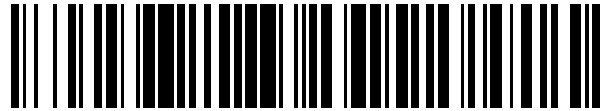


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 398 420**

51 Int. Cl.:

G06F 12/02 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **16.07.2004 E 04778511 (8)**

97 Fecha y número de publicación de la concesión europea: **21.11.2012 EP 1652094**

54 Título: **Procedimiento y aparato para mejorar el rendimiento de recuperación de espacio inutilizado usando memoria caché de la traza de pila**

30 Prioridad:

01.08.2003 US 632474

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

19.03.2013

73 Titular/es:

**INTEL CORPORATION (100.0%)
2200 MISSION COLLEGE BOULEVARD
SANTA CLARA, CA 95052, US**

72 Inventor/es:

**GANSHA, WU y
LUEH, GUEI-YUAN**

74 Agente/Representante:

CARPINTERO LÓPEZ, Mario

ES 2 398 420 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 y aparato para mejorar el rendimiento de recuperación de espacio inutilizado usando memoria caché de la traza de pila

Reserva de propiedad intelectual

5 Una porción de la divulgación del presente documento de patente contiene materiales que están sometidos a protección de propiedad intelectual. El propietario de la propiedad intelectual no tiene objeción en la reproducción de facsímiles tanto del documento de patente como de la divulgación de patente, como figura en los archivos y registros de patentes de la Oficina de Patentes y Marcas, pero, por el contrario, se reserva todos los derechos de propiedad intelectual.

10 **Antecedentes**

1.- Campo

La presente invención se refiere en general a la recuperación de espacio inutilizado, y más específicamente, a la enumeración del conjuntos de raíces en un procedimiento de recuperación de espacio inutilizado.

2.- Descripción

15 La función de recuperación de espacio inutilizado, es decir, la recuperación automática de almacenamiento informático, es encontrar objetos de datos que ya no se usan y hacer que su espacio esté disponible para su reutilización por programas en ejecución. La recuperación de espacio inutilizado es importante para evitar complicaciones innecesarias e interacciones sutiles creadas para la asignación explícita de almacenamiento, para reducir la complejidad de depuración de programas, y de este modo favorecer una programación totalmente modular y aumentar la portabilidad de aplicaciones de software. Debido a su importancia, la recuperación de espacio
20 inutilizado se está convirtiendo en parte integrante de entornos de tiempo de ejecución controlados.

El funcionamiento básico de un recuperador de espacio inutilizado puede comprender tres fases, En la primera fase, se pueden identificar todas las referencias directas a objetos de los programas actualmente en ejecución para todas las amenazadas. Estas referencias se denominan raíces, o juntas un conjunto de raíces, y un procedimiento de
25 identificación de todas tales referencias pueden denominarse enumeración del conjunto de raíces. En la segunda fase, se pueden buscar todos los objetos que se pueden alcanzar desde el conjunto de raíces ya que estos objetos se pueden usar en el futuro. Un objeto que se puede alcanzar desde cualquier referencia en el conjunto de raíces es considerado un objeto operativo; de otro modo se considera un objeto de espacio inutilizado. Un objeto que se puede alcanzar desde un objeto operativo también está operativo. El procedimiento para encontrar todos los objetos operativos que se pueden alcanzar desde el conjunto de raíces pueden denominarse como trazado de objetos (o
30 marcado y exploración). En la tercera fase, el espacio de almacenamiento de objetos de espacio inutilizados puede ser recuperado (recuperación de espacio inutilizado). Esta fase puede ser llevada a cabo bien por un recuperador de espacio inutilizado o una aplicación en ejecución (normalmente denominado mutador). En la práctica, estas tres fases, especialmente las dos últimas fases, pueden intercalarse funcionalmente o temporalmente y una técnica de recuperación puede depender en gran medida de una técnica de trazado de objetos operativos. Dependiendo de donde se produzca la enumeración del conjunto de raíces, la enumeración del conjunto de raíces se puede denominar enumeración del conjunto de raíces de registro (en lo sucesivo enumeración de registro) enumeración del conjunto de raíces de montón (en lo sucesivo enumeración de montón), o enumeración del conjunto de raíces de pila (en lo sucesivo enumeración de pila). En comparación con la enumeración de pila, las sobrecargas producidas por la enumeración del conjunto de raíces en otras zonas de almacenamiento son habitualmente pequeñas en una
35 aplicación típica y pueden ignorarse.

Cuando se ejecuta espacio de almacenamiento libre por debajo de un límite, se puede utilizar la recuperación de espacio inutilizado y se pueden suspender todas las amenazas para que de este modo la enumeración del conjunto de raíces para cada amenaza pueda empezar (para recuperación de espacio inutilizado concurrente, algunas
45 amenazas no pueden suspenderse para utilizar la enumeración del conjunto de raíces). Para la enumeración de pila para una amenaza, la trama de pila (en la pila de hilo) donde está suspendido el hilo se convierte en una trama actual a partir de la cual puede empezar la enumeración de pila. Todas las referencias vivas en la trama actual pueden ser identificadas y enumeradas. Después de enumerar la trama actual, la siguiente trama de pila (es decir, una trama de interlocutores) en una pila de llamadas se convierte en una trama actual en la que las referencias vivas pueden ser identificadas. Este procedimiento, que se denomina desenrollado, sigue hasta que todas las tramas en una cadena de llamadas son encaminadas y enumeradas.

Un mecanismo de desenrollado de pila implicado en la enumeración de pila en un recuperador de espacio inutilizado desenrolla o encamina tramas de pila de una pila de llamadas, una trama a la vez, para identificar referencias
55 actualmente activas; es decir referencias para formar un conjunto de raíces. Para algunas aplicaciones, especialmente las que tienen un gran número de hilos y una cadena de llamadas profunda por hilo, el desenrollado de pila incurre en una sobrecarga importante de tiempo de ejecución para la recuperación de espacio inutilizado. Cuando más amenas hay, y cuanto más profunda es la cadena de llamadas por hilo, mayor es la sobrecarga de

tiempo de ejecución que se puede usar. Por lo tanto, es deseable mejorar la eficiencia de enumeración de pila reduciendo la sobrecarga producida por el desenrollado de la pila.

El documento US 2003/0126352 divulga tal recuperación de espacio inutilizado de desenrollado de pila.

5 El documento US 6523141 divulga un procedimiento y un aparato para detectar e informar de fugas de memoria asociadas a un sistema operativo. Un procedimiento para identificar una sección de memoria asignada dinámicamente que no puede liberarse explícitamente incluye el procesamiento de información asociada al fallo de un sistema operativo y la identificación de un sitio de llamada que está asociado a la sección de la memoria asignada dinámicamente usando la información.

Breve descripción de los dibujos

10 Las características y ventajas de la presente invención se pondrán de manifiesto a partir de la siguiente descripción detallada de la presente invención en los que:

La figura 1 representa un esquema de alto nivel de un sistema de tiempo de ejecución controlado a modo de ejemplo usando al menos una memoria caché de la traza de pila para mejorar el rendimiento de recuperación de espacio inutilizado, según una realización de la presente invención.

15 La figura 2 es un diagrama de flujo a modo de ejemplo de un procedimiento de alto nivel en el que se usa una memoria caché de la traza de pila durante la enumeración del conjunto de raíces para la recuperación de espacio inutilizado en una sistema de tiempo de ejecución controlado, según una realización de la presente invención

20 La figura 3 es un diagrama de bloques funcional de alto nivel de un mecanismo de enumeración de pila usando una memoria caché de la traza de pila, según una realización de la presente invención.

La figura 4 es un diagrama de flujo a modo de ejemplo de un procedimiento en el que se crea una memoria caché de pila y se usa para mejorar el rendimiento de enumeración del conjunto de raíces durante la recuperación de espacio inutilizado para un hilo, según una realización de la presente invención; y

25 Las figuras 5(a) – (d) son ilustraciones esquemáticas del estado de una memoria caché de la traza de pila durante diferentes sesiones de enumeración de pila para un hilo, según una realización de la presente invención

Descripción detallada

Una realización de la presente invención es un procedimiento y un aparato para mejorar el rendimiento de enumeración del conjunto de raíces para recuperación de espacio inutilizado usando al menos una memoria caché de la traza de pila. La presente invención se puede usar para reducir la sobrecarga de enumeración de pila durante la recuperación de espacio inutilizado en aplicaciones de software con un gran número de hilos y una cadena de llamadas profunda por hilo, sin mucho coste. En algunas aplicaciones de software, una cadena de llamadas funcional es un hilo que puede ser repetitivo, es decir la relación entre llamante y llamado puede no cambiar mucho, de una sesión de recuperación de espacio inutilizado a la siguiente. De este modo una memoria caché de la traza de pila se puede usar para el hilo de almacenar información de la traza de pila, lo cual refleja la relación llamante-llamado en una cadena de llamada. La información de la traza de pila puede comprender una lista de tramas de pila, información de contexto asociada a cada trama, indicador de instrucción actual (IP), y/o información de número de línea fuente. En la primera enumeración de pila (la primera sesión de recuperación de espacio inutilizado), se puede llevar a cabo un desenrollado completo de pila y se puede crear una memoria caché de la traza de pila para almacenar información de la traza para cada trama en una pila. En la segunda o última enumeración (la segunda o última sesión de recuperación de espacio inutilizado), una parte o la totalidad del trabajo implicado en el desenrollado completo de pila se puede evitar recuperando simplemente porciones repetitivas de información de la traza de pila desde la memoria caché de la traza de pila. Asimismo, en la segunda o última enumeración de pila, la memoria caché de la traza de pila se puede modificar para alojar nuevas trazas y/o para actualizar trazas antiguas.

