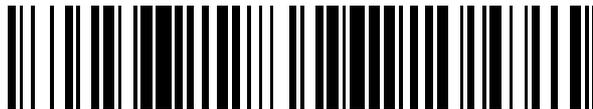


19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 766 448**

51 Int. Cl.:

**G06F 16/30** (2009.01)

**G06F 16/17** (2009.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **10.07.2006 PCT/US2006/026926**

87 Fecha y número de publicación internacional: **25.01.2007 WO07011585**

96 Fecha de presentación y número de la solicitud europea: **10.07.2006 E 06786914 (9)**

97 Fecha y número de publicación de la concesión europea: **06.11.2019 EP 1902394**

54 Título: **Movimiento de datos desde un archivo en volumen de almacenamiento a una ubicación alternativa para liberar espacio**

30 Prioridad:

**14.07.2005 US 181063**

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

**12.06.2020**

73 Titular/es:

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

72 Inventor/es:

**ROBERT, CHRISTOPHE FRANCK;  
SURIYANARAYANAN, GUHAN;  
HAVEWALA, SAROSH CYRUS y  
MOHAMED, AHMED H.**

74 Agente/Representante:

**CARPINTERO LÓPEZ, Mario**

ES 2 766 448 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

## DESCRIPCIÓN

Movimiento de datos desde un archivo en volumen de almacenamiento a una ubicación alternativa para liberar espacio

### Campo técnico

5 La presente invención se refiere a una arquitectura y a procedimientos que permiten que datos de un archivo de ordenador almacenado en un volumen de almacenamiento sean movidos o creen una imagen fantasma en una ubicación alternativa para el espacio libre en el volumen de almacenamiento. Más particularmente, la presente invención se refiere a una arquitectura y a procedimientos de este tipo por los cuales el resto del archivo de imagen fantasma permanece en el volumen de almacenamiento y los datos fantasmas en la ubicación alternativa pueden recuperarse, si es necesario, y volver a colocarse en el archivo de imagen fantasma para dar como resultado una eliminación archivo que ha dejado de ser una imagen fantasma.

### Antecedentes de la invención

15 En un dispositivo informático tal como un ordenador personal, un servidor de ordenador, o similar, y como se sabe, los datos normalmente se almacenan persistentemente en el dispositivo informático en la forma de archivos de ordenador locales residentes en uno o más volúmenes de almacenamiento locales del dispositivo informático. Cada volumen de almacenamiento de este tipo puede residir en un disco duro del dispositivo informático o similar, y dicho volumen de almacenamiento puede estar organizado, ser accedido y ser controlado de otro modo por un sistema de archivos que se ejecuta en el dispositivo informático, como también se conoce.

20 A veces, puede darse el caso de que algunos, muchos, y tal vez incluso la mayoría de los archivos de ordenador en el volumen de almacenamiento no sean de interés y se puede considerar que se han 'enfriado'. Es decir, no se ha accedido a dichos archivos fríos durante un período prolongado de tiempo y/o probablemente no se accederá a ellos durante un período prolongado de tiempo, por ejemplo, y por lo tanto tienen poco o ningún valor real en el volumen.

25 Por supuesto, este tipo de archivos fríos podrían simplemente ser eliminados del volumen de almacenamiento, sobre todo si hay una necesidad de espacio en dicho volumen. Sin embargo, debe apreciarse que la mayoría de los usuarios detestan eliminar archivos simplemente para crear espacio. Además, se considera una mala práctica eliminar un archivo frío simplemente por la falta de uso percibida. En cualquier caso, puede y probablemente sea el caso que un archivo frío, aunque no se considere necesario y no se prevea como sea necesario, puede llegar a ser necesario en algún momento futuro.

30 En tal situación, entonces, sería útil poder crear espacio en el volumen moviendo datos de tales archivos fríos a una ubicación alternativa, mientras todavía permite que dichos archivos fríos mantengan una presencia en el volumen. Es decir, sería útil poder 'formar una imagen fantasma' de un archivo frío moviendo datos de dichos archivos fríos a la ubicación alternativa o copiar todo el archivo a la ubicación alternativa, mientras se deja el archivo frío en una forma de 'imagen fantasma' más pequeña en el volumen. Por lo tanto, si de hecho se necesita el archivo de imagen fantasma en el dispositivo informático, los datos para dicho archivo de imagen fantasma se pueden recuperar de la ubicación alternativa, el archivo de imagen fantasma se puede reconstituir en función del mismo, y luego se puede emplear dicho archivo reconstituido.

35 En otro escenario, que puede ser el caso de que los archivos de datos de una organización o similares sean replicados en múltiples lugares de ramificación de la organización, tal vez desde una ubicación central, tal como un concentrador. Por ejemplo, una empresa de diseño arquitectónico puede desear tener todos los archivos de diseño arquitectónico disponibles en cualquiera de las varias sucursales de dicha empresa.

40 En una situación de este tipo, un sistema en red puede construirse para replicar una copia de cada archivo para cada sucursal, y conservar todos los archivos en todas las sucursales actuales. En dicho sistema, por ejemplo, un servidor central centralizado almacenaría cada uno de esos archivos, y se emplearía un servicio de replicación para distribuir una copia de cada archivo en el concentrador a través de una red a un servidor de sucursal o similar en cada sucursal. Sin embargo, debe apreciarse que a medida que aumenta el número de archivos de la organización, y a medida que aumenta el tamaño de cada archivo, y a medida que aumenta el número de sucursales, también aumenta la cantidad de tráfico en la red, tal vez hasta un punto que excede el ancho de banda disponible. Además, a medida que aumenta el tamaño general de todos los archivos en el servidor central, de hecho puede ocurrir que cada servidor de sucursal no tenga suficiente espacio disponible para almacenar todos los archivos replicados desde el servidor central.

45 Sin embargo, similar al escenario anterior, puede darse el caso de que algunos, muchos, y quizás incluso la mayoría de los archivos de ordenador en el servidor de sucursal de una sucursal particular no sean de interés y puedan considerarse superfluos. Tales archivos superfluos pueden, por ejemplo, relacionarse con asuntos que no son relevantes para la sucursal particular, probablemente no se accederá a través de la sucursal particular y, por lo tanto, tienen poco o ningún valor real para ser replicados en el servidor de la sucursal para la sucursal particular. Por ejemplo, una sucursal de una firma de diseño arquitectónico en Wilkes-Barre, Pensilvania, probablemente tiene muy poca o ninguna necesidad de tener archivos de diseño arquitectónico relacionados con un proyecto manejado por

una sucursal de la firma en Boynton Beach, Florida.

5 En una situación así, entonces, y similar al escenario anterior, sería útil ser capaz de almacenar en su totalidad en un servidor de sucursal de una sucursal particular solo los archivos correspondientes a la sucursal particular, mientras que solo se almacenan parcialmente en dicho servidor de sucursal de dicha sucursal particular todos los demás archivos no relevantes disponibles en el servidor central para la organización. Por lo tanto, de una manera similar a la del escenario anterior, sería útil poder "hacer una imagen fantasma" los archivos no relevantes en el servidor de sucursal de la sucursal en particular, de modo que cada archivo no relevante permanezca en el servidor de sucursal en una forma de 'imagen fantasma' más pequeña. Por lo tanto, y de nuevo, si de hecho se necesita el archivo de imagen fantasma en el servidor de sucursal, los datos de dicho archivo de imagen fantasma se pueden recuperar del servidor central, el archivo de imagen fantasma se puede reconstituir en función del mismo, y dicho archivo reconstituido se puede emplear.

15 En consecuencia, existe la necesidad de un procedimiento y un mecanismo mediante el cual un archivo en una fuente, tal como un volumen local o un servidor de sucursal, pueda replicarse u ocultarse de manera tal que los datos del mismo se almacenen en un sumidero, tal como una ubicación alternativa o un servidor central, y el archivo en la fuente está por lo tanto en una forma reducida o de imagen fantasma que se puede reconstituir si es necesario. En particular, existe la necesidad de tal procedimiento y mecanismo por el cual se pueda formar y reconstituir dicho archivo de imagen fantasma, según sea necesario.

20 El documento US 2004/172423 A1 se refiere a la sincronización de datos compartidos entre dispositivos informáticos pares. Los objetos de datos seleccionados (archivos y carpetas) se copian en los dispositivos informáticos seleccionados. Un servicio de sincronización de datos que se ejecuta en cada dispositivo seleccionado monitoriza los objetos de datos seleccionados para detectar cambios, en algunos ejemplos interceptando llamadas al sistema de archivos del dispositivo. Cuando se detecta un cambio, el servicio de sincronización de datos envía una notificación del cambio a los otros dispositivos para que puedan actualizar sus copias del objeto de datos. Por lo tanto, las copias del objeto de datos se mantienen sincronizadas en todos los dispositivos seleccionados. Un usuario puede acceder a un objeto de datos desde cualquiera de los dispositivos seleccionados, sabiendo que recuperará la última versión del objeto de datos. Si un dispositivo no está disponible temporalmente, aún se puede acceder a la última versión desde otro dispositivo. Algunos archivos pueden ser muy grandes, tal como clips de audio o video. En lugar de incurrir en los costos de almacenar dicho archivo en cada dispositivo informático, un usuario puede elegir "hacer una imagen fantasma" del archivo en algunos dispositivos. Un dispositivo de imágenes fantasma almacena solo metadatos sobre el archivo (tal como su nombre y fecha de versión) en lugar de todo el archivo. El usuario aún puede acceder al archivo a través de la imagen fantasma: las solicitudes de acceso se envían a un dispositivo que contiene los contenidos reales, y esos contenidos se presentan al usuario como si estuvieran almacenados localmente.

35 El documento US 2004/039891 A1 se refiere a técnicas para optimizar automáticamente la utilización de la capacidad de almacenamiento entre múltiples unidades de almacenamiento en un entorno de almacenamiento en base a los costos de almacenamiento de datos asociados con las unidades de almacenamiento. En ejemplos, se determina automáticamente cuándo se necesita el movimiento de datos para optimizar la utilización del almacenamiento para un grupo de unidades de almacenamiento. Según un ejemplo, para optimizar la utilización del almacenamiento y los costos de almacenamiento, los archivos se mueven desde la unidad de almacenamiento de origen a una unidad de almacenamiento de destino que tiene un costo de almacenamiento de datos más bajo asociado que la unidad de almacenamiento de origen.

45 El documento US 2005/021566 A1 se refiere a técnicas para facilitar operaciones de copia de seguridad y restauración en un entorno de almacenamiento de archivos que comprende archivos recreados. Las operaciones de copia de seguridad y restauración en archivos recreados se realizan sin activar la recuperación mientras se mantiene la integridad de los datos.

El documento US 6 161 104 A se refiere a transferencia a alta velocidad y al uso compartido de unidades de disco y otros dispositivos de almacenamiento en un sistema de procesamiento de datos digital en red.

### **Sumario de la invención**

50 Es un objeto de la presente invención mejorar los sistemas de la técnica anterior. Este objeto se soluciona mediante la materia objeto de las reivindicaciones independientes. Realizaciones preferidas se definen mediante las reivindicaciones dependientes.

55 Las necesidades mencionadas anteriormente se satisfacen al menos en parte por la presente invención, en la que se proporciona un procedimiento en relación con un dispositivo informático que tiene un volumen de almacenamiento, un sistema de archivos que gestiona el volumen de almacenamiento, y una pluralidad de archivos almacenados en el volumen de almacenamiento por el sistema de archivos y es accedido por dicho sistema de archivos. Cada archivo se define para incluir datos y metadatos relacionados con los datos, y al menos una parte de los datos del archivo se ha eliminado del archivo y se ha almacenado en una de una pluralidad de ubicaciones alternativas de modo que los datos no ocupen sustancialmente espacio en el volumen y el archivo está en una forma

reducida y de imagen fantasma. Además, el dispositivo informático tiene para cada una de la pluralidad de ubicaciones alternativas un administrador de imágenes fantasma correspondiente, donde cada administrador de imágenes fantasma se debe emplear para comunicarse con la ubicación alternativa correspondiente al mismo.

5 Para reconstituir un archivo de imagen fantasma para su uso, un solo filtro de imágenes fantasma en el dispositivo informático localiza información fantasma en los metadatos del archivo de imagen fantasma y localiza dentro de la información fantasma una identificación de un administrador de imágenes fantasma particular del dispositivo informático, donde el administrador de imágenes fantasma identificado es responsable del archivo de imagen fantasma. A continuación, el filtro de imágenes fantasma comunica al administrador de imágenes fantasma identificado una solicitud para obtener datos del archivo de imagen fantasma de la ubicación alternativa correspondiente, y el administrador de imágenes fantasma identificado de hecho obtiene dichos datos solicitados de dicha ubicación alternativa. Al recibir los datos solicitados del archivo de imagen fantasma del administrador de imágenes fantasma identificado, el filtro de imágenes fantasma reconstituye los datos recibidos en el archivo de imagen fantasma. Por lo tanto, el filtro de imágenes fantasma único es genérico para todos los administradores de imágenes fantasma del dispositivo informático.

### 15 **Breve descripción de los dibujos**

El sumario anterior, así como la siguiente descripción detallada de las realizaciones preferidas de la presente invención, se entenderá mejor cuando se lea conjuntamente con los dibujos adjuntos. Con el fin de ilustrar la invención, se muestran en los dibujos realizaciones que se prefieren actualmente. Sin embargo, como debe entenderse, la invención no se limita a las disposiciones e instrumentos precisos mostrados. En los dibujos:

20 La figura 1 es un diagrama de bloques que representa un sistema informático de propósito general en el que pueden incorporarse aspectos de la presente invención y/o partes de la misma;  
 La figura 2 es un diagrama de bloques que muestra un archivo de imagen fantasma en una fuente de tal manera que sus datos se almacenan en un sumidero de acuerdo con realizaciones de la presente invención;  
 25 Las figuras 3A y 3B son diagramas de bloques que muestran el archivo y los datos de la figura 2, por lo que solo los datos se almacenan en el sumidero (figura 3A) y por el cual todo el archivo se almacena en el sumidero (figura 3B) de acuerdo con realizaciones de la presente invención;  
 La figura 4 es un diagrama de flujo que muestra etapas clave realizadas para crear un archivo de imagen fantasma en el sumidero de la figura 2 de acuerdo con una realización de la presente invención;  
 La figura 5 es un diagrama de flujo que muestra etapas clave realizadas para reconstituir el archivo de imagen fantasma del sumidero de la figura 2 de acuerdo con una realización de la presente invención;  
 30 La figura 6 es un diagrama de flujo que muestra con más detalle etapas clave realizadas para reconstituir el archivo de imagen fantasma del sumidero de la figura 2 de acuerdo con una realización de la presente invención;  
 La figura 7 es un diagrama de flujo que muestra etapas clave realizadas para reconstituir parcialmente el archivo de imagen fantasma del sumidero de la figura 2 de acuerdo con una realización de la presente invención;  
 35 La figura 8 es un diagrama de bloques que muestra varios estados para los datos del archivo de la figura 2 de acuerdo con realizaciones de la presente invención;  
 La figura 9 es un diagrama de flujo que muestra etapas clave realizadas para reconstituir eficientemente al menos una parte del archivo de imagen fantasma del sumidero de la figura 2 en función del estado de los segmentos de los datos de la parte tal como se establece en la figura 8 en de acuerdo con una realización de la presente invención;  
 40 La figura 10 es un diagrama de flujo que muestra etapas clave realizadas cuando se vuelven a formar imágenes fantasma de archivos en el sumidero de la figura 2 de acuerdo con una realización de la presente invención;  
 La figura 11 es un diagrama de bloques que muestra la fuente de la figura 2 asociada con una pluralidad de sumideros, por lo que la fuente tiene un único filtro de imágenes fantasma genérico y un administrador de imágenes fantasma correspondiente a cada sumidero, de acuerdo con una realización de la presente invención;  
 45 y  
 La figura 12 es un diagrama de flujo que muestra etapas clave realizadas por el filtro de imágenes fantasma de la figura 11 para crear imágenes fantasma y reconstituir un archivo en la fuente de acuerdo con una realización de la presente invención.

### 50 **Descripción detallada de la invención**

#### ENTORNO INFORMÁTICO

La figura 1 y la siguiente descripción pretenden proporcionar una breve descripción general de un entorno informático adecuado en el cual se puede implementar la presente invención y/o porciones de la misma. Aunque no es obligatorio, la invención se describe en el contexto general de instrucciones ejecutables por ordenador, tales como módulos de programa, ejecutados por un ordenador, tal como una estación de trabajo cliente o un servidor. Generalmente, los módulos de programa incluyen rutinas, programas, objetos, componentes, estructuras de datos, y similares, que realizan tareas particulares o implementan tipos particulares de datos abstractos. Además, debe apreciarse que la invención y/o partes de la misma se pueden practicar con otras configuraciones de sistemas informáticos, incluyendo dispositivos de mano, sistemas multiprocesadores, productos electrónicos de consumo basados en microprocesadores o programables, PC en red, miniordenadores, ordenadores centrales y similares. La

invención puede también ponerse en práctica en un entorno informático distribuido en el que las funciones se realizan por dispositivos de procesamiento remotos que están enlazados a través de una red de comunicaciones. En un entorno informático distribuido, los módulos de programa pueden ubicarse en dispositivos de almacenamiento de memoria local y remota.

5 Como se muestra en la figura 1, un sistema informático de propósito general, a modo de ejemplo, incluye un ordenador 120 personal convencional o similar, incluyendo una unidad 121 de procesamiento, una memoria 122 de sistema, y un bus 123 del sistema que acopla varios componentes del sistema incluyendo el sistema de memoria a la unidad 121 de procesamiento. El bus 123 del sistema puede ser cualquiera de varios tipos de estructuras de bus que incluyen un bus de memoria o controlador de memoria, un bus periférico y un bus local usando cualquiera de una variedad de arquitecturas de bus. La memoria del sistema incluye leer solo la memoria 124 de solo lectura (ROM) y la memoria 125 de acceso aleatorio (RAM). Un sistema 126 básico de entrada/salida (BIOS), que contiene las rutinas básicas que ayudan a transferir información entre los elementos dentro del ordenador 120 personal, como durante el arranque, se almacena en la ROM 124.

15 El ordenador 120 personal puede incluir además una unidad 127 de disco duro para leer y escribir en un disco duro (no mostrado), una unidad 128 de disco magnético para leer o escribir en un disco 129 magnético extraíble y una unidad 130 de disco óptico para leer o escribir en un disco 131 óptico extraíble, tal como un CD-ROM u otro medio óptico. La unidad 127 de disco duro, la unidad 128 de disco magnético y la unidad 130 de disco óptico están conectadas al bus 123 del sistema por una interfaz 132 de unidad de disco duro, una interfaz 133 de unidad de disco magnético y una interfaz 134 de unidad óptica, respectivamente. Las unidades y sus medios de legibles por ordenador asociados proporcionan almacenamiento no volátil de instrucciones legibles por ordenador, estructuras de datos, módulos de programa y otros datos para el ordenador 120 personal.

