

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 764 761**

51 Int. Cl.:

G06F 16/22 (2009.01)

G06F 16/23 (2009.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **03.04.2018** **E 18165366 (8)**

97 Fecha y número de publicación de la concesión europea: **20.11.2019** **EP 3385867**

54 Título: **Gestión de identificadores únicos en una base de datos**

30 Prioridad:

05.04.2017 FR 1752977
05.04.2017 US 201715479880

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

04.06.2020

73 Titular/es:

AMADEUS S.A.S. (100.0%)
485 Route du Pin Montard, Sophia Antipolis
06410 Biot, FR

72 Inventor/es:

VENEZIA, ALESSANDRO;
DEVEDEUX, THIBAUT;
BURDESE, SAMUEL y
SARAULLI, STEFANO

74 Agente/Representante:

ELZABURU, S.L.P

ES 2 764 761 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

Gestión de identificadores únicos en una base de datos

Campo de la invención

5 La presente invención se refiere a la gestión de bases de datos accesibles a gran escala para un gran número de clientes con una alta tasa de actualizaciones.

Más específicamente, la presente invención se refiere a la asignación de identificadores a objetos almacenados en la base de datos, siendo asociado el identificador con un objeto dado, en cualquier momento, único en la base de datos.

Antecedentes de la invención

10 En los sistemas de información a gran escala, a los objetos almacenados en una base de datos generalmente se les asigna un identificador que consiste, por ejemplo, en una cadena de caracteres alfanuméricos.

El identificador se puede utilizar como un medio para identificar un objeto dado con una estructura de datos de tamaño limitado. Al procesar el identificador en lugar del propio objeto, se optimizan los recursos de los sistemas generales. Yuan Shi et al., en "object identifier reuse mechanism for Totem" (22nd conference on data Engineering Workshops 2006), describen un sistema para reutilizar identificadores de objetos con el fin de ahorrar recursos.

15 Los objetos procesados y almacenados en bases de datos pueden no ser estáticos en esencia. Los objetos se pueden crear, borrar y modificar a varias velocidades. Por lo tanto, los identificadores también se pueden crear, borrar y modificar correspondientemente.

20 Sin embargo, para las bases de datos que procesan una gran cantidad de datos a los que accede un gran número de clientes, crear, borrar, modificar y también asignar identificadores se puede convertir en una tarea que afecta al rendimiento general del sistema. Un ejemplo de un parámetro de rendimiento que puede verse afectado es el tiempo de respuesta. Otros ejemplos son la memoria necesaria para almacenar los identificadores y su asociación con los objetos correspondientes, y el ancho de banda de comunicación para actualizar los identificadores y sus correspondencias.

25 En algunas aplicaciones, a cada objeto se le asigna un identificador único que representa el objeto de una forma única en todo el sistema.

Un campo en el que es importante el uso de identificadores únicos en una base de datos es la gestión de reservas de viaje.

Las compañías de viajes, como por ejemplo las compañías aéreas y las empresas hoteleras, utilizan bases de datos y *software* de reservas muy sofisticados para gestionar las reservas de los clientes.

30 Una reserva de viaje requiere la implicación de un gran número de participantes: agentes comerciales, compañías de transporte, aduanas y otros organismos administrativos, etc. Por regla general, estos participantes también están distribuidos en grandes regiones geográficas.

35 Por lo tanto, es necesario utilizar una base de datos y una arquitectura de comunicación comunes para soportar una gestión adecuada de la gran cantidad de información que representan las especificaciones de viaje de cada reserva por parte de cada cliente.

Por lo tanto, una gran cantidad de dispositivos cliente (en las compañías de viajes) accede simultáneamente, y desde una gran cantidad de ubicaciones, a la base de datos común.

40 El TPF (acrónimo de "Transaction Processing Facility" - "Centro de Procesamiento de Transacciones") es un ejemplo de un sistema operativo en tiempo real desarrollado por IBM™ para procesar transacciones de datos que intervienen en sistemas de reserva de viajes. Existen otros sistemas, como entenderá el experto en la materia.