45 La referencia en la memoria a “una realización” de la presente invención significa que un rasgo, estructura o característica particular descrita junto con la realización está incluida en al menos una realización de la presente invención. De este modo, los aspectos de la expresión “en una realización” que aparece en varios lugares a lo largo de toda la memoria no se refieren necesariamente a la misma realización.

50 La figura 1 ilustra un esquema de alto nivel de un sistema de tiempo de ejecución controlado a modo de ejemplo usando al menos una memoria caché de la traza de pila para mejorar el rendimiento de recuperación de espacio inutilizado, según una realización de la presente invención. El sistema 100 de tiempo de ejecución controlado puede comprender una máquina de núcleo virtual (VM) 110, al menos un compilador Justo a tiempo (JIT) 120, un mecanismo 130, de enumeración del conjunto de raíces al menos una memoria caché 140 de la traza de pila y un recuperador de espacio inutilizado 150. El núcleo VM 110 es una máquina informática abstracta implementada en software en la parte superior de una plataforma de hardware y un sistema operativo. El uso de una VM hace que los programas de software sean independientes de diferentes sistemas operativos y hardware. Una VM puede denominarse Máquina Virtual de Java (JVM) para programas de Java, cualquiera puede denominarse con otros nombres tales como, por ejemplo, Infraestructura de Lenguaje Común (CLI) para programa C#. Para usar una VM,

se debe en primer lugar compilar un programa en un formato de distribución de arquitectura neutra, es decir un lenguaje intermedio tal como, por ejemplo, código de bytes para un código Java. La VM interpreta el lenguaje intermedio y ejecuta el código sobre una plataforma informática. Sin embargo, la interpretación por la VM impone típicamente una penalización de rendimiento inaceptable para la ejecución de un código de bytes debido a un gran procesamiento de sobrecarga de tiempo de ejecución. Un compilador de JIT ha sido diseñado para mejorar el rendimiento de la VM. El compilador de JIT 120 compila el lenguaje intermedio de un procedimiento dado en un código nativo de la máquina subyacente antes de llamar por primera vez al procedimiento. El código nativo del procedimiento se almacena en memoria y cualquiera de las llamadas posteriores al procedimiento puede ser gestionada por este código nativo más rápido, en lugar de por la interpretación de VM.

El mecanismo 130 de enumeración del conjunto de raíces puede identificar referencias vivas iniciales (raíces) en una pila, registro, y otras zonas de almacenamiento. El mecanismo de enumeración del conjunto de raíces puede ser una parte de la VM de núcleo 110, una parte de un recuperador de espacio inutilizado, una parte tanto de la VM de núcleo como del recuperador de espacio inutilizado. El mecanismo de enumeración del conjunto de raíces puede comprender un mecanismo 135 de enumeración de pila para identificar referencias vivas en una pila, un componente de enumeración de registro para identificar referencias vivas en registros, y otros componentes para identificar referencias vivas en otras zonas de almacenamiento. Puesto que la enumeración de pila puede producir más sobrecarga que la enumeración en registros y otras zonas, una memoria caché de la traza de pila 140 puede usarse para reducir la sobrecarga de enumeración de pila para un hilo. Una memoria caché de la traza de pila puede comprender un espacio de almacenamiento en memoria. En una realización la memoria de traza de pila puede ser implementada usando estructuras de datos dinámicos y escalables tales como, por ejemplo, una lista de enlaces.

El mecanismo de enumeración de pila 135 puede usar una memoria caché de la traza de pila para almacenar información de traza de pila para un hilo. La información de traza de pila puede comprender una lista de tramas de pila, información de contexto asociada a cada trama, indicador de instrucción actual (IP), y/o información de número de línea. Durante la primera enumeración de pila (para la primera sesión de recuperación de espacio inutilizado), el mecanismo de enumeración de pila puede crear una memoria caché de la traza de pila y almacenar en caché la información de traza de pila en la memoria caché de la traza de pila, mientras el mecanismo de enumeración de pila encamina y enumera cada trama en una pila, es decir, desenrolla la pila entera. Durante la segunda o última enumeración de pila (para la segunda o última sesión de recuperación de espacio inutilizado), el mecanismo de enumeración de pila puede en primer lugar empezar a encaminar cada trama en la pila. Para una trama, el mecanismo de enumeración de pila puede detectar si la información de traza que empieza a partir de la trama está almacenada en caché en la memoria caché de la traza de pila. Si la información de traza que empieza a partir de la trama está almacenada en caché, puede significar que el mecanismo de enumeración de pila puede usar directamente la información de traza sin desenrollar, además, la pila. Por otra parte, si la información de traza que empieza a partir de la trama no está almacenada en caché, o si ha cambiado desde la última enumeración de pila, el mecanismo de enumeración de pila puede modificar la memoria caché de la traza de pila para almacenar en caché la información de traza nueva o actualizada. Después de la segunda o última enumeración de pila, el mecanismo de enumeración de pila puede actualizar información de pila en la memoria caché de la traza de pila.

El recuperador de espacio inutilizado 150 puede comprender un mecanismo de trazado de objetos operativos 152 y un mecanismo de recuperación de espacio inutilizado 154. El mecanismo de trazado de objetos operativos 152 puede buscar un espacio de almacenamiento para encontrar todos los objetos alcanzables a partir de un conjunto de raíces. Puestos de estos objetos se pueden usar en el futuro, son considerados objetos operativos. El mecanismo de trazado de objetos operativos puede marcar estos objetos como objetos operativos y posteriormente explorar para buscar cualesquiera otros objetos operativos que estos objetos pueden alcanzar. Los objetos distintos de los objetos operativos son considerados objetos de espacio inutilizado cuyo espacio de almacenamiento puede ser recuperado por el mecanismo de recuperación de espacio inutilizado 154. En la práctica, el mecanismo de trazado de objetos operativos y el mecanismo de recuperación de espacio inutilizado pueden intercarse funcionalmente y una técnica de recuperación puede depender en gran medida de una técnica de marcado de objetos operativos. El mecanismo de recuperación de espacio inutilizado puede recuperar espacio de objetos de espacio inutilizado con o sin desplazar objetos operativos a una zona contingente en un extremo del espacio de almacenamiento. En una realización, el mecanismo de recuperación de espacio inutilizado puede ser una parte de un mutador en lugar de un recuperador de espacio inutilizado.

La figura 2 es un diagrama de flujo a modo de ejemplo de un procedimiento de nivel alto en el que una memoria caché de la traza de pila se usa durante la enumeración del conjunto de raíces para la recuperación de espacio inutilizado en un sistema de tiempo de ejecución controlado, según una realización de la presente invención. En el bloque 210, se puede iniciar una sesión de recuperación de espacio inutilizado. En el bloque 220, se puede llevar a cabo una enumeración de pila usando una memoria caché de la traza de pila. Se puede crear una memoria caché de la traza de pila durante la primera enumeración de pila para un hilo para almacenar en caché información de traza de pila cuando se lleva a cabo el desenrollado total de pila. Durante una última sesión de enumeración de pila para el hilo, una parte o la totalidad de la información de traza de pila almacenada en caché se puede recuperar sin llevar a cabo el desenrollado total de pila, si no ha cambiado desde la última sesión de enumeración de pila. Para estas tramas cuya información de traza ha cambiado, se puede llevar a cabo el desenrollado de pila, y la memoria caché de la traza de pila puede modificarse para alojar información de traza de pila actualizada. En el bloque 230, se

puede llevar a cabo enumeración del conjunto de raíces en otras zonas de almacenamiento para encontrar referencias vivas en otras zonas de almacenamiento tales como los registros. Los bloques 220 y 230 pueden realizarse de manera concurrente, o en un orden inverso sin afectar al espíritu de la presente invención. En el bloque 240, las referencias vivas obtenidas durante la enumeración de pila y la enumeración del conjunto de raíces en otras zonas de almacenamiento pueden ponerse juntas para formar un conjunto de raíces. En el bloque 250, se puede buscar un espacio de almacenamiento para marcar y explorar todos los objetos operativos alcanzables a partir de referencias en el conjunto de raíces. En el bloque 260, todos los objetos en el espacio de almacenamiento que no están marcados pueden ser considerados espacio inutilizado y su espacio se puede recuperar.

