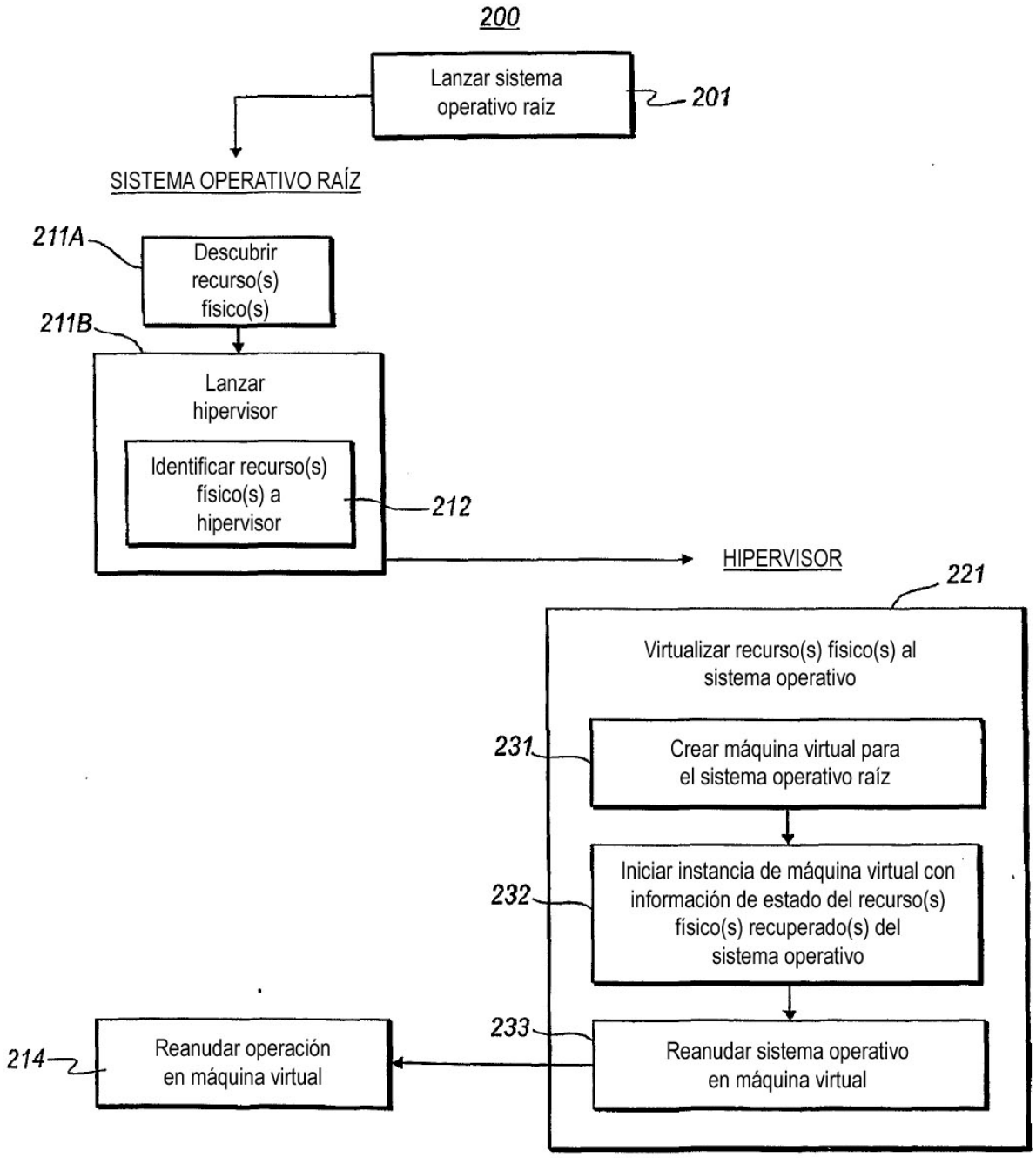
200
Lanzar sistema operativo raíz
201
SISTEMA OPERATIVO RAÍZ
211A
Descubrir recurso(s) físico(s)
211B
Lanzar hipervisor
Identificar recurso(s) físico(s) a hipervisor
212
HIPERVISOR
221
Virtualizar recurso(s) físico(s) al sistema operativo
231
Crear máquina virtual para el sistema operativo raíz
232
Iniciar instancia de máquina virtual con información de estado del recurso(s) físico(s) recuperado(s) del sistema operativo
233
Reanudar sistema operativo en máquina virtual
214
Reanudar operación en máquina virtual