Para compartir información con respecto a una reserva dada, la industria de viajes ha desarrollado descriptores comunes y estructuras de datos para utilizarlos en los sistemas de procesamiento de transacciones de datos.

45 Históricamente, la industria de las aerolíneas definía el PNR (acrónimo de "Passenger Name Record" - "Registro de Nombre de Pasajero"), que es un registro en una base de datos que contiene información (incluyendo información de reserva) sobre un viajero o un grupo de viajeros que viajan juntos. El objetivo del PNR consistía en permitir que las compañías aéreas intercambiaran información de reserva si el viaje involucraba a varias aerolíneas para varios destinos. IATA (acrónimo de "International Air Transportation Association" - "Asociación de Transporte Aéreo Internacional"), la asociación comercial que agrupa las compañías aéreas de todo el mundo, es el organismo que define la norma para el PNR (contenido, formato, etc.).

Aunque los PNR se habían definido inicialmente para la industria de las aerolíneas, también se utilizan para las reservas de hotel, alquiler de coches, reservas de billetes de tren, etc.

Para facilitar el acceso a los PNR se asocian identificadores únicos con los PNR. Los identificadores únicos generalmente se denominan "localizadores de registros", pero existen otras denominaciones.

- 5 Los localizadores de registros se definen generalmente como una cadena de caracteres alfanuméricos, por ejemplo 6 caracteres (números y letras). El uso de una cadena alfanumérica es el patrón actual en la industria.

La gestión de los localizadores de registros ha de satisfacer una gran cantidad de restricciones técnicas. Las siguientes son restricciones ejemplares:

- Un alto rendimiento, por ejemplo, más de 50 solicitudes de identificador por segundo.
- 10 - Un bajo retardo de respuesta, por ejemplo al menos el 90% de las solicitudes de identificador han de estar procesadas en menos de un milisegundo.
- Estar asociados de forma única fiable con un PNR activo en un momento dado y evitar fallos debidos a conflictos de identificación.
- Ser reutilizables una vez que se ha agotado un uso previo del identificador.
- 15 - Garantizar que se evitan los identificadores "perdidos", por ejemplo identificadores asignados a un PNR inactivo sin posibilidad de recuperar el identificador (por ejemplo, para reutilizarlo).
- Simplicidad de operación, evitación de intervenciones manuales.
- Respeto de las restricciones de comunicación de la arquitectura del sistema (restricciones de mensajes y similares).
- 20 - Interoperabilidad entre los sistemas de procesamiento de datos de transacciones (TPF desarrollado por IBM™ y otros sistemas).
- Respeto de las restricciones de normalización.

25 Para hacer frente a estas restricciones, existe la necesidad de una evolución desde unidades principales centralizadas hasta sistemas distribuidos, en donde servidores independientes y redundantes (llamados "procesadores especializados" - "*backends*") puedan realizar asignaciones de localizadores de registros de forma simultánea e independiente para registros de reserva recién creados.

30 En algunas unidades principales centralizadas, los localizadores de registros no solo sirven como identificadores para los PNR, sino que también sirven como una dirección de bajo nivel en la base de datos. Para generar los localizadores de registros, los sistemas centralizados implementan algoritmos que pueden convertir una dirección de la base de datos en un localizador de registros que cumpla con la norma.

Este método de generación para identificadores únicos no es conveniente para arquitecturas distribuidas. La generación basada en direcciones no se puede implementar en cada procesador especializado sin un aumento de la complejidad del sistema mientras se siguen cumpliendo las restricciones impuestas a los identificadores (tamaño corto con un número limitado de caracteres, alto rendimiento, compatibilidad, etc.).

- 35 Por lo tanto, es necesario mejorar la generación de identificadores únicos para los sistemas de bases de datos.

La presente invención se encuentra dentro de este contexto.

