

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 586 907**

51 Int. Cl.:

G06F 12/08 (2006.01)

H04L 29/08 (2006.01)

G06F 9/45 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **03.06.2011 E 11783092 (7)**

97 Fecha y número de publicación de la concesión europea: **18.05.2016 EP 2562991**

54 Título: **Procedimiento de extracción previa de datos, nodo y sistema para sistema de memoria de tabla hash distribuida (DHT)**

30 Prioridad:

13.01.2011 CN 201110006813

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

19.10.2016

73 Titular/es:

**HUAWEI TECHNOLOGIES CO., LTD. (100.0%)
Huawei Administration Building, Bantian
Longgang District, Shenzhen, Guangdong
518129, CN**

72 Inventor/es:

**BAO, DONG y
YANG, DEPING**

74 Agente/Representante:

LEHMANN NOVO, María Isabel

ES 2 586 907 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

Procedimiento de extracción previa de datos, nodo y sistema para sistema de memoria de tabla *hash* distribuida (DHT)

5 Campo de la invención

La presente invención se refiere al campo de las tecnologías de información y, en particular, a un procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT), un nodo y un sistema.

10

Antecedentes de la invención

A medida que se desarrollan las redes, la cantidad de información aumenta considerablemente. Un sistema de almacenamiento de igual a igual (P2P) basado en la tecnología de tabla *hash* distribuida (DHT) tiene una gran capacidad de adaptación y puede permitir un almacenamiento masivo de datos. Sin embargo, en un sistema de almacenamiento P2P, un segmento de datos consecutivos se divide en varios bloques y se almacena en varios nodos del sistema, de modo que un cliente no puede leer datos de manera consecutiva desde el sistema, generándose así una baja eficacia en la lectura de datos.

15

En un procedimiento existente de extracción previa de datos para un sistema de almacenamiento DHT, un cliente determina un contenido y una ubicación de almacenamiento de datos que van a usarse a continuación según una regla de acceso de un programa de aplicación y, después, el cliente lee datos de cada unidad de almacenamiento pertinente y almacena los datos en el cliente, de modo que cuando una aplicación de capa superior necesita los datos, la aplicación de capa superior puede adquirir directamente los datos desde el cliente sin su transmisión a través de la red, mejorándose así la eficacia de lectura. Sin embargo, la ejecución de la operación de extracción previa de datos, por parte del sistema de almacenamiento DHT, depende del cliente, de modo que el procedimiento de extracción previa simplemente mejora la eficacia de las comunicaciones desde el cliente a cada nodo, pero la eficacia de la lectura de datos del sistema no varía, es decir, la eficacia de la lectura de datos del sistema sigue siendo baja.

20

25

30

El documento de ZUONING YIN ET AL: "*DHT Based Collaborative Multimedia Streaming and Caching Service*", TENCON 2005 IEEE REGION 10, IEEE, PI, 1 de noviembre de 2005 (1/11/2005), páginas 1 a 6, XP031015759, ISBN: 978-0-7803-9311-0, presenta un esquema de genera un servicio de almacenamiento en caché y de flujo continuo de datos multimedia en una red superpuesta basada en una tabla *hash* distribuida. Utilizan DHT para establecer un servicio de almacenamiento en caché descentralizado. La información de ubicación de segmentos almacenados en caché se distribuye entre diferentes nodos homólogos y la información acerca de segmentos del mismo vídeo se organiza en un grupo de multidifusión a nivel de aplicación.

35

El documento US 2004/117398 proporciona un sistema en el que se genera una DB en un entorno de virtualización, un servidor de gestión obtiene información de procesamiento de la DB, tal como un plan de ejecución de procesamiento de DB y un grado de prioridad de procesamiento de un DBMS, predice los datos a los que se va a acceder próximamente y la orden de tales accesos en función de la información, y ordena la lectura en memorias caché de datos de dispositivos de almacenamiento a los que se va a acceder próximamente en función de los resultados de predicción.

40

45

El documento WO2005-066767 proporciona técnicas de gestión de memoria caché. Se recibe una solicitud entrante para acceder a un primer bloque de datos. Se determina la probabilidad de acceder a un segundo bloque de datos en función del acceso del primer bloque de datos, y el segundo bloque de datos se extrae previamente tras determinar que la probabilidad supera el umbral de lectura predefinido.

50

El documento de ZHENHUA LI ET AL: "*Continu Streaming: Achieving high playback continuity of Gossip-based Peer-to-Peer streaming*", 2008 IEEE INTERNATIONAL SYMPOSIUM ON PARALLEL & DISTRIBUTED PROCESSING: [IPDPS 2008]; MIAMI, FL, del 14 al 18 de abril de 2008, IEEE, PISCATAWAY, NJ, EE.UU., 14 de abril de 2008 (14/04/2008), páginas 1 a 12, XP031268224, ISBN: 978-1-4244-1693-6, describe el diseño de *ContinuStreaming*, un sistema de flujo continuo P2P basado en sondeos que puede mantener una gran flexibilidad y una baja sobrecarga, proporcionando al mismo tiempo una cobertura novedosa y crucial para todas las propiedades de la diseminación de datos. Con la ayuda de DHT, los segmentos de datos que pueden perderse debido a la planificación de datos basada en sondeos pueden extraerse rápidamente por los datos bajo demanda. El documento WO2010-076536 se refiere a un sistema de procesamiento de una solicitud para obtener datos emitidos por un cliente, que incluye: una pluralidad de nodos organizados en una red P2P, adecuada para almacenar un conjunto de datos distribuidos a través de los diversos nodos y que almacena una tabla de identificación adecuada para identificar el nodo, entre la pluralidad de nodos, que puede suministrar los datos en respuesta a la solicitud; un servidor *proxy* se usa para almacenar dicha tabla y para identificar el nodo que puede suministrar los datos en respuesta a una solicitud basada en la tabla.

55

60

65

Resumen de la invención

Las formas de realización de la presente invención proporcionan un procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT), un nodo y un sistema, con el fin de resolver el problema de la baja eficacia de lectura en los sistemas de almacenamiento DHT de la técnica anterior.

Una forma de realización de la presente invención proporciona las siguientes soluciones técnicas según las reivindicaciones independientes. Las reivindicaciones dependientes se refieren a formas de implementación preferidas de la presente invención.

Según el procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT), el nodo y el sistema proporcionados en las formas de realización de la presente invención, el bloque de datos que va a extraerse previamente puede determinarse según el bloque de datos leído y el umbral fijado de bloques de datos que van a extraerse previamente, y, después, el bloque de datos que va a extraerse previamente presente en el disco se descarga en la memoria intermedia, de modo que cuando la solicitud de lectura se recibe desde el cliente, el bloque de datos que va a extraerse previamente puede leerse directamente desde la memoria intermedia, mejorándose así la eficacia de acceso de datos para cada nodo de almacenamiento del sistema, y ahorrándose recursos del sistema.

Breve descripción de los dibujos

Para ilustrar más claramente las soluciones técnicas según las formas de realización de la presente invención o de la técnica anterior, a continuación se exponen brevemente los dibujos adjuntos requeridos para describir las formas de realización o la técnica anterior.

La FIG. 1 es un diagrama de flujo de un ejemplo de un procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en la presente invención.

La FIG. 2 es un diagrama esquemático de una estructura de almacenamiento de datos en un sistema de almacenamiento de tabla *hash* distribuida (DHT).

La FIG. 3 es un diagrama de flujo de interacción para llevar a cabo el procedimiento de extracción previa por medio de cada nodo de almacenamiento en el procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT) mostrado en la FIG. 1.

La FIG. 4 es un diagrama de flujo de una forma de realización de un procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en la presente invención.

La FIG. 5 es un diagrama estructural y esquemático de un ejemplo de un nodo de almacenamiento de un sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en la presente invención.

La FIG. 6 es un diagrama estructural y esquemático de una forma de realización de un nodo de almacenamiento de un sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en la presente invención.

La FIG. 7 muestra un proceso principal para procesar, por medio de un nodo de almacenamiento de un sistema de almacenamiento de tabla *hash* distribuida (DHT), una solicitud de lectura proporcionada en una forma de realización de la presente invención.

La FIG. 8 muestra un proceso principal llevado a cabo por un nodo de almacenamiento de un sistema de almacenamiento de tabla *hash* distribuida (DHT) tras recibir una solicitud de extracción previa proporcionada en una forma de realización de la presente invención.

Descripción detallada de las formas de realización

Con el fin de entender mejor los objetivos, las soluciones técnicas y las ventajas de las formas de realización de la presente invención, las soluciones técnicas según las formas de realización de la presente invención se describen a continuación de manera concisa y completa haciendo referencia a los dibujos adjuntos en las formas de realización de la presente invención. La FIG. 1 es un diagrama de flujo de un ejemplo de un procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en la presente invención. Como se muestra en la FIG. 1, el procedimiento incluye:

S101: Un primer nodo de almacenamiento recibe una solicitud de lectura enviada por un cliente.

S102: El primer nodo de almacenamiento determina un bloque de datos que va a extraerse previamente y un segundo nodo de almacenamiento donde reside el bloque de datos que va a extraerse previamente según un bloque de datos leído y un umbral fijado de bloques de datos que van a extraerse previamente, y envía una solicitud de

extracción previa al segundo nodo de almacenamiento, donde el primer nodo de almacenamiento y el segundo nodo de almacenamiento son el mismo o diferentes nodos de almacenamiento, la solicitud de extracción previa incluye información de identificación del bloque de datos que va a extraerse previamente, y la información de identificación se usa para identificar de manera única el bloque de datos que va a extraerse previamente.

5 S103: El segundo nodo de almacenamiento lee de un disco el bloque de datos que va a extraerse previamente según la solicitud de extracción previa, y almacena el bloque de datos que va a extraerse previamente en una memoria intermedia local, de modo que un cliente lee el bloque de datos que va a extraerse previamente de la memoria intermedia local del segundo nodo de almacenamiento.