FIG. 2

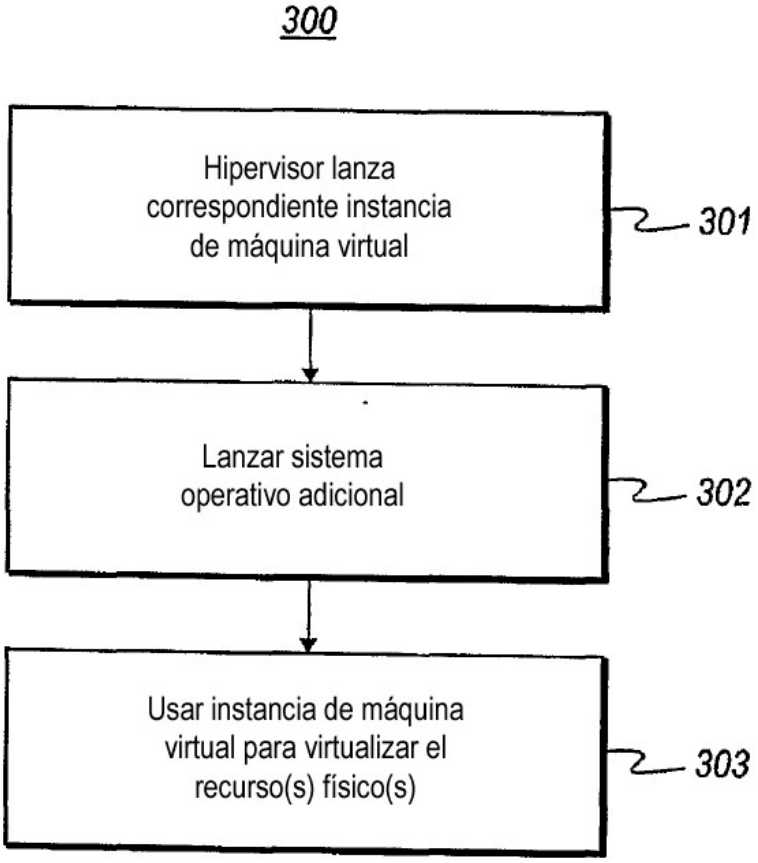
300
Hipervisor lanza correspondiente instancia de máquina virtual
301
Lanzar sistema operativo adicional
302
Usar instancia de máquina virtual para virtualizar el recurso(s) físico(s)
303