La figura 3 es un diagrama de bloques funcional de nivel alto de un mecanismo de enumeración de pila usando una memoria caché de la traza de pila, según una realización de la presente invención. El mecanismo de enumeración de pila 135 puede comprender un clasificador de tramas de pila 310, un mecanismo de almacenamiento en caché de información de traza 320, un mecanismo de desenrollado de pila 330, un mecanismo de detección de traza sin cambios 340 y un mecanismo de recuperación de traza sin cambios 350. El clasificador de tramas de pila 310 puede usar el valor de una etiqueta "almacenado en caché" de una trama de pila en una pila 370 para decidir si la trama de pila está recién generada (es decir, no almacenada en caché) o ya ha estado en una memoria caché de la traza de pila 140 (es decir, almacenada en caché). La etiqueta "almacenado en caché" puede ser configurada por un compilador cuando se inicia una llamada. Cuando la trama de pila asociada a la llamada se crea por primera vez, el compilador puede establecer que el valor de la etiqueta "almacenado en caché" sea falso. Cuando el clasificador de traza de pila clasifica una trama de pila como "no almacenado en caché", el mecanismo de desenrollado de pila 330 puede llevar a cabo el desenrollado normal de pila para esta trama (es decir, enumera referencias en esta trama y va a la siguiente trama) y guarda referencias vivas en un conjunto de raíces 360. Por consiguiente, el mecanismo de almacenamiento en caché de información de traza 320 puede almacenar en caché la información de traza de la trama en la memoria caché de traza de pila y cambiar el valor de la etiqueta "almacenada en caché" de la trama de pila de falso a verdadero, es decir, la trama de pila se convierte en almacenada en caché. El mecanismo de almacenamiento en caché de información de traza puede comprender un creador de memoria caché 325 que crea una memoria caché de la traza de pila cuando se crea lleva a cabo por primera vez la enumeración de pila.

En la segunda o última sesión de enumeración de pila. El creador de memoria caché puede modificar la memoria caché de la traza de pila para alojar tramas de pilas recién generadas y/o actualizadas. La memoria caché de la traza de pila puede comprender dos zonas: una zona de identificación para almacenar identificadores de tramas de pila y una zona de lista para almacenar referencias vivas enumeradas en cada trama de pila. En una realización, se puede representar información de traza de pila almacenada en caché por la siguiente estructura de datos

```
struct CachedStackFrame {
    <eip, esp> id; /* eip and esp represent IP and SP, respectively */
    Ref_List ref_list; /* cached enumerated references for this frame */
} cache[MAX_CALL_DEPTH];
```

donde la matriz cache[MAX_CALL_DEPT] es un objeto local de hilo, es decir, cada hilo tiene su propia memoria caché de la traza de pila.

El mecanismo de almacenamiento en caché de información de traza 320 puede comprender también un componente de identificación para identificar cada trama de pila en una memoria caché de la traza de pila con valores de un par de registros, es decir, registro indicador de instrucción (IP) y registro indicador de pila (SP). El IP puede asimismo denominarse contador de programa (PC) y puede ser suficiente para indicar localización de fuente y nombre de procedimiento de una trama de pila. El SP indica la posición de pila actual. Puesto que puede haber múltiples llamadas (con diferentes posiciones de pila) en un mismo procedimiento en la pila con el mismo IP (por ejemplo llamadas recursivas), IP solo no puede identificar una trama de pila y SP puede diferenciar tramas asociadas a estas llamadas. De este modo usando <IP, SP> puede ser deseable identificar únicamente una trama de pila.

Cuando el clasificador de trama de pila 310 clasifica una trama de pila en la pila 370 como "almacenada en caché" en la segunda o última enumeración de pila, el mecanismo de detección de traza sin cambio 340 puede buscar la memoria caché de la traza de pila de arriba abajo y detectar una trama de partida de una porción sin cambios de una traza de pila. La trama de partida de la porción sin cambios en la memoria caché de la traza de pila puede ser detectada cuando valores de <IP, SP> de una trama en la memoria caché de la traza de pila resultan ser iguales a valores de <IP, SP> de una trama actual. Cuando la trama de partida de la porción sin cambios es detectada en la memoria caché de la traza de pila, el mecanismo de recuperación de porción sin cambios 350 puede copiar referencias en la porción sin cambios directamente en el conjunto de raíces 360 sin desenrollado adicional de la pila.

La figura 4 es un diagrama de flujo a modo de ejemplo de un procedimiento en el que se crea una memoria caché de la traza de pila y se usa para mejorar el rendimiento de enumeración del conjunto de raíces durante la recuperación de espacio inutilizado para un hilo, según una realización de la presente invención. En el bloque 410, se puede iniciar una enumeración de pila. En el bloque 415, se puede adoptar una decisión de si es la primera enumeración de pila para el hilo. Si es la primera enumeración de pila para el hilo, se puede crear una memoria caché de la traza de pila para el hilo en el bloque 420 antes de evaluar una trama en una pila para el inicio de enumeración en el

bloque 425, en caso contrario, se pueden evaluar tramas en la pila (una a una) que se inician directamente en el bloque 425. En el bloque 425, se puede verificar una trama en la pila para ver si la trama está etiquetada “almacenada en caché” o “no almacenada en caché”. Si la trama está etiquetada “no almacenada en caché”, se puede llevar a cabo una enumeración de pila normal en el bloque 430, donde se puede iniciar el desenrollado de pila hasta que se alcance una trama “almacenada en caché” en la pila. En el bloque 435, información de traza acerca de tramas que han sido enumeradas en el bloque 430 se puede almacenar en caché y las tramas pueden etiquetarse como “almacenadas en caché”. La información de traza puede comprender un identificador para cada trama y referencias vivas enumeradas en cada trama. El identificador de una trama puede comprender valores de IP y SP. En el bloque 440, la pila es verificada para ver si hay algunas tramas sin enumerar. En caso de que haya, la siguiente trama en la pila puede evaluarse en el bloque 425 y el procedimiento entre el bloque 425 y el bloque 455 (que no incluye el bloque 455) puede reiterarse hasta que no se deje ninguna trama en la pila. Si la pila se encuentra como “almacenada en caché” en el bloque 425, la memoria caché de la traza de pila puede buscarse para detectar un punto de inicio para una porción sin cambios de una traza de pila en el bloque 445. Una vez que el punto de inicio de la porción sin cambios se ha detectado, se pueden copiar referencias en tramas siguiendo el punto de inicio en la memoria caché de la traza de pila en un conjunto de rices en el bloque 450, sin un desenrollado adicional de la pila. Un bloque 455, se puede obtener referencias de todas las tramas en la pila.

Las ventajas de usar una memoria caché de la traza de pila para enumeración de pila puede, además, ilustrarse comparando el procedimiento de enumeración de pila normal con el procedimiento de enumeración de pila usando una memoria de traza de pila. El anterior se ilustra mediante Pseudo código 1, y este último se ilustra mediante Pseudo Código 2.

© 2003 INTEL CORPORATION

```

1  Frame_Context context;
2  Ref_List allref = {}; /*stack enumeration result list */
3  ... ..
4  context.initialize_to_gc_suspend_point();
5  while (not finished) {
6      normalize_if_necessary (&context);
7      /* perform normal stack enumeration */
8      Ref_List rl = enumerate (&context);
9      /* copy enumerated references into result list */
10     allref.append (rl);
11     finished = unwind_to_next_frame (&context);
12 }

```

Pseudo Código 1. Enumeración de Pila Normal

En el Pseudo Código1, “Frame_Context” en la línea 1 representa una estructura de datos de una trama de llamada y contiene información tal como, por ejemplo, registros guardados datos vertidos. Antes de que se inicie la enumeración de pila para un hilo, se inicia “context” en una trama donde el hilo se suspende a través de “initilize_to_gc_suspend_point()” en la línea 4 “normalize_if_necessary()” en la línea 6 normaliza un SP de trama desde de su posición suspendida a su posición base, si fuese necesario. La enumeración de referencias en esta trama puede iniciarse a partir de la posición SP normalizada. Al final de la enumeración para esta trama, “unwind_to_next_frame()” desarrolla la pila de una trama de llamada en una cadena de llamada y recupera el contexto de llamante. El procedimiento de enumeración continúa hasta que todas las tramas en la cadena de llamada son enumeradas. Dicho de otro modo, Pseudo Código 1 ilustra un procedimiento de desenrollado completo de pila

© 2003 INTEL CORPORATION

```

1  Frame_Context context;
2  Ref_List allref = {}; /* initialize stack enumeration result list */
3  /* temporary cache to save stack frames not in cache */
4  CachedStackFrame notcached[MAX_CALL_DEPTH];
5  ... ..
6  context.initialize_to_gc_suspend_point();
7  while (not finished) {
8      if (context.in_cache == false) {
9          normalize_if_necessary (&context);
10         /* perform a normal enumeration */
11         Ref_List rl = enumerate (&context);
12         /* copy enumerated references into result list */
13         allref.append (rl);
14         /* add information of this frame to temporary cache */
15         add_to_cache (notcached, <context.eip, context.esp>, rl);
16         finished = unwind_to_next_java_frame (&context);
17     } else {
18         /* find the starting point of reusable trace in cache */
19         for (j = top; j >= 0; j--)
20             if (context.<eip, esp> == cache[j].<eip, esp>) break;
21         /* copy enumerated references in cache to result list */
22         for (; j >= 0; j--)
23             allref.append (cache[j].ref_list);
24         /* stop unwinding, and jump out of loop */
25         break;
26     }
27 }
28 /* update the cache with information in temporary cache */
29 update_cache (cache, notcached);

```

Pseudo Código 2. Enumeración de Pila Usando Memoria Caché de Traza de Pila

Pseudo Código 2 difiere de Pseudo Código 1 en la utilización de una memoria caché de la traza de pila (mostrada en las líneas 18-25) para aumentar la oportunidad de evitar una parte o la totalidad del desenrollado de pila. Por lo tanto, las sobrecargas producidas por enumeración de pila usando una memoria caché de la traza de pila puede ser inferior a las sobrecargas producidas por enumeración de pila a través del desenrollado completo de pila.