10 El procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en la presente invención puede aplicarse a un sistema de almacenamiento de tabla *hash* distribuida (DHT). Este tipo de sistema de almacenamiento almacena datos según la DHT y, durante el almacenamiento de datos, el sistema de almacenamiento divide generalmente un segmento de datos consecutivos (es decir, datos no procesados) en varios bloques, y almacena los bloques de datos divididos en varios nodos de almacenamiento del sistema. Por ejemplo, en un sistema de almacenamiento DHT mostrado en la FIG. 2, los bloques de datos divididos D1, D2 ... D8 corresponden a un segmento de datos consecutivos, y los datos no procesados se almacenan en discos de los nodos de almacenamiento N1, N2, N3 y N4. Por lo general, un procedimiento de acceso de datos para cada bloque de datos del sistema de almacenamiento puede ser 'obtener (clave, valor)', donde la clave es una identificación única de un bloque de datos del sistema de almacenamiento, y el valor es un contenido de datos del bloque de datos. El disco de almacenamiento de datos puede ser, por ejemplo, un disco duro común de electrónica de unidad integrada (IDE), un disco duro de conexión de tecnología avanzada serie (SATA) o un disco duro de estado sólido.

25 El primer nodo de almacenamiento, un cuerpo de ejecución de S101 y S102, puede ser cualquier nodo de almacenamiento del sistema de almacenamiento DHT. Tras recibir una solicitud de lectura para leer un determinado bloque de datos D1 enviada por el cliente, el primer nodo de almacenamiento lleva a cabo una operación de extracción previa. Específicamente, puesto que los bloques de datos de los datos no procesados están enumerados secuencialmente, lo que puede reflejarse específicamente por los valores de clave de los bloques de datos, el primer nodo de almacenamiento puede determinar un valor de clave de un bloque de datos que se leerá después de D1 según el valor de clave del bloque de datos leído D1, y también puede llevar a cabo una operación DHT en el valor de clave del bloque de datos que va a leerse, con el fin de adquirir un segundo nodo de almacenamiento donde reside el bloque de datos y enviar una solicitud de extracción previa al segundo nodo de almacenamiento, donde la solicitud de extracción previa incluye información de identificación (un valor de clave) de un bloque de datos que va a extraerse previamente, de manera que el segundo nodo de almacenamiento lleva a cabo una operación de extracción previa en el bloque de datos que va a leerse después de D1 según la información de identificación.

40 Debe observarse que el segundo nodo de almacenamiento no hace referencia a un nodo del sistema, sino que puede ser múltiples nodos de almacenamiento del sistema. Esto se debe a que un bloque de datos determinado que va a extraerse previamente puede ser múltiple. Los bloques de datos que van a extraerse previamente pueden almacenarse en segundos nodos de almacenamiento diferentes. Por lo tanto, después de determinar los bloques de datos que van a extraerse previamente, cada segundo nodo de almacenamiento puede leer, desde un disco, un bloque de datos que va a adquirirse previamente y después almacenar el bloque de datos en una memoria intermedia local, es decir, el segundo nodo de almacenamiento descarga en la memoria intermedia local el bloque de datos que va a extraerse previamente presente en el disco, de modo que el cliente puede leer directamente el bloque de datos de la memoria intermedia local. Además, el segundo nodo de almacenamiento y el primer nodo de almacenamiento pueden ser el mismo nodo de almacenamiento, es decir, el al menos un bloque de datos que va a extraerse previamente determinado por el primer nodo de almacenamiento puede estar ubicado en una unidad de almacenamiento local. En este caso, el primer nodo de almacenamiento lee de un disco local el bloque de datos que va a extraerse previamente según la información de identificación del bloque de datos que va a extraerse previamente incluida en la solicitud de extracción previa, y almacena en una memoria intermedia local el bloque de datos que va a extraerse previamente.

55 Puesto que la operación de extracción previa se lleva a cabo realmente para ahorrar tiempo en la transmisión de red de cada bloque de datos, la capacidad de la memoria intermedia en cada nodo de almacenamiento es limitada y se tienen en cuenta factores, por ejemplo el factor de que si se almacenan en la memoria intermedia más bloques de datos que van a extraerse previamente, la velocidad de lectura se reduce, se fija generalmente un umbral de bloques de datos que van a extraerse previamente y el cómputo de bloques de datos que van a extraerse previamente está limitado por el umbral de bloques de datos que van a extraerse previamente. El umbral de bloques de datos que van a extraerse previamente es, por lo general, mayor o igual que una relación del tiempo de lectura de un único bloque de datos con respecto a un tiempo de transmisión de red de un único bloque de datos. Además, el umbral de bloques de datos que van a extraerse previamente también puede determinarse según la capacidad de memoria intermedia del nodo de almacenamiento y según las condiciones de otros dispositivos de hardware en cada nodo de almacenamiento, por ejemplo un disco, una unidad de procesamiento central (CPU), una placa maestra y una tarjeta de red.

Después de recibir la solicitud de lectura enviada por el cliente, cada nodo de almacenamiento del sistema comprueba si el bloque de datos que va a leerse existe en la memoria intermedia, es decir, si el bloque de datos que va a leerse se ha adquirido previamente. Si el bloque de datos que va a leerse existe en la memoria intermedia, el nodo de almacenamiento puede leer directamente el bloque de datos de la memoria intermedia y devolver el bloque de datos al cliente; y si el bloque de datos que va a leerse no existe en la memoria intermedia, esto indica que el bloque de datos no se ha adquirido previamente, y el nodo de almacenamiento puede leer el bloque de datos del disco. Durante la lectura del bloque de datos o tras la lectura del bloque de datos, cada nodo de almacenamiento puede iniciar además un nuevo proceso de extracción previa como el descrito anteriormente, el cual no se repite de nuevo en el presente documento.

En el procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en este ejemplo, el bloque de datos que va a extraerse previamente puede determinarse según el bloque de datos leído y el umbral fijado de bloques de datos que van a extraerse previamente, y después el bloque de datos que va a extraerse previamente presente en el disco se descarga a la memoria intermedia, de modo que tras la recepción de la solicitud de lectura desde el cliente, el bloque de datos que va a extraerse previamente puede leerse directamente desde la memoria intermedia, mejorándose así la eficacia del acceso de datos para cada nodo de almacenamiento del sistema.

Para ilustrar más claramente el ejemplo del procedimiento de extracción previa de datos proporcionado en la presente invención, la FIG. 3 proporciona un diagrama de flujo de interacción para ejecutar, por medio de cada nodo de almacenamiento de un sistema de almacenamiento, un procedimiento de extracción previa. Como se muestra en la FIG. 3, esta descripción de ejemplo se realiza con referencia a un ejemplo en el que un cliente envía a un nodo de almacenamiento A una solicitud de lectura para leer un bloque de datos D_n , y el nodo de almacenamiento A inicia una operación de extracción previa. El procedimiento incluye:

S201: Un cliente encuentra un nodo de almacenamiento A según la clave de D_n y envía una solicitud de lectura al nodo de almacenamiento A.

Un procedimiento para encontrar, por medio del cliente, el nodo de almacenamiento según la clave del bloque de datos puede ser un procedimiento en el que se usa un algoritmo DHT general, el cual no se describe en detalle en el presente documento.

El cliente puede iniciar una operación de lectura de bloque de datos para el nodo de almacenamiento A de múltiples maneras; por ejemplo, el cliente puede iniciar una operación de lectura de datos para el nodo A usando una interfaz de almacenamiento 'obtener (clave, valor)' proporcionada por un sistema de almacenamiento DHT, o iniciar la operación de lectura de datos según un intervalo de una clave, o de otras maneras que no se mencionan en el presente documento.

S202: El nodo de almacenamiento A consulta el bloque de datos D_n de una memoria intermedia; si el bloque de datos D_n existe en la memoria intermedia, el nodo de almacenamiento A recupera directamente D_n de la memoria intermedia y devuelve D_n al cliente.

S203: Si D_n no existe en la memoria intermedia, lee D_n de un disco y devuelve D_n al cliente.

S204: El nodo de almacenamiento A inicia una operación de extracción previa. Para facilitar la descripción, en este ejemplo se supone que el nodo de almacenamiento A determina, según D_n y un umbral fijado de bloques de datos que van a extraerse previamente, que es necesario extraer previamente dos bloques de datos D_{n+1} y D_{n+2} (dos bloques de datos subsiguientes a D_n de datos no procesados y que tienen números consecutivos a D_n de datos no procesados) y descubre, según un algoritmo DHT, que D_{n+1} y D_{n+2} están almacenados en un nodo de almacenamiento X y un nodo de almacenamiento Y, respectivamente. Después, se llevan a cabo las etapas S205 y S206.

S205: El nodo de almacenamiento A envía una solicitud de extracción previa del bloque de datos D_{n+1} al nodo de almacenamiento X.

S206: El nodo de almacenamiento A envía una solicitud de extracción previa del bloque de datos D_{n+2} al nodo de almacenamiento Y.

Las etapas S205 y S206 no siguen un orden particular y pueden llevarse a cabo de manera síncrona o asíncrona.

S207: El nodo de almacenamiento X recibe la solicitud de extracción previa del bloque de datos D_{n+1} y, después, lleva a cabo un proceso de descarga del bloque de datos desde un disco a una memoria intermedia.

S208: El nodo de almacenamiento Y recibe la solicitud de extracción previa del bloque de datos D_{n+2} y, después, lleva a cabo un proceso de descarga del bloque de datos desde un disco a una memoria intermedia.

En S207 y S208, tras recibirse las solicitudes de extracción previa enviadas por el nodo de almacenamiento A, el nodo de almacenamiento X y el nodo de almacenamiento Y pueden descargar directamente los bloques de datos correspondientes desde los discos a las memorias intermedias.

5 Sin embargo, para mejorar la precisión de extracción previa del sistema, el nodo de almacenamiento que recibe la solicitud de extracción previa puede determinar además si se requiere la extracción previa, y una operación para determinar si se requiere la extracción previa se describe en una forma de realización posterior.

10 S209: Cuando el cliente necesita leer el bloque de datos Dn+1, el cliente encuentra el nodo de almacenamiento X según la clave de Dn+1 e inicia un operación de lectura de bloque de datos en el nodo de almacenamiento X usando la interfaz de almacenamiento 'obtener (clave, valor)' proporcionada por el sistema de almacenamiento DHT.