Compendio de la invención

40 Según un **primer aspecto de la invención** se proporciona un sistema para asociar identificadores con objetos representados en el sistema, en el que, en un momento dado, un objeto dado representado en una base de datos del sistema se asocia con un identificador único,

comprendiendo el sistema:

- una base de datos, comprendiendo dicha base de datos:
 - una unidad de memoria configurada para almacenar dichos objetos y sus respectivas asociaciones con dichos identificadores,
 - 45 • una unidad de índice configurada para almacenar un estado de disponibilidad de los identificadores del sistema, y

- una unidad de punteros de control configurada para almacenar punteros para datos en la unidad de índice;
- al menos una unidad de gestión de identificadores configurada para acceder a dicha base de datos; y
- al menos una unidad de actualización configurada para actualizar el estado de disponibilidad de los identificadores del sistema,

5 en donde:

- dichos identificadores pertenecen a un conjunto finito de identificadores organizados, dentro de dicha unidad de índice, de acuerdo con páginas de identificadores,
 - estando representada cada página por un mapa de bits,
 - en donde cada bit del mapa de bits corresponde a un valor de identificador respectivo, y
- 10 • cada valor de bit corresponde a la disponibilidad del identificador correspondiente,
- dichos punteros para datos en el índice comprenden al menos
 - un primer puntero para una última página cargada por una unidad de gestión de identificadores del sistema, y
 - un segundo puntero para una última página actualizada por una unidad de actualización del sistema, y

15 en donde:

- dicha al menos una unidad de gestión de identificadores está configurada además para:
 - cargar una primera página de identificadores basada en dicho primer puntero para asignar un identificador de dicha página de identificadores a un objeto que ha de ser almacenado en la base de datos, sobre la base de un mapa de bits correspondiente a dicha primera página cargada, y
- 20 - dicha al menos una unidad de actualización está configurada además para:
 - determinar identificadores correspondientes a una segunda página de identificadores basada en dicho segundo puntero,
 - acceder a dicha unidad de memoria para comprobar la disponibilidad de los identificadores determinados, y
- 25 • actualizar un mapa de bits que representa dicha segunda página cargada sobre la base de la disponibilidad de dichos identificadores determinados, y
- dicha base de datos está configurada además para:
 - almacenar en dicha unidad de memoria un objeto y su identificador asignado tras su activación en el sistema, y
- 30 • borrar de dicha unidad de memoria un objeto y su identificador asignado tras su desactivación en el sistema.

Algunas realizaciones proporcionan una respuesta de tiempo corto mientras mantienen la complejidad de la implementación en un nivel bajo. Algunas realizaciones también proporcionan compatibilidad con bases de datos existentes.

35 Según algunas realizaciones, dicha al menos una unidad de gestión de identificadores está configurada además para descartar dicha primera página de identificadores cargada, después de un tiempo que no supera una duración máxima de la vida útil de la página.

Según algunas realizaciones, dicha al menos una unidad de actualización está configurada además para esperar dicha duración máxima de la vida útil de la página antes de determinar un identificador correspondiente a dicha segunda página de identificadores.

40 Según algunas realizaciones, al menos una unidad de gestión de identificadores o una unidad de actualización, o ambas, están configuradas además para comprobar si dicho primer puntero y dicho segundo puntero apuntan simultáneamente a una misma página de identificadores.

Según algunas realizaciones, dichas páginas de identificadores se ordenan de forma cíclica.

Según algunas realizaciones, antes de asignar un identificador a un objeto que ha de ser almacenado en la base de datos, se comprueban los punteros para verificar que el segundo puntero no esté apuntando a la página de identificadores situada junto a la página a la que apunta el primer puntero.

Por ejemplo, los punteros se comprueban cada vez que se carga una página.

- 5 Según algunas realizaciones, antes de actualizar un mapa de bits de una página de identificadores, se comprueban los punteros para verificar que el primer puntero no esté apuntando a la página de identificadores situada junto a la página a la que apunta el segundo puntero.