Las figuras 5(a)-(d) son ilustraciones esquemáticas del estado de una memoria caché de la traza de pila en diferentes sesiones de enumeración de pila para un hilo, según una realización de la presente invención. La figura 5(a) ilustra una pila antes de que se inicie la primera sesión de enumeración de pila. Cuando se recurre a la enumeración de pila por primera vez para un hilo, todas las tramas en la pila se etiquetan "no almacenada en caché" por un compilador. De este modo, se lleva a cabo el desenrollado completo de pila durante la primera enumeración de pila, después de lo cual se etiquetan todas las tramas con "almacenada en caché" porque se ha creado una memoria caché de la traza de pila y se han almacenado en caché referencias enumeradas para cada trama en la memoria caché de traza de pila, como se muestra en la figura 5(b). Cada trama en la memoria caché de la traza de pila tiene una zona de identificación (id) para almacenar identificadores de trama (es decir, valores de IP y SP), y una zona de lista para almacenar referencias enumeradas para cada trama. La figura 5(c) ilustra el estado de cada trama en la pila y la memoria caché de la traza de pila en una última sesión de enumeración de pila. Cuando esta última sesión de enumeración de pila se inicia, puede no ser necesario llevar a cabo el desenrollado completo de pila porque algunas tramas (por ejemplo, 4 tramas que corresponden a los procedimientos T. principal, A.a, B.b, y C.c en la figura 5(c) permanecen sin cambios en la pila y pueden recuperarse directamente de la memoria caché de la traza de pila sin llevar a cabo el desenrollado de pila. Solo información para otras tres tramas (es decir, tramas que corresponden a procedimientos G.g, H.h e I.i) no están almacenadas en caché. De este modo solo es necesario el desenrollado de pila parcial (hasta la trama que corresponde al procedimiento B.b). Adicionalmente, la memoria caché de la traza de pila se puede modificar para almacenar en caché información en tramas que se enumeran durante la enumeración de pila parcial. La figura 5(d) ilustra el estado de la pila u la memoria caché de la traza de

pila después de su última sesión de enumeración de pila.

5 Aunque la presente invención se refiere al uso de memorias caché de la traza de pila para enumeración del conjunto de raíces en una pila durante la recuperación de espacio inutilizado, el experto en la técnica apreciará fácilmente que la presente invención se puede usar para reducir sobrecargas producidas por cualquier procedimiento que implica el desenrollado de pila tal como, por ejemplo, una manipulación de excepción, detección de relación llamante-llamado, etc. Adicionalmente, la presente invención se puede usar para recuperación automática de espacio inutilizado en cualquier sistema tal como, por ejemplo, entonos de tiempo de funcionamiento controlados que ejecutan Java, C#, y/o otros lenguajes de programación.

10 Aunque una realización a modo de ejemplo de la presente invención se describe con referencia a diagramas de bloque y de flujo en las figuras 1-5(d) y Pseudo Códigos 1-2, el experto en la técnica apreciará fácilmente que muchos otros procedimientos de implementación de la presente invención se pueden usar de manera alternativa. Por ejemplo, se puede cambiar el orden de ejecución de los bloques en los diagramas de flujo o se pueden cambiar las etapas en los pseudo códigos, y/o se pueden cambiar, eliminar o combinar algunos de los bloques en los diagramas de bloques/flujo y las etapas en los pseudo códigos.

15 En la descripción anterior, se han descrito varios aspectos de la presente invención. Con fines explicativos, se han expuesto números específicos, sistemas y configuración para proporcionar una comprensión exhaustiva de la presente invención. Sin embargo, es evidente que el experto en la técnica que tiene el beneficio de esta divulgación, que la presente invención se puede poner en práctica sin los detalles específicos. En otros casos, se han omitido, simplificado, combinado o dividido características, componentes o módulos bien conocidos para no entorpecer la
20 presente memoria.

Realizaciones de la presente invención pueden implementarse en cualquier plataforma informatizada, que comprende hardware, y sistemas operativos. El hardware puede comprender un procesador, una memoria, un bus, y un concentrador E/S para periféricos. El procesador puede ejecutar un compilador para compilar cualquier software a las instrucciones específicas de procesador. El procesamiento requerido por las realizaciones se puede llevar a
25 cabo por un ordenador universal solo o en combinación con un ordenador especial. Tal procesamiento puede llevarse a cabo por una única plataforma o por una plataforma de procesamiento distribuida. Además, tal procesamiento y tal funcionalidad pueden implementarse en forma de hardware universal o en forma de software.

En realizaciones de la presente invención se implementan en software, el software puede almacenarse en un medio o dispositivo de almacenamiento (por ejemplo, unidad de disco duro, disquete, memoria de solo lectura (ROM), dispositivo de CD-ROM, memoria ultrarrápida, dispositivo de memoria, disco versátil digital (DVD) u otro dispositivo de almacenamiento) legible por un sistema de procesamiento programable universal o especial para llevar a cabo
30 los procedimientos descritos en el presente documento. Realizaciones de la invención pueden asimismo ser consideradas que se implementan como medio de almacenamiento legible por ordenador, configuradas para su uso con un sistema de procesamiento, donde el medio de almacenamiento así configurado hace que el sistema de
35 procesamiento funcione de una manera específica y predefinida para llevar a cabo las funciones descritas en el presente documento.

Aunque la presente invención se ha descrito con referencia a realizaciones ilustrativas, la presente descripción no está destinada a ser interpretada en un sentido limitativo.

REIVINDICACIONES

1.- Un procedimiento de recuperación automática de objetos de espacio inutilizado, que comprende:

5 efectuar (220) la enumeración de un conjunto de raíces en una pila de un hilo;
 obtener (240) un conjunto de raíces combinando referencias obtenidas a partir de enumeración del conjunto
 de raíces para todos los hilos;
 trazar (250) objetos operativos que se pueden alcanzar a partir de referencias en el conjunto de raíces; y
 recuperar (260) espacio de almacenamiento ocupado por objetos distintos de los objetos operativos; y
 caracterizado por:

10 la realización (220) de la enumeración del conjunto de raíces en una pila de un hilo por:

 almacenamiento en una memoria caché de la traza de pila (140) información de traza de pila de tramas de
 pila;
15 uso de la información de traza almacenada en la memoria caché de la traza de pila (140) sin desenrollado
 adicional de la pila si la información de traza que se inicia a partir de una trama se almacena en caché en la
 memoria caché de la traza de pila (140); y la modificación de la memoria de traza de pila para almacenar en
 caché la información de traza nueva o actualizada si la información de traza que se inicia a partir de una
20 trama no se almacena en caché, o si la información de traza de pila ha cambiado desde la última
 enumeración de pila

2.- El procedimiento de la reivindicación 1, en el que el conjunto de raíces comprende referencias actualmente operativas en las pilas, registros y otras zonas de almacenamiento para todos los hilos.

3.- El procedimiento de la reivindicación 1 o la reivindicación 2, en el que la ejecución (220) de la enumeración del conjunto de raíces en una pila de un hilo usando una memoria caché (140) de traza de pila comprende:

25 Determinar si una trama de pila se almacena en caché en la memoria caché (140) de la traza de pila;
 Detectar (445) una parte de la información de traza de pila que no ha cambiado desde la última
 enumeración de pila en la memoria caché (140) de traza de pila y recuperar (450) la parte no modificada a
 partir de la memoria caché (140) de la traza de pila, si la trama de pila es almacenada en caché en la
 memoria caché (140) de la traza de pila; y
30 Efectuar (430) una enumeración normal del conjunto de raíces en la pila, si la trama de pila no está
 almacenada en caché en la memoria caché (140) de la traza de pila.

4.- El procedimiento de cualquiera de las reivindicaciones 1 a 3, en el que el trazado (250) de los objetos operativos comprende el marcado de los objetos operativos y de otros objetos de exploración que se pueden alcanzar a partir de los objetos marcados.

35 5.- Un procedimiento según cualquiera de las reivindicaciones anteriores, en el que la ejecución (220) de la
 enumeración del conjunto de raíces en una pila de hilo comprende:

 Iniciar una lista de enumeración de la pila;
 Crear (420) una memoria caché (140) de la traza de pila para el hilo; y
 Ejecutar (220) una enumeración del conjunto de raíces en la pila utilizando la memoria caché (140) de traza
40 de pila para producir una lista de enumeración de pila actualizada.