15 S210: Tras recibir una solicitud para leer el bloque de datos Dn+1, el nodo de almacenamiento X determina que Dn+1 existe en la memoria intermedia y devuelve directamente al cliente el bloque Dn+1 presente en la memoria intermedia.

20 S211: Cuando el cliente necesita leer el bloque de datos Dn+2, el cliente encuentra el nodo de almacenamiento Y según la clave de Dn+2 e inicia un operación de lectura de bloque de datos en el nodo de almacenamiento Y usando la interfaz de almacenamiento 'obtener (clave, valor)' proporcionada por el sistema de almacenamiento DHT.

S212: Tras recibir una solicitud para leer el bloque de datos Dn+2, el nodo de almacenamiento Y determina que Dn+2 existe en la memoria intermedia y devuelve directamente al cliente el bloque Dn+2 presente en la memoria intermedia.

25 La FIG. 4 es un diagrama de flujo de una forma de realización de un procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en la presente invención. Como se muestra en la FIG. 4, en el procedimiento de extracción previa de datos para un sistema DHT proporcionado en esta forma de realización, tras recibir una solicitud de extracción previa, un nodo de almacenamiento necesita determinar además si se requiere una operación de extracción previa con el objetivo de mejorar la precisión de
30 extracción previa del sistema. Específicamente, tras llevarse a cabo una operación de lectura de un bloque de datos, cada nodo de almacenamiento registra el bloque de datos leído y una indicación de tiempo correspondiente en una tabla de registros de lectura, donde la indicación de tiempo se usa para identificar el tiempo de lectura del bloque de datos. Cuando recibe la solicitud de extracción previa, el nodo de almacenamiento determina si llevar a cabo una operación de extracción previa según el cómputo del bloque de datos leído en la tabla de registros de lectura y una
35 indicación de tiempo con el tiempo de lectura más reciente. Si el cómputo del bloque de datos leído es mayor o igual que un umbral fijado de bloques de datos leídos y la diferencia entre el tiempo de lectura previa actual y el tiempo de lectura más reciente mostrado por la indicación de tiempo es inferior o igual al tiempo fijado, se lleva a cabo la operación de extracción previa.

40 La razón de fijar el umbral de bloques de datos leídos en el registro de lectura es que si un cliente lee solamente muy pocos bloques de datos, esto no indica necesariamente que el cliente lea siempre secuencialmente los bloques de datos de los datos no procesados, y esto puede generar una determinación no precisa de la lectura secuencial y, por consiguiente, una extracción previa innecesaria. Sin embargo, el umbral de bloques de datos leídos no puede fijarse a un valor muy elevado, ya que esto puede provocar una extracción previa de baja eficacia en una fase
45 anterior y aumentar el consumo del sistema durante la transferencia y almacenamiento del registro de lectura. Por lo tanto, el umbral de bloques de datos leídos puede fijarse, por ejemplo, según el cómputo de los bloques de datos de los datos no procesados y según configuraciones de hardware del sistema.

50 Además, la diferencia entre el tiempo de lectura previa actual y el tiempo de lectura más reciente mostrado por la indicación de tiempo tiene que ser inferior o igual a un determinado tiempo, lo que indica que un bloque de datos que va a extraerse previamente y los bloques de datos leídos corresponden a una operación de lectura de datos no procesados actual del cliente, en lugar de una operación de lectura de datos no procesados anterior del cliente.

55 Esta forma de realización proporciona un proceso de lectura D1-D4 iniciado por el cliente, donde se supone que la operación de extracción previa se lleva a cabo cuando el umbral fijado de bloques de datos leídos del registro de lectura es 3. El procedimiento incluye específicamente:

S301: Un cliente solicita a un nodo de almacenamiento A que lea un bloque de datos D1.

60 S302: Se supone que D1 es un primer bloque de datos leído por el cliente, el nodo de almacenamiento A lee el bloque de datos D1 de un disco y devuelve el bloque de datos D1 al cliente.

S303: El nodo de almacenamiento A registra <D1> (incluyendo la indicación de tiempo de D1) en un registro de lectura, donde la tabla de registros de lectura corresponde al mismo segmento de datos no procesados.

65

S304: El nodo de almacenamiento A analiza y determina que D2 es posterior al bloque de datos D1, calcula un nodo de almacenamiento B que almacena D2, envía una solicitud de extracción previa de D2 al nodo de almacenamiento B y transporta el registro de lectura de bloque <D1>.

5 Un proceso en el que el nodo de almacenamiento A analiza D2 puede ser: determinar el cómputo de bloques de datos que van a extraerse previamente según un umbral fijado de bloques de datos que van a extraerse previamente. Puesto que un bloque de datos leído y el bloque de datos que va a extraerse previamente pertenecen a los mismos datos no procesados, el bloque de datos leído y el bloque de datos que va a extraerse previamente tienen el mismo prefijo, y el bloque de datos leído y el bloque de datos que va a extraerse previamente están numerados secuencialmente en los datos no procesados, donde el prefijo es una identificación única de los datos no procesados a los que pertenece un bloque de datos, y un número identifica la ubicación de los datos en los datos no procesados. Por lo tanto, el nodo de almacenamiento A puede determinar un prefijo y el número del bloque de datos que va a extraerse previamente según un prefijo y el número del bloque de datos leído D1, y después lleva a cabo una operación *hash* en el bloque de datos que va a extraerse previamente según el prefijo y el número del bloque de datos que va a extraerse previamente, con el fin de determinar un nodo de almacenamiento del bloque de datos que va a extraerse previamente.

S305: Después de recibir la solicitud de extracción previa desde el nodo de almacenamiento A, el nodo de almacenamiento B determina que el bloque de datos leído del registro de lectura es solamente D1, y el nodo de almacenamiento B solo guarda el registro de lectura y no ejecuta una operación de extracción previa.

S306: El cliente solicita al nodo de almacenamiento B que lea el bloque de datos D2.

S307: Puesto que el nodo de almacenamiento B no adquiere previamente D2, el nodo de almacenamiento B lee el bloque de datos D2 desde un disco y devuelve el bloque de datos D2 al cliente.

S308: El nodo de almacenamiento B actualiza el registro de lectura a <D1, D2> (incluyendo la indicación de tiempo de D2).

30 S309: El nodo de almacenamiento B analiza D3 y determina que D3 es posterior al bloque de datos D2, calcula la ubicación de D3, envía una solicitud de extracción previa de D3 a un nodo de almacenamiento C donde reside D3 y transmite el registro con el historial de bloques <D1, D2>.

35 S310: Después de recibir la solicitud de extracción previa desde el nodo de almacenamiento B, el nodo de almacenamiento C determina que el registro de lectura incluye 2 bloques de datos leídos, lo que es inferior a un umbral de bloques de datos leídos de 3, y el nodo de almacenamiento C solo guarda el registro de lectura y no ejecuta una operación de extracción previa.

S311: El cliente solicita al nodo de almacenamiento C que lea el bloque de datos D3.

40 S312: Puesto que el nodo de almacenamiento C no adquiere previamente D3, el nodo de almacenamiento C lee el bloque de datos D3 de un disco y devuelve el bloque de datos D3 al cliente.

45 S313: El nodo de almacenamiento C actualiza el registro de lectura a <D1, D2, D3> (incluyendo la indicación de tiempo de D3).

50 S314: El nodo de almacenamiento C analiza D4 y determina que D4 es posterior al bloque de datos D3, calcula la ubicación de D4, envía una solicitud de extracción previa de D4 a un nodo de almacenamiento D donde reside D4 y transmite el registro de bloques <D1, D2, D3>.

55 S315: Tras recibir la solicitud de extracción previa desde el nodo de almacenamiento C, el nodo de almacenamiento D determina que el registro de lectura incluye 3 bloques de datos leídos, número que es igual al umbral de bloques de datos leídos de valor 3, y el nodo de almacenamiento D determina además si la diferencia entre el tiempo de lectura previa actual y el tiempo de lectura más reciente mostrado por la indicación de tiempo es inferior o igual al tiempo fijado, y en el presente documento se supone que la diferencia es inferior o igual al tiempo fijado, de modo que el nodo de almacenamiento guarda el registro de lectura, lee D4 de un disco y almacena D4 en una memoria intermedia.

S316: El cliente solicita al nodo de almacenamiento D que lea el bloque de datos D4.

60 S317: Puesto que el nodo de almacenamiento D ha adquirido previamente D4, el nodo de almacenamiento D lee directamente el bloque de datos D4 de la memoria intermedia y devuelve el bloque de datos D4 al cliente.

65 Después, el nodo de almacenamiento D sigue llevando a cabo una operación de extracción previa de un bloque de datos, lo que no se repite de nuevo en el presente documento.

Además, si el cómputo de los bloques de datos leídos del registro de lectura sea igual al umbral fijado de bloques de datos leídos, puede realizarse la lectura para mejorar el tiempo de llevar a cabo la determinación de extracción previa por parte del nodo de almacenamiento, y para ahorrar espacio de almacenamiento de la tabla de registros de lectura; si el cómputo de los bloques de datos en la tabla de registros de lectura es mayor que el umbral fijado de bloques de datos leídos, un bloque de datos leído previamente se borra según una indicación de tiempo de modo que el cómputo de los bloques de datos en la tabla de registros sea igual al umbral fijado de bloques de datos leídos.

Debe observarse que para garantizar el acuerdo de un contenido con un bloque de datos en un disco, si se reescribe un bloque de datos que va a extraerse previamente presente en una memoria intermedia, el mismo bloque de datos del disco tiene que reescribirse de manera síncrona; asimismo, si un bloque de datos del disco se reescribe, el mismo bloque de datos que va a extraerse previamente presente en la memoria intermedia tiene que reescribirse de manera síncrona. Puede usarse una determinada estructura de datos para registrar una relación entre el bloque de datos que va a extraerse previamente presente en la memoria intermedia y el bloque de datos del disco, de manera que cuando el bloque de datos del disco cambia, el bloque de datos que va a extraerse previamente puede modificarse de manera síncrona.