10 Según un **segundo aspecto de la invención** se proporciona un método para asociar identificadores con objetos representados en un sistema, en el que, en un momento dado, un objeto dado representado en una base de datos del sistema se asocia con un identificador único,

comprendiendo el sistema:

- una base de datos, comprendiendo dicha base de datos:
 - una unidad de memoria configurada para almacenar dichos objetos y sus respectivas asociaciones con dichos identificadores,
 - 15 • una unidad de índice configurada para almacenar un estado de disponibilidad de los identificadores del sistema, y
 - una unidad de punteros de control configurada para almacenar punteros para datos en la unidad de índice,
- al menos una unidad de gestión de identificadores configurada para acceder a dicha base de datos, y
- al menos una unidad de actualización configurada para actualizar el estado de disponibilidad de los 20 identificadores en el sistema,

en donde:

- dichos identificadores pertenecen a un conjunto finito de identificadores organizados, dentro de dicha unidad (102) de índice, de acuerdo con páginas de identificadores,
 - estando representada cada página por un mapa de bits,
 - 25 • en donde cada bit del mapa de bits corresponde a un valor de identificador respectivo, y
 - cada valor de bit corresponde a la disponibilidad del identificador correspondiente,
- dichos punteros para datos en el índice incluyen al menos
 - un primer puntero para una última página cargada por una unidad de gestión de identificadores del sistema, y
 - 30 • un segundo puntero para una última página actualizada por una unidad de actualización del sistema,

comprendiendo el método las siguientes etapas, realizadas por dicha al menos una unidad de gestión de identificadores:

- cargar una primera página de identificadores basada en dicho primer puntero,
- asignar un identificador de dicha página de identificadores a un objeto que ha de ser almacenado en la base de 35 datos, sobre la base de un mapa de bits correspondiente a dicha primera página cargada,

comprendiendo el método las siguientes etapas, realizadas por dicha al menos una unidad de actualización:

- determinar identificadores correspondientes a una segunda página de identificadores sobre la base de dicho segundo puntero,
- acceder a dicha unidad de memoria para comprobar la disponibilidad de los identificadores determinados,
- 40 - actualizar un mapa de bits que representa dicha segunda página cargada sobre la base de la disponibilidad de dichos identificadores determinados, y

comprendiendo el método las siguientes etapas, realizadas por dicha base de datos:

- almacenar en dicha unidad de memoria un objeto y su identificador asignado tras su activación en el sistema, y
- borrar de dicha unidad de memoria un objeto y su identificador asignado tras su desactivación en el sistema.

5 Según algunas realizaciones, dicha al menos una unidad de gestión de identificadores descarta además dicha primera página de identificadores cargada, después de un tiempo que no supera una duración máxima de la vida útil de la página.

Según algunas realizaciones, dicha al menos una unidad de actualización espera además dicha duración máxima de la vida útil de la página antes de determinar un identificador correspondiente a dicha segunda página de identificadores.

10 Según algunas realizaciones, al menos una unidad de gestión de identificadores o una unidad de actualización, o ambas, comprueban además si dicho primer puntero y dicho segundo puntero apuntan simultáneamente a una misma página de identificadores.

Según algunas realizaciones, dichas páginas de identificadores se ordenan de forma cíclica.

15 Según algunas realizaciones, antes de asignar un identificador a un objeto que ha de ser almacenado en la base de datos, se comprueban los punteros para verificar que el segundo puntero no esté apuntando a la página de identificadores situada junto a la página a la que apunta el primer puntero.

Según algunas realizaciones, antes de actualizar un mapa de bits de una página de identificadores, se comprueban los punteros para verificar que el primer puntero no esté apuntando a la página de identificadores situada junto a la página a la que apunta el segundo puntero.

20 Según un **tercer aspecto de la invención** se proporciona un producto de programa informático que comprende instrucciones para implementar un método de acuerdo con el segundo aspecto de la invención cuando el programa se carga y ejecuta mediante un aparato programable.