6.- El procedimiento de cualquier reivindicación anterior, que comprende, además, etiquetar una trama de pila para indicar si la trama de pila se ha almacenado en caché en la memoria caché (140) de la traza de pila o si se ha generado recientemente en la pila y no ha sido almacenada en caché en la memoria caché (140) de la traza de pila.

45 7.- El procedimiento de la reivindicación 5, en el cual la lista de enumeración de pila comprende referencias
 actualmente operativas en la pila.

8.- El procedimiento de la reivindicación 5, en el cual la creación (420) de la memoria caché (140) de la traza de pila comprende:

 determinar (415) si la enumeración del conjunto de raíces en la pila se efectúa por primera vez;
 crear de manera dinámica la memoria caché (140) de la traza de pila y el almacenamiento en caché de
50 información de tramas de pila en la memoria caché (140) de la traza de pila, si la enumeración del conjunto
 de raíces en la pila se efectúa por primera vez; y
 en caso contrario, reutilizar la memoria caché (140) de la traza de pila para la enumeración del conjunto de
 raíces en la pila

9.- El procedimiento de cualquier reivindicación anterior, en el que el almacenamiento en caché de información de

tramas de pila comprende añadir un identificador a cada trama utilizando los valores de un registro de indicador de instrucciones y de un registro de indicador de pila.

10.- El procedimiento de la reivindicación 8, en el que la reutilización de la memoria caché (140) de la traza de pila comprende al menos una de lo siguiente:

5 usar la memoria caché (140) de la traza de pila sin cambios y modificar la memoria caché (140) de la traza de pila para recibir tramas de pila recientes y actualizadas.

11.- El procedimiento de cualquier reivindicación anterior, en el que la ejecución (220) de la enumeración del conjunto de raíces en la pila utilizando la memoria caché (140) de la traza de pila comprende:

10 determinar si una trama de pila está almacenada en caché en la memoria caché (140) de la traza de pila basada en una etiqueta de la trama de pila;
copiar una parte no modificada de la información de traza de pila de la memoria caché (140) de la traza de pila en la lista de enumeración de pila, si el valor de la etiqueta indica que la trama de pila está almacenada en caché en la memoria caché (140) de la traza de pila; y
ejecutar en caso contrario una enumeración normal del conjunto de raíces en la pila.

15 12.- El procedimiento de la reivindicación 11, en el que el copiado de la parte sin cambios de la información de traza de pila comprende la detección de una parte de la información de traza de pila que no ha cambiado desde la última enumeración de pila en la memoria caché (140) de la traza de pila y la recuperación de la parte sin cambios de la memoria caché (140) de la traza de pila.

20 13.- El procedimiento de la reivindicación 11, en el que la ejecución de una enumeración normal de la pila comprende:

desenrollar la pila hasta alcanzar una trama que se almacena en caché en la memoria caché (140) de la traza de pila;
almacenar en caché información de tramas no desenrolladas en la memoria caché (140) de la traza de pila;
y
25 añadir referencias enumeradas a partir de las tramas desenrolladas en la lista de enumeración de pila.

14.- Un sistema (100) de tiempo de ejecución controlado, que comprende

una máquina virtual (110);
un compilador Justo En Tiempo (120);
un mecanismo (130) de enumeración de un conjunto de raíces capaz de obtener (240) un conjunto de raíces;
30 un recuperador (150) de espacio inutilizado capaz de trazar (250) objetos operativos que se pueden alcanzar a partir del conjunto de raíces y recuperar (260) un espacio de almacenamiento ocupado por objetos distintos de los objetos operativos; y
caracterizado por

35 el mecanismo (130) de enumeración del conjunto de raíces que comprende:

un mecanismo de enumeración de pila; y
al menos una memoria caché de la traza de pila; y **caracterizado por**:

40 efectuar (220) una enumeración del conjunto de raíces en una pila de hilo por:

el almacenamiento en una memoria caché (140) de la traza de pila de información de traza de pila de las tramas de pila;
45 el uso de información de traza almacenadas en la memoria caché (140) de pila sin desenrollado adicional de la pila si la información de traza que se inicia a partir de una trama se almacena en caché en la memoria caché (140) de la traza de pila; y
la modificación de la memoria caché de la traza de pila para almacenar en caché información de traza recientes o actualizadas si la información de traza que se inicia a partir de una trama no está almacenada
50 en caché, o si la información de traza de pila se ha modificado desde la última enumeración de pila.

15.- El sistema de la reivindicación 14, en el que la memoria caché (140) de la traza de pila comprende una zona de identificación para almacenar identificadores de trama de pila y una zona para almacenar referencias enumeradas para tramas de pila.

55 16.- El sistema de la reivindicación 14 o la reivindicación 15, en el que el mecanismo (130) de enumeración del conjunto de raíces comprende:

un clasificador (310) de trama de pila capaz de clasificar una trama de pila basándose en una etiqueta de la

- trama de pila;
un mecanismo (320) de almacenamiento en caché de información de traza capaz de almacenar en caché información de la trama de pila en una memoria caché (140) de la traza de pila, si el valor de la etiqueta indica que la trama de la pila no está almacenada en memoria;
- 5 un mecanismo (340) de detección de traza sin cambio capaz de detectar una parte de la información de traza de pila que no han cambiado desde la última enumeración de pila en la memoria caché (140) de la traza de pila, si el valor de la etiqueta indica que la trama de pila está almacenada en caché; y
un mecanismo (350) de recuperación de traza sin cambios capaz de recuperar la parte sin cambios de la información de traza de pila de la memoria caché (140) de la traza de pila.
- 10 17.- El sistema de cualquiera de las reivindicaciones 14 a 16, que comprende, además, un mecanismo (330) de desenrollado de pila capaz de desenrollar una pila hasta alcanzar una trama de pila que está almacenada en caché en la memoria caché (140) de la traza de pila.
- 18.- El sistema de cualquiera de las reivindicaciones 14 a 17, en el que el mecanismo (320) de almacenamiento en caché de información de traza comprende:
- 15 un creador (325) de memoria caché capaz de crear la memoria caché (140) de la traza de pila cuando la enumeración del conjunto de raíces en la pila se efectúa por primera vez; y
un componente de identificación de trama capaz de identificar una trama en la memoria caché (140) de la traza de pila.
- 20 19.- Un programa informático que comprende un medio de código de programa informático adaptado para llevar a cabo todas las etapas de cualquiera de las reivindicaciones 1 a 13 cuando dicho programa es ejecutado en un ordenador.