20 Aunque el entorno ejemplar descrito en el presente documento emplea un disco duro, un disco magnético extraíble 129, y un disco óptico extraíble 131, se debe apreciar que otros tipos de medios legibles por ordenador que pueden almacenar datos que sean accesibles por un ordenador pueden también utilizarse en el entorno operativo ejemplar. Tales otros tipos de medios incluyen un casete magnético, una tarjeta de memoria flash, un disco de video digital, un cartucho de Bernoulli, una memoria de acceso aleatorio (RAM), una memoria de solo lectura (ROM) y similares.

25 Un número de módulos de programa pueden almacenarse en el disco duro, disco 129 magnético, disco 131 óptico, ROM 124 o RAM 125, incluyendo un sistema 135 operativo, uno o más programas 136 de aplicación, otros módulos 137 de programa y datos 138 de programa. Un usuario puede introducir comandos e información en el ordenador 700 personal a través de dispositivos de entrada, tales como un teclado 140 y un dispositivo 142 apuntador. Otros dispositivos de entrada (no mostrados) pueden incluir un micrófono, una palanca de mando, una almohadilla para juegos, una antena parabólica, un escáner o similar. Estos y otros dispositivos de entrada a menudo están conectados a la unidad 121 de procesamiento a través de una interfaz 146 de puerto serie que está acoplada al bus del sistema, pero pueden estar conectados por otras interfaces, tales como un puerto paralelo, puerto de juego o bus serie universal (USB). Un monitor 147 u otro tipo de dispositivo de visualización también está conectado al bus 123 del sistema a través de una interfaz, tal como un adaptador 148 de video. Además del monitor 147, un ordenador personal típicamente incluye otros dispositivos de salida periféricos (no mostrados), tales como altavoces e impresoras. El sistema ejemplar de la figura 1 también incluye un adaptador 155 de hospedador, un bus 156 de interfaz de sistema de ordenador pequeño (SCSI) y un dispositivo 162 de almacenamiento externo conectado al bus 40 156 de SCSI.

45 El ordenador 120 personal puede operar en un entorno de red utilizando conexiones lógicas a uno o más ordenadores remotos, tales como un ordenador 149 remoto. El ordenador 149 remoto puede ser otro ordenador personal, un servidor, un enrutador, un PC en red, un dispositivo par u otro nodo de red común, y típicamente incluye muchos o todos los elementos descritos anteriormente con relación al ordenador 120 personal, aunque solo un dispositivo 150 de almacenamiento de memoria se ha ilustrado en la figura 1. Las conexiones lógicas representadas en la figura 1 incluyen una red 151 de área local (LAN) y una red 152 de área extensa (WAN). Dichos entornos de red son comunes en oficinas, redes informáticas de toda la empresa, intranets e Internet.

50 Cuando se utiliza en un entorno de red LAN, el ordenador 120 personal está conectado a la LAN 151 a través de una interfaz de red o adaptador 153. Cuando se usa en un entorno de red WAN, el ordenador 120 personal incluye típicamente un módem 154 u otro medio para establecer comunicaciones a través de la red 152 de área amplia, tal como Internet. El módem 154, que puede ser interno o externo, se conecta al bus 123 del sistema a través de la interfaz 146 de puerto serie. En un entorno de red, los módulos de programa representados con respecto al ordenador 120 personal, o partes de los mismos, pueden almacenarse en el dispositivo de almacenamiento de memoria remota. Se apreciará que las conexiones de red mostradas son ejemplares y se pueden usar otros medios para establecer un enlace de comunicaciones entre los ordenadores.

#### FORMACIÓN DE IMAGEN FANTASMA Y RECONSTITUCIÓN DE UN ARCHIVO

En la presente invención, y volviendo ahora a la figura 2, un dispositivo 10 informático, tal como un ordenador personal o un servidor de ordenador o similar tiene un volumen 12 de almacenamiento, tal como un disco duro o una unidad de RAM persistente o similar, el volumen 12 ha almacenado en el mismo una cantidad de archivos 14 de

ordenador, y los archivos 14 en el volumen 12 están organizados, accedidos y controlados por un sistema 16 de archivos que se ejecuta en el dispositivo 10 informático. Como puede apreciarse, el dispositivo 10 informático, el volumen 12, los archivos 14 y el sistema 16 de archivos pueden ser cualquier tipo de dispositivo informático, volumen, archivo y sistema de archivos sin apartarse del alcance de la presente invención.

5 En una realización de la presente invención, el dispositivo 10 informático hay un ordenador personal o similar, y al menos algunos de los archivos 14 en el volumen 12 de los mismos se han determinado que son fríos en que tales archivos 14, por ejemplo, no han sido accedidos durante un período prolongado de tiempo y/o probablemente no se accederá durante un período prolongado de tiempo, y por lo tanto tienen poco o ningún valor real en el volumen 12. Sin embargo, debe tenerse en cuenta que la frialdad de un archivo 14 puede definirse de cualquier manera apropiada sin apartarse del alcance de la presente invención.

10 En cualquier caso, al determinarse que es frío, un archivo 14 no se elimina del volumen 12, pero en su lugar se reduce en tamaño en el volumen 12 mediante la eliminación de al menos algunos datos 20 de los mismos en una ubicación 18 alternativa. Dicha ubicación 18 alternativa puede ser local para el dispositivo informático o puede ser remota desde el mismo. En general, dicha ubicación 18 alternativa está acoplada apropiadamente al dispositivo 10 informático y puede ser cualquier ubicación de almacenamiento apropiada sin apartarse del alcance de la presente invención. Por ejemplo, la ubicación 18 alternativa puede ser otro volumen 12 en el dispositivo 10 de almacenamiento, otro volumen 12 en otro dispositivo 10 informático, un almacén de archivos en un servidor, un dispositivo de almacenamiento a largo plazo en un servidor remoto, o similares.

20 Con la ubicación 18 alternativa, entonces, el espacio en el volumen 12 se puede liberar moviendo los datos 20 antes mencionados de dichos archivos 14 en frío a dicha ubicación 18 alternativa. Significativamente, aunque los datos 20 de dicho archivo 14 frío se han movido, dicho archivo 14 frío permanece como una presencia o 'imagen fantasma' en el volumen 12, aunque en forma reducida o 'fantasma'. Por lo tanto, si de hecho el archivo 14 de imagen fantasma es necesario en el dispositivo 10 informático, dicho archivo 14 de imagen fantasma se reconstituye recuperando sus datos 20 de la ubicación 18 alternativa y volviendo a asociar dichos datos 20 recuperados con dicho archivo 14 de imagen fantasma para formar el archivo 14 reconstituido. Como puede apreciarse, entonces, una vez reconstituido, el archivo 14 puede de hecho ser empleado.

25 En otra realización de la presente invención, el dispositivo 10 informático es un servidor de sucursal o similar, y al menos algunos de los archivos 14 en el volumen 12 del mismo se ha determinado que es irrelevante el hecho de que dichos archivos 14, por ejemplo, no son relevante para una sucursal asociada con dicho servidor de sucursal. Por supuesto, la irrelevancia con respecto a un archivo 14 puede definirse de cualquier manera apropiada sin apartarse del alcance de la presente invención.

30 En cualquier caso, y como antes, un archivo 14 irrelevante se mantiene en el volumen 12 del servidor 10 de sucursal en un formato reducido y sin al menos algunos datos 20 del mismo. Aquí, dichos datos 20 se almacenan en una ubicación 18 alternativa, tal como un servidor concentrador centralizado mantenido por una organización de la cual la oficina sucursal es una sucursal. De nuevo, dicho servidor 18 concentrador está acoplado apropiadamente al servidor 10 de sucursal.

35 Con el servidor 18 de concentrador, entonces, el espacio usado en el volumen 12 del servidor 10 de sucursal se puede minimizar, y el ancho de banda necesario para rellenar dicho espacio también puede minimizarse, mediante el almacenamiento de tales datos 20 mencionados anteriormente de tales archivos 14 irrelevantes en dicho servidor 40 18 concentrador. Significativamente, y similar a antes, aunque los datos 20 de un archivo 14 irrelevante no están presentes en el servidor 10 de sucursal, dicho archivo 14 irrelevante permanece como una presencia o 'imagen fantasma' en el volumen 12 del servidor de sucursal, aunque en una forma reducida o 'fantasma'. Por lo tanto, si de hecho el archivo 14 de imagen fantasma es necesario en el servidor 10 de sucursal, dicho archivo 14 de imagen fantasma se reconstituye recuperando sus datos 20 del servidor 18 de concentrador y volviendo a asociar dichos datos 20 recuperados con dicho archivo 14 de imagen fantasma para formar el archivo 14 reconstituido. Nuevamente, una vez reconstituido, el archivo 14, de hecho, puede emplearse.

40 Para generalizar a continuación, ya sea en una disposición que implica un dispositivo 10 informático, tal como un ordenador personal y una ubicación 18 alternativa, un servidor 10 de sucursal y un servidor 18 de concentrador, o de otra manera, 'imagen fantasma' implica un archivo 14 en un volumen 12 en una fuente 10, como se ve en la figura 2, donde se ha determinado que el archivo 14 solo debe estar presente en el volumen 12 en un formato reducido que no incluya una parte de los datos 20 asociados con dicho archivo 14. En consecuencia, y como se ve en la figura 2, dichos datos 20 se almacenan en un sumidero 18 adecuadamente acoplado a la fuente 10.

45 Con tal sumidero 18, entonces, el espacio usado en el volumen 12 de la fuente 10 se puede minimizar en que un archivo 14 de imagen fantasma en el volumen 12 está presente en una forma 'código auxiliar' o reducida. Por lo tanto, si de hecho el archivo 14 de imagen fantasma es necesario en la fuente 10, dicho archivo 14 de imagen fantasma se reconstituye recuperando sus datos 20 del sumidero 18 y volviendo a asociar dichos datos 20 recuperados con dicho archivo 14 de imagen fantasma para formar el archivo 14 reconstituido. Una vez más, una vez reconstituido, el archivo 14, de hecho, puede emplearse en la fuente 10 según sea necesario.

En una realización de la presente invención, el código auxiliar de un archivo 14 de imagen fantasma como presente en el volumen de una fuente 10 se almacena en la misma ubicación en el volumen 12 que el archivo 14 original. Por lo tanto, ya sea que el sistema 16 de archivos organice el volumen 12 de acuerdo con un formato de directorio o de otra manera, el sistema 16 de archivos que busca en una ubicación particular un archivo 14 que ha sido imagen fantasma debe encontrar el código auxiliar para el archivo 14, es decir, el archivo 14 de imagen fantasma, en tal ubicación. Significativamente, el archivo 14 de imagen fantasma como residente en el volumen 12 de la fuente 10 contiene todos los metadatos 22 del archivo 14 original, no fantasma, y también contiene información 24 de imagen fantasma que puede emplearse para recuperar los datos 20 del archivo 14 desde el sumidero 18, entre otras cosas. Como puede apreciarse, dicha información 24 de imagen fantasma puede agregarse a los metadatos 22 para el archivo 14 o puede almacenarse en otra ubicación del archivo 14.

En consecuencia, un usuario o similar puede explorar todo el volumen 12 de la fuente 10 a través del sistema 16 de archivos incluso si algunos de los archivos 14 en el volumen 12 son imágenes fantasma y algunos de los archivos 14 en el volumen 12 no son imágenes fantasma. En particular, los metadatos 22 asociados con un archivo 14 de imagen fantasma pueden emplearse durante dicha exploración para, entre otras cosas, identificar el archivo 14 de imagen fantasma, proporcionar información de tamaño, información de fecha y similares. Cuando el usuario intenta acceder a un archivo 14 de imagen fantasma, los datos 20 del mismo se recuperan en base a la información 24 de imagen fantasma presente en el archivo 14 de imagen fantasma, el archivo 14 de imagen fantasma se reconstituye en un archivo 14 que no es imagen fantasma y el archivo 14 que no es imagen fantasma puede, de hecho, ser accedido por el usuario.

Volviendo ahora a la figura 3A, se ve que un archivo 14 que no es imagen fantasma como puede ser residente en un volumen 12 en la fuente 10 incluye metadatos 22 que pueden estar organizados en un encabezado o similares, donde tales metadatos 22 incluyen información relacionada con los datos 20 del archivo 14 que no es imagen fantasma, tal como un tamaño de archivo lógico, un tamaño en el volumen 12, un tiempo de creación, un tiempo de modificación, un tiempo de acceso, permisos de acceso y varios atributos de archivo. Además, y como debería ser evidente, dicho archivo 14 que no es fantasma como residente en el volumen 12 en la fuente 10 incluye los datos 20 reales del archivo 14. Debe tenerse en cuenta aquí que dichos datos 20 reales pueden organizarse como datos primarios y datos alternativos, donde los datos primarios son los datos 20 del archivo 14 que pueden ser empleados por una aplicación 30 o similar (figura 2), mientras que los datos alternativos son los datos del archivo 14 que se han creado para otros usos. Como solo un ejemplo, dichos datos alternativos pueden incluir una representación gráfica como una 'miniatura' que puede emplearse al mostrar una representación del archivo 14, quizás por la aplicación 30 o el sistema 16 de archivos en la fuente 10.

En cualquier caso, es probable el caso de que los datos primarios sea el mayor de los datos 20 por tamaño. En consecuencia, cuando el archivo 14 que no es imagen fantasma es, de hecho, imagen fantasma, de hecho, puede ser el caso de que solo los datos primarios de los datos 20 se eliminen del resto del archivo 14 que no es imagen fantasma para formar el archivo 14 de imagen fantasma, y que solo dichos datos primarios se almacenan en el sumidero 18, tal vez junto con una identificación del archivo 14 de imagen fantasma, como se muestra en la figura 3A. Por supuesto, otras partes del archivo 14 también pueden eliminarse durante la creación de imágenes fantasma de un archivo 14 sin apartarse del alcance de la presente invención.

Sin embargo, como se puede apreciar, cuando el archivo 14 que no es imagen fantasma es de hecho imagen fantasma en el contexto de un servidor 10 de sucursal y un servidor 18 concentrador, donde el servidor 18 concentrador replica los archivos 14 en el servidor 10 de sucursal, de hecho, puede ser el caso de que, si bien los datos primarios de los datos 20 se eliminan del resto del archivo 14 replicado en la fuente 18 para formar el archivo 14 de imagen fantasma, la totalidad de dicho archivo 14 se almacena en el sumidero 18, tal vez junto con una identificación del archivo 14 de imagen fantasma, como se muestra en la figura 3B.

Para convertir un archivo 14 que no es imagen fantasma en un archivo 14 de imagen fantasma (es decir, en un archivo 14 de imagen fantasma), y haciendo referencia ahora a las figuras 3 y 4, el archivo 14 se identifica primero (etapa 401), y luego se identifican los datos 20 que se eliminarán del archivo 14 (etapa 403). Nuevamente, tales datos 20 son típicamente los datos primarios de los datos 20, pero podrían ser otros datos de dichos datos 20. En cualquier caso, los datos 20 identificados se trasladan desde la fuente 10 al sumidero 18 por medio de un mecanismo y conducto de transporte apropiado (etapa 405), y dichos datos 20 movidos se almacenan en el sumidero 18 en un formato apropiado (etapa 407). Como puede apreciarse, el almacenamiento de los datos 20 en el sumidero 18 puede realizarse en cualquier formato y manera apropiados sin apartarse del alcance de la presente invención. Por ejemplo, los datos 20 pueden comprimirse y/o cifrarse si así se desea, y dichos datos 20 deben almacenarse de manera que dichos datos 20 puedan recuperarse con relativa facilidad cuando se solicite la reconstitución del archivo 14. Por ejemplo, dichos datos 20 pueden almacenarse de acuerdo con una ID empleada para la recuperación, como se expondrá con más detalle a continuación.

Una vez los datos 20 se han movido como en la etapa 405, el espacio ocupado por tales datos 20 en el volumen 12 de la fuente 10 no necesita permanecer asignado al archivo 14, como puede apreciarse. En consecuencia, dicho espacio asignado se puede marcar como libre (etapa 409), con el resultado de que el archivo 14 ahora imagen fantasma se marca como escaso. Significativamente, aunque dicho espacio asignado ha sido liberado del uso por el archivo 14 de imagen fantasma, el tamaño lógico del archivo 14 ahora fantasma no debe cambiarse en los

metadatos 22 para dicho archivo 14 o en otro lugar, aunque el tamaño en el volumen 12 para, de hecho, el archivo 14 puede cambiarse para reflejar dicho espacio liberado.

Además, en una realización de la presente invención, los metadatos 22 para el archivo 14 ahora imagen fantasma se modifica para demostrar que existe el archivo 14 que es ahora imagen fantasma (etapa 411). Para hacerlo, por ejemplo, se puede establecer un atributo de 'imagen fantasma' en los metadatos 22. Como puede apreciarse, dicho atributo de imagen fantasma establecido puede emplearse principalmente como una señal para una aplicación 30 de consulta o similar de que el archivo 14 es de hecho imagen fantasma. Con tal señal, entonces, la aplicación 30 de consulta puede apreciar no solo que el archivo 14 es de hecho imagen fantasma sino que acceder a dicho archivo 14 implica un costo en el sentido de que dicho archivo 14 debe reconstituirse primero en una forma que no es imagen fantasma. Como debe entenderse, dicho costo puede estar en términos del ancho de banda necesario para acceder a los datos 20 para el archivo 14 desde el sumidero 18, el espacio requerido para los datos 20 en el volumen 12 y/o la latencia para acceder a los datos 20 en el sumidero 18 y reconstituir el archivo 14 basado en el mismo, entre otras cosas.

Además, en una realización de la presente invención, los metadatos 22 para el archivo 14 que es ahora imagen fantasma se modifica para incluir la mencionada información 24 de imagen fantasma (etapa 413). Una vez más, dicha información 24 de imagen fantasma incluye información que puede emplearse para recuperar los datos 20 para el archivo 14 del sumidero 18, entre otras cosas. Por ejemplo, dicha información 24 de imagen fantasma puede incluir la ID almacenada con y que identifica los datos 20 en el sumidero 18, así como una cantidad de almacenamiento que puede emplearse para almacenar otra información relevante para el archivo 14 de imagen fantasma, incluyendo información sobre cómo localizar el sumidero 18, cómo reconstituir el archivo 14 y/o similares. Este almacenamiento en la información 24 de imagen fantasma puede ser una cantidad variable o fija, y en este último caso podría estar limitado a 16 kilobytes más o menos. Como puede apreciarse, dicha información 24 de imagen fantasma se emplea principalmente por cualquier construcción empleada para reconstituir el archivo 14, y no es utilizada típicamente por la aplicación 30 de consulta mencionada anteriormente o similar, aunque dicho uso puede ocurrir sin apartarse del alcance de la presente invención.