También se proporciona un medio de almacenamiento de información no transitoria legible por un ordenador o por un microprocesador, que almacena instrucciones de un programa informático para implementar un método de acuerdo con el segundo aspecto de la invención, cuando el ordenador o microprocesador carga y ejecuta el programa.

25 Los objetos según el segundo y el tercer aspecto de la invención proporcionan al menos las mismas ventajas que las proporcionadas por el sistema según el primer aspecto.

Breve descripción de los dibujos

Las características y ventajas de la invención se pondrán de manifiesto a partir de la siguiente descripción de realizaciones ejemplares no limitativas, con referencia a los dibujos adjuntos, en los que:

- 30 - la **Figura 1** ilustra un contexto general de implementación de realizaciones;
- las **Figuras 2A y 2B** ilustran localizadores de Registros según realizaciones;
- la **Figura 3** ilustra esquemáticamente una tabla que representa PNR de acuerdo con realizaciones;
- la **Figura 4** ilustra esquemáticamente una tabla de índice de acuerdo con realizaciones;
- 35 - las **Figuras 5A-5B** ilustran esquemáticamente una tabla almacenada en la unidad de punteros de control de acuerdo con realizaciones;
- la **Figura 6** es un flujo de datos que ilustra etapas para asignar un localizador de registros a un dispositivo cliente de acuerdo con realizaciones;
- la **Figura 7** es un flujo de datos que ilustra etapas para actualizar el sistema de base de datos de acuerdo con realizaciones; y
- 40 - la **Figura 8** ilustra esquemáticamente un dispositivo de acuerdo con realizaciones.

Descripción detallada de la invención

La **Figura 1** ilustra un contexto general de implementación de realizaciones.

45 Se proporciona un sistema **100** de base de datos para almacenar objetos que pueden ser proporcionados y recuperados por una pluralidad de dispositivos externos. El sistema de base de datos se puede implementar como una base de datos relacional o un sistema de gestión de base de datos relacional de objetos (ORDBMS). Por ejemplo, el ORDBMS puede ser un sistema como los comercializados por Oracle Corporation™.

Por ejemplo, el sistema **100** de base de datos puede almacenar PNR creados por sistemas de gestión implementados por ordenadores de agencias de viajes.

La base de datos comprende una unidad **101** de memoria para almacenar los objetos (tales como los PNR) y su asociación con un identificador único. El identificador único puede ser un localizador de registros, por ejemplo.

5 Según algunas realizaciones, la unidad **101** de memoria almacena solo objetos que están activos en el sistema global. En el caso de un PNR, la unidad **101** de memoria solo tendrá en la memoria PNR que estén activos. Por lo tanto, en cuanto ya no se necesita un PNR (por ejemplo, el recorrido que representa ha terminado), se elimina de la unidad **101** de memoria. El objeto se puede borrar o desactivar de acuerdo con cualquier otro mecanismo. Dicho mecanismo puede ser realizado por una unidad de eliminación de PNR externa (no representada). Las acciones de la unidad de
10 eliminación de PNR pueden ser activadas por la unidad **150** de control descrita más abajo.

El hecho de que la unidad **101** de memoria comprenda solo objetos activos permite determinar, a partir de la asociación entre un objeto y su identificador único, si un identificador único está disponible o no para su asignación a otro objeto. En otras palabras, dado que la unidad **101** de memoria comprende solo objetos activos, se puede inferir que, en cuanto un identificador único se asocia con un objeto en la unidad **101** de memoria, ya no está disponible para otro objeto.

15 El sistema **100** de base de datos comprende además una unidad **102** de índice. La unidad **102** de índice comprende una síntesis del estado de disponibilidad de todos los identificadores únicos en el sistema. Cada identificador único del sistema está marcado como disponible o no disponible en la unidad **102** de índice.