Los expertos en la técnica deben entender que todos o una parte de los procesos del procedimiento según las formas de realización pueden llevarse a cabo mediante un hardware pertinente siguiendo las instrucciones de un programa informático. El programa puede estar almacenado en un medio de almacenamiento legible por ordenador. Cuando el programa se ejecuta, se lleva a cabo el proceso del procedimiento según las formas de realización. El medio de almacenamiento puede ser un disco, una memoria de solo lectura (ROM), una memoria de acceso aleatorio (RAM), etc.

La FIG. 5 es un diagrama estructural y esquemático de un ejemplo de un nodo de almacenamiento de un sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en la presente invención. Como se muestra en la FIG. 5, el aparato incluye un módulo de análisis 11, un módulo de notificación 12 y un módulo de extracción previa 13.

El módulo de análisis 11 está configurado para recibir una solicitud de lectura enviada por un cliente, y para determinar un bloque de datos que va a extraerse previamente y un nodo de almacenamiento donde reside el bloque de datos que va a extraerse previamente según un bloque de datos leído y un umbral fijado de bloques de datos que van a extraerse previamente.

El módulo de notificación 12 está configurado para enviar una solicitud de extracción previa al nodo de almacenamiento, donde la solicitud de extracción previa incluye información de identificación del bloque de datos que va a extraerse previamente, la información de identificación se usa para identificar de manera única el bloque de datos que va a extraerse previamente, y al menos un bloque de datos es el bloque de datos que va a extraerse previamente.

El módulo de extracción previa 13 está configurado para leer de un disco el bloque de datos que va a extraerse previamente determinado por el módulo de análisis 11 según la solicitud de extracción previa, y para almacenar en una memoria intermedia local el bloque de datos que va a extraerse previamente, de modo que el cliente lee de la memoria intermedia local el bloque de datos que va a extraerse previamente.

Según el nodo de almacenamiento del sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en este ejemplo, el bloque de datos que va a extraerse previamente puede determinarse según el bloque de datos leído y el umbral fijado de bloques de datos que van a extraerse previamente, y después el bloque de datos que va a extraerse previamente presente en el disco se descarga a la memoria intermedia, de modo que cuando la solicitud de lectura se recibe desde el cliente, el bloque de datos que va a extraerse previamente puede leerse directamente desde la memoria intermedia, mejorándose así la eficacia del acceso de datos para cada nodo de almacenamiento del sistema y ahorrándose recursos de sistema.

La FIG. 6 es un diagrama estructural y esquemático de una forma de realización de un nodo de almacenamiento de un sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en la presente invención. Como se muestra en la FIG. 6, el aparato incluye un módulo de análisis 11, un módulo de notificación 12 y un módulo de extracción previa 13.

En función de la forma de realización anterior, el nodo de almacenamiento puede incluir además:

un módulo de registro 14, configurado para registrar el bloque de datos leído y una indicación de tiempo del mismo en una tabla de registros de lectura, donde la indicación de tiempo se usa para identificar el tiempo de lectura del bloque de datos.

La solicitud de extracción previa enviada por el módulo de notificación 12 incluye además la tabla de registros de lectura.

Además, el nodo de almacenamiento puede incluir además:

5 un módulo de determinación 15, configurado para determinar si el bloque de datos que va a extraerse previamente tiene que adquirirse previamente según un cómputo del bloque de datos leído y una indicación de tiempo con el tiempo de lectura más reciente de la tabla de registros de lectura registrado por el módulo de registro 14.

10 Si el módulo de determinación 15 determina que el cómputo de los bloques de datos leídos es mayor o igual que un umbral fijado de bloques de datos leídos, y la diferencia entre el tiempo de lectura previa actual y el tiempo de lectura más reciente mostrado por la indicación de tiempo es inferior o igual al tiempo fijado, el módulo de extracción previa 13 lee del disco el bloque de datos que va a extraerse previamente y almacena en la memoria intermedia local el bloque de datos que va a extraerse previamente.

15 Además, el nodo de almacenamiento puede incluir además:

un módulo de actualización 16, configurado para borrar un bloque de datos leído previamente según una indicación de tiempo si el cómputo del bloque de datos registrado en la tabla de registros de lectura por el módulo de registro 14 es mayor que el umbral fijado de bloques de datos leídos, de modo que el cómputo del bloque de datos registrado en la tabla de registros es igual al umbral fijado de bloques de datos leídos.

20 Además, el módulo de análisis 11 puede configurarse específicamente para:

25 determinar un cómputo del bloque de datos que va a extraerse previamente según el umbral fijado de bloques de datos que van a extraerse previamente;
 determinar la información de identificación del bloque de datos que va a extraerse previamente según la información de identificación del bloque de datos leído, donde la información de identificación es un prefijo y un número, el prefijo es una identificación única de datos no procesados a los que pertenece el bloque de datos, y el número identifica la ubicación del bloque de datos en los datos no procesados; y
 30 llevar a cabo una operación *hash* en el bloque de datos que va a extraerse previamente según el prefijo y el número del bloque de datos que va a extraerse previamente, con el fin de determinar el nodo de almacenamiento del bloque de datos que va a extraerse previamente.

Además, el módulo de extracción previa 13 puede estar configurado además específicamente:

35 para determinar si el bloque de datos que va a extraerse previamente existe en la memoria intermedia local según la información de identificación; y
 si el bloque de datos que va a extraerse previamente no existe en la memoria intermedia local, para leer del disco el bloque de datos que va a extraerse previamente según la información de identificación, y para almacenar en la memoria intermedia local el bloque de datos que va a extraerse previamente, de modo que el
 40 cliente lee de la memoria intermedia local el bloque de datos que va a extraerse previamente.

El aparato incluye además:

45 un módulo de reescritura 17, configurado para reescribir en la memoria intermedia local el bloque de datos que va a extraerse previamente, y para reescribir de manera síncrona el mismo bloque de datos en el disco; y configurado además para reescribir el bloque de datos en el disco, y para reescribir de manera síncrona en la memoria intermedia local el mismo bloque de datos que va a extraerse previamente.

50 El nodo de almacenamiento de un sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en la FIG. 5 y en la FIG. 6 puede ser cualquier nodo de almacenamiento de un sistema de almacenamiento DHT, y el hardware de almacenamiento mostrado en la FIG. 5 y en la FIG. 6 incluye, por ejemplo, la memoria intermedia, el disco, una CPU, una placa maestra y una tarjeta de red. En la descripción de las formas de realización anteriores puede haberse hecho referencia a operaciones específicas realizadas por el nodo de almacenamiento para ejecutar un procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT).
 55 La FIG. 7 muestra un proceso principal en el que un nodo de almacenamiento de un sistema de almacenamiento de tabla *hash* distribuida (DHT) procesa una solicitud de lectura proporcionada en una forma de realización de la presente invención. La FIG. 7 no muestra todos los módulos del nodo de almacenamiento ni todas las operaciones relevantes del mismo, y los detalles de un proceso para ejecutar, por medio de un nodo de almacenamiento, una operación de extracción previa pueden encontrarse en los procedimientos de las anteriores formas de realización.
 60 Haciendo referencia a la FIG. 7, el proceso incluye específicamente:

S401: Un cliente solicita a un nodo de almacenamiento que lea un bloque de datos D1.

65 Debe observarse que puesto que lo que se proporciona con la presente invención es un procedimiento de extracción previa de bloques de datos para un sistema de almacenamiento DHT, un módulo con una función pertinente para ejecutar una operación de extracción previa se describe principalmente en las formas de realización de los nodos de

almacenamiento de un sistema de almacenamiento DHT proporcionado en la presente invención, y un módulo con una función pertinente para ejecutar una solicitud de lectura no se describe de manera específica. Sin embargo, el módulo para ejecutar la operación de extracción previa y el módulo para ejecutar la operación de lectura pueden ser realmente la misma unidad de funcionamiento del nodo de almacenamiento. Para facilitar la descripción, un módulo de extracción previa 13 proporcionado en la forma de realización de la presente invención puede servir como un módulo de ejecución para ejecutar la operación de extracción previa y la operación de lectura.

Por lo tanto, en S401, un módulo que recibe una solicitud para leer el bloque de datos D1 enviada por el cliente es el módulo de extracción previa 13.

S402: El módulo de extracción previa 13 del nodo de almacenamiento consulta D1 en una memoria intermedia y determina si el bloque de datos D1 se ha adquirido previamente.

S403: Si el bloque de datos D1 se ha adquirido previamente, el módulo de extracción previa 13 adquiere directamente de la memoria intermedia el bloque de datos D1 y devuelve el bloque de datos D1 al cliente.

S404: Si el bloque de datos D1 no se ha adquirido previamente, el módulo de extracción previa 13 lee el bloque de datos D1 de un disco.

S405: Devolver el bloque de datos D1 al cliente.

S406: Un módulo de registro 14 registra <D1> (incluyendo la indicación de tiempo de D1) en un registro de lectura.

S407: Un módulo de extracción previa 13 notifica el bloque de datos leído a un módulo de análisis 11.

S408: Después, el módulo de análisis 11 analiza el cómputo de bloques de datos que tienen que adquirirse previamente y un nodo de almacenamiento; si un bloque de datos que va a extraerse previamente está ubicado en el nodo de almacenamiento local, se llevan a cabo las etapas S407 y S408; y si el bloque de datos que va a extraerse previamente está ubicado en otro nodo de almacenamiento, se lleva a cabo la etapa S409.

S409: Un módulo de notificación 12 envía una solicitud de extracción previa al módulo de extracción previa local 13.

S410: Un módulo de determinación 15 determina si adquirir previamente el bloque de datos que va a extraerse previamente; y si es así, se lleva a cabo la etapa S411.

S411: El módulo de extracción previa 13 lee el bloque de datos del disco local y lo almacena en la memoria intermedia.

S412: El módulo de notificación 12 puede enviar una solicitud de extracción previa a otro nodo de almacenamiento.

En esta forma de realización, solo se describen la operación de lectura realizada después de que algún nodo de almacenamiento de un sistema de almacenamiento DHT reciba la solicitud de lectura enviada por el cliente y el proceso principal para iniciar la operación de extracción previa, y los detalles de un proceso para llevar a cabo, por medio del nodo de almacenamiento, la operación de extracción previa pueden encontrarse en los procedimientos de las formas de realización anteriores, los cuales no se repiten de nuevo en el presente documento.