Como se debe apreciara ahora, un archivo 14 de imagen fantasma como imagen fantasma puede residir sin perturbaciones en el volumen 12 de la fuente 10 durante un corto periodo de tiempo, durante un largo periodo de tiempo, para un periodo de tiempo muy largo, o para siempre, al menos hasta que el volumen 12 ya no permanece en operación. Sin embargo, en cualquier momento, y volviendo ahora a la figura 5, debe apreciarse que una solicitud para acceder a los datos 20 almacenados del archivo 14 de imagen fantasma puede recibirse desde una aplicación 30, un usuario en dicha aplicación 30, etc. (etapa 501). Por supuesto, antes de dicha solicitud, también es posible que también se reciban una o más solicitudes para acceder a los metadatos 22 de dicho archivo 14 de imagen fantasma. Por ejemplo, la aplicación 30 mencionada anteriormente puede acceder al atributo de imagen fantasma establecido en los metadatos 22 de dicho archivo 14 de imagen fantasma para determinar que el archivo 14 es realmente imagen fantasma, o el sistema 16 de archivos que controla el volumen 12 puede acceder a los metadatos 22 en el curso de compilar un listado de directorio o similar. En el último caso, y en particular, el sistema 16 de archivos, la aplicación 30 u otra entidad también pueden acceder a los datos alternativos de los datos 22 para obtener el gráfico antes mencionado que puede emplearse al mostrar una representación del archivo 14.

En cualquier caso, en respuesta a una solicitud de acceso a los datos 20 almacenados del archivo 14 de imagen fantasma como en la etapa 501, el archivo 14 de imagen fantasma, de hecho, se reconstituye de la siguiente manera. Preliminarmente, el archivo 14 se encuentra en el volumen 12 (etapa 503), y luego se identifica la información 24 de imagen fantasma en los metadatos 22 del archivo 14 (etapa 505). Como puede apreciarse, los datos 20 del archivo 14 almacenados en el sumidero 18 se ubican en base a dicha información 24 de imagen fantasma identificada (etapa 507), y dichos datos 20 ubicados pueden entonces moverse desde el sumidero 18 a la fuente 10 mediante un mecanismo y conducto de transporte apropiado (etapa 511).

Por supuesto, para almacenar tales datos 20 con el archivo 14 en la fuente 10, el espacio para ser ocupado por tales datos 20 en el volumen 12 de la fuente 10 deben ser asignados (etapa 509), con el resultado de que el archivo 14 ya no es imagen fantasma ya no está marcado como escaso. Como se puede apreciar ahora, después de que los datos 20 se hayan movido al archivo 14 para reconstituirlos, los metadatos 22 para el archivo 14 que ya no es imagen fantasma se modifican para mostrar que dicho archivo 14 no es imagen fantasma (etapa 513), tal como, por ejemplo, restableciendo el atributo de imagen fantasma en dichos metadatos 22. Además, los metadatos 22 para el archivo 14 que ya no es imagen fantasma se modifican para eliminar la información 24 de imagen fantasma (etapa 515). En consecuencia, el archivo 14 ahora no es en una forma de imagen fantasma.

Como se debe apreciar ahora, hay dos escenarios principales que dan lugar a la creación de un archivo 14 de imagen fantasma. En el primer escenario, el archivo 14 se crea directamente como un archivo 14 de imagen fantasma en el volumen 12. Esta creación de imagen fantasma puede ser típica en una arquitectura de replicación como la establecida anteriormente con un servidor 10 de sucursal y un servidor 18 concentrador. En dicha arquitectura de replicación, llenar cada uno de varios servidores 10 de sucursal con copias que no son imagen fantasma de todos los archivos 14 desde un servidor 18 concentrador requeriría enormes cantidades de ancho de banda. En consecuencia, y en su lugar, un motor de replicación puede decidir simplemente crear copias de imagen fantasma de los archivos 14 en los servidores 10 de sucursal. Los datos 20 asociados con cualquier archivo 14 de

imagen fantasma particular en cualquier servidor 10 de sucursal particular, entonces, pueden recuperarse del servidor 18 concentrador a pedido cuando sea necesario reconstituir dicho archivo 14 de imagen fantasma particular en un archivo 14 que no es imagen fantasma en tal caso servidor 10 de sucursal.

5 En el segundo escenario, el archivo 14 se crea como un archivo 14 que no es imagen fantasma en el volumen 12 y en algún momento posterior se convierte en un archivo 14 de imagen fantasma en tal volumen 12. Dicha creación que no es imagen fantasma puede ser típica en una arquitectura que ahorra espacio, como la establecida anteriormente con un ordenador personal o similar y una ubicación 18 alternativa. En una arquitectura que ahorra espacio, el espacio en un volumen 12 del ordenador personal o similar se recupera moviendo los datos 20 asociados con los archivos 14 que, por ejemplo, rara vez se usan, a la ubicación 18 alternativa, que puede ser otro medio o a 10 otro sistema. Al igual que en el primer escenario, los datos 20 asociados con cualquier archivo 14 de imagen fantasma en particular en el ordenador personal o similar se pueden recuperar de la ubicación 18 alternativa a pedido cuando sea necesario para reconstituir dicho archivo 14 de imagen fantasma en un archivo 14 que no es imagen fantasma.

15 En una realización de la presente invención, las operaciones de la figura 5 que se realizan en la fuente 10 en nombre de un usuario o aplicación 30 y con respecto a un archivo 14 de imagen fantasma, de hecho, se realiza con la ayuda de un filtro 26 de imágenes fantasma, como se puede ver en la figura 2. Dichas operaciones pueden realizarse de forma transparente o con aviso al usuario o aplicación 30, aunque es probable que la transparencia sea preferible a dicho usuario o aplicación 30. En particular, cuando un usuario o aplicación 30 intenta acceder a los 20 datos 20 eliminados de un archivo 14 de imagen fantasma, el sistema 16 de archivos al recibir una solicitud de dicho acceso descubrirá que el archivo 14 de imagen fantasma no contiene los datos 20 eliminados y, por lo tanto, devuelve un error que interceptará el filtro 26 de imágenes fantasma. Después de tal interceptación, entonces, el filtro 26 de imágenes fantasma emplea el sistema 16 de archivos para obtener la información 24 de imagen fantasma de los metadatos 22 para el archivo 14 de imagen fantasma, y en base a la información 24 de imagen fantasma obtenida, dicho filtro 26 de imágenes fantasma activa la reconstitución de dicho archivo 14 de imagen fantasma, tal 25 como se expondrá con más detalle a continuación, de modo que la solicitud de acceso pueda cumplirse en última instancia.

30 Como se puede apreciar, el filtro 26 de imágenes fantasma puede ser un constructo de nivel inferior sin mucha funcionalidad y sin acceso a recursos de red tales como el sumidero 18. En tal situación, y como se muestra en la figura 2, el filtro 26 de imágenes fantasma puede interactuar con una construcción de nivel superior tal como un administrador 28 de imagen fantasma, donde tal administrador 28 de imagen fantasma incluye funcionalidad adicional de imagen fantasma y acceso directo a recursos de red tales como el sumidero 18. Además, en tal situación, debe apreciarse que cuando el filtro 26 de imágenes fantasma activa tal reconstitución de dicho archivo 14 de imagen fantasma, el administrador 28 de imagen fantasma realiza la mayor parte de dicha funcionalidad reconstituyente para la fuente 10 como se muestra en conexión con la figura 5, como se expondrá con más detalle a 35 continuación. Dicho administrador 28 de imágenes fantasmas también puede realizar la mayor parte de la funcionalidad de imágenes fantasmas para la fuente 10 como se muestra en relación con la figura 4.

#### APLICACIÓN 30 QUE SOLICITA DATOS 20 DEL ARCHIVO 14 DE IMAGEN FANTASMA

40 Aunque la reconstitución de un archivo 14 de imagen fantasma ya se ha detallado anteriormente en relación con la figura 4, es instructivo volver a visitar tal proceso desde el punto de vista de una aplicación 30 o similar que desee leer una parte de los datos 20 de tal archivo 14 de imagen fantasma. Como se puede apreciar, entonces, dicha aplicación 30 que desea leer dichos datos 20 de dicho archivo 14 típicamente logra dicha función emitiendo un comando abierto al sistema 16 de archivos con respecto al archivo 14, y luego emitiendo un comando de lectura para el sistema 16 de archivos con respecto a dicho archivo 14 abierto. En una realización de la presente invención, dicha aplicación 30 continúa emitiendo dichos comandos de apertura y lectura, aunque dichos comandos se 45 interpretan de manera algo diferente en función del archivo 14 en cuestión que está siendo una imagen fantasma. Dichas diferencias son transparentes y poco preocupantes para la aplicación 30, aunque la aplicación 30 puede experimentar cierta latencia en el curso de la reconstitución del archivo 14 de imagen fantasma, donde tal latencia probablemente se deba en gran parte a mover los datos 20 del sumidero 18 a la fuente 10.

50 En cualquier caso, y volviendo ahora a la figura 6, la aplicación 30 inicia el proceso mediante la emisión de la orden de apertura con respecto a un archivo 14 de imagen fantasma particular, en el volumen 12, y tal comando de apertura en última instancia, es recibido por el sistema 16 de archivos (etapa 601). Notablemente, no se espera que la aplicación 30 al emitir dicho comando de apertura tenga conocimiento de que el archivo 14 particular es de hecho una imagen fantasma, aunque la aplicación 30 podría de hecho tomar esa determinación ordenando a que el sistema 16 de archivos informe si el atributo de imagen fantasma está configurado para el archivo 14.

55 Como se señaló anteriormente, el sistema 16 de archivos, al recibir el comando de apertura aprecia que el archivo 14 de imagen fantasma no contiene los datos 20 eliminados y, por lo tanto, devuelve un error (etapa 603), y el filtro 26 de imágenes fantasma intercepta tal error devuelto y percibe en base al mismo que el archivo 14 en cuestión está de hecho en un formato de imagen fantasma (etapa 605). En consecuencia, el filtro 26 de imágenes fantasma ordena al sistema 16 de archivos que recupere la información 24 de imagen fantasma de dicho archivo 14 de 60 imagen fantasma y de hecho recibe dicha información 24 de imagen fantasma (etapa 607). Basándose en dicha

información 24 de imagen fantasma, entonces, el filtro 26 de imágenes fantasma crea una memoria intermedia y un identificador correspondiente al archivo 14 de imagen fantasma, y entrega el controlador a la aplicación 30 como la respuesta (normal) al comando de apertura del mismo (etapa 609).

5 Obsérvese que el controlador para el archivo 14 de imagen fantasma tal como se recibe por la aplicación 30 representa un ejemplo del archivo 14 de imagen fantasma como se abrió, y se emplea por la aplicación 30 como la referencia a tal archivo 14 cuando la emisión de otros comandos con respecto a tal archivo 14, tal como, por ejemplo, un comando de lectura o un comando de cierre. Sin embargo, y de manera significativa, dicho controlador para el archivo 14 de imagen fantasma creado por el filtro 26 de imágenes fantasma y como empleado como parámetro de controlador de un comando de la aplicación 30 hará que dicho comando se entregue directamente al filtro 26 de imágenes fantasma y no al sistema 16 de archivos.

10 Debe apreciarse también que la memoria intermedia para el archivo 14 de imagen fantasma se emplea por el filtro 26 de imágenes fantasma para almacenar información relativa al archivo 14 de imagen fantasma, tal como, por ejemplo, la información 24 de imagen fantasma de la misma. Por lo tanto, en el curso del archivo 14 de imagen fantasma que está siendo manipulado por el filtro 26 de imágenes fantasma, los cambios en dicha información 24 de imagen fantasma pueden registrarse en la memoria intermedia y no es necesario volver a escribirlos en el archivo 14. Por supuesto, después de que el filtro 26 de imágenes fantasma haya terminado de manipular el archivo 14 de imagen fantasma, dichos cambios registrados en la memoria intermedia se pueden volver a escribir en la información 24 de imagen fantasma para el archivo 14, si es necesario.

15 En cualquier caso, con el controlador para el archivo 14 de imagen fantasma desde el filtro 26 de imágenes fantasma, la aplicación 30 puede proceder mediante la emisión de la orden de lectura con respecto al archivo 14 de imagen fantasma particular, en el volumen 12, donde dicha orden de lectura tiene parámetros que incluyen el controlador, un desplazamiento en los datos 20 para el archivo, una longitud de lectura con respecto a dichos datos 20 a partir de dicho desplazamiento, y similares. En consecuencia, dicho comando de lectura basado en tener dicho controlador como un parámetro del mismo es finalmente recibido por el filtro 26 de imágenes fantasma y no por el sistema 16 de archivos (etapa 611). Sin embargo, y como debería ser evidente, los datos 20 de hecho no están presentes en el archivo 14 de imagen fantasma, y en consecuencia el filtro de imágenes fantasma almacena el comando de lectura en una cola o similar para esperar un procesamiento adicional (etapa 613).

20 Tal procesamiento adicional, como debe entenderse, comprende principalmente obtener los datos 20 para el archivo 14 de imagen fantasma del sumidero 18. En particular, el filtro 26 de imágenes fantasma inicia la reconstitución del archivo 14 de imagen fantasma basándose en los datos 20 del mismo en el sumidero 18 (etapa 615), donde dicha reconstitución puede producirse de la siguiente manera.

25 Recordando que el filtro 26 de imágenes fantasma puede interactuar con el administrador 28 de imágenes fantasma para realizar una funcionalidad de imagen fantasma de nivel superior y acceso directo a recursos de red tales como el sumidero 18, el filtro 26 de imágenes fantasma emite al administrador 28 de imágenes fantasma una solicitud para obtener alguna porción de los datos 20 para el archivo 14 de imagen fantasma del sumidero 18 (etapa 615a), donde dicha solicitud incluye la identificación de la información 24 de imagen fantasma del archivo 14 de imagen fantasma, un desplazamiento en los datos 20 y una longitud de datos 20 para se obtendrá a partir de dicho desplazamiento. A continuación, el administrador 28 de imágenes fantasma de hecho obtiene la porción solicitada de dichos datos 20 del sumidero 18 (etapa 615b). Presumiblemente, el administrador 28 de imágenes fantasma incluye toda la funcionalidad necesaria para obtenerlo, y lo hace de una manera que es conocida o debería ser evidente para el público relevante y que no necesita establecerse en detalle en el presente documento. Por consiguiente, se puede emplear cualquier procedimiento particular para obtenerlo sin apartarse del alcance de la presente invención.

30 Debe indicarse que puede ser el caso de que en el transcurso del administrador 28 de imágenes fantasma, la obtención de la porción solicitada de datos 20 del sumidero 18 como en la etapa 615b, puede haber cierto retraso. Por ejemplo, puede darse el caso de que dichos datos 20 se obtengan a través de una red, en cuyo caso puede haber un retraso de la red, tal vez del orden de unos pocos milisegundos, unos pocos segundos o incluso más. Por lo tanto, el administrador 28 de imágenes fantasma, el filtro 26 de imágenes fantasma, y/u otra entidad pueden desear mantener una lista de datos pendiente que represente cada porción de datos 20 que se ha solicitado al sumidero 18 y que aún no se ha obtenido del mismo, tal vez junto con un tiempo requerido correspondiente. Como puede apreciarse, una identificación de cada porción de datos solicitada se agrega a la lista de datos pendientes al ser solicitada, y se elimina de dicha lista de datos pendientes al recibirse y almacenarse con el archivo 14 de imagen fantasma. Con dicha lista de datos pendientes, entonces, puede darse el caso de que todas las solicitudes pendientes de datos 20 sean monitorizadas por edad, y que las solicitudes que no se satisfagan dentro de un cierto período de tiempo se puedan agotar.

35 Una vez que el administrador 28 de imágenes fantasma, de hecho, ha obtenido tal porción solicitada de tales datos 20 del sumidero 18, como administrador 28 de imágenes fantasma entrega tal porción solicitada al filtro 26 de imágenes fantasma (etapa 615C), y tal dicho filtro 26 de imágenes fantasma luego escribe dicha porción solicitada a una ubicación adecuada del archivo 14 en cuestión mediante un comando apropiado para el sistema 16 de archivos (etapa 615d). Escribir dicha porción solicitada en la ubicación adecuada del archivo 14 en cuestión es conocido o debería ser evidente para el público relevante y, por lo tanto, no es necesario exponerlo en el presente documento

con ningún detalle. Por consiguiente, se puede emplear cualquier procedimiento particular para escribir sin apartarse del alcance de la presente invención.

Por supuesto, puede ser necesario repetir las etapas 615a-615d un número de veces hasta que todos los datos 20 para el archivo 14 se escriban en el mismo, y por lo tanto tales etapas se repiten según sea necesario. Una vez que todos los datos 20 para el archivo 14 se escriben en el mismo, el filtro 26 de imágenes fantasma emite comandos según sea necesario para el sistema 14 de archivos para corregir los metadatos 22 de dicho archivo 14 ahora reconstituido, incluyendo la eliminación de la información 24 de imagen fantasma y restableciendo el atributo de imagen fantasma (etapa 617). Además, el filtro 26 de imágenes fantasma recupera todos los comandos de lectura para el archivo 14 de la cola y pasa dichos comandos de lectura al sistema 16 de archivos para su posterior procesamiento (etapa 619), por lo que el comando de lectura de la aplicación 30 se responde con los datos 20 en cuestión.

Obsérvese que el controlador para el archivo 14 abierto permanece asociado con el filtro 26 de imágenes fantasma y continúa dirigiendo comandos de la aplicación 30 en relación con dicho archivo 14 ahora reconstituido para filtrar tal filtro 26 de imágenes fantasma. En consecuencia, puede ser el caso de que el filtro 26 de imágenes fantasma reenvíe dichos comandos al sistema 16 de archivos, puede ser el caso de que el filtro 26 de imágenes fantasma vuelva a asociar el identificador con el sistema 16 de archivos, o puede ser el caso de que el filtro de imágenes fantasma ordene otra construcción para volver a asociar el identificador con el sistema 16 de archivos o similar.