El sistema **100** de base de datos comprende además una unidad **103** de punteros de control. La unidad **103** de punteros de control comprende punteros para datos en la unidad **102** de índice.

20 El sistema **100** de base de datos se conecta además a través de una red de comunicaciones (no representada) con varios dispositivos que pueden leer y/o escribir datos en y/o desde el sistema de base de datos. La red de comunicaciones es una red basada en el Protocolo de Internet, por ejemplo.

La unidad **110** de gestión de transacciones es una unidad de procesamiento conectada al sistema **100** de base de datos encargada de la actualización en lo que respecta a los objetos almacenados en éste. A petición de un dispositivo
25 cliente, crea el objeto que ha de ser almacenado en la unidad **101** de memoria sobre la base de la información proporcionada por el dispositivo cliente (incluyendo el identificador único que ha de ser asociado con el mismo). La unidad **110** de gestión de transacciones también transmite los objetos creados al sistema de base de datos de gestión para su almacenamiento en la unidad **101** de memoria. La unidad **110** de gestión de transacciones también puede intervenir en el borrado y/o la modificación de objetos en la unidad **101** de memoria del sistema **100** de base de datos.

30 Los identificadores únicos se proporcionan a los dispositivos cliente a través de una o más unidades **120**, **121**, **122** de gestión de identificadores conectadas al sistema **100** de base de datos. Las unidades de gestión de identificadores son unidades **120**, **121**, **122** independientes ("procesadores especializados") que pueden acceder independientemente al sistema **100** de base de datos para leer datos. Uno o más dispositivos cliente **130**, **131**, **132** también pueden acceder independientemente a las unidades **120**, **121**, **122** de gestión de identificadores ("procesadores especializados") para
35 obtener identificadores únicos.

La sincronización del uso de los identificadores únicos se realiza a través de una unidad **140** de actualización conectada al sistema **100** de base de datos. Según un objetivo general del sistema general, en un momento dado no se debe asignar un identificador único a más de un objeto activo. Si un identificador está asociado con dos o más
40 objetos activos, se ha producido un error en el sistema. Según algunas realizaciones, un identificador puede estar presente en el sistema sin estar asociado con un objeto (como una reserva), por ejemplo en un caso en el que se filtra intencionalmente (por ejemplo, el identificador representa una mala palabra o el objeto activo todavía no está oficialmente en el sistema).

En una realización ejemplar, la unidad **140** de actualización regenera el estado de disponibilidad de los identificadores
45 únicos en un modo por lotes. La actualización del estado de disponibilidad del identificador único puede ser gestionada por una unidad **150** de control conectada a la unidad **140** de actualización. La unidad **150** de control se puede implementar utilizando el *software* de automatización de carga de trabajo Control-M™ de BMC software™.

En el contexto general de implementación de realizaciones arriba descrito, los dispositivos cliente **130**, **131**, **132** se conectan a las unidades **120**, **121**, **122** de gestión de identificadores para obtener identificadores únicos que han de ser asociados con objetos que han de ser almacenados en el sistema de base de datos. Los dispositivos cliente **130**,
50 **131**, **132** también se conectan a la unidad **110** de gestión de transacciones para la creación de dichos objetos sobre la base de la información proporcionada por los dispositivos cliente y para el almacenamiento de los objetos en el sistema **100** de base de datos en asociación con los identificadores únicos.

A continuación se describirá el ejemplo de un sistema de gestión de viajes. Sin embargo, la invención no se limita a este tipo de aplicaciones. Algunas realizaciones tienen otros tipos de aplicaciones en sistemas en los que se han de
55 asociar objetos con identificadores únicos y en los que el número de identificadores únicos disponibles es limitado.

A continuación, los identificadores únicos se pueden designar como "localizadores de registros", "recloc" o "rloc". Los objetos se pueden designar como "PNR". El sistema de base de datos se puede designar como el acrónimo "RDBMS".

Los localizadores de registros según algunas realizaciones se describen con referencia a las **Figuras 2A y 2B**.