FIG. 3

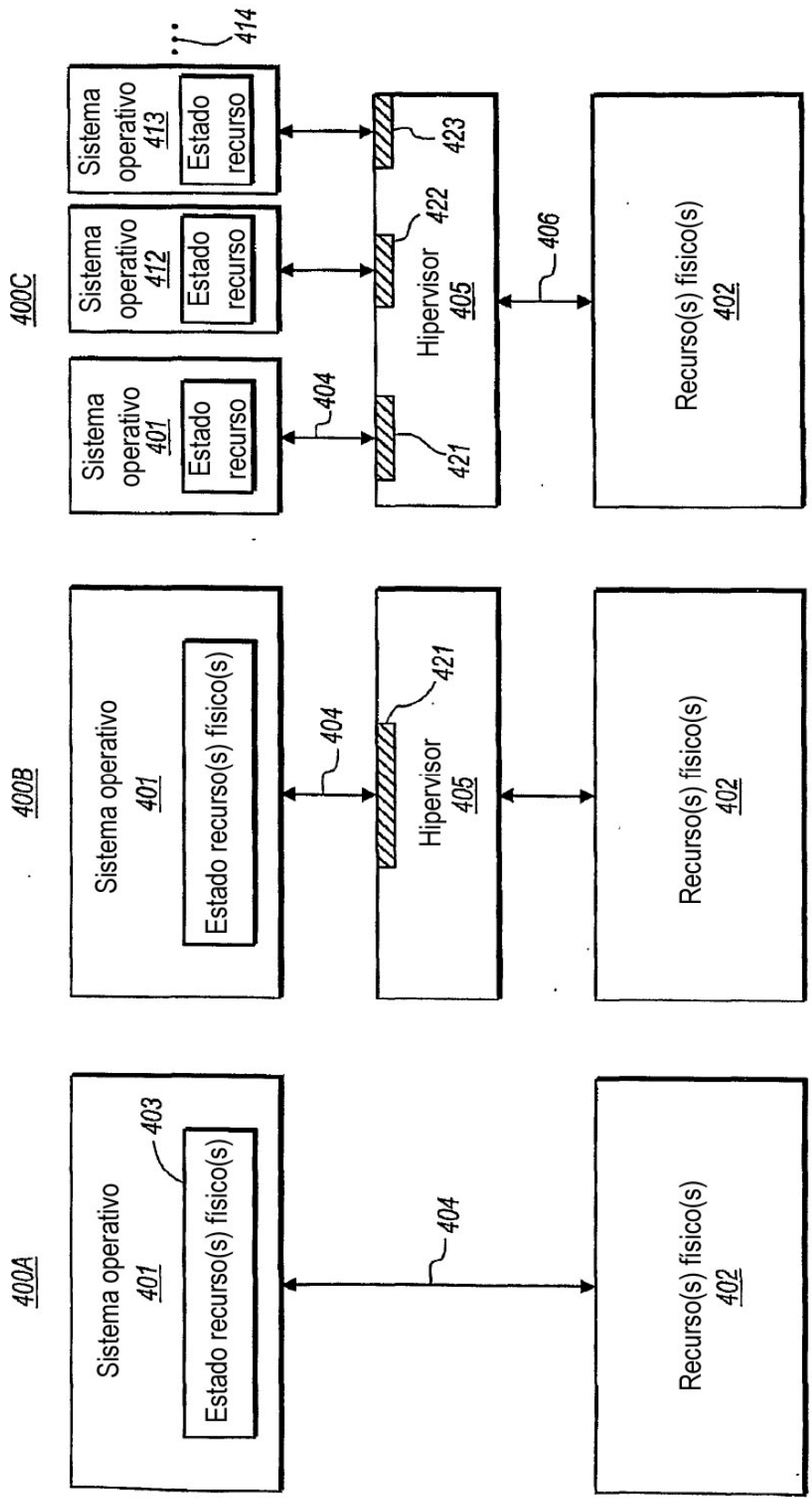

FIG. 4A

FIG. 4B

FIG. 4C

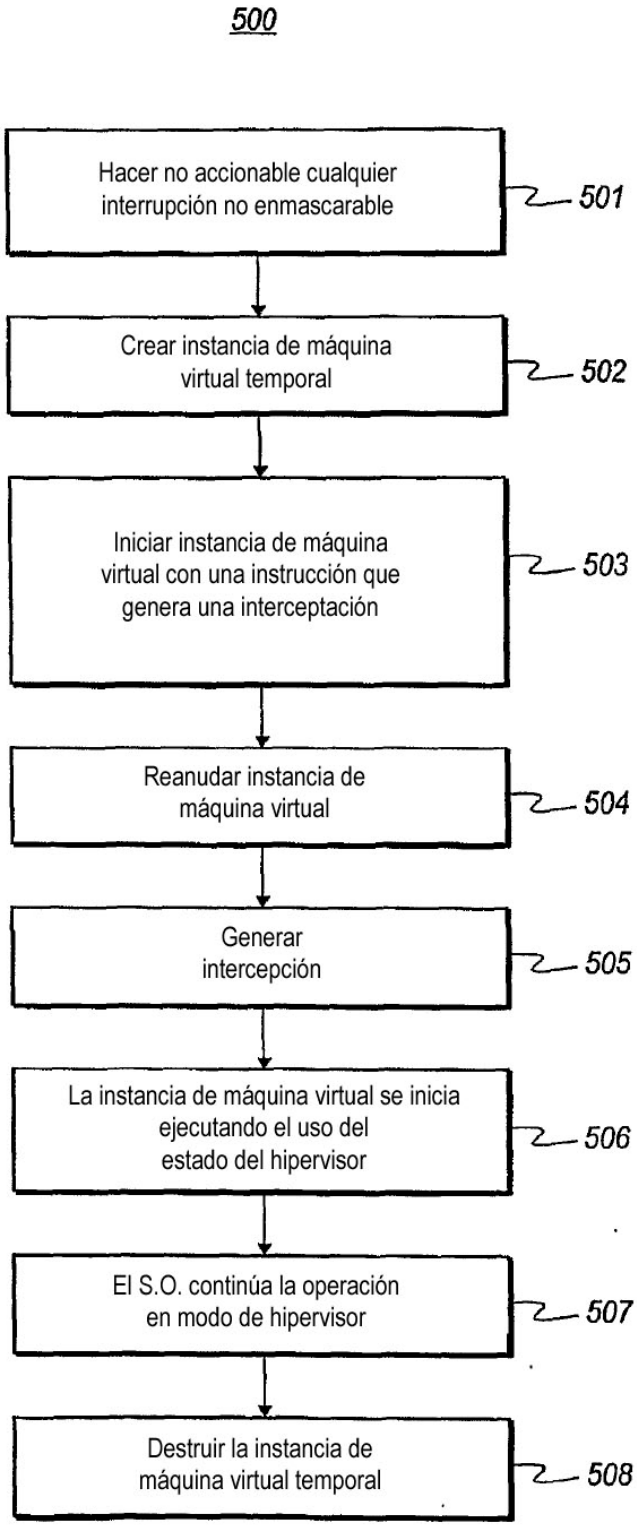
500
Hacer no accionable cualquier interrupción no enmascarable
501
Crear instancia de máquina virtual temporal
502
Iniciar instancia de máquina virtual con una instrucción que genera una interceptación
503
Reanudar instancia de máquina virtual
504
Generar intercepción
505
La instancia de máquina virtual se inicia ejecutando el uso del estado del hipervisor
506
El S.O. continúa la operación en modo de hipervisor
507
Destruir la instancia de máquina virtual temporal
508

FIG. 5