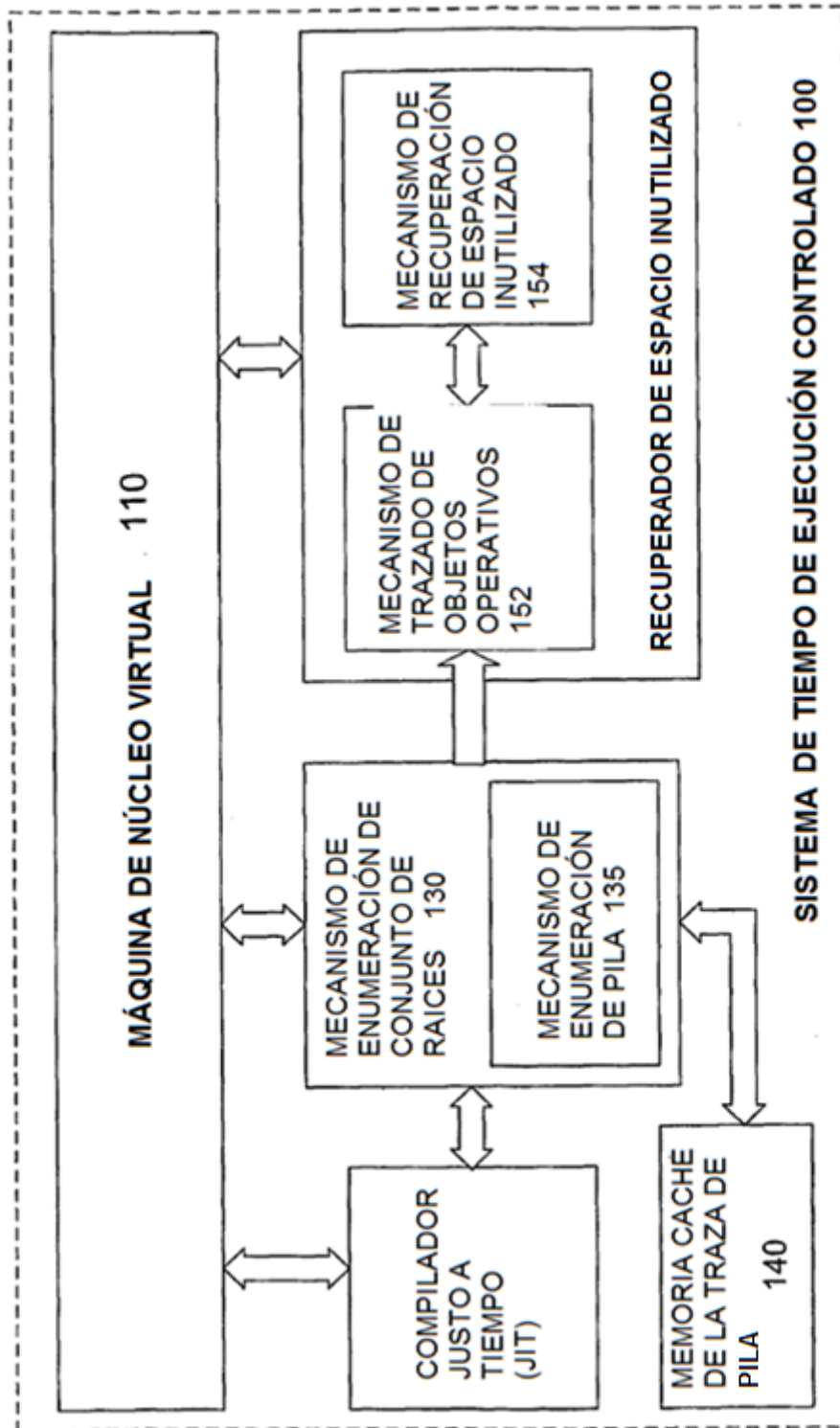


FIGURA 1

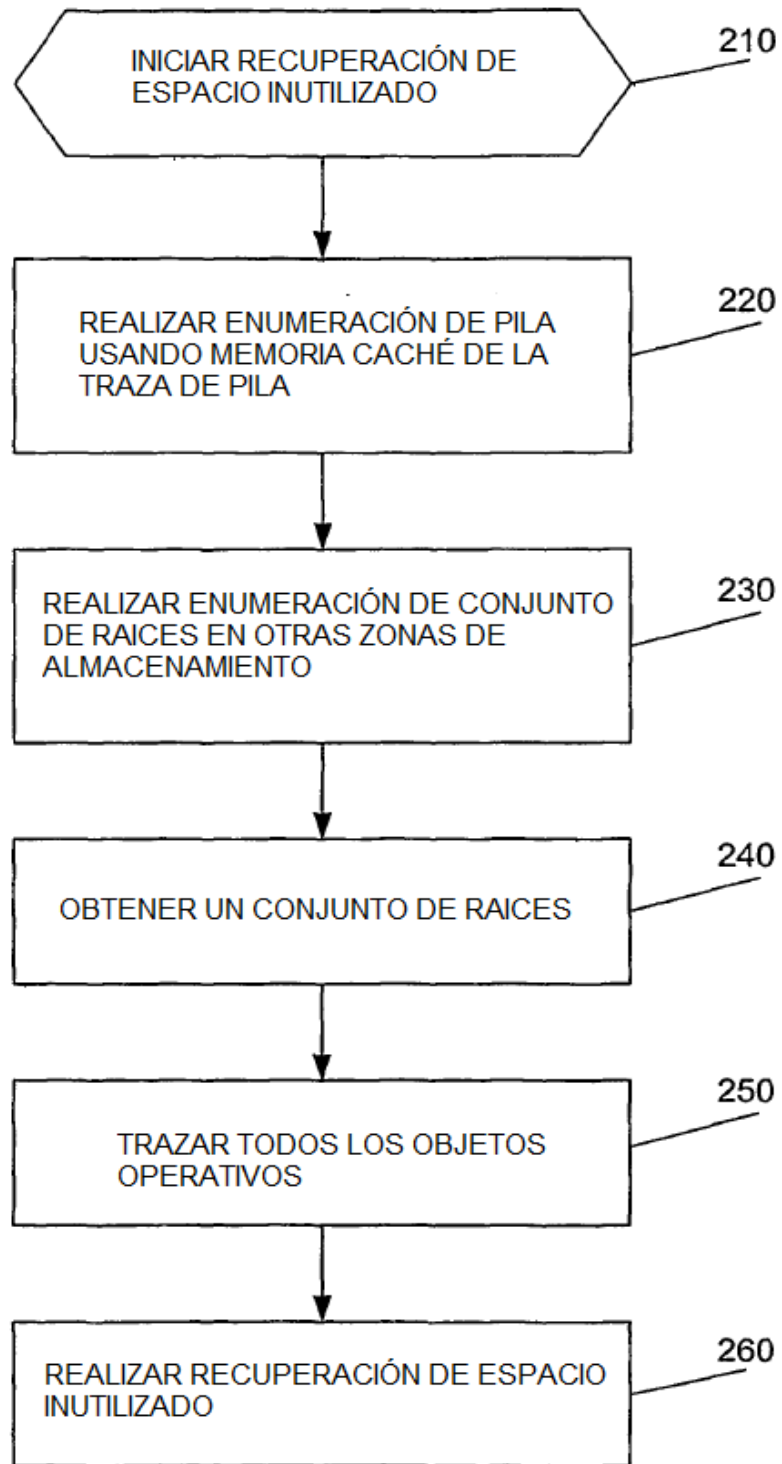


FIGURA 2

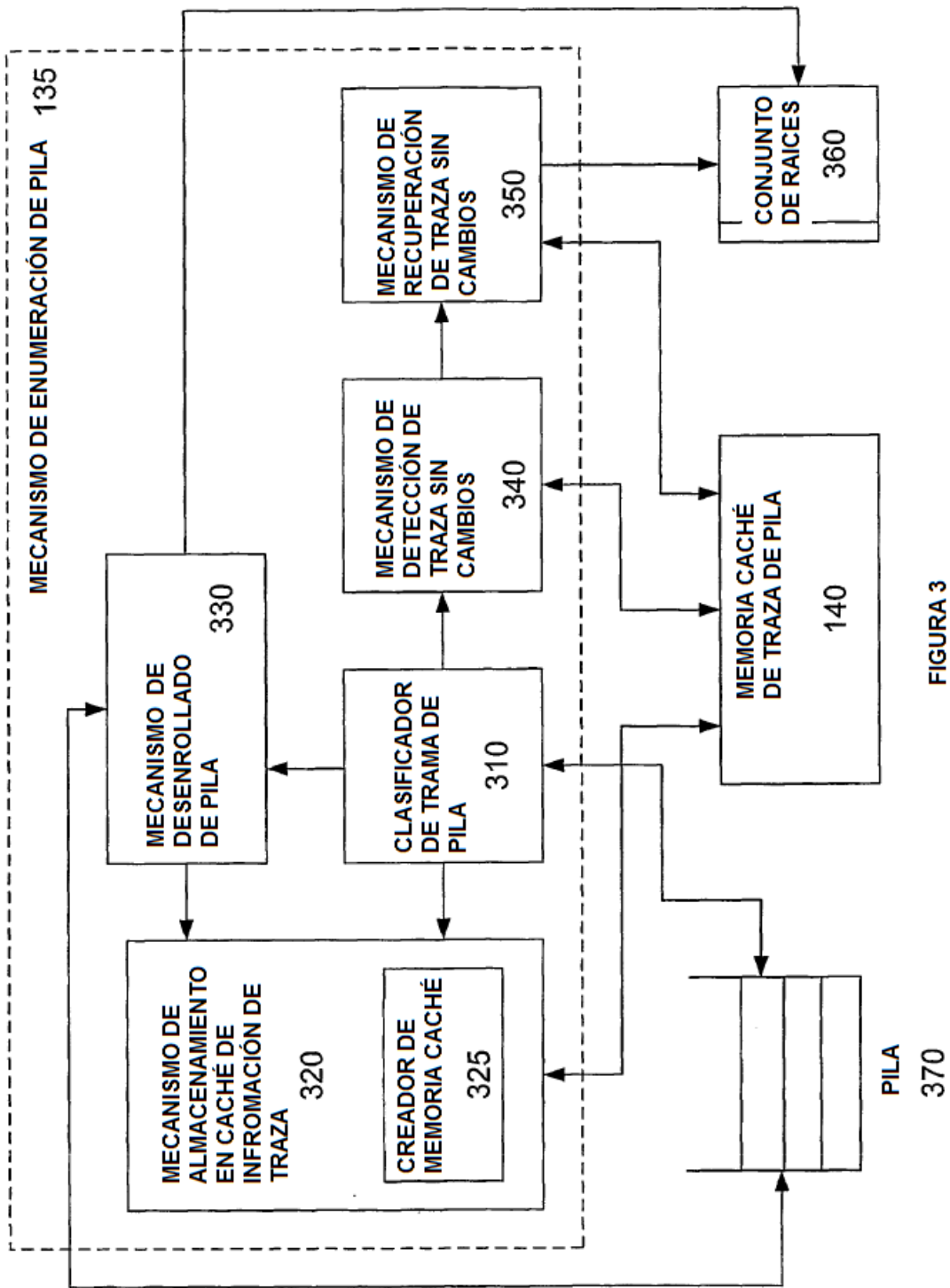


FIGURA 3