5 Los localizadores de registros están representados por un número finito de números enteros "RLC". Por ejemplo, se expresan en base 34 con relleno a la izquierda para ajustar 6 caracteres. Por lo tanto: $0_{10} = 222\ 222_{34}$, $1_{10} = 222\ 223_{34}$... hasta $ZZZ\ ZZZ_{34}$.

10 El conjunto finito de localizadores de registros se divide lógicamente en las páginas **P1**, ..., **P5**. Cada página puede comprender un mismo número de localizadores de registros. También pueden comprender diferentes números de localizadores de registros. Por ejemplo, cada página comprende 16.000 localizadores de registros. Las páginas pueden estar numeradas secuencialmente. Además, por ejemplo, las páginas pueden ser adyacentes, es decir, una página comprende localizadores de registros sucesivos. Alternativamente, las páginas pueden comprender localizadores de registros no adyacentes.

En el sistema de base de datos, algunas operaciones pueden procesar páginas enteras. Otras operaciones, fuera del sistema de base de datos, pueden procesar localizadores de registros individuales.

15 Cada localizador de registros puede estar asociado con un valor que representa su disponibilidad en el sistema. Por ejemplo, un estado de bit que tiene un valor "1" cuando el localizador de registros está disponible y "0" cuando no está disponible.

En la **Figura 2A**, los localizadores de registros están representados en un segmento finito, por ejemplo desde 222 222 hasta $ZZZ\ ZZZ$.

20 Sin embargo, según algunas realizaciones, los localizadores de registros se representan como valores cíclicos, como en la **Figura 2B**. Los localizadores de registros tienen un número limitado y hay un "vuelco": cuando se "pasa" el número más grande, el recuento se reinicia desde el primer número. En otras palabras, los localizadores de recursos satisfacen la siguiente relación $ZZZ\ ZZZ + 222\ 223 = 222\ 222$. El localizador de registros situado junto al último es el primero.

25 Por lo tanto, los localizadores de registros tienen un número limitado, y están ordenados de forma cíclica y clasificados de acuerdo con las páginas. También tienen una caracterización de uso para indicar si están disponibles o no.

La **Figura 3** ilustra esquemáticamente la tabla que representa los PNR en la unidad **101** de memoria.

30 La tabla **300** tiene una primera columna **301** que almacena los PNR. Por ejemplo, solo los PNR activos en el sistema se almacenan en la tabla. Alternativamente se puede utilizar una estructura de datos para diferenciar entre PNR activos y no activos.

Una segunda columna **302** almacena los localizadores de registros. Dado que la tabla contiene solo PNR activos, los localizadores de registros en la tabla están en uso y no están disponibles. En caso de que la tabla diferencie entre PNR activos y no activos, la diferenciación también se aplica a los localizadores de registros.

35 Por lo tanto, en cada línea **303**, **304** de la tabla, un PNR está asociado de forma exclusiva con un localizador de registros único.

La **Figura 4** ilustra esquemáticamente la tabla **400** de índice que representa la síntesis (almacenada en la unidad **102**) del estado de disponibilidad de los localizadores de registros en la unidad **101** de memoria.

Los localizadores de registros se representan de acuerdo con un almacenamiento en modo tampón.

40 La columna **401** comprende las páginas ("Pg") de localizadores de registros. La columna **402** comprende un mapa de bits ("BM") asociado con la página correspondiente en la columna **401** que representa el estado de disponibilidad de cada localizador de registros en la página.

45 El valor "1" representa un localizador de registros que está disponible. El valor "0" representa un localizador de registros que está en uso y no está disponible. Dicho de otro modo, un localizador de registros que está presente en la unidad **101** de memoria como asociado con un PNR activo está representado por un valor "0" en el mapa de bits correspondiente a la página a la que pertenece. En caso de que el localizador de registros esté ausente de la unidad **101** de memoria, está representado por "1".