#### RECONSTITUCIÓN PARCIAL DEL ARCHIVO 14 DE IMAGEN FANTASMA

Como se puede apreciar, no debería ser necesario reconstituir completamente un archivo 14 de imagen fantasma en circunstancias en las que menos de la totalidad de los datos 20 de imagen fantasma desde el archivo 14 se requieren por una aplicación 30 de solicitud. Por lo tanto, si una aplicación 30 puede determinar que solo se necesitan 1, 2, 12 o 100 kilobytes en un desplazamiento particular de 2 gigabytes de datos 20, no debería ser necesario obtener los 2 gigabytes de datos 20 del sumidero 18, sino que, en cambio, solo los kilobytes de datos 20 necesarios en el desplazamiento particular. Además, en tal situación, al reconstituir solo parcialmente un archivo 14 de imagen fantasma, se evita la necesidad de transmitir una cantidad considerable de datos innecesarios 20 desde el sumidero 18 a la fuente 10 y el ancho de banda requerido para hacerlo se reduce correspondientemente.

Al reconstituir solo parcialmente un archivo 14 de imagen fantasma, una aplicación 30 puede leer solo unos pocos bytes de un archivo si es necesario sin activar una reconstitución completa del archivo 14. Por lo tanto, y por ejemplo, si solo se necesita el primer fotograma de un archivo 14 de video, dicho primer fotograma se obtendría de los datos 20 para el archivo 14 en el sumidero 18, y no la cantidad completa de datos 20 para el archivo 14 de imagen fantasma, que podría ser del orden de 10 o incluso 100 gigabytes. La reconstitución parcial, entonces, obtiene solo la mayor cantidad de datos 20 en el sumidero 18 para un archivo 14 de imagen fantasma como sea necesario, para satisfacer una solicitud de lectura particular de una aplicación 30, y no más.

Obsérvese que cuando se reconstituye parcialmente un archivo 14 de imagen fantasma, que puede o puede no ser el caso que la porción de los datos 20 obtenidos a partir del sumidero 18, se elimina realmente de tal sumidero 18. Por un lado, la porción obtenida se almacena en la fuente 10 y, por lo tanto, ya no necesita mantenerse en el sumidero 18. Sin embargo, por otro lado, en realidad puede requerir más esfuerzo eliminar de hecho la porción obtenida del sumidero 18. Además, en al menos algunas circunstancias, dicha porción obtenida debe permanecer en el sumidero 18 para el acceso por otras fuentes 10, tal como, por ejemplo, si el sumidero es un servidor 18 concentrador y la fuente es un servidor 10 de sucursal.

Como se puede apreciar, si un archivo 14 de imagen fantasma es, de hecho parcialmente reconstituido, un registro se debe mantener para indicar qué porciones del archivo 14 de imagen fantasma, de hecho, se reconstituyen, de manera que el filtro 26 de imágenes fantasma puede determinar si tales porciones están presentes en el archivo 14 de imagen fantasma. De acuerdo con esto, y en una realización de la presente invención, dicho registro se mantiene en la información 24 de imagen fantasma en los metadatos 22 de dicho archivo 14 de imagen fantasma. En particular, para cada sección contigua de datos 20 de un archivo 14 de imagen fantasma que se reconstituye y, por lo tanto, está presente en el archivo 14 de imagen fantasma, la información 24 de imagen fantasma incluye para ello una referencia de sección que incluye un desplazamiento que describe el comienzo de la sección y una longitud que describe una cantidad contigua de tal sección.

Típicamente, la información 24 de imagen fantasma para un archivo 14 de imagen fantasma que incluye los registros antes mencionados de datos 20 parcialmente reconstituidos se mantiene mediante el filtro 26 de imágenes fantasma. Como se señaló anteriormente, aunque dicho filtro 26 de imágenes fantasma podría mantener y actualizar dicha información 24 de imagen fantasma directamente en los metadatos 22 del archivo 14 de imagen fantasma por medio del sistema 16 de archivos, tal mantenimiento y actualización pueden interferir con otras operaciones realizadas por el sistema 16 de archivos. Por lo tanto, el filtro 26 de imágenes fantasma en el transcurso de la manipulación del archivo 14 de imagen fantasma inicialmente obtiene dicha información 24 de imagen fantasma de dichos metadatos 22 y almacena dicha información 24 de imagen fantasma en una memoria intermedia creada en conexión con el archivo 14 de imagen fantasma como en la etapa 609 de la figura 6, luego mantiene y actualiza dicha información 24 de imagen fantasma durante el transcurso de la manipulación del archivo 14 de imagen

fantasma, y al concluir escribe dicha información 24 de imagen fantasma desde la memoria intermedia a los metadatos 22 para el archivo 14 de imagen fantasma.

Obsérvese que, en ocasiones, la reconstitución parcial de un archivo 14 de imagen fantasma puede ser interrumpida, por ejemplo, por una pérdida de potencia o conectividad de red en la fuente 12. Del mismo modo, también puede darse el caso de que la memoria intermedia para el archivo 14 de imagen fantasma se pierda, especialmente en el caso de una pérdida de potencia y donde la memoria intermedia se mantiene en una RAM volátil o similar. En tal situación, la información 24 de imagen fantasma que se mantiene y actualiza se pierde y no se escribe desde la memoria intermedia en los metadatos 22 para el archivo 14, y en efecto no solo se pierde dicha información 24 de imagen fantasma, sino todos los datos 20 parcialmente reconstituidos asociados con la misma también se pierde a pesar de estar físicamente presente en la fuente 10, especialmente en la medida en que dichos datos 20 no pueden ubicarse sin dicha información 24 de imagen fantasma. Por consiguiente, en una realización de la presente invención, dicha información 24 de imagen fantasma, tal como se mantiene y actualiza, se escribe periódicamente desde la memoria intermedia a los metadatos 22 para el archivo 14, tal como, por ejemplo, una vez cada minuto aproximadamente. Por lo tanto, a lo sumo, solo un minuto más o menos de dicha información 24 de imagen fantasma y los datos 20 parcialmente reconstituidos asociados con la misma pueden perderse debido a una pérdida de la memoria intermedia.

En una realización de la presente invención, un archivo 14 de imagen fantasma se reconstituye solo parcialmente sobre la base de una orden apropiada de la aplicación 30 de solicitud de datos 20 de tal archivo 14 de imagen fantasma. En consecuencia, dicha aplicación 30 primero se debe verificar con el sistema 16 de archivos para determinar si el atributo fantasma está configurado para un archivo 14 particular para determinar si dicho archivo 14 es realmente una imagen fantasma, y si es así, la aplicación 30 puede entonces, mediante un comando apropiado, solicitar una reconstitución parcial de una porción o porciones de los datos 20 para dicho archivo 14 de imagen fantasma.

En vista de la reconstitución parcial, entonces, y en una realización de la presente invención, el filtro 26 de imágenes fantasma en respuesta a un comando de lectura con respecto a archivo 14 de imagen fantasma como el que figura en la etapa 611 de la figura 6 responde a la misma revisando primero cualquier referencia de sección en la información 24 de imagen fantasma para dicho archivo 14 de imagen fantasma almacenado en una memoria intermedia correspondiente, y luego determina a partir de las referencias de sección si los datos 20 solicitados o una porción de los mismos ya está presente en el archivo 14 de imagen fantasma en la fuente 10. Si todos los datos solicitados 20 están de hecho presentes, dichos datos 20 se leen del archivo 14 de imagen fantasma en la fuente 10 sin necesidad de obtener dichos datos 20 del sumidero 18. Si solo está presente una porción de los datos 20 solicitados, dicha porción actual de los datos 20 se lee del archivo 14 de imagen fantasma en la fuente 10 sin necesidad de obtener dichos datos 20 del sumidero 18 y se obtiene el resto de los datos 20 del sumidero 18 y luego los lee como se indica anteriormente. Si ninguno de los datos 20 solicitados está presente, todos estos datos 20 se obtienen del sumidero 18 y luego se leen como anteriormente.

En resumen, entonces, y volviendo ahora a la figura 7, en respuesta a un comando desde una aplicación 30 para abrir un archivo 14 de imagen fantasma de la que puede ser necesaria reconstitución parcial (etapa 701), el filtro 26 de imágenes fantasma crea de nuevo una memoria intermedia y un identificador correspondiente al archivo 14 de imagen fantasma, y entrega el identificador a la aplicación 30 como la respuesta (normal) al comando de apertura del mismo como en la etapa 609 (etapa 703). Además, el filtro 26 de imágenes fantasma obtiene inicialmente la información 24 de imagen fantasma de los metadatos 22 para el archivo 14 de imagen fantasma y almacena dicha información 24 de imagen fantasma en la memoria intermedia creada (etapa 705).

Como anteriormente, con el controlador para el archivo 14 de imagen fantasma desde el filtro 26 de imágenes fantasma, la aplicación 30 emite un comando de lectura para una porción de los datos 20 del archivo 14 de imagen fantasma particular, en el volumen 12, donde dicho comando de lectura incluye el controlador, y define el desplazamiento y la longitud de la porción como en la etapa 611 (etapa 707). Aquí, en respuesta al comando de lectura, el filtro 26 de imágenes fantasma determina, basándose en la información 24 de imagen fantasma en la memoria intermedia para el archivo 14 de imagen fantasma, si dicha porción tal como está definida ya está presente al menos parcialmente en el archivo 14 de imagen fantasma como residente en la fuente 10 (etapa 709). Hacer tal determinación es conocido o debería ser evidente para el público relevante y, por lo tanto, no es necesario exponerlo en detalle en el presente documento. Por consiguiente, un procedimiento de hacer tal determinación puede emplearse sin apartarse del alcance de la presente invención.

De nuevo, si la totalidad de la porción de los datos 20, de hecho, está presente, dicha porción de estos datos no tiene que obtenerse desde el sumidero 18 (etapa 711a). Si solo está presente una parte de la porción de los datos 20, el resto de la porción de los datos 20 se obtiene del sumidero 18 (etapa 711b). Si no está presente ninguna porción de los datos 20, toda esa porción de los datos 20 se obtiene del sumidero 18 (etapa 711c). Significativamente, al obtener cualquier parte de los datos 20 del sumidero 18 tal como en las etapas 711b o 711c, el filtro 26 de imágenes fantasma actualiza la información 24 de imagen fantasma para el archivo 14 de imagen fantasma almacenado en la memoria intermedia para reflejar adecuadamente que la parte de los datos 20 ahora es residente y está reconstituida en el archivo 14 de imagen fantasma (etapa 713). En cualquier caso, con la porción solicitada de los datos 20 ahora residente en la fuente 10, la aplicación 30 puede leer dicha porción solicitada, de

hecho, como en la etapa 619 (etapa 715).

Como se ha establecido anteriormente, el filtro 26 de imágenes fantasma escribe periódicamente la información 24 de imagen fantasma en su forma más actual de la memoria intermedia a los metadatos 22 para el archivo 14 de manera que dicha información 24 de imagen fantasma y los datos 20 parcialmente reconstituidos asociados con el mismo no se pierdan totalmente en el caso de una pérdida de la memoria intermedia (etapa 717). Además, una vez que se cierra el archivo 14 de imagen fantasma, tal como, por ejemplo, al comando de la aplicación 30 (etapa 719), el filtro 26 de imágenes fantasma cierra la memoria intermedia escribiendo la información 24 de imagen fantasma en su forma más actual desde dicha memoria intermedia a los metadatos 22 para el archivo 14 (etapa 721), suponiendo, por supuesto, que el archivo 14 de imagen fantasma no se haya reconstituido por completo. Por lo tanto, dicha información 24 de imagen fantasma en su forma más actual puede recuperarse nuevamente en algún momento posterior como en la etapa 705.

#### LECTURA RÁPIDA DEL ARCHIVO 14 DE IMAGEN FANTASMA PARCIALMENTE RECONSTITUIDO

Una aplicación 30 en la emisión de un comando de lectura con respecto a un archivo 14 de imagen fantasma no tiene en consideración el estado del archivo 14 de imagen fantasma, y en particular, no tener en cuenta si el archivo 14 de imagen fantasma ya se ha reconstituido parcialmente y/o ya está en proceso de reconstitución parcial. Es decir, y volviendo ahora a la figura 8, puede darse el caso de que un comando de lectura particular con respecto a un archivo 14 de imagen fantasma especifique una porción particular de los datos 20, y que dicha porción particular corresponda a un primer segmento que comprende datos 20 que ya están presentes en el archivo 14 de imagen fantasma, un segundo segmento que comprende datos 20 que están pendientes de copiarse desde el sumidero 18 al archivo 14 de imagen fantasma, y un tercer segmento que comprende datos 20 que no están presentes en el archivo 14 de imagen fantasma, pero en su lugar están almacenados en el sumidero 18 solamente.

Como se puede apreciar, si un comando de lectura a procesar de tal manera que la totalidad de la porción particular de los datos 20 fueron copiados desde el sumidero 18 en el archivo 14 de imagen fantasma, tal procesamiento sería duplicado y derrochador con respecto a al menos el primer y el segundo segmento correspondientes a dicha porción particular. En particular, y como también se puede apreciar, copiar los datos 20 correspondientes al primer segmento es innecesario en la medida en que dichos datos 20 ya están presentes en el archivo 14 de imagen fantasma, y copiar los datos 20 correspondientes al segundo segmento es innecesario en la medida en que los datos 20 ya están pendientes de ser copiados desde el sumidero 18 al archivo 14 de imagen fantasma. De hecho, solo los datos 20 correspondientes al tercer segmento deben copiarse del sumidero 18 al archivo 14 de imagen fantasma, ya que dichos datos 20 no están presentes en el archivo 14 de imagen fantasma y aún no se ha solicitado que se copien a dicho archivo 14 de imagen fantasma.

En una realización de la presente invención, entonces, el filtro 26 de imágenes fantasma procesa un comando de lectura para una porción particular de datos 20 identificando primero con respecto a dicha parte particular de datos 20 correspondientes a segmentos en el archivo 14 de imagen fantasma que ya están presentes (es decir, primeros segmentos), segmentos correspondientes en el archivo 14 de imagen fantasma que ya están pendientes (es decir, segundos segmentos) y segmentos correspondientes en el archivo 14 de imagen fantasma que no están presentes o pendientes (es decir, terceros segmentos), y luego, de hecho, obteniendo solo dichos terceros segmentos. En efecto, entonces, el filtro 26 de imágenes fantasma elimina del comando de lectura todos los segmentos que de hecho no necesitan leerse desde el sumidero 18. Al realizar dicha acción de eliminación, el comando de lectura se gestiona más rápido, ya que dicha 'lectura rápida' obtiene solo los datos 20 del sumidero que de hecho se necesitan y no los datos 20 que ya están presentes en el archivo 14 de imagen fantasma o que están pendientes. Por lo tanto, la lectura rápida realizada por el filtro 26 de imágenes fantasma da como resultado una respuesta más rápida al comando de lectura desde la aplicación 30, y al mismo tiempo reduce la cantidad de ancho de banda necesaria para dicho comando de lectura.

En una realización de la presente invención, el filtro 26 de imágenes fantasma identifica a cada primer segmento en el archivo 14 de imagen fantasma que ya está presente con referencia a la información 24 de imagen fantasma mantenida para dicho archivo 14 de imagen fantasma. Como se estableció anteriormente, se puede hacer referencia a dicha información 24 de imagen fantasma como localizada en los metadatos 22 para dicho archivo 14 de imagen fantasma, o se puede referir como ubicada en una memoria intermedia correspondiente a dicho archivo 14 de imagen fantasma. Del mismo modo, en una realización de la presente invención, el filtro 26 de imágenes fantasma identifica cada segundo segmento en el archivo 14 de imagen fantasma que ya está pendiente con referencia a la información en la lista de datos pendientes divulgada anteriormente en relación con la etapa 615b de la figura 6. La identificación de dichos segmentos primero y segundo con referencia a dicha información 24 de imagen fantasma y la lista de datos pendientes debe ser conocida o evidente para el público relevante y, por lo tanto, no es necesario exponerla en el presente documento con ningún detalle. Por consiguiente, dicha identificación puede realizarse de cualquier manera apropiada sin apartarse del alcance de la presente invención.

Como ahora puede apreciarse, una vez que se identifican los segmentos primero y segundo, el resto es el tercer segmento(s), y por proceso de eliminación, entonces, estos terceros segmentos del archivo 14 de imagen fantasma que no están presentes o pendientes son identificados. Por supuesto, una vez identificados, dichos terceros segmentos pueden solicitarse desde el sumidero 18 y, una vez recibidos, copiarse en el archivo 14 de imagen

fantasma. Debe tenerse en cuenta que una vez solicitado, cada uno de estos terceros segmentos en efecto se convierten en un segundo segmento pendiente. Debe tenerse en cuenta, también, que una vez recibido y copiado al archivo 14 de imagen fantasma, cada segundo segmento en efecto se convierte en un primer segmento presente. Debe tenerse en cuenta, finalmente, que una vez que un segmento se convierte en un primer segmento, dicho primer segmento está disponible para ser leído por la aplicación 30 en respuesta al comando de lectura del mismo.

En al menos algunas circunstancias, se puede apreciar que, aunque una aplicación 30 no ha emitido un comando de lectura para particular, los datos 20 en un archivo 14 de imagen fantasma, el filtro 26 de imágenes fantasma, sin embargo, debe mover tales datos 20 desde el sumidero 18 al archivo 14 de imagen fantasma en anticipación de tal comando de lectura, especialmente si el filtro 14 de imágenes fantasma no está ocupado de otra manera. Como solo un ejemplo, cuando se transmite contenido como un archivo 14 de audio de imagen fantasma, un archivo 14 de video de imagen fantasma o un archivo 14 multimedia de imagen fantasma, es razonable anticipar que se seguirá un comando de lectura para los datos 20 en un período de tiempo T0 particular mediante un comando de lectura para datos 20 en un período de tiempo siguiente T1. En tal situación, el filtro 26 de imágenes fantasma, si no está ocupado de otra manera, puede aprovechar la oportunidad de obtener dichos datos 20 para el período de tiempo T1 del sumidero 18 incluso sin un comando de lectura específico de la aplicación 30. Por supuesto, al hacerlo, el filtro 26 de imágenes fantasma puede realizar tal acción como una lectura rápida de la manera expuesta anteriormente.

En resumen, entonces, y volviendo ahora a la figura 9, el filtro 26 de imágenes fantasma realiza una lectura rápida en respuesta a una orden de lectura desde una aplicación 30 de la siguiente manera. Preliminarmente, el comando de lectura se recibe desde la aplicación 30, donde dicho comando de lectura especifica una porción o rango de los datos 30 que se leerán desde un archivo 14 de imagen fantasma (etapa 901). Típicamente, y nuevamente, dicho rango se expresa como un desplazamiento y una longitud con respecto a los datos 30.