La FIG. 8 muestra un proceso principal llevado a cabo por un nodo de almacenamiento de un sistema de almacenamiento de tabla *hash* distribuida (DHT) tras recibir una solicitud de extracción previa proporcionada en una forma de realización de la presente invención. La FIG. 8 no muestra todos los módulos del nodo de almacenamiento ni todas las operaciones relevantes del mismo, y los detalles de un proceso para ejecutar, por medio del nodo de almacenamiento, una operación de extracción previa pueden encontrarse en los procedimientos de las anteriores formas de realización. Como se muestra en la FIG. 8, el proceso incluye específicamente:

S501: Un módulo de extracción previa 13 del nodo de almacenamiento recibe una solicitud para leer previamente Dn enviada por otro nodo de almacenamiento, donde la solicitud transporta el registro de lectura de datos no procesados.

S502: El módulo de extracción previa 13 determina si el bloque de datos Dn está ubicado en una memoria intermedia; si el bloque de datos Dn está en la memoria intermedia, esto indica que se ha llevado a cabo una operación de extracción previa en el bloque de datos, y la operación de extracción previa detiene su ejecución; en caso contrario, se lleva a cabo la etapa S503.

S503: El módulo de extracción previa 13 suministra el registro de lectura de los datos no procesados a un módulo de determinación 15.

S504: El módulo de determinación 15 determina si es necesario descargar Dn desde un disco local a una memoria intermedia; y, si es necesario, se lleva a cabo la etapa S505.

5 S505: El módulo de determinación 15 suministra una instrucción de ejecución de extracción previa al módulo de extracción previa 13.

S506: El módulo de extracción previa 13 lee Dn desde el disco local.

10 S507: El módulo de extracción previa 13 almacena el Dn leído en la memoria intermedia local.

La presente invención proporciona además una forma de realización de un sistema de almacenamiento de tabla *hash* distribuida (DHT) y el sistema incluye múltiples nodos de almacenamiento proporcionados en las formas de realización de la presente invención. Diagramas estructurales del sistema pueden observarse en la FIG. 5 y en la FIG. 6.

15 En una forma de realización a modo de ejemplo, el sistema de almacenamiento DHT puede incluir un primer nodo de almacenamiento y un segundo nodo de almacenamiento.

20 El primer nodo de almacenamiento está configurado para recibir una solicitud de lectura enviada por un cliente, determinar un bloque de datos que va a extraerse previamente y un segundo nodo de almacenamiento donde reside el bloque de datos que va a extraerse previamente según un bloque de datos de lectura y un umbral fijado de bloques de datos que van a extraerse previamente, y enviar una solicitud de extracción previa al segundo nodo de almacenamiento, donde el primer nodo de almacenamiento y el segundo nodo de almacenamiento son el mismo o diferentes nodos de almacenamiento, la solicitud de extracción previa incluye información de identificación del bloque de datos que va a extraerse previamente, y la información de identificación se usa para identificar de manera

25 única el bloque de datos que va a extraerse previamente.

30 El segundo nodo de almacenamiento está configurado para leer de un disco el bloque de datos que va a extraerse previamente según la solicitud de extracción previa enviada por el primer nodo de almacenamiento, y para almacenar en una memoria intermedia local el bloque de datos que va a extraerse previamente, de modo que el cliente lee el bloque de datos que va a extraerse previamente de la memoria intermedia local del segundo nodo de almacenamiento.

35 Además, el primer nodo de almacenamiento puede estar configurado además para registrar el bloque de datos leído y una indicación de tiempo del mismo en una tabla de registros de lectura, donde la indicación de tiempo se usa para identificar el tiempo de lectura del bloque de datos, y la tabla de registros de lectura corresponde al mismo segmento de datos no procesados.

40 La solicitud de extracción previa enviada por el primer nodo de almacenamiento al segundo nodo de almacenamiento incluye además la tabla de registros de lectura.

45 Además, el segundo nodo de almacenamiento puede estar configurado además para determinar si el bloque de datos que va a extraerse previamente tiene que adquirirse previamente según el cómputo del bloque de datos leído y una indicación de tiempo con el tiempo de lectura más reciente de la tabla de registros de lectura.

50 Si el cómputo del bloque de datos leído es mayor o igual que un umbral fijado de bloques de datos leídos, y la diferencia entre el tiempo de lectura previa actual y el tiempo de lectura más reciente mostrado por la indicación de tiempo es inferior o igual al tiempo fijado, el bloque de datos que va a extraerse previamente se lee del disco, y el bloque de datos que va a extraerse previamente se almacena en la memoria intermedia local.

Además, el primer nodo de almacenamiento puede estar configurado específicamente para determinar el cómputo del bloque de datos que va a extraerse previamente según el umbral fijado de bloques de datos que van a extraerse previamente.

55 La información de identificación del bloque de datos que va a extraerse previamente se determina según la información de identificación del bloque de datos leído, donde la información de identificación es un prefijo y un número, el prefijo es una identificación única de los datos no procesados a los que pertenece el bloque de datos, y el número identifica la ubicación del bloque de datos en los datos no procesados.

60 La operación *hash* se lleva a cabo en el bloque de datos que va a extraerse previamente según el prefijo y el número del bloque de datos que va a extraerse previamente, para determinar el bloque de datos que va a extraerse previamente y el segundo nodo de almacenamiento donde reside el bloque de datos que va a extraerse previamente, y la solicitud de extracción previa se envía al segundo nodo de almacenamiento.

65 En el sistema de almacenamiento de tabla *hash* distribuida (DHT) proporcionado en esta forma de realización, cada nodo de almacenamiento interactúa para ejecutar un proceso de una operación de extracción previa de datos, al que

puede haberse hecho referencia en el procedimiento de extracción previa de datos para una tabla *hash* distribuida (DHT) proporcionado en la presente invención, y no se repite de nuevo en el presente documento.

5 Finalmente, debe observarse que las anteriores formas de realización se proporcionan simplemente para describir las soluciones técnicas de la presente invención, pero no pretenden limitar la presente invención.

REIVINDICACIONES

1. Un procedimiento de extracción previa de datos para un sistema de almacenamiento de tabla *hash* distribuida (DHT), que comprende:

5 recibir (S101), por medio de un primer nodo de almacenamiento, una solicitud de lectura enviada por un cliente;
 registrar, por medio del primer nodo de almacenamiento, un bloque de datos leído y una indicación de tiempo del bloque de datos leído en una tabla de registros de lectura, donde la indicación de tiempo se usa para
 10 identificar el tiempo de lectura del bloque de datos leído, y la tabla de registros de lectura corresponde a un mismo segmento de datos no procesados; y
 determinar (S102), por medio de un primer nodo de almacenamiento, un bloque de datos que va a extraerse previamente y un segundo nodo de almacenamiento donde reside el bloque de datos que va a extraerse
 15 previamente según el bloque de datos leído y un umbral fijado de bloques de datos que van a extraerse previamente, y enviar una solicitud de extracción previa al segundo nodo de almacenamiento, donde la solicitud de extracción previa comprende información de identificación del bloque de datos que va a extraerse
 previamente y la tabla de registros de lectura, y la información de identificación se usa para identificar de manera única el bloque de datos que va a extraerse previamente;
 20 determinar, por medio del segundo de almacenamiento, según un cómputo del bloque de datos leído y la indicación de tiempo con el tiempo de lectura más reciente de la tabla de registros de lectura, si es necesario adquirir previamente el bloque de datos que va a extraerse previamente; y
 leer (S103) de un disco, por medio del segundo nodo de almacenamiento, el bloque de datos que va a extraerse previamente según la solicitud de extracción previa, y almacenar en una memoria intermedia local
 25 el bloque de datos que va a extraerse previamente si el cómputo del bloque de datos leído es mayor o igual que un umbral fijado de bloques de datos leídos y la diferencia entre el tiempo de lectura previa actual y el tiempo de lectura más reciente mostrado por la indicación de tiempo es inferior o igual al tiempo fijado, de modo que el cliente lee de la memoria intermedia local del segundo nodo de almacenamiento el bloque de datos que va a extraerse previamente.

30 2. El procedimiento según la reivindicación 1, que comprende además:

si el cómputo del bloque de datos en la tabla de registros de lectura es mayor que el umbral fijado de bloques de datos leídos, borrar un bloque de datos leído previamente según una indicación de tiempo, de modo que el
 35 cómputo del bloque de datos en la tabla de registros es igual al umbral fijado de bloques de datos leídos.

3. El procedimiento según una cualquiera de las reivindicaciones 1 a 2, en el que determinar, por medio del primer
 40 nodo de almacenamiento, el bloque de datos que va a extraerse previamente y el segundo nodo de almacenamiento donde reside el bloque de datos que va a extraerse previamente según el bloque de datos leído y el umbral fijado de bloques de datos que van a extraerse previamente, y enviar la solicitud de extracción previa al segundo nodo de almacenamiento comprenden:

determinar, por medio del primer nodo de almacenamiento, el cómputo del bloque de datos que va a extraerse previamente según el umbral fijado de bloques de datos que van a extraerse previamente;
 45 determinar, por medio del primer nodo de almacenamiento, la información de identificación del bloque de datos que va a extraerse previamente según la información de identificación del bloque de datos leído, donde la información de identificación es un prefijo y un número, el prefijo es una identificación única de los datos no procesados a los que pertenece el bloque de datos leído, y el número identifica la ubicación del bloque de datos en los datos no procesados; y
 50 llevar a cabo una operación *hash* en el bloque de datos que va a extraerse previamente según el prefijo y el número del bloque de datos que va a extraerse previamente, para determinar el bloque de datos que va a extraerse previamente y el segundo nodo de almacenamiento donde reside el bloque de datos que va a extraerse previamente, y enviar la solicitud de extracción previa al segundo nodo de almacenamiento.

4. El procedimiento según la reivindicación 3, en el que leer del disco, por medio del segundo nodo de
 55 almacenamiento, el bloque de datos que va a extraerse previamente según la solicitud de extracción previa, y almacenar en la memoria intermedia local el bloque de datos que va a extraerse previamente, es específicamente:

determinar, por medio del segundo nodo de almacenamiento, si el bloque de datos que va a extraerse
 60 previamente existe en la memoria intermedia local según la información de identificación; y
 si el bloque de datos que va a extraerse previamente no existe en la memoria intermedia local, leer del disco, por medio del segundo nodo de almacenamiento, el bloque de datos que va a extraerse previamente según la información de identificación, y almacenar en la memoria intermedia local el bloque de datos que va a extraerse previamente.