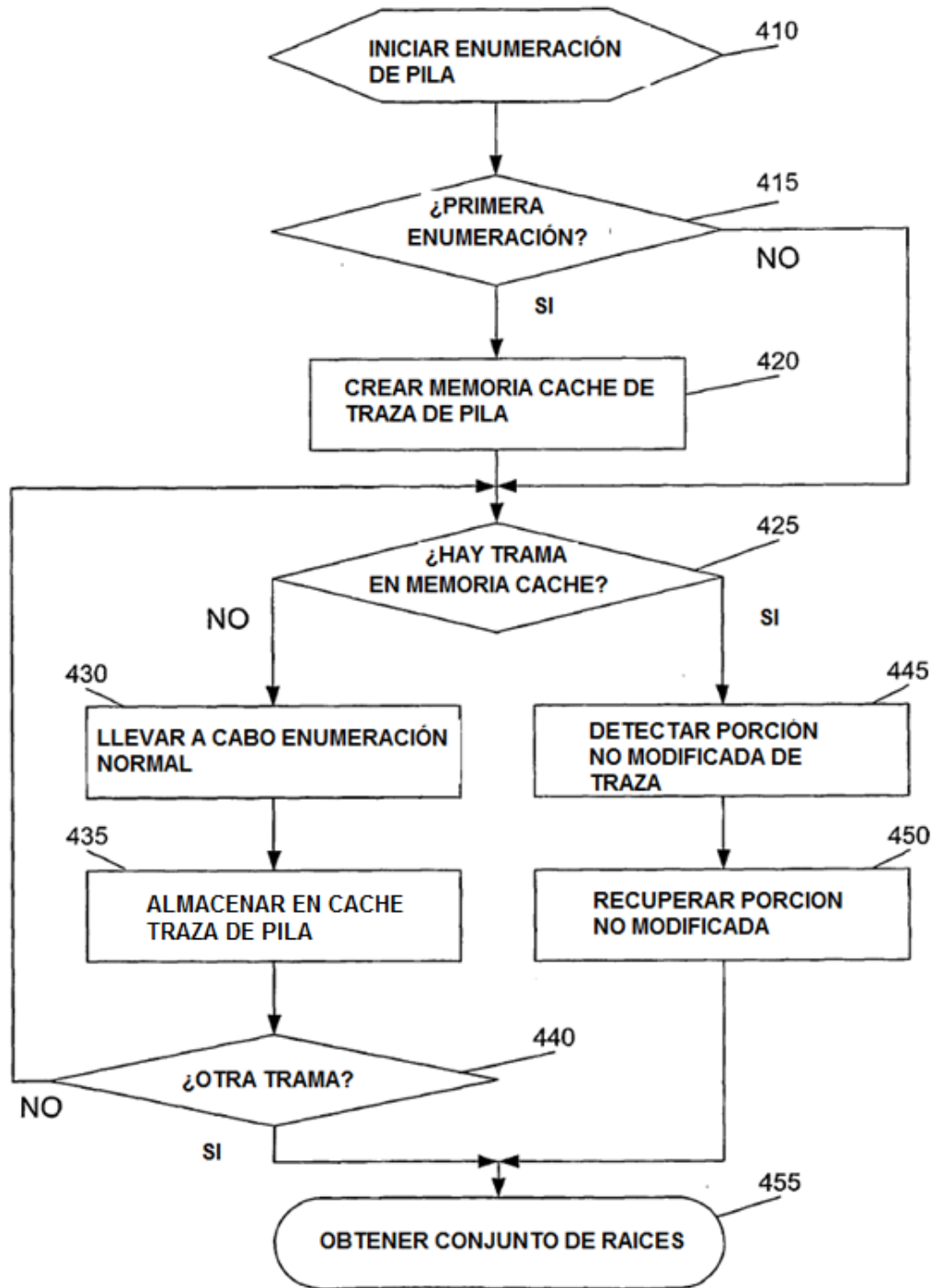


FIGURA 4

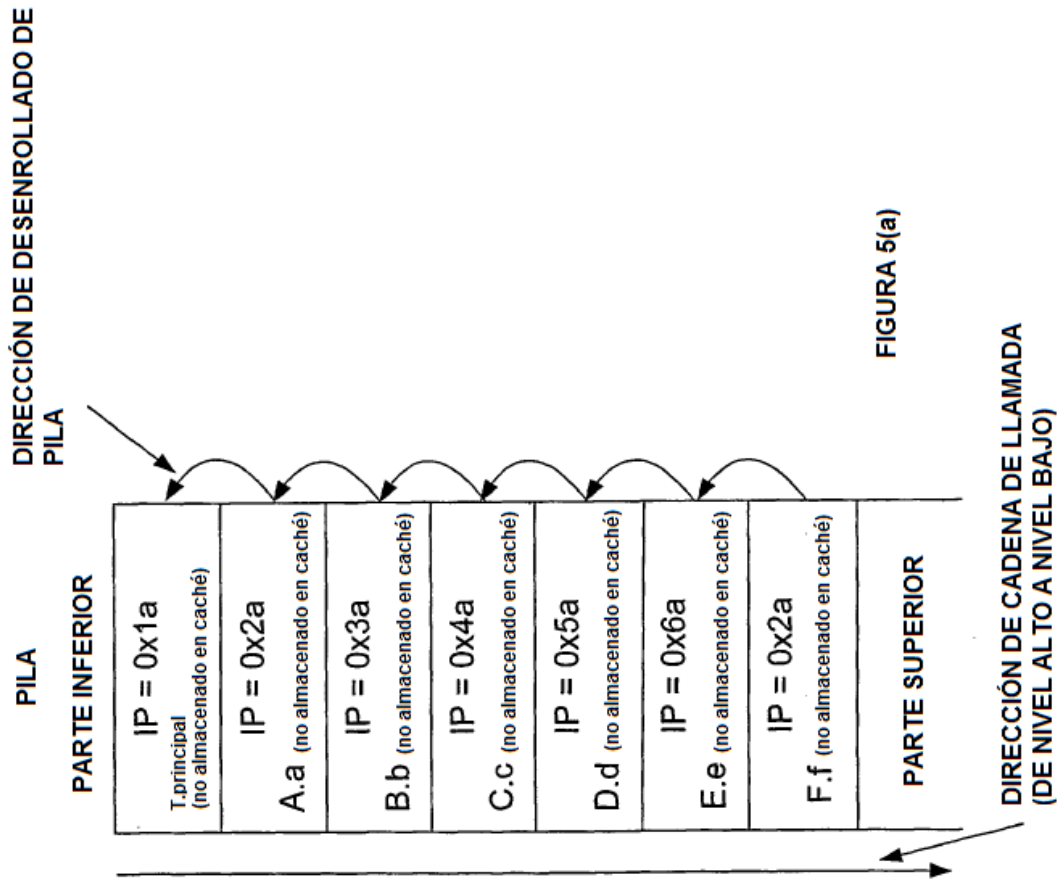


FIGURA 5(a)

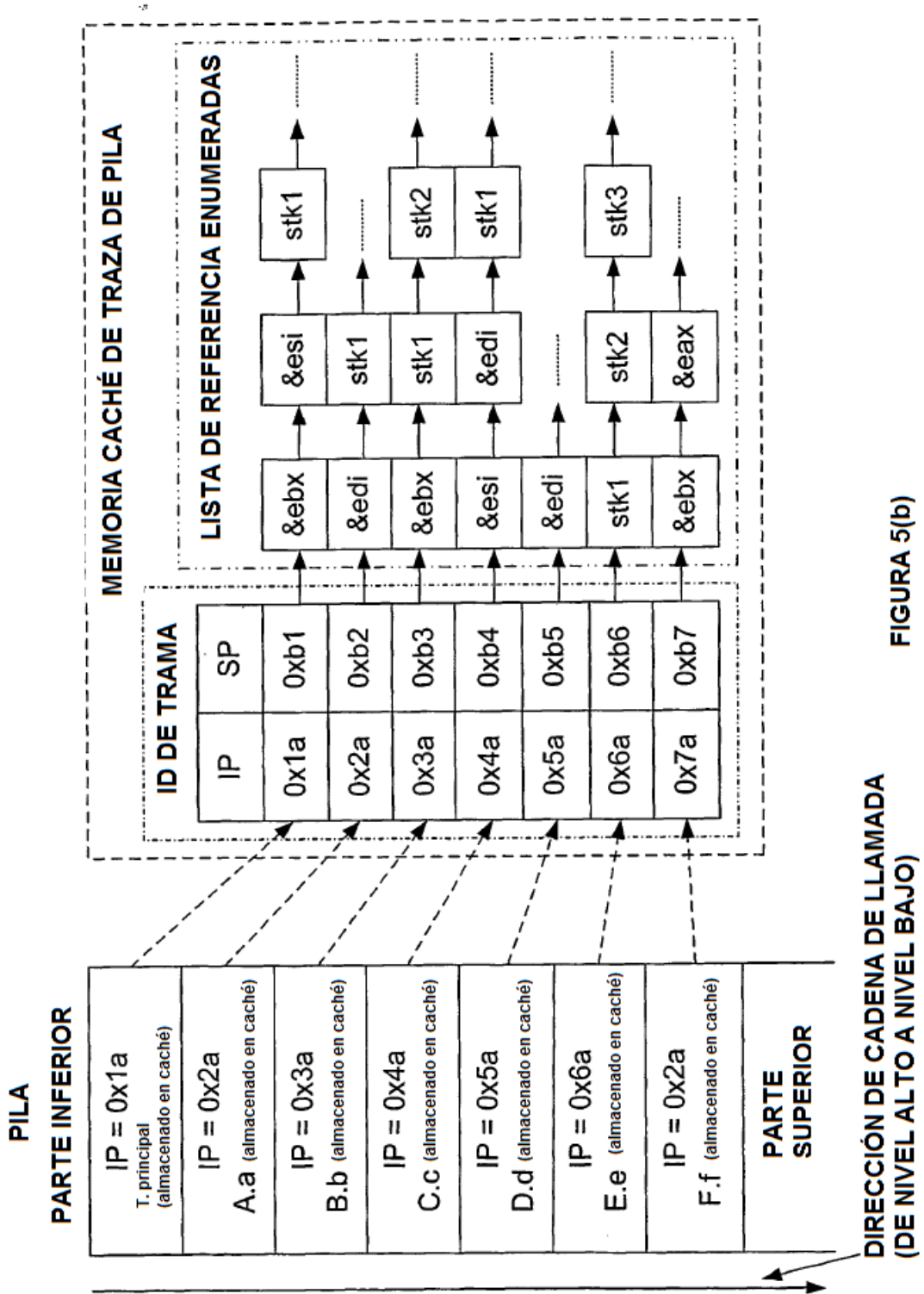


FIGURA 5(b)