A partir de entonces, el filtro 26 de imágenes fantasma primero identifica dentro de tal rango de datos 20 los primeros segmentos de datos 20 correspondientes en el archivo 14 de imagen fantasma que ya están presentes (etapa 903). Nuevamente, dicha identificación se realiza con referencia a la información 24 de imagen fantasma mantenida para dicho archivo 14 de imagen fantasma, ya sea en los metadatos 22 para dicho archivo 14 de imagen fantasma o en una memoria intermedia correspondiente a dicho archivo 14 de imagen fantasma. Si los primeros segmentos identificados comprenden todo el rango solicitado de datos 20, el archivo 14 de imagen fantasma ya se ha reconstituido en la medida necesaria para satisfacer el comando de lectura, y dicho comando de lectura puede completarse sin más copias de los datos 20 desde el sumidero 18 y sin esperar ningún dato pendiente 20 (etapa 905).

Sin embargo, si los primeros segmentos identificados no comprenden toda la gama de datos 20 solicitados, el archivo 14 de imagen fantasma debe, de hecho, reconstituirse en la medida necesaria para satisfacer el comando de lectura, al menos basado en datos 20 pendientes, y en consecuencia, el filtro 26 de imágenes fantasma calcula un primer conjunto que comprende el rango de cada segmento del rango solicitado de datos 20 que no es un primer segmento (etapa 907). A partir de entonces, y de forma similar a antes, el filtro 26 de imágenes fantasma identifica dentro del primer conjunto los segundos segmentos de datos 20 correspondientes en el archivo 14 de imagen fantasma que están pendientes (etapa 909). Nuevamente, dicha identificación se realiza con referencia a la lista de datos pendientes. Si los segundos segmentos identificados comprenden todo el primer conjunto, el archivo 14 de imagen fantasma está pendiente de ser reconstituido en la medida necesaria para satisfacer el comando de lectura, y dicho comando de lectura puede completarse cuando todos los datos 20 pendientes de hecho se copian en el archivo 14 de imagen fantasma (etapa 911).

Sin embargo, como debería apreciarse ahora, si los segundos segmentos identificados no comprenden todo el primer conjunto, el archivo 14 de imagen fantasma debe reconstituirse en la medida necesaria para satisfacer el comando de lectura, en base a los datos 20 a copiar desde el sumidero 18, y en consecuencia, el filtro 26 de imagen fantasma calcula un segundo conjunto que comprende el rango de cada segmento del rango de datos 20 solicitado que no es un primer segmento y no es un segundo segmento (etapa 913), es decir, los terceros segmentos. A partir de entonces, el filtro 26 de imágenes fantasma solicita que el segundo conjunto/terceros segmentos se copien desde el sumidero 18 al archivo 14 de imagen fantasma (etapa 915).

En el curso del tiempo, y de nuevo, cada tercer segmento conforme a lo solicitado convertido en un segundo segmento pendiente, y cada segundo segmento como de hecho copiado se convierte en un primer segmento presente, hasta que toda la copia se complete (etapa 917). Posteriormente, el rango solicitado del comando de lectura está disponible para ser leído por la aplicación 30 en respuesta a dicho comando de lectura (etapa 919).

#### FORMACIÓN ADICIONAL/FORMACIÓN DE IMÁGENES FANTASMA DE UN ARCHIVO 14

Como se puede apreciar, un archivo 14 de imagen fantasma que se ha reconstituido o parcialmente reconstituido puede en algún momento volverse a formar como imagen fantasma, por ejemplo, si se requiere espacio adicional en la fuente 10. Además, y como también se puede apreciar, un archivo 14 que nunca ha sido una imagen fantasma de la misma manera puede ser una imagen fantasma en algún momento por razones similares.

En particular, una vez que un archivo 14 de imagen fantasma ha sido parcial o completamente reconstituido o una

vez que un archivo 14 se ha instalado en la fuente 10, tal archivo 14 continúa residiendo en la fuente 10 en tal forma a menos que tal archivo 14 se vuelva a formar o se forme una imagen fantasma (en adelante, 'volver a formar imagen fantasma'). Tal cambio de imagen se desencadena por algún evento, que puede ser, por ejemplo, una determinación de la necesidad de espacio en la fuente 10, una determinación de que no se ha accedido al archivo 14 durante un período de tiempo, etc. Además, al volver a formar la imagen fantasma de un archivo 14, especialmente en el contexto de un servidor 10 de sucursal y un servidor 18 concentrador, cualquier cambio en el archivo 14 en la fuente 10 o el sumidero 18 puede replicarse en el sumidero 18 o la fuente 10, respectivamente, para mantener el archivo 14 actualizado.

En una realización de la presente invención, a continuación, volver a formar la imagen fantasma en la fuente 10 se lleva a cabo de acuerdo con un algoritmo de volver a formar la imagen fantasma que tiene en cuenta factores tales como la última vez de acceso, el espacio libre restante en el volumen, la frecuencia de acceso a los datos, modificaciones al archivo 14 que tienen lugar en otro lugar, y/o similares. El algoritmo de volver a formar la imagen fantasma al considerar tales factores debería tener como objetivo reducir la probabilidad de que un archivo 14 de imagen fantasma sea reconstituido nuevamente poco después de volver a formarse la imagen fantasma. Por lo tanto, el ancho de banda necesario para tal formación de imagen fantasma y volver a formar la imagen fantasma puede minimizarse, si no se reduce. Típicamente, aunque no necesariamente, el algoritmo de volver a formar la imagen fantasma se realiza mediante una aplicación 30 en la fuente 10 o por el administrador 28 de volver a formar la imagen fantasma en la fuente 10, aunque otras entidades pueden realizar dicho algoritmo de volver a formar la imagen fantasma sin apartarse del alcance de la presente invención.

El algoritmo de volver a formar la imagen fantasma como se emplea a los archivos 14 de volver a formar la imagen fantasma a una fuente 10 puede definirse al menos parcialmente basada en la política configurable. En particular, dicho algoritmo de volver a formar la imagen fantasma puede activarse al menos parcialmente en función de una política configurable, y dicho algoritmo de volver a formar la imagen fantasma puede decidir volver a formar la imagen fantasma de un archivo 14 particular al menos parcialmente en base a una política configurable. En cada caso, dicha política puede configurarse por un usuario de la fuente 10, por un administrador de la fuente 10 o similar.

Los posibles parámetros de activación empleados en conexión con un algoritmo de volver a formar la imagen fantasma incluyen, pero no se limitan a:

- un activador periódico que se activa cuando ha transcurrido un período de tiempo predeterminado;
- un activador de espacio que se activa cuando el espacio libre en el volumen 12 en la fuente 10 cae por debajo de una cierta cantidad o cuando el espacio utilizado en el volumen 12 en la fuente 10 excede una cierta cantidad;
- un activador de espacio periódico que se activa cuando ha transcurrido un período de tiempo predeterminado, pero solo si el espacio libre en la fuente 10 cae por debajo de una cierta cantidad o el espacio utilizado en la fuente excede una cierta cantidad;
- un activador de volumen completo que se activa cuando el filtro 26 de imágenes fantasma observa un error de volumen completo devuelto por el sistema 16 de archivos para indicar que el volumen 12 no tiene espacio libre restante;
- un activador de bytes descargados que se activa cuando se descarga un número predeterminado de bytes a la fuente 10; y
- un activador manual que puede ser activado por un usuario, un administrador o similar.

Por supuesto, la activación del algoritmo de volver a formar la imagen fantasma puede producirse en función de uno solo de dichos activadores o una combinación de dichos activadores.

Los posibles factores de selección para determinar si volver a formar una imagen fantasma para un archivo 14 particular incluyen, pero no se limitan a:

- la última vez que se accedió al archivo 14, por lo que un archivo 14 con un tiempo de acceso anterior se puede volver a formar la imagen fantasma preferentemente;
- el tiempo de descarga asociado con el archivo 14, mediante el cual un archivo 14 con un tiempo de descarga anterior puede volver a formar la imagen fantasma preferentemente;
- el tamaño de archivo para el archivo 14, por lo que un archivo 14 más grande puede volver a formar la imagen fantasma preferentemente;
- el tipo de archivo para el archivo 14, mediante el cual un archivo 14 con una extensión particular se puede volver a formar la imagen fantasma preferentemente;
- los atributos de archivo para el archivo 14, por lo que, por ejemplo, un archivo 14 que es un archivo del sistema que no se vuelve a formar la imagen fantasma, pero en un archivo 14 que se oculta se vuelve a formar la imagen fantasma;
- si están presentes múltiples archivos 14 similares, por lo que los archivos 14 que se consideran similares pueden volverse a formar la imagen fantasma preferentemente;
- si el archivo 14 se ha modificado en el sumidero 18, por lo que dicho archivo 14 se modifica para eliminar los

datos 20 que puedan considerarse obsoletos;

- si el archivo 14 se ha modificado o creado en la fuente 10, por lo que dicho archivo 14 no puede seleccionarse para volver a formar imágenes fantasma para preservar tales modificaciones/creación o puede seleccionarse para volver a formar imágenes para copiar modificaciones/creación al sumidero 18;
- 5 – frecuencia de acceso al archivo 14 durante un período de tiempo particular;
- número de accesos al archivo 14 durante un período de tiempo particular;
- frecuencia de acceso/número de accesos/último tiempo de acceso de otros archivos 14 relacionados, tal como, por ejemplo, en la misma carpeta, en el mismo conjunto de contenido, etc.;
- 10 – entrada externa de archivos específicos o tipos de archivos que se van a modificar preferentemente, tal como, por ejemplo, mediante una lista, un procedimiento, un archivo XML, etc.

Por supuesto, el algoritmo de volver a formar una imagen fantasma puede emplear un único factor de selección o una combinación de dichos factores de selección.

Debe tenerse en cuenta que los archivos 14 seleccionados para volver a formar una imagen fantasma pueden de hecho volver a formar una imagen fantasma o que pueden en cambio simplemente ser designados como candidatos para volver a formar una imagen fantasma preferencial. En este último caso en particular, el volver a formar una imagen fantasma puede realizarse solo hasta que se haya alcanzado un activador de detención particular. Como puede apreciarse, dicho activador de detención puede corresponder al parámetro activador que inició una sesión de volver a formar una imagen fantasma, puede corresponder a criterios para seleccionar archivos 14 candidatos para volver a formar una imagen fantasma, o puede ser otro activador sin apartarse del alcance de la presente invención.

Debe indicarse, además, que en el caso en que el volver a formar una imagen fantasma solo se realiza hasta que se alcance algún activador de detención, puede ser aconsejable en función de los criterios de selección de archivos 14 que son candidatos para volver a formar una imagen fantasma, suavizar el volver a formar una imagen fantasma a través de tales archivos 14. Por ejemplo, si los archivos 14 candidatos se seleccionan en función de tener una última fecha de acceso de al menos dos semanas en la fuente 10, puede ser aconsejable volver a formar una imagen fantasma primero de todos esos candidatos con una última fecha de acceso superior a dos meses, luego si es necesario volver a formar una imagen fantasma de todos los candidatos con una última fecha de acceso superior a un mes, luego, si es necesario, tres semanas, etc., hasta que se active el activador de detención. Del mismo modo, si los archivos 14 candidatos se seleccionan en función de tener un tamaño de archivo mínimo de 10 megabytes, puede ser aconsejable volver a formar una imagen fantasma primero de todos esos candidatos con un tamaño de archivo superior a 1 gigabyte, y luego, si es necesario, volver a formar una imagen fantasma de tales candidatos con un tamaño de archivo superior a 100 megabytes, luego si es necesario 50 megabytes, etc. hasta que se active el activador de detención. Como puede apreciarse, en cualquier escenario, se requiere un cierto procesamiento para crear una lista de los archivos 14 candidatos, para ordenar la lista, para generar una o más rondas de volver a formar una imagen fantasma, y similares.

Debe indicarse, además, que el volver a formar una imagen fantasma se puede emplear para mantener la consistencia de los archivos 14 en la fuente 10 y en el sumidero 18, especialmente en la situación en la que las diferentes versiones del archivo 14 pueden estar situadas en ambos lugares. Por lo tanto, si el archivo 14 se reconstituye en la fuente 10 y los datos 20 del mismo se modifican, pero los datos 20 correspondientes en el sumidero 18 no se modifican, se puede realizar una reorganización para reemplazar los datos 20 en el sumidero 18 con los datos en la fuente 10. Del mismo modo, si los datos 20 del archivo 14 en la fuente 10 no se modifican, pero los datos 20 correspondientes en el sumidero 18 se modifican, se puede volver a formar una imagen fantasma solo para eliminar dichos datos 20 en la fuente 10, con la expectativa de que una reconstitución posterior de dicho archivo copiará los datos 20 en el sumidero 18 a la fuente 10. Por supuesto, si los datos 20 del archivo 14 en la fuente 10 se modifican y los datos 20 correspondientes en el sumidero 18 también se modifican, existe un conflicto y, en consecuencia, se deberán consultar las reglas de conflicto apropiadas para determinar si es así cómo volver a formar una imagen fantasma del archivo 14.

En resumen, entonces, y volviendo ahora a la figura 10, el volver a formar una imagen fantasma de los archivos 14 en un volumen de 12 puede llevarse a cabo de acuerdo con un algoritmo de volver a formar una imagen fantasma, donde tal algoritmo de volver a formar una imagen fantasma se realiza mediante alguna entidad de la siguiente manera. Preliminarmente, el algoritmo de volver a formar una imagen fantasma se activa por algún evento (etapa 1001), donde dicho activador puede ser generado internamente por la entidad o recibido externamente por dicha entidad. Después de tal activación, entonces, el algoritmo de volver a formar una imagen fantasma selecciona archivos 14 entre los archivos 14 reconstituidos total y parcialmente y los archivos 14 de los que nunca se ha formado una imagen fantasma en base a algunos criterios de selección (etapa 1003).

En este punto, el algoritmo de volver a formar una imagen fantasma simplemente puede iniciarse volviendo a formar una imagen fantasma de los archivos 14 seleccionados (etapa 1005), o en su lugar puede considerar los archivos 14 seleccionados para ser archivos 14 candidatos para una posible vuelta a formar una imagen fantasma (etapa 1007). En el último caso, los archivos 14 candidatos se seleccionan para volver a formar imágenes fantasma en una o más rondas hasta que se active un disparador de detención. En particular, para cada ronda, se selecciona un grupo de los archivos 14 candidatos (etapa 1009), para dicho grupo seleccionado se vuelve a formar una imagen fantasma

(etapa 1011), y se determina si el activador de detención se ha activado. Si es así, el procedimiento termina (etapa 1013). Si no, el procedimiento continúa volviendo a seleccionar otro grupo como en la etapa 1009 (etapa 1015).

#### FORMACIÓN DE IMAGEN FANTASMA GENÉRICA

5 Como hasta ahora se expone en el presente documento, todos los archivos 14 en que se ha formado una imagen fantasma en una fuente 10 particular, en particular, se forma una imagen fantasma de tal manera que los datos 20 de la misma residen en un único sumidero 18. Sin embargo, y como puede apreciarse, podría darse el caso de que los datos 20 de cada archivo 14 de imagen fantasma en una fuente 10 particular puedan residir en cualquiera de una pluralidad de tales sumideros 18, como se muestra en la figura 11.

10 En particular, y como se puede apreciar, el filtro 26 de imágenes fantasma de la presente invención no necesita necesariamente estar restringido a trabajar con un único administrador 28 de imágenes fantasma a los datos 20 de imagen fantasma desde un archivo 14 a un solo sumidero 18. En cambio, el filtro fantasma puede de hecho trabajar con una pluralidad de administradores 28 de imágenes fantasma, en el que cada uno de dichos administradores 28 de imágenes fantasma realiza funciones de imágenes fantasmas con respecto a uno particular de una pluralidad de sumideros 18.

15 En el contexto de un servidor 10 de sucursal y un servidor 18 concentrado, por ejemplo, un servidor 10 de sucursal particular puede conectarse con múltiples servidores 18 de concentrador. Por lo tanto, puede ser que un servidor 18 concentrador interconectado con el servidor 10 derivado particular tenga datos 20 de una primera fuente mientras que otro servidor 18 concentrador interconectado con dicho servidor 10 derivado particular tenga datos 20 de una segunda fuente. Asimismo, en el contexto de un dispositivo 10 informático y una ubicación 18 alternativa, por ejemplo, un dispositivo 10 informático particular puede almacenar datos 20 en múltiples ubicaciones alternativas. Por lo tanto, puede ser que una ubicación 18 alternativa para un dispositivo 10 informático particular se designe como almacenamiento 20 de datos de archivos 14 de un primer tipo particular, mientras que otra ubicación 18 alternativa para el dispositivo 10 informático particular se designe como almacenamiento 20 de datos de archivos 14 de un segundo tipo particular. Por supuesto, el número de sumideros 18 empleados y los criterios para dividir los datos 20 entre tales sumideros 18 pueden ser cualquier número y criterio apropiados sin apartarse del alcance de la presente invención.

20 En cualquier caso, si se van a emplear múltiples sumideros 18 con una fuente particular 18, se requiere un mecanismo para identificar qué sumidero 18 tiene datos 20 de un archivo 14 de imagen fantasma particular de dicha fuente 10, e identificar qué administrador 28 de imágenes fantasma correspondiente debe emplearse para acceder a dicho sumidero 18. En consecuencia, y en una realización de la presente invención, dicha identificación se mantiene en la información 24 de imagen fantasma asociada con dicho archivo 14 particular, como se muestra en la figura 11.

25 Como resultado, el filtro 26 de imágenes fantasma al encontrar un archivo 14 de imagen fantasma particular y leer la información 24 de imagen fantasma del mismo puede obtener de dicha información 24 de imagen fantasma la identificación del administrador 28 de imágenes fantasma para ser empleado en conexión con dicho archivo 14 de imagen fantasma particular, y en base a esto, puede comunicarse con dicho administrador 28 de imágenes fantasma identificado para acceder a dicho sumidero 18 según sea apropiado. En efecto, entonces, el filtro 26 de imágenes fantasma es genérico para todos los administradores 28 de imágenes fantasma, y el administrador 28 de imágenes fantasma identificado controla o 'posee' dicho archivo 14 de imagen fantasma particular en la medida en que el filtro 26 de imágenes fantasma nunca debería comunicarse con ningún otro administrador 28 de imágenes fantasma en relación con dicho archivo 14 de imagen fantasma particular.