65 5. Un sistema de almacenamiento de tabla *hash* distribuida (DHT), que comprende:

un primer nodo de almacenamiento, que está configurado para recibir una solicitud de lectura enviada por un cliente, para registrar un bloque de datos leído y una indicación de tiempo del bloque de datos leído en una tabla de registros de lectura, donde la indicación de tiempo se usa para identificar el tiempo de lectura del bloque de datos y la tabla de registros de lectura corresponde a un mismo segmento de datos no procesados, para determinar un bloque de datos que va a extraerse previamente y un segundo nodo de almacenamiento donde reside el bloque de datos que va a extraerse previamente según el bloque de datos leído y un umbral fijado de bloques de datos que van a extraerse previamente, y para enviar una solicitud de extracción previa al segundo nodo de almacenamiento, donde la solicitud de extracción previa comprende información de identificación del bloque de datos que va a extraerse previamente y la tabla de registros de lectura, y la información de identificación se usa para identificar de manera única el bloque de datos que va a extraerse previamente; y
 el segundo nodo de almacenamiento, que está configurado para determinar si es necesario adquirir previamente el bloque de datos que va a extraerse previamente según un cómputo del bloque de datos leído y la indicación de tiempo con el tiempo de lectura más reciente de la tabla de registros de lectura, para leer de un disco el bloque de datos que va a extraerse previamente según la solicitud de extracción previa, y para almacenar en una memoria intermedia local el bloque de datos que va a extraerse previamente si el cómputo del bloque de datos leído es mayor o igual que un umbral fijado de bloques de datos leídos y la diferencia entre el tiempo de lectura previa actual y el tiempo de lectura más reciente mostrado por la indicación de tiempo es inferior o igual al tiempo fijado, de modo que el cliente lee de la memoria intermedia local del segundo nodo de almacenamiento el bloque de datos que va a extraerse previamente.

6. El sistema de almacenamiento de tabla *hash* distribuida (DHT) según la reivindicación 5, en el que el segundo nodo de almacenamiento está configurado para borrar un bloque de datos leído previamente según la indicación de tiempo si el cómputo del bloque de datos de la tabla de registros de lectura es mayor que el umbral fijado de bloques de datos leídos.

7. El sistema de almacenamiento de tabla *hash* distribuida (DHT) según la reivindicación 5 o 6, en el que el primer nodo de almacenamiento está configurado además para:

determinar el cómputo del bloque de datos que va a extraerse previamente según el umbral fijado de bloques de datos que van a extraerse previamente;
 determinar la información de identificación del bloque de datos que va a extraerse previamente según la información de identificación del bloque de datos leído, donde la información de identificación es un prefijo y un número, el prefijo es una identificación única de los datos no procesados a los que pertenece el bloque de datos, y el número identifica la ubicación del bloque de datos en los datos no procesados; y
 llevar a cabo una operación *hash* en el bloque de datos que va a extraerse previamente según el prefijo y el número del bloque de datos que va a extraerse previamente, con el fin de determinar el nodo de almacenamiento del bloque de datos que va a extraerse previamente.

8. El sistema de almacenamiento de tabla *hash* distribuida (DHT) según la reivindicación 7, en el que el segundo nodo de almacenamiento está configurado además para:

determinar si el bloque de datos que va a extraerse previamente existe en la memoria intermedia local según la información de identificación; y
 si el bloque de datos que va a extraerse previamente no existe en la memoria intermedia local, leer del disco el bloque de datos que va a extraerse previamente según la información de identificación, y almacenar en la memoria intermedia local el bloque de datos que va a extraerse previamente, de modo que el cliente lee de la memoria intermedia local el bloque de datos que va a extraerse previamente.