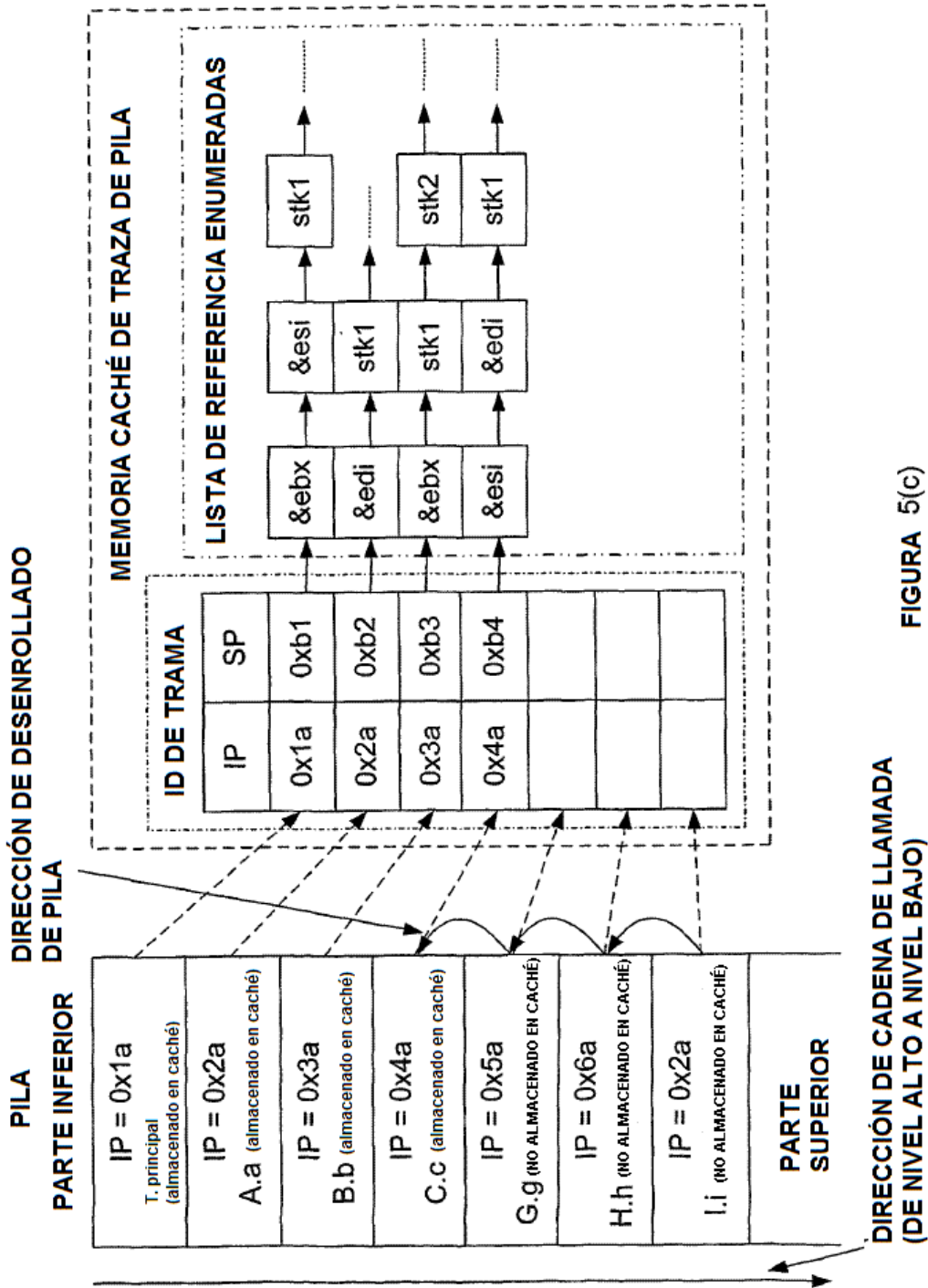


FIGURA 5(c)

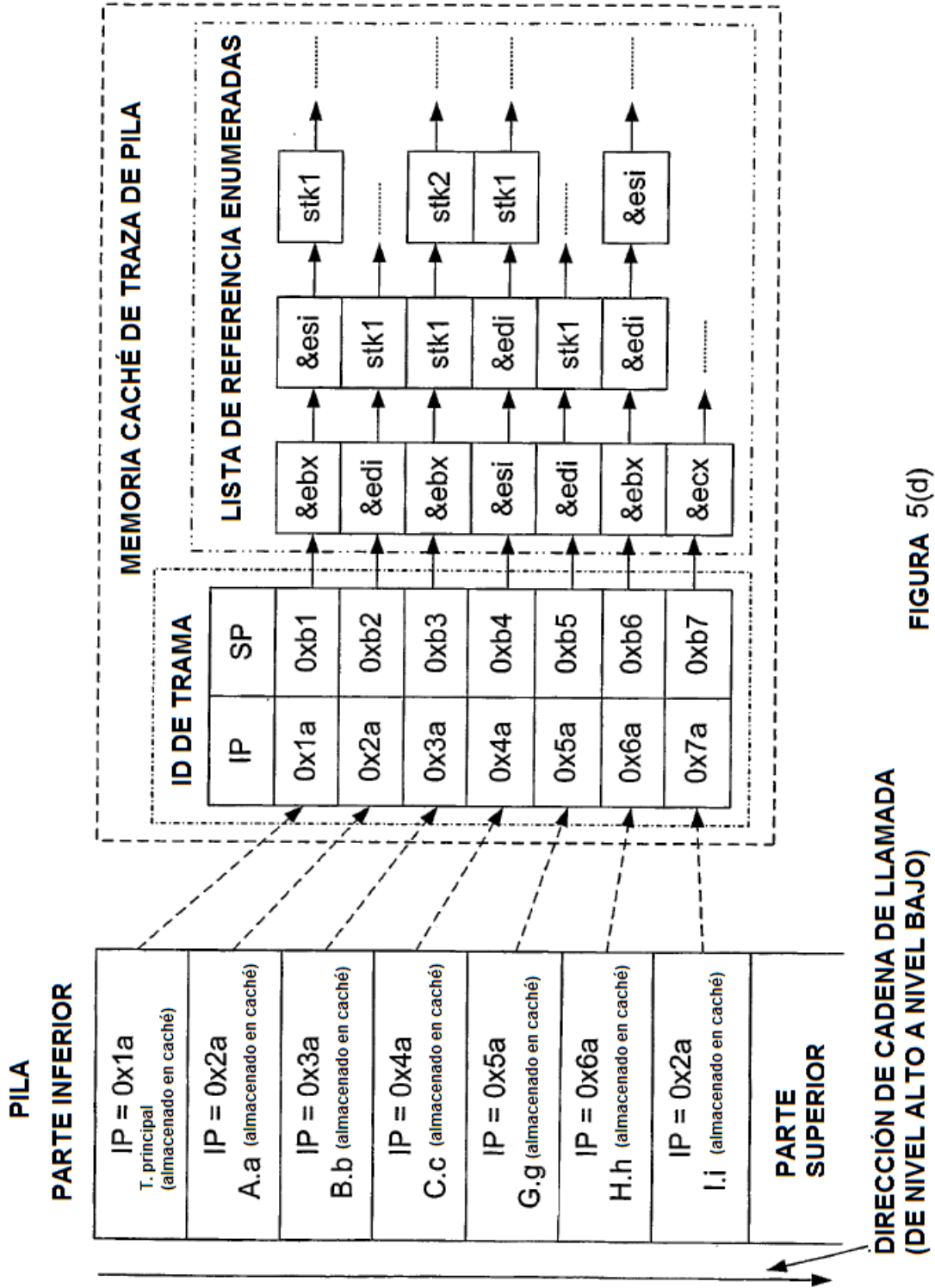


FIGURA 5(d)