30 Presumiblemente, cada administrador 28 de imágenes fantasma incluye toda la funcionalidad e información necesaria para comunicarse con el sumidero 18 correspondiente al mismo, de modo que el filtro 26 de imágenes fantasma no necesita preocuparse por tales asuntos. Para el filtro 26 de imágenes fantasma, entonces, la comunicación con el administrador 28 de imágenes fantasma identificado en relación con un archivo 14 de imagen fantasma particular es todo lo que se requiere para acceder al sumidero 18 correspondiente, y dicho filtro 26 de imágenes fantasma no tiene que preocuparse realmente de cómo se identifica el administrador 28 de imágenes fantasmas que se comunica con dicho sumidero 18 correspondiente, cómo dicho administrador 28 de imágenes fantasma localiza dicho sumidero 18 correspondiente, o similar.

35 Similar a los procedimientos mostrados en las figuras 4 y 5, entonces, y ahora con referencia a la figura 12, la formación de imagen fantasma genérica y la reconstitución de un archivo 14 se realiza de la siguiente manera. Presumiblemente, dicha formación de imagen fantasma se inicia mediante un administrador 28 de imágenes fantasma particular que tiene una identificación particular (ID) y de hecho se realiza mediante el filtro 28 de imágenes fantasma, y dicho administrador 26 de imágenes fantasma envía una solicitud de imagen fantasma apropiada al filtro 26 de imágenes fantasma con la ID y una identificación del archivo 14 en el que se va a hacer la imagen fantasma (etapa 1201), y luego el filtro 26 de imágenes fantasma elimina los datos 20 del archivo 14 según sea necesario (etapa 1203). El filtro 26 de imágenes fantasma reenvía dichos datos 20 eliminados según sea necesario al administrador 28 de imágenes fantasma solicitante por medio de su ID (etapa 1205), y dicho administrador 28 de imágenes fantasma puede reenviar dichos datos eliminados al sumidero 18 correspondiente en función de los protocolos que para esto se han incluido con dicho administrador 28 de imágenes fantasma.

5 Debe tenerse en cuenta a este respecto que, dependiendo del tipo de sumidero 18 y el archivo 14, el administrador 28 de imágenes fantasma puede elegir no enviar tales datos eliminados al sumidero 18, tal como, por ejemplo, si el sumidero 18 replica el archivo 14 en cuestión a la fuente 10 de manera de solo lectura. Por supuesto, en tal caso puede ser que la solicitud de creación de imágenes fantasma del administrador 28 de imágenes fantasma informara al filtro de imágenes fantasma para que, de hecho, no realice la etapa 1205 por ser innecesaria.

10 En cualquier caso, y como antes, el filtro 26 de imágenes fantasma modifica los metadatos 22 para el archivo ahora de imagen fantasma configurando el atributo de 'imagen fantasma' y agregando la información 24 de imagen fantasma (etapa 1207). Significativamente, dicha información de imagen fantasma debe incluir la ID del administrador 28 de imágenes fantasma para su uso posterior al reconstituir el archivo 14 de imagen fantasma. Por lo tanto, en algún momento posterior cuando el administrador 28 de imágenes fantasmas o una aplicación 30 solicite acceso a los datos 20 de dicho archivo 14 de imagen fantasma, la solicitud finalmente llega al filtro 26 de imágenes fantasma como se estableció anteriormente (etapa 1209) y dicho filtro 26 de imágenes fantasmas localiza la información 24 de imagen fantasma en los metadatos 22 del archivo 14 (etapa 1211).

15 Nuevamente, los datos 20 del archivo 14 almacenados en el sumidero 18 se ubican en base a dicha información 24 de imagen fantasma identificada, aunque en este caso el filtro 26 de imagen fantasma localiza primero dentro de dicha información 24 de imagen fantasma la ID del administrador 28 de imágenes fantasma responsable de dicho archivo 14 de imagen fantasma (etapa 1213), y con dicha ID comunica al administrador 28 de imágenes fantasma correspondiente una solicitud para obtener de hecho tales datos 20 del sumidero 18 correspondiente (etapa 1215). Presumiblemente, dicho administrador 28 de imágenes fantasmas de hecho obtiene dichos datos 20 de dicho sumidero 18 y los suministra al filtro 26 de imágenes fantasma (etapa 1217), y dicho filtro de imágenes fantasma luego reconstituye dichos datos 20 en el archivo 14 en cuestión (etapa 1219).

#### CONCLUSIÓN

25 La programación necesaria para efectuar los procesos realizados en relación con la presente invención es relativamente sencilla y deben ser evidentes para el público de programación correspondiente. En consecuencia, dicha programación no se adjunta al presente documento. Cualquier programación particular, entonces, puede emplearse para efectuar la presente invención.

30 En la descripción anterior, se puede ver que la presente invención comprende un procedimiento nuevo y útil y mecanismo por el que un archivo 14 a una fuente 10 tal como un volumen 12 local, un dispositivo 10 informático o un servidor 10 de sucursal pueden replicarse o formar una imagen fantasma de modo que sus datos 20 se almacenen en un sumidero 18, tal como una ubicación 18 alternativa o un servidor 18 concentrador, y el archivo 14 en la fuente, por lo tanto, está en una forma reducida o de imagen fantasma que se puede reconstituir si es necesario. El archivo 14 de imagen fantasma puede formarse y reconstituirse, según sea necesario.

**REIVINDICACIONES**

1. Un procedimiento adecuado para un dispositivo (10) informático que tiene un volumen (12) de almacenamiento, un filtro (26) de imágenes fantasma, un administrador (28) de imágenes fantasma, una aplicación (30) y un sistema (16) de archivos que administra el volumen de almacenamiento, comprendiendo el procedimiento las etapas de:

5 recibir (501, 601) mediante el sistema de archivos desde la aplicación un comando abierto con respecto a un archivo de imagen fantasma en el volumen de almacenamiento, en el que un archivo que no es imagen fantasma se ha convertido en el archivo de imagen fantasma eliminando datos del archivo que no es imagen fantasma, en el que los datos se han movido a un sumidero, y en el que los metadatos del archivo de imagen fantasma comprenden un atributo que indica que el archivo de imagen fantasma es un archivo de imagen fantasma e información (24) de imagen fantasma, comprendiendo la información de imagen fantasma una ID e información sobre cómo ubicar el sumidero, almacenándose la ID con e identificando los datos eliminados en el sumidero; interceptar (605) mediante el filtro de imágenes fantasma un error devuelto por el sistema de archivos, devolviéndose el error porque el archivo de imagen fantasma no contiene los datos; recuperar (503, 607) a través del sistema de archivos la información de imagen fantasma en los metadatos del archivo de imagen fantasma; crear mediante el filtro de imágenes fantasma una memoria intermedia y un controlador correspondiente al archivo de imagen fantasma y entregar (609) el controlador a la aplicación como respuesta al comando de apertura; recibir (611) mediante el filtro de imágenes fantasma un comando de lectura para el archivo de imagen fantasma desde la aplicación, teniendo el comando de lectura el controlador como parámetro; almacenar (613) mediante el filtro de imágenes fantasma el comando de lectura en una cola; e iniciar (615) mediante el filtro de imágenes fantasma que reconstituye el archivo de imagen fantasma, en el que la reconstitución comprende:

emitir (615A) al administrador de imágenes fantasma una solicitud para obtener una porción de los datos del archivo de imagen fantasma desde el sumidero, con lo que el administrador de imágenes fantasma de hecho obtiene (615B) dicha porción solicitada de los datos desde el sumidero; recibir (615C) la porción solicitada de los datos del archivo de imagen fantasma desde el administrador de imágenes fantasma identificado en el filtro de imágenes fantasma; y escribir mediante el filtro (615D) de imágenes fantasma la porción recibida de los datos en el archivo de imagen fantasma a través del sistema de archivos.

2. El procedimiento de la reivindicación 1, en el que los metadatos del archivo de imagen fantasma como residentes en el volumen incluyen todos los metadatos del archivo que no es imagen fantasma.

3. El procedimiento de la reivindicación 1 o 2, mediante el cual el filtro de imágenes fantasmas es genérico para una pluralidad de administradores de imágenes fantasma del dispositivo informático.

35 4. Un dispositivo (16) informático, que tiene:

un volumen (12) de almacenamiento;  
 un sistema (16) de archivos que gestiona el volumen de almacenamiento;  
 un filtro (26) de imágenes fantasma;  
 un administrador (28) de imágenes fantasma; y  
 una aplicación (30),  
 en el que el sistema está adaptado para:

recibir (501, 601) mediante el sistema de archivos desde la aplicación un comando abierto con respecto a un archivo de imagen fantasma en el volumen de almacenamiento, en el que un archivo que no es imagen fantasma se ha convertido en el archivo de imagen fantasma eliminando datos del archivo que no es imagen fantasma, en el que los datos se han movido a un sumidero, y en el que los metadatos del archivo de imagen fantasma comprenden un atributo que indica que el archivo de imagen fantasma es un archivo de imagen fantasma e información (24) de imagen fantasma, comprendiendo la información de imagen fantasma una ID e información sobre cómo ubicar el sumidero, almacenándose la ID con e identificando los datos eliminados en el sumidero; interceptar (605) mediante el filtro de imágenes fantasma un error devuelto por el sistema de archivos, devolviéndose el error porque el archivo de imagen fantasma no contiene los datos; recuperar (503, 607) a través del sistema de archivos la información de imagen fantasma en los metadatos del archivo de imagen fantasma; crear mediante el filtro de imágenes fantasma una memoria intermedia y un controlador correspondiente al archivo de imagen fantasma y entregar (609) el controlador a la aplicación como respuesta al comando de apertura; recibir (611) mediante el filtro de imágenes fantasma un comando de lectura para el archivo de imagen fantasma desde la aplicación, teniendo el comando de lectura el controlador como parámetro; almacenar (613) mediante el filtro de imágenes fantasma el comando de lectura en una cola; e

iniciar (615) mediante el filtro de imágenes fantasma que reconstituye el archivo de imagen fantasma, en el que la reconstitución comprende:

- 5 emitir (615A) al administrador de imágenes fantasma una solicitud para obtener una porción de los datos del archivo de imagen fantasma desde el sumidero, con lo que el administrador de imágenes fantasma de hecho obtiene (615B) dicha porción solicitada de los datos desde el sumidero;  
recibir (615C) la porción solicitada de los datos del archivo de imagen fantasma desde el administrador de imágenes fantasma identificado en el filtro de imágenes fantasma; y  
escribir mediante el filtro (615D) de imágenes fantasma la porción recibida de los datos en el archivo de imagen fantasma a través del sistema de archivos.
- 10 5. El dispositivo de la reivindicación 4, en el que los metadatos del archivo de imagen fantasma como residentes en el volumen incluyen todos los metadatos del archivo que no es imagen fantasma.