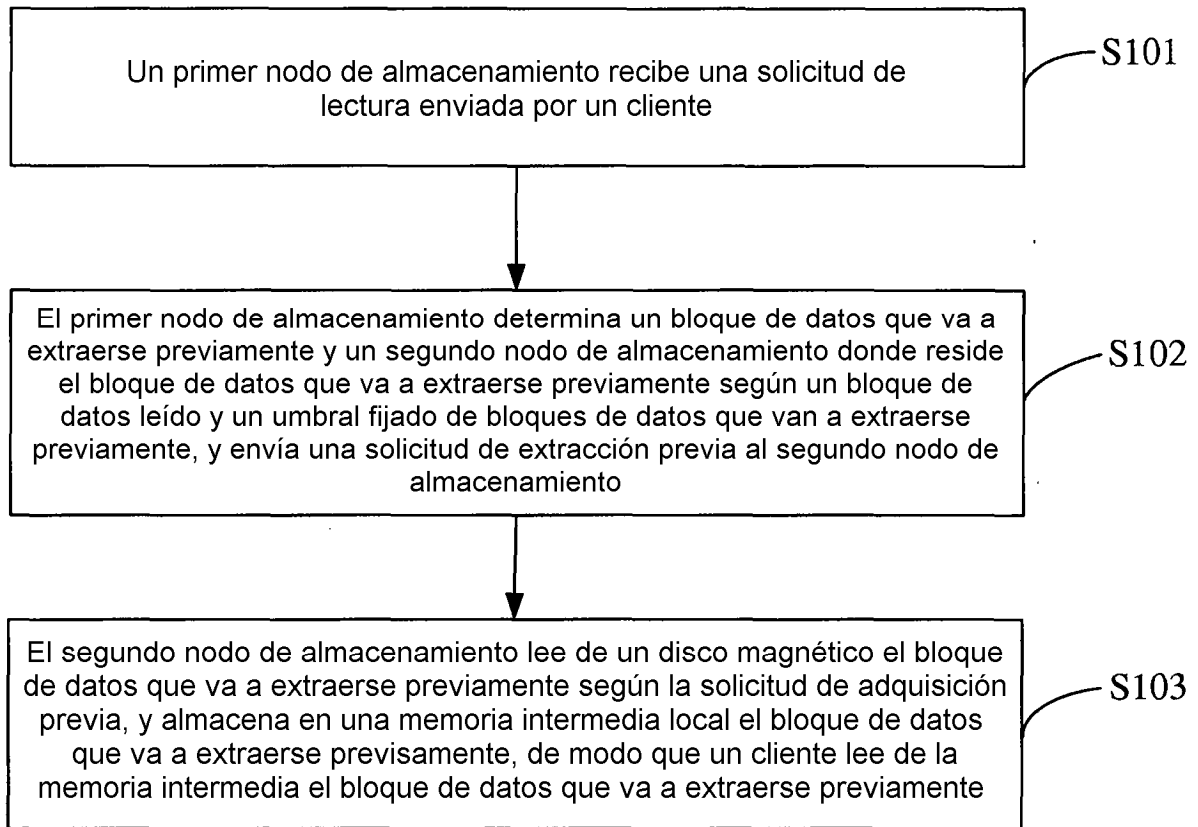


FIG. 1

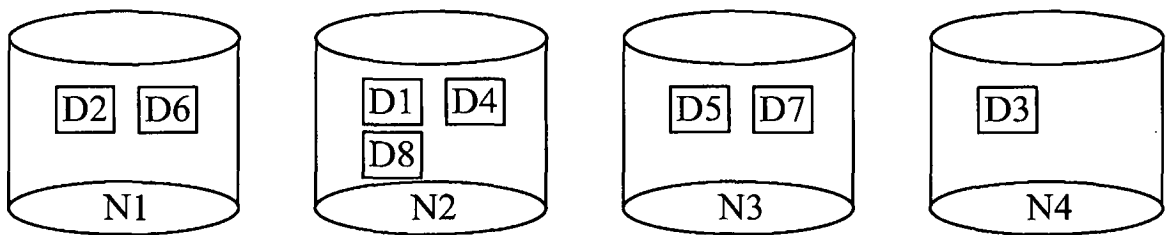


FIG. 2

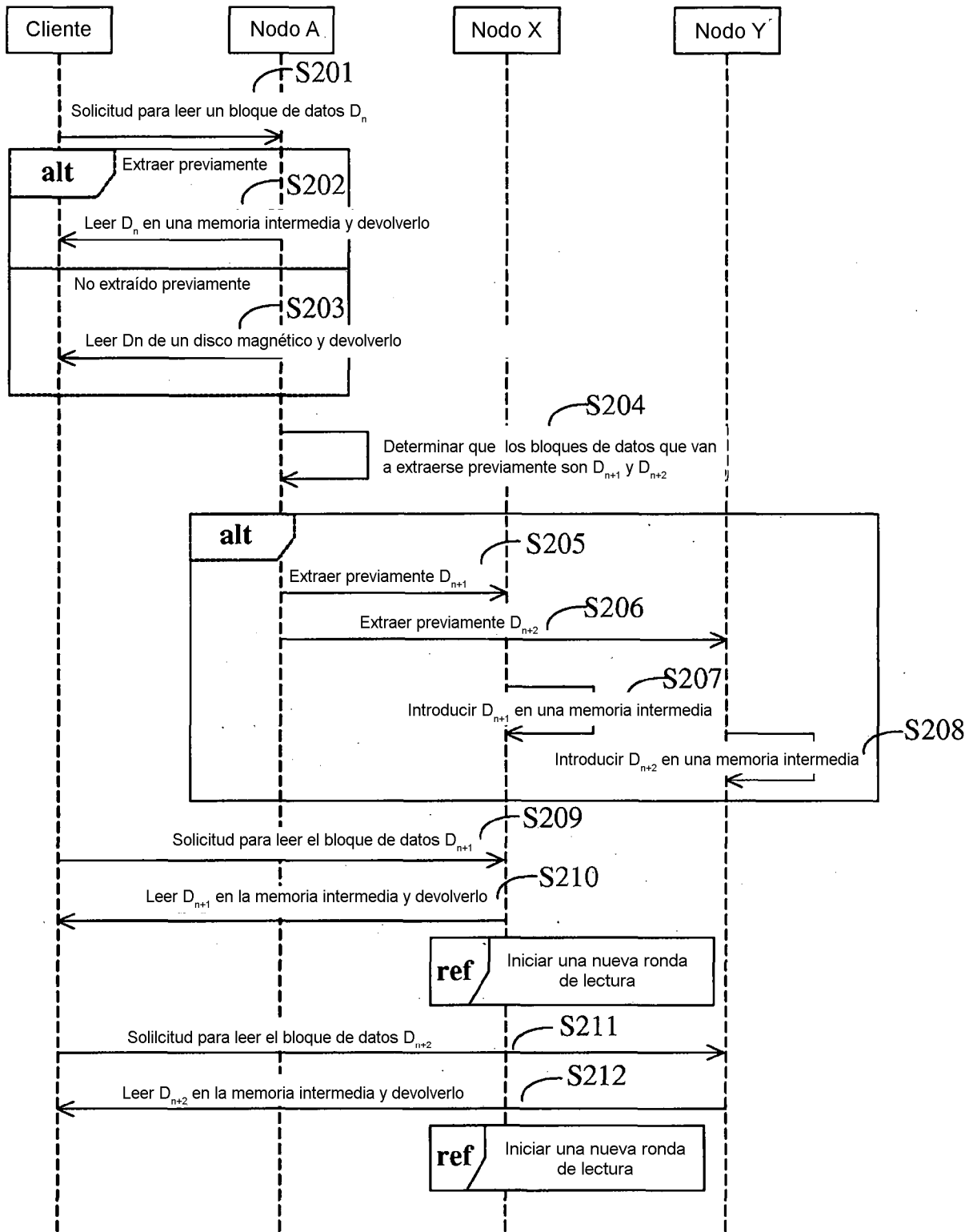


FIG. 3

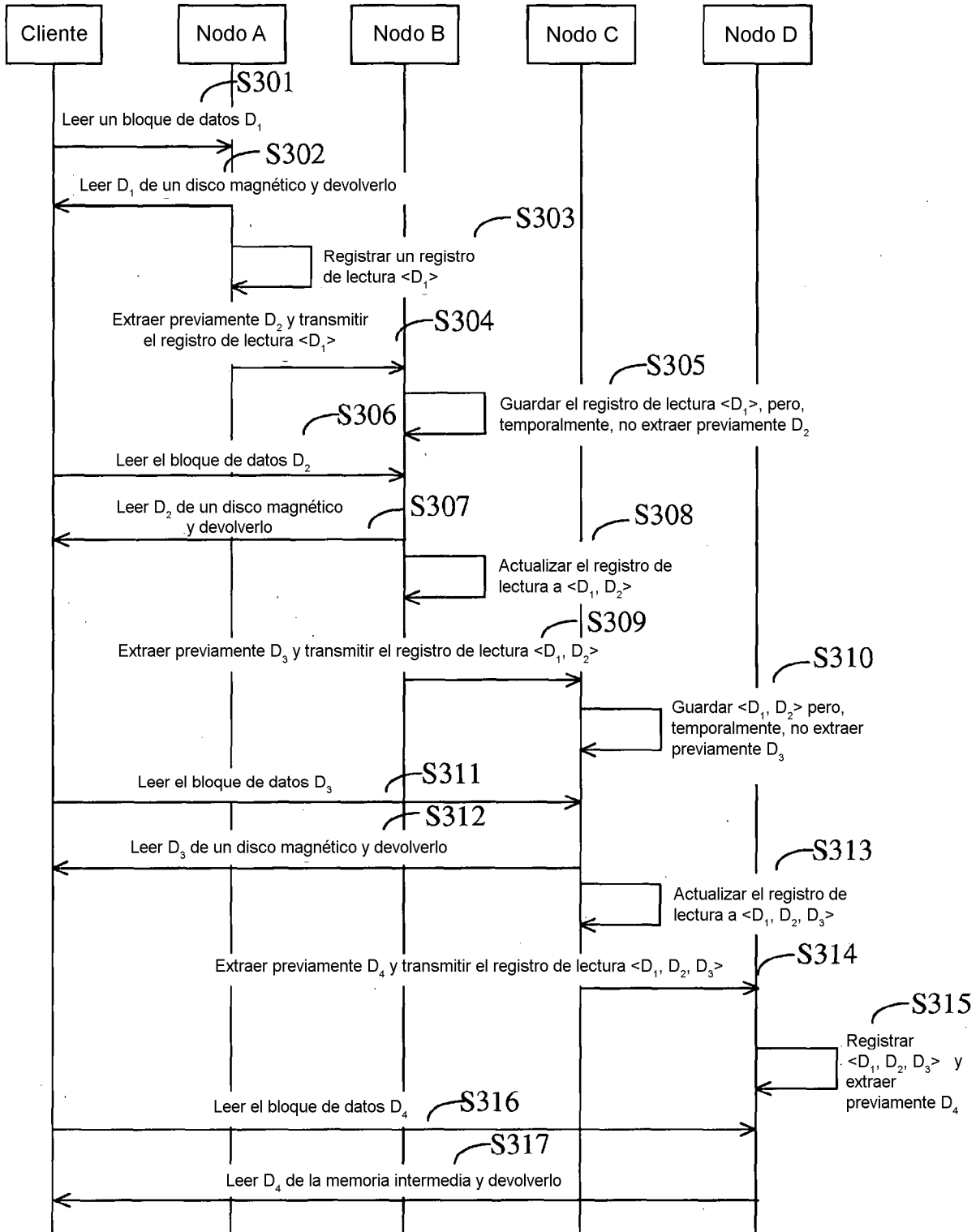


FIG. 4

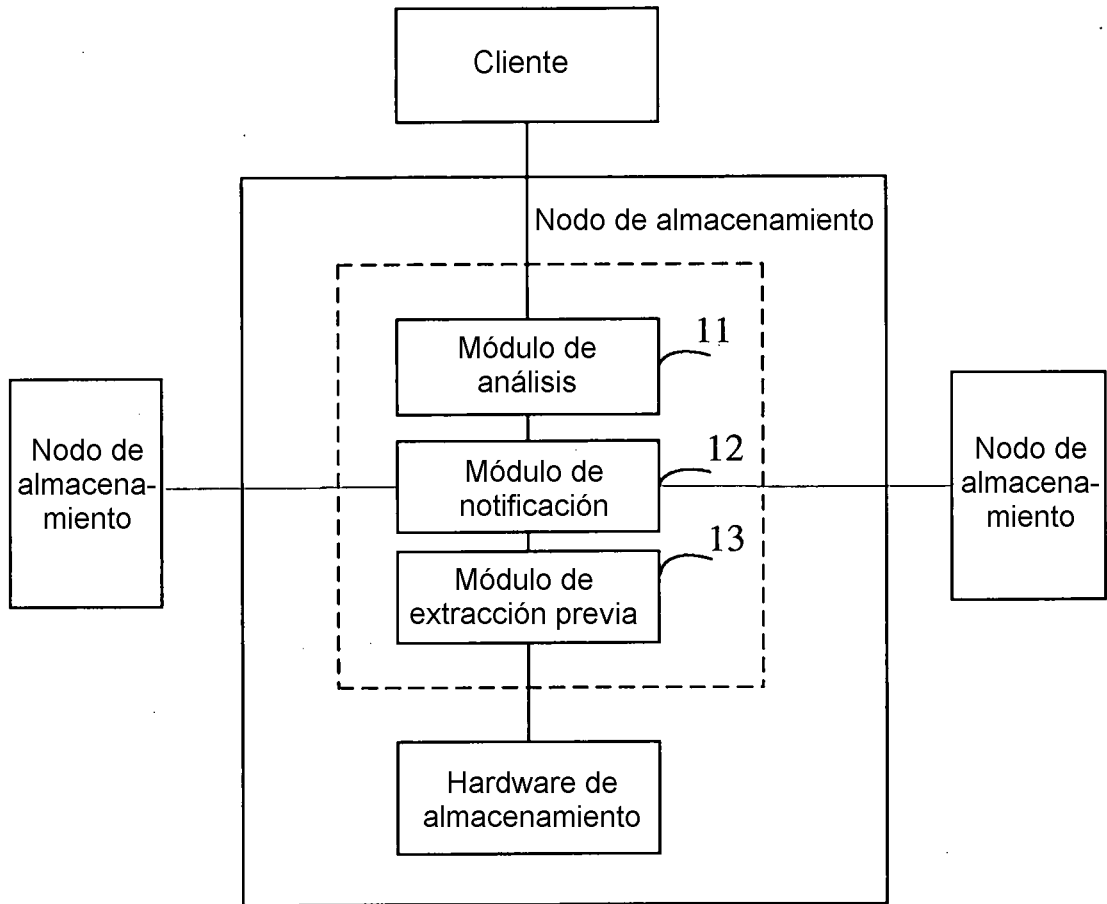


FIG. 5

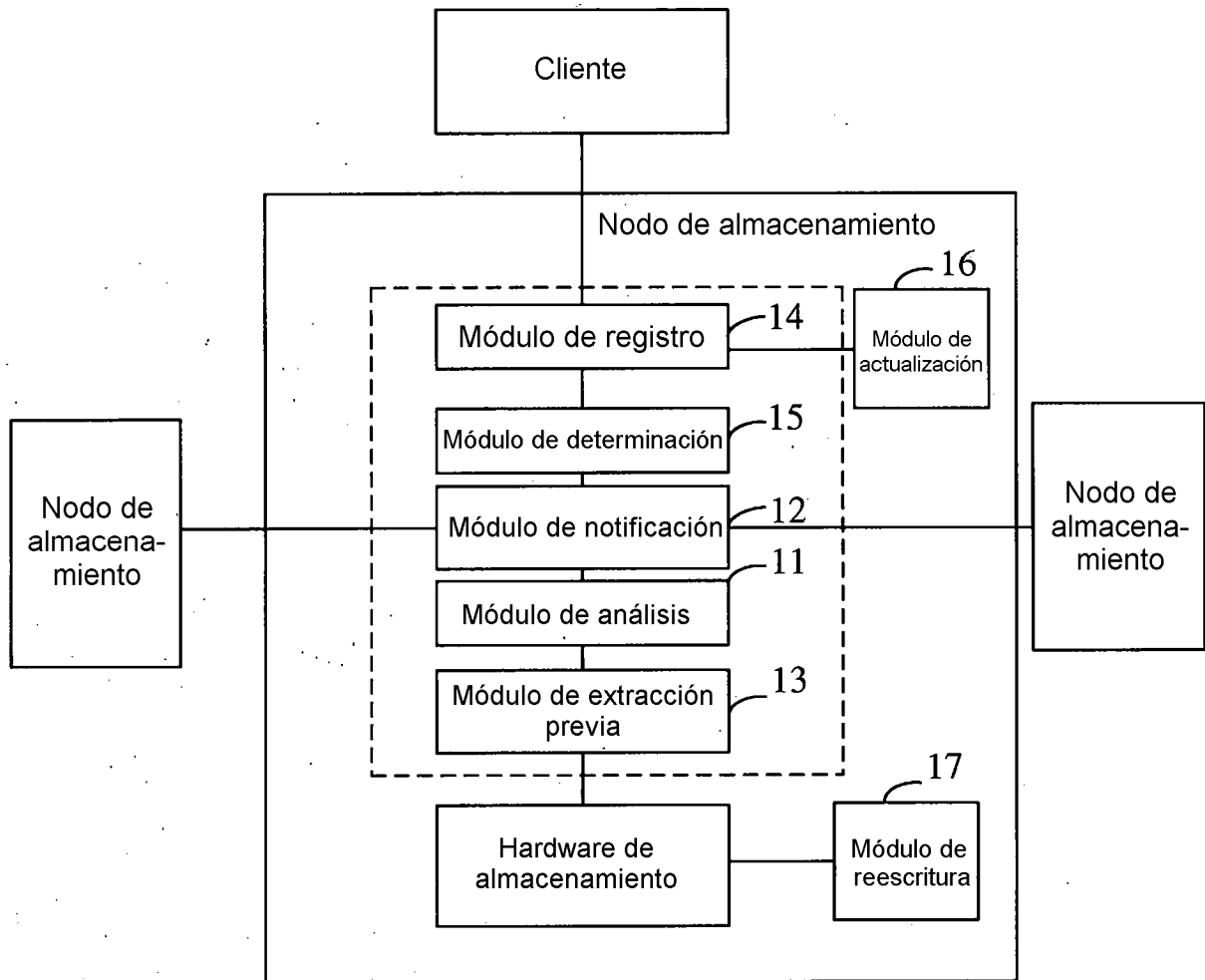


FIG. 6

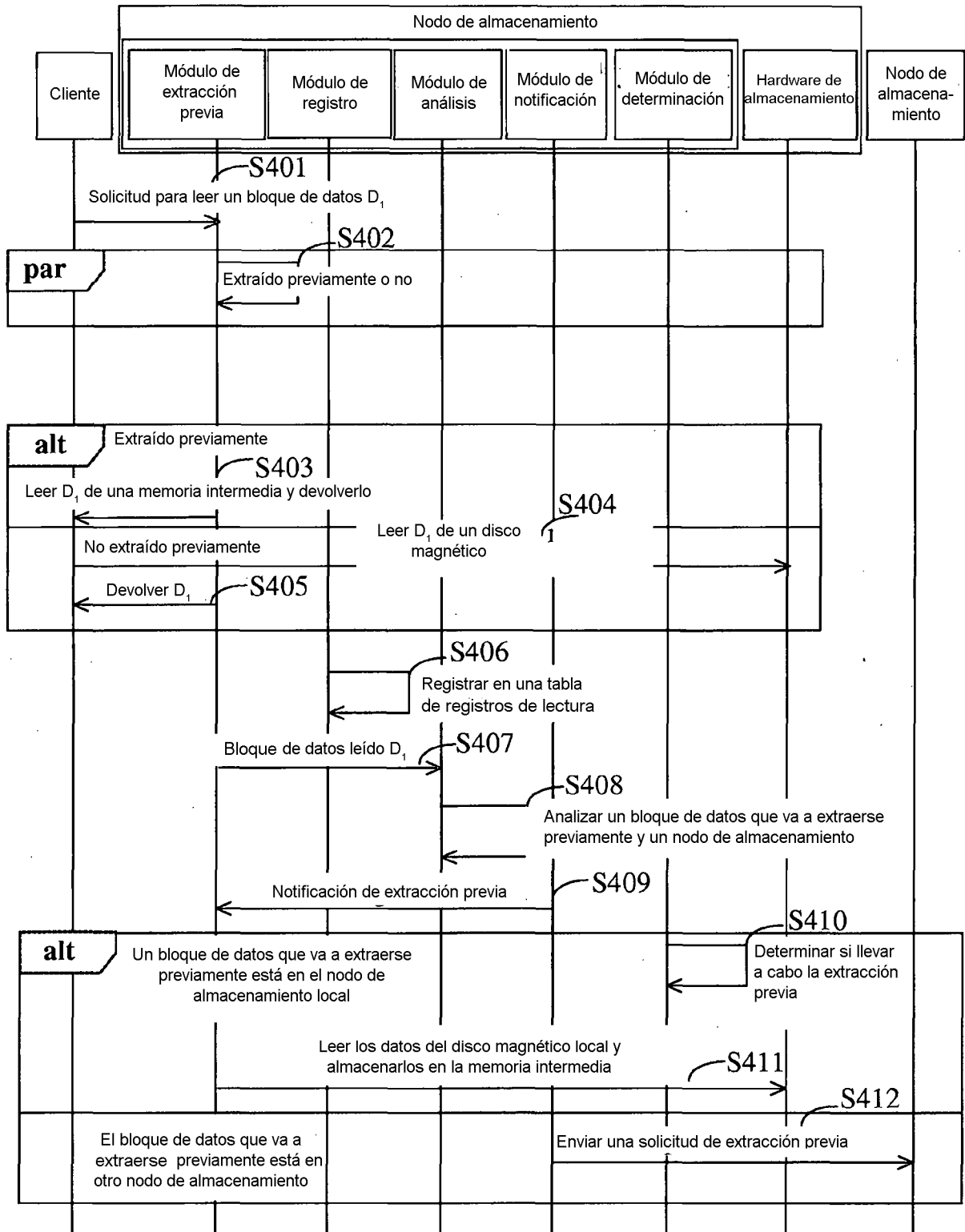


FIG. 7

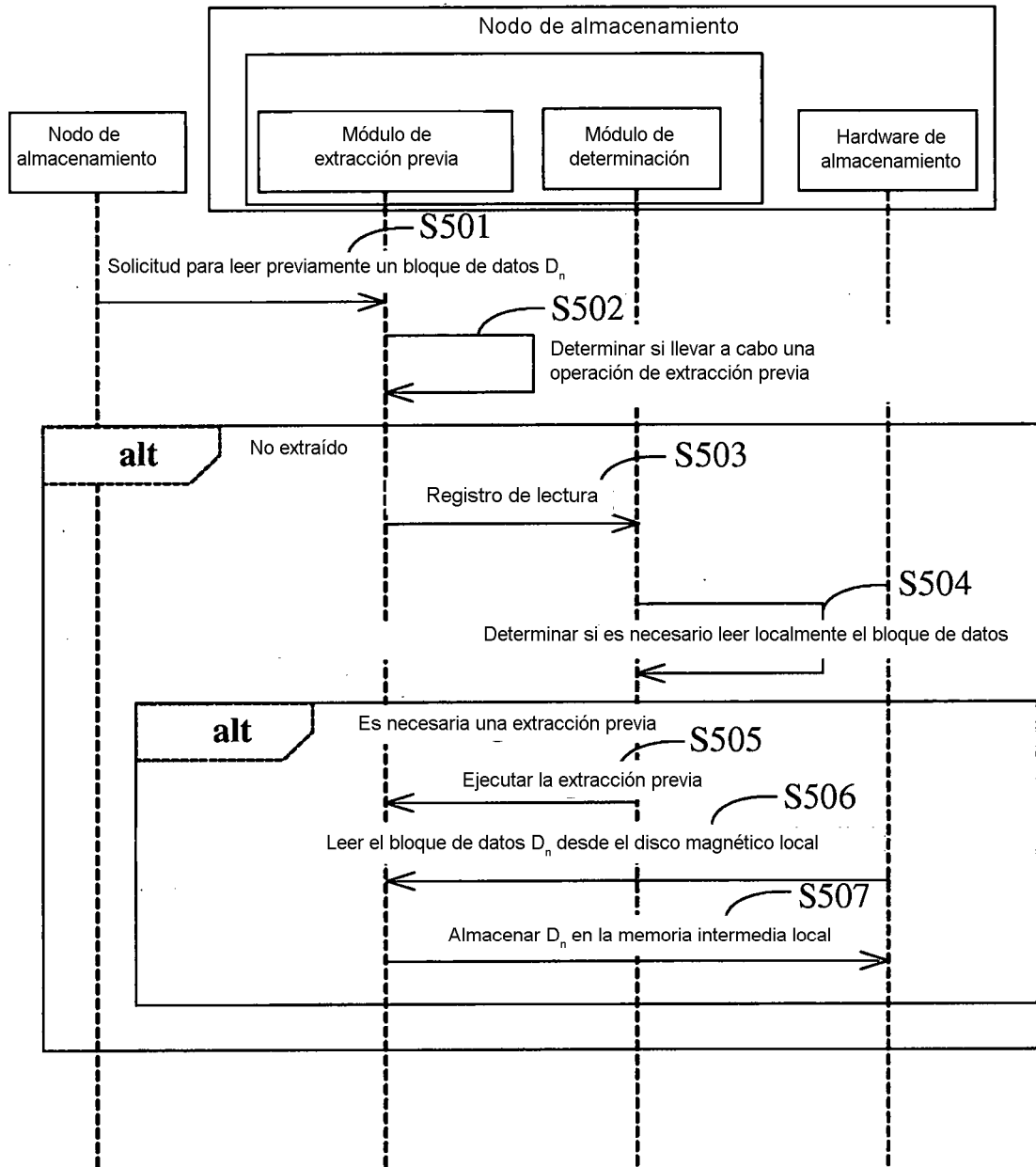


FIG. 8