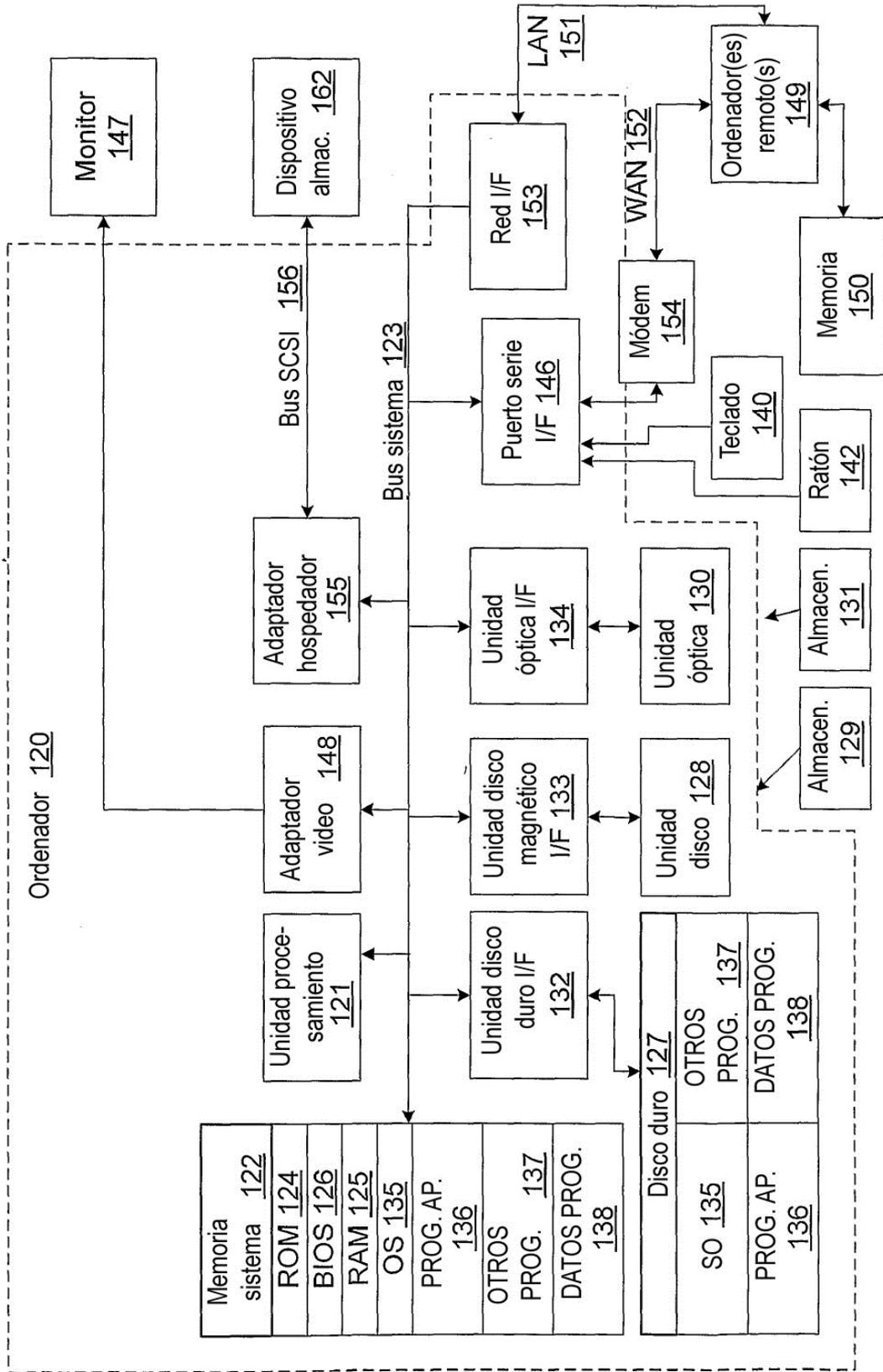


Fig. 1

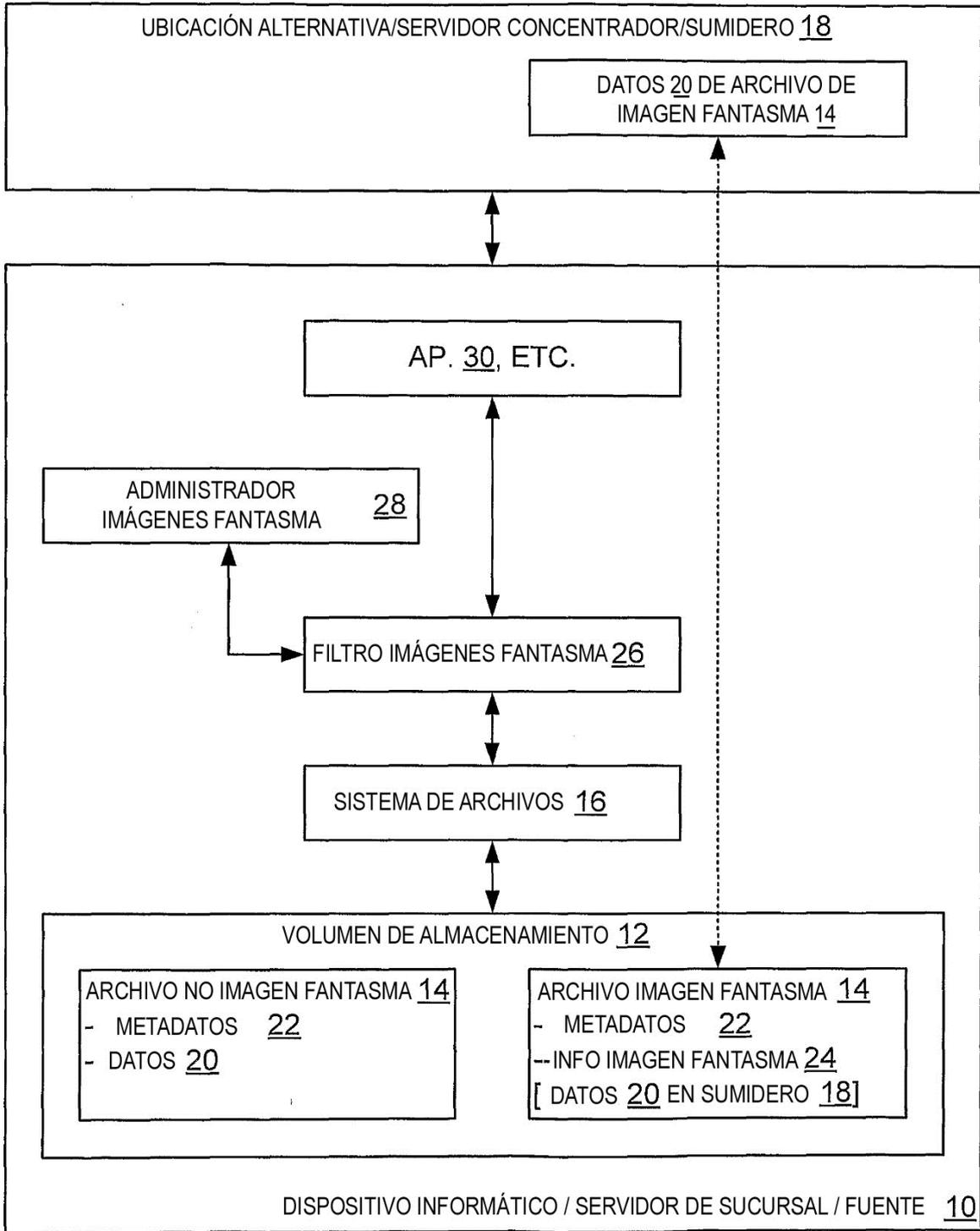


Fig. 2

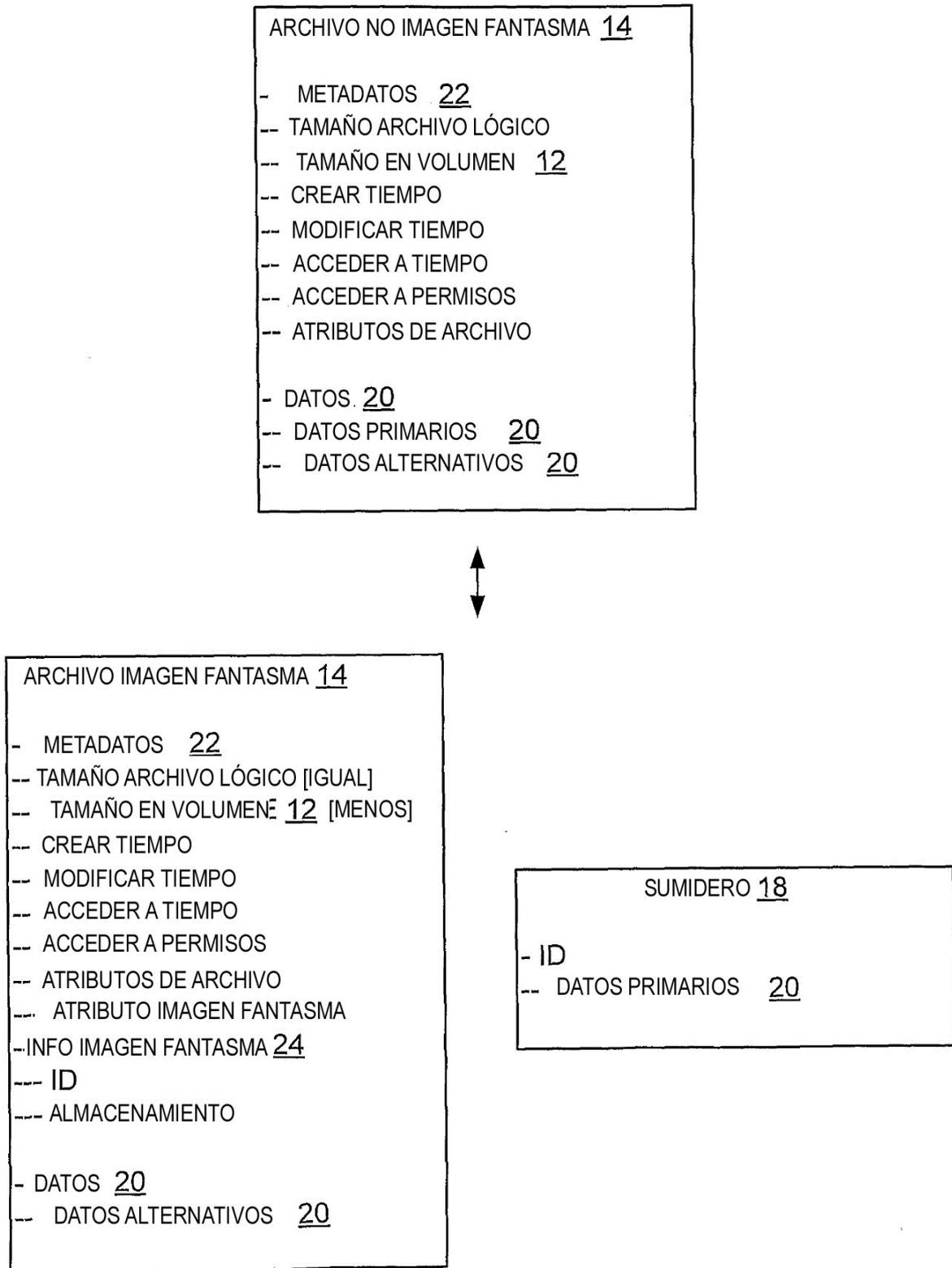


FIG. 3A

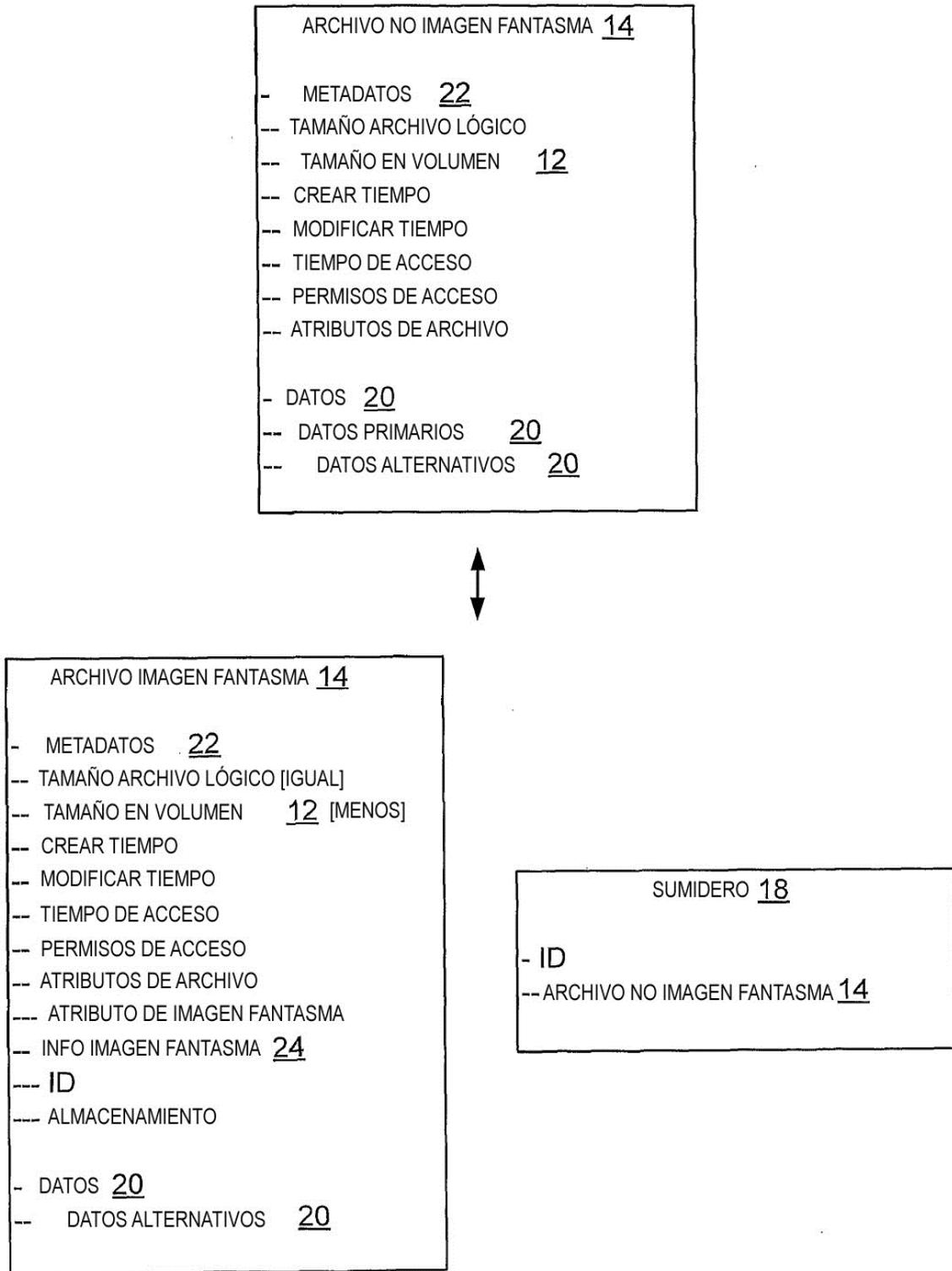
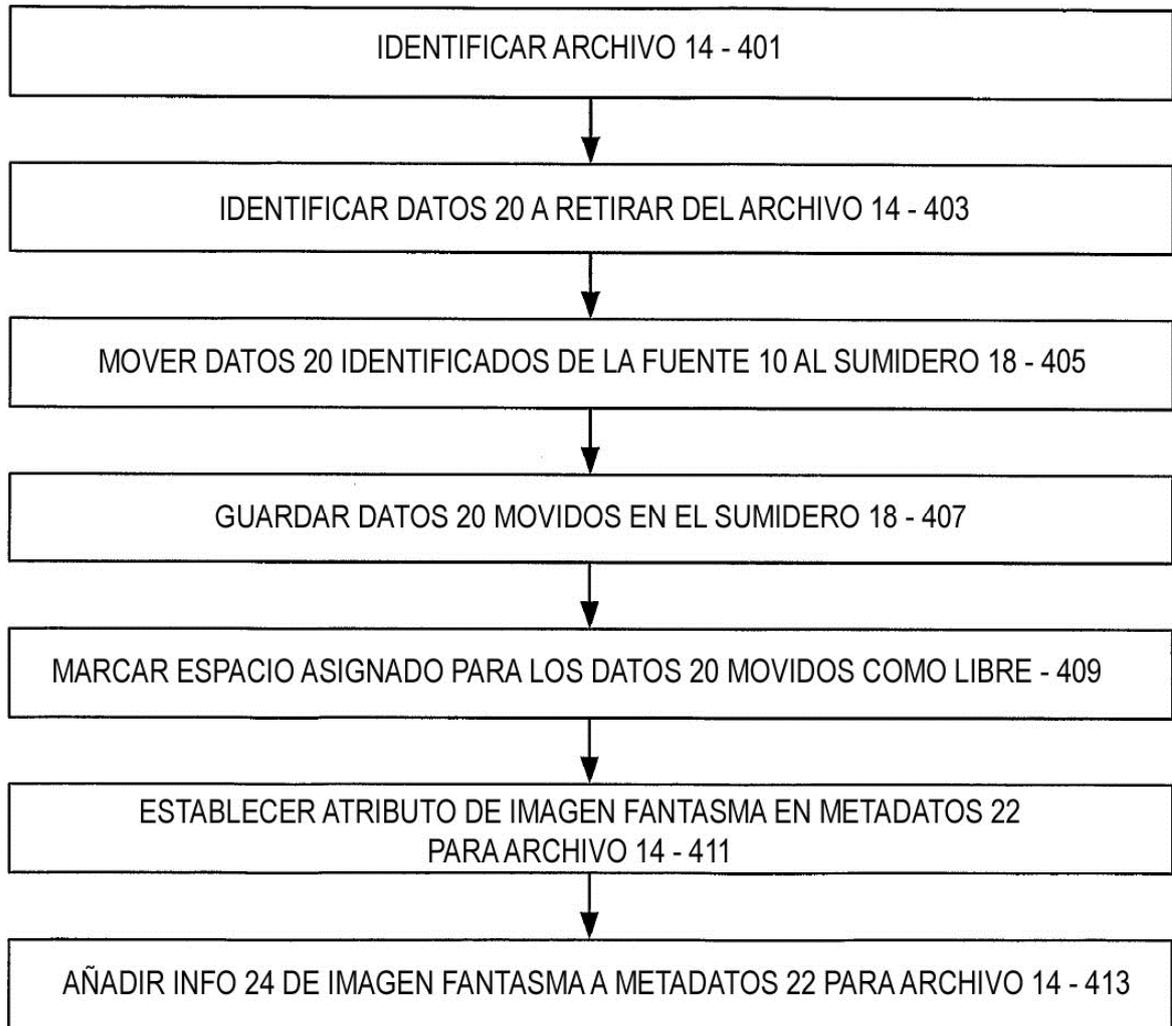
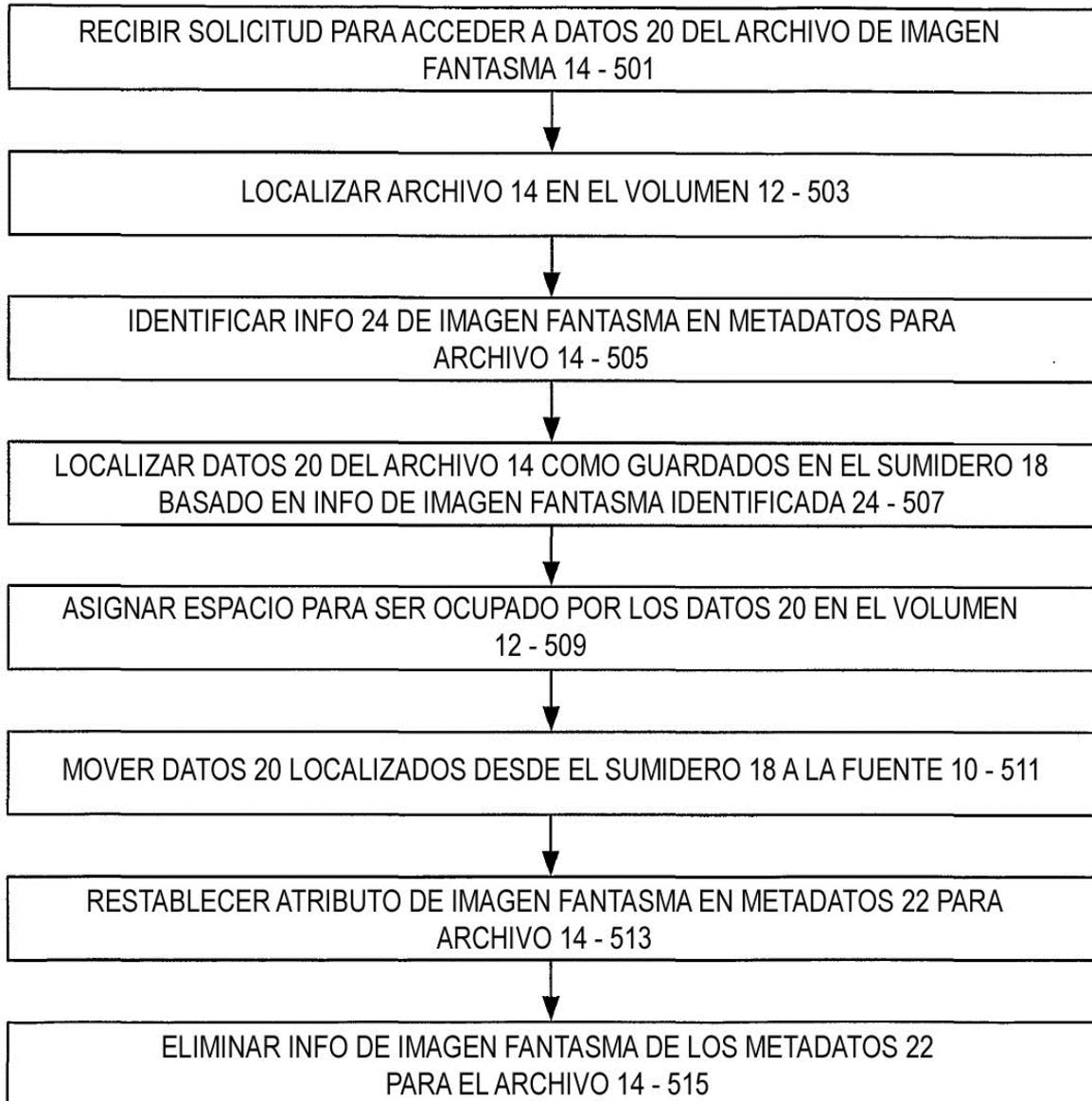


FIG. 3B



**FIG. 4**



**FIG. 5**

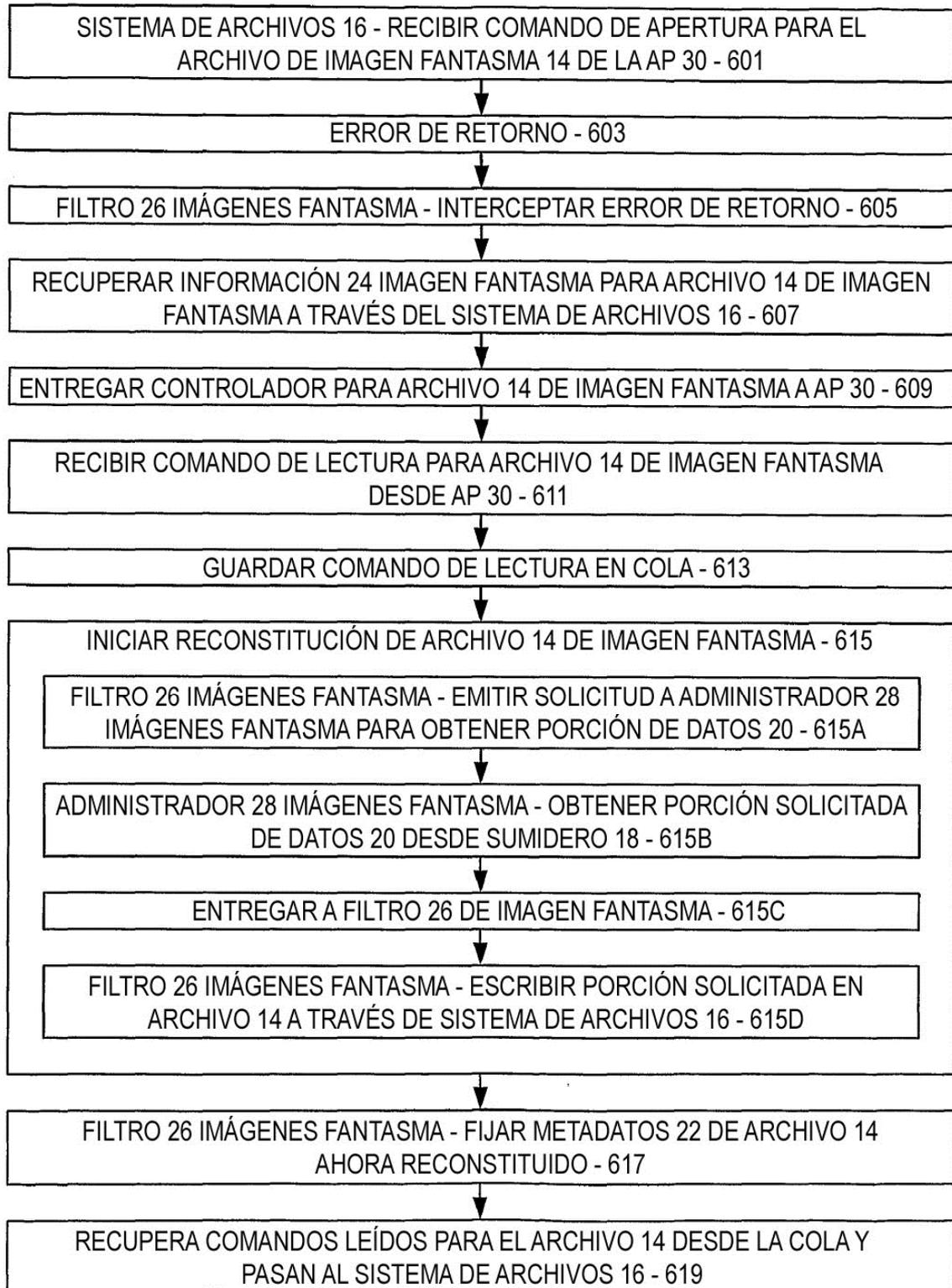
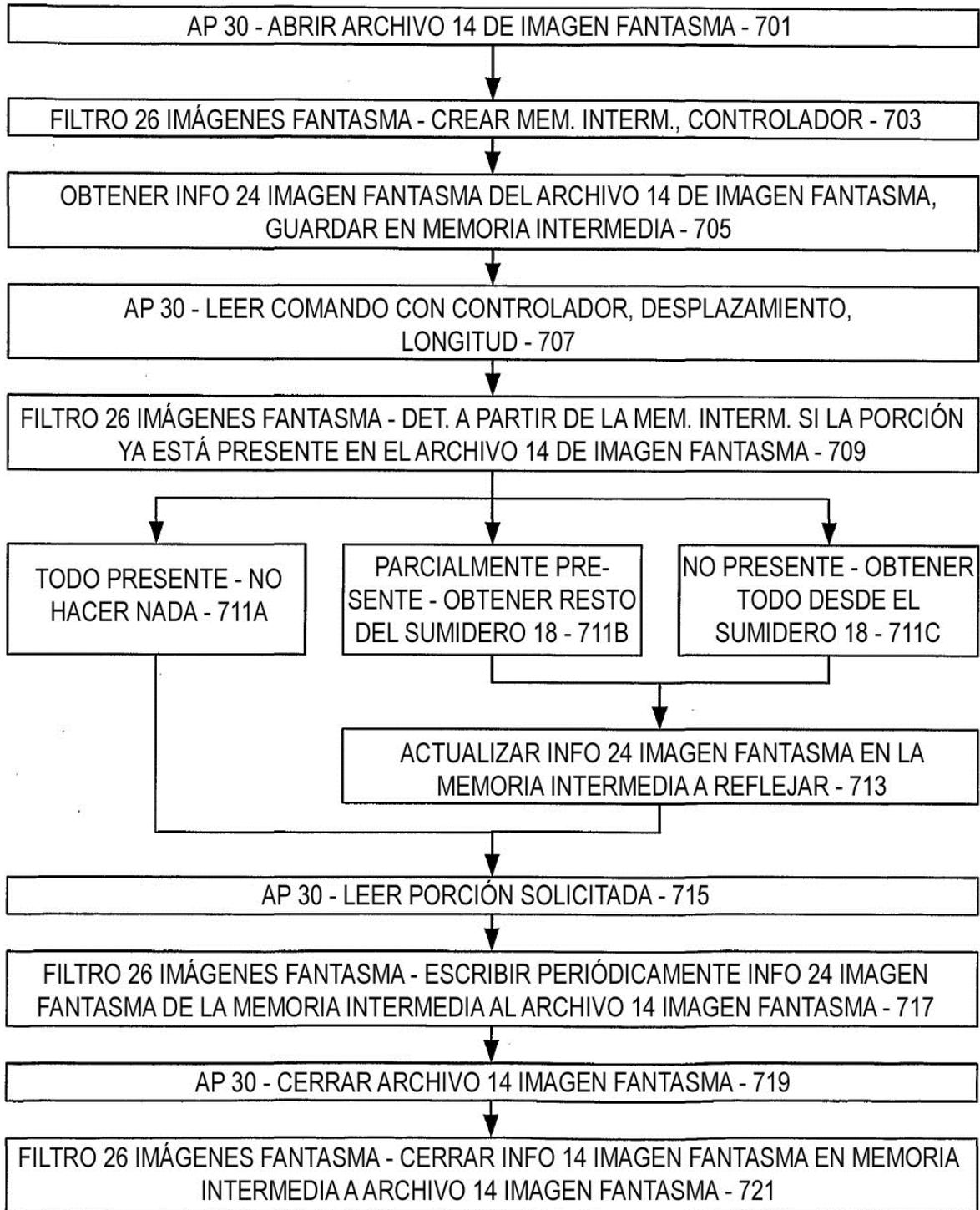


FIG. 6



**FIG. 7**

DATOS 20 EN ARCHIVO 14 DE IMAGEN FANTASMA



**FIG. 8**

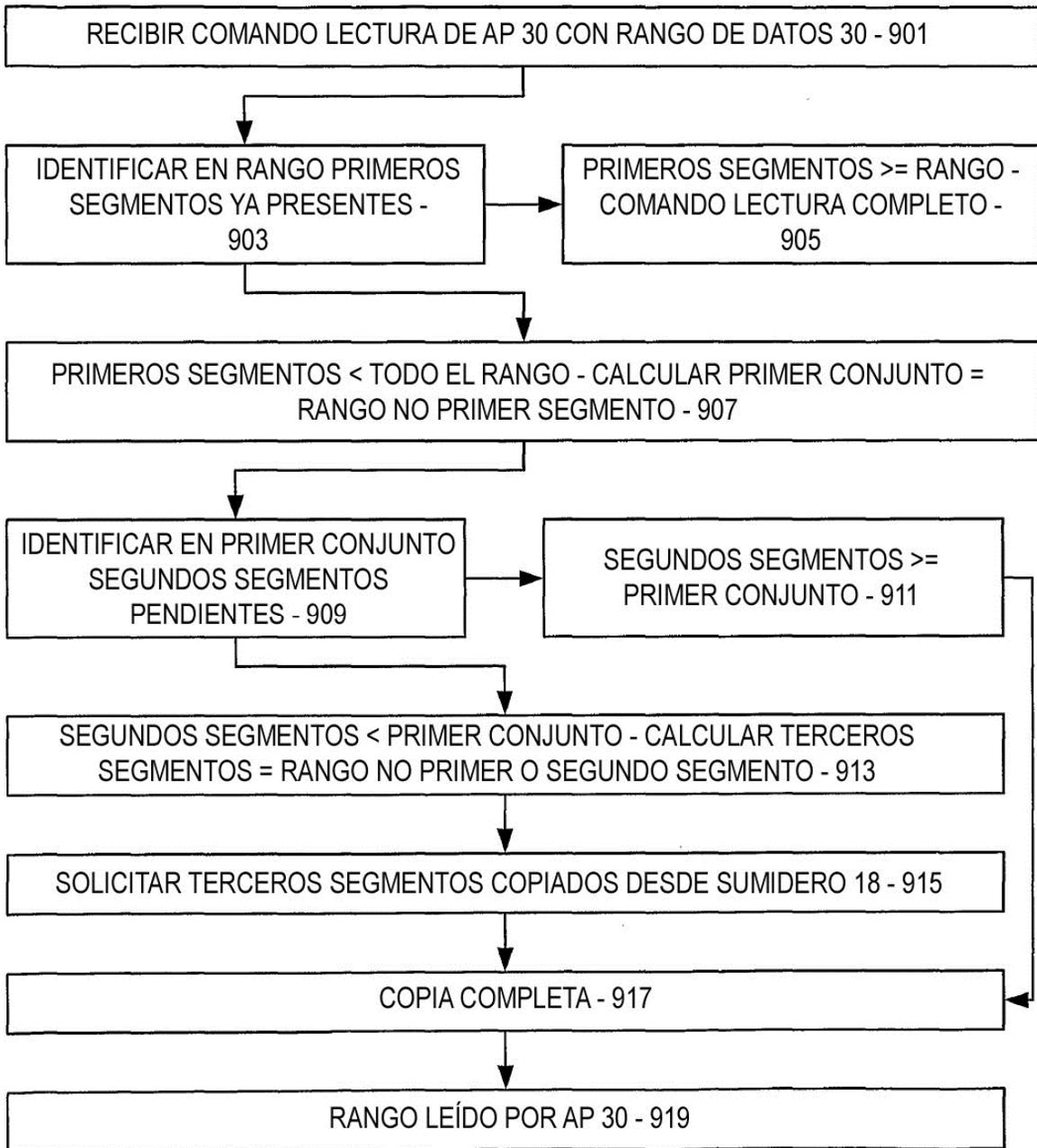


FIG. 9

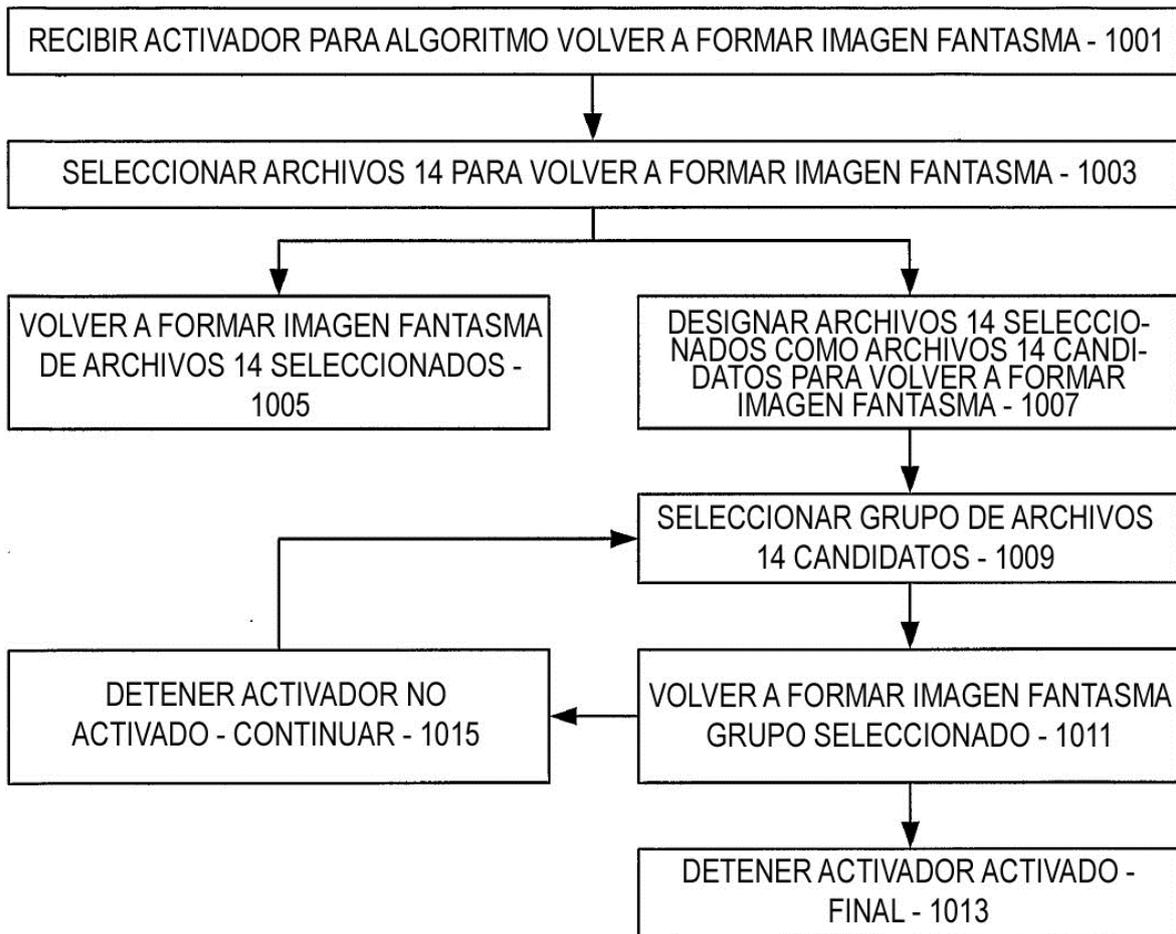


FIG. 10

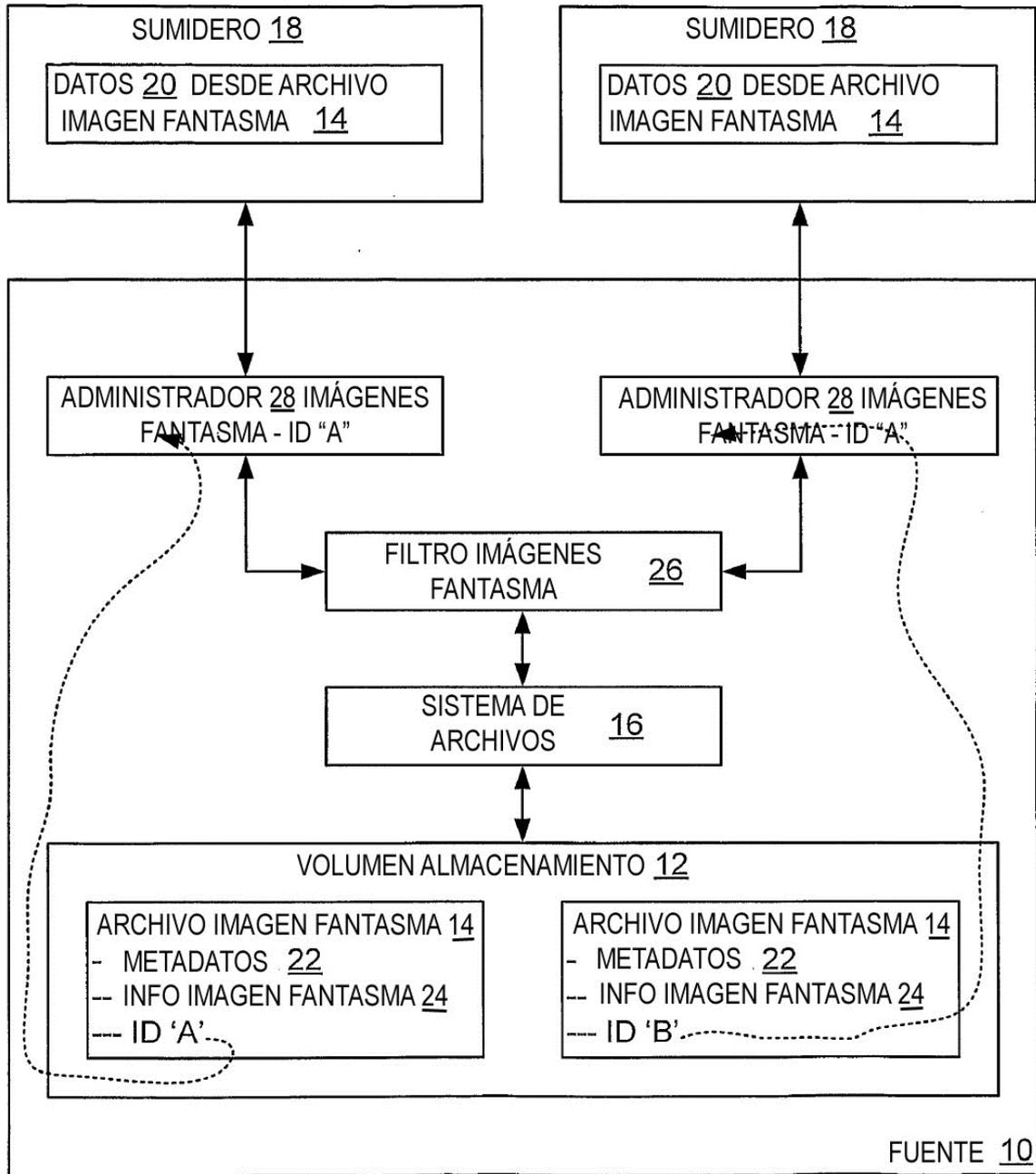
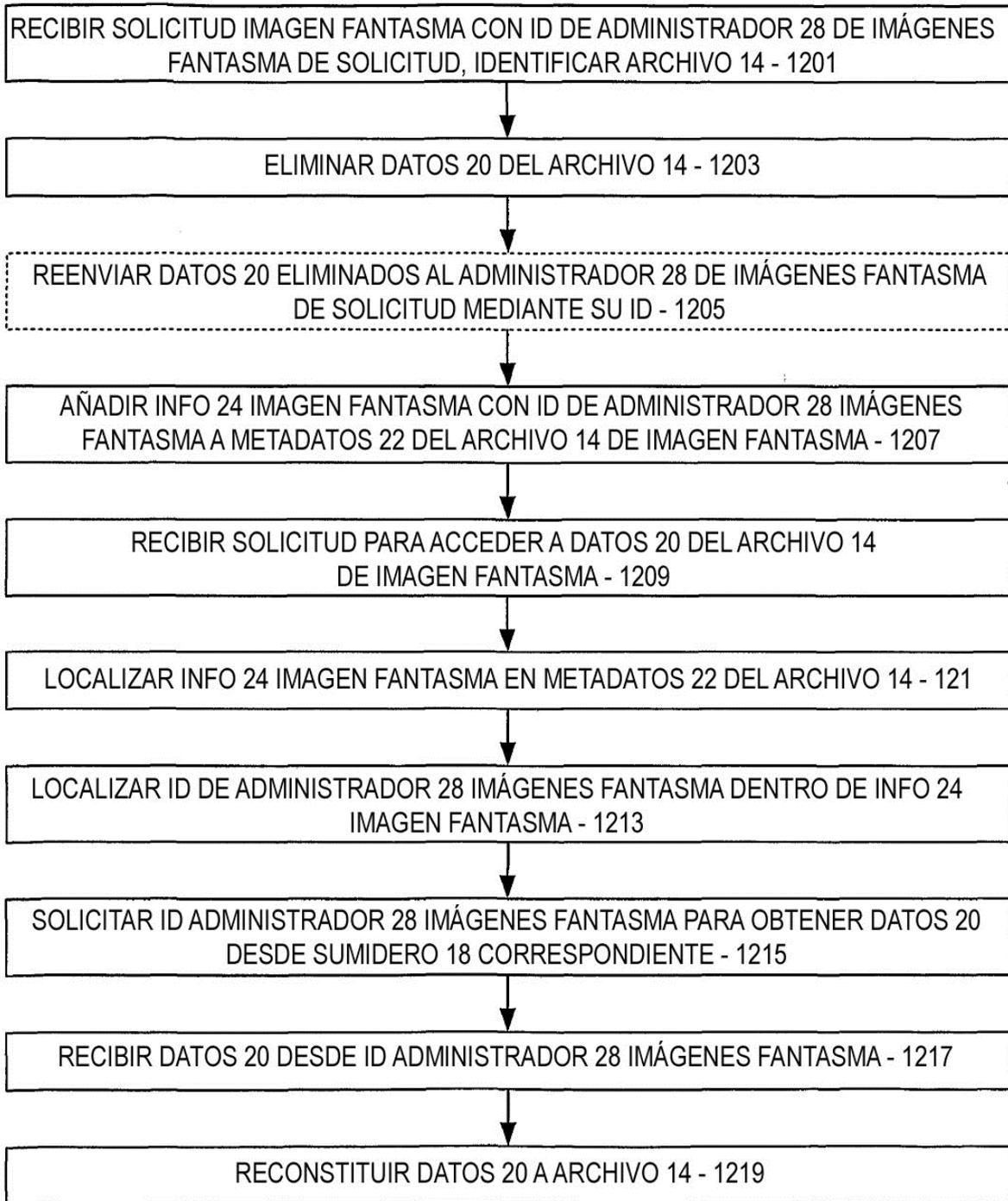


Fig. 11



**FIG. 12**