En cada mapa de bits de la columna **401**, un localizador de registros tiene la posición que presenta en la página a la que pertenece. Dicho de otro modo, el primer bit del mapa de bits corresponde al localizador de registros más bajo de la página, el último corresponde al localizador de registros más alto de la página.

50 Según algunas realizaciones, para una página dada representada en la tabla **400** de índice, se utilizan todos los localizadores de registros.

- 5 Esto significa que todos los bits de los mapas de bits son significativos. Sin embargo, durante el proceso de asignación, las unidades **120**, **121**, **122** de gestión de identificadores pueden omitir algunos de los bits si los localizadores de registros correspondientes están contenidos en una tabla de la unidad **101** de memoria (o cualquier sistema de almacenamiento similar). Por lo tanto, la tabla puede contener localizadores de registros que el sistema no debe generar por razones técnicas y/o funcionales.
- Las **Figuras 5A-5B** ilustran esquemáticamente la tabla **500** almacenada en la unidad **103** de punteros de control.
- La tabla **500** almacena punteros para páginas en la tabla **400** de índice. La fila **501** representa los punteros para páginas específicas almacenadas en la tabla **400** de índice. La fila **502** representa estas páginas específicas.
- 10 En algunas implementaciones, la tabla **500** puede tener diferentes filas para diferentes rangos de localizadores de registros. La organización de la tabla **500** con filas y columnas tal como se representa en la **Figura 5A** permite dicha adición de fila y dichas implementaciones.
- 15 El puntero **503** ("B" por "*Begin*" - "Comenzar") apunta a la primera página que comprende localizadores de registros autorizados en el sistema. Esto significa que los localizadores de registros que pertenecen a una página clasificada entre la página con el número más bajo y la página a la que apunta el puntero B no pueden ser utilizados. En caso de que las páginas comprendan valores sucesivos, esto significa que los localizadores de registros que tienen un valor inferior al valor del primer localizador de registros en la página a la que apunta el puntero B no pueden ser utilizados.
- 20 El puntero **504** ("E" por "*End*" - "Terminar") apunta a la página después de la última página que comprende localizadores de registros autorizados en el sistema. Esto significa que los localizadores de registros que pertenecen a una página clasificada entre la página a la que apunta el puntero E y la página con el número más alto o que pertenecen a la página a la que apunta el puntero E no pueden ser utilizados. En caso de que las páginas comprendan valores sucesivos, esto significa que los localizadores de registros que tienen un valor superior o igual al valor del primer localizador de registros en la página a la que apunta el puntero E no pueden ser utilizados.
- 25 El puntero **505** ("R" por "*Read*" - "Leer") apunta a la última página cargada por una unidad **120**, **121**, **122** de gestión de identificadores.
- El puntero **506** ("W" por "*Write*" - "Escribir") apunta a la página justo después de la última página actualizada por la unidad **140** de actualización.
- Tal como se describe a continuación, las unidades de gestión de identificadores leen los localizadores de registros página por página. Las unidades de administración de identificadores únicos cargan una página completa para uno o más dispositivos cliente determinados que podrán usar localizadores de registros disponibles de la página cargada.
- 30 Se implementan reglas de control que prevén que los punteros R y W nunca apunten a la misma página al mismo tiempo. Por ejemplo, estas reglas de control pueden ser implementadas por las unidades **120**, **121**, **122** de gestión de identificadores. También pueden ser implementadas por la unidad **140** de actualización.
- Además, las reglas prevén que la página a la que apunta el puntero R esté en uso (o sea cargada) por una unidad de gestión de identificadores únicos.
- 35 Los punteros se comprueban para determinar si la siguiente relación es verdadera: $R + 1 = W$. Esto se puede comprobar regularmente. Esto también se puede comprobar cada vez que se carga una página. En caso de que la página a la que apunta el puntero R sea la página inmediatamente anterior a la que apunta el puntero W, esto significa que no hay ninguna página con localizadores de registros disponibles. No se pueden crear (o utilizar) PNR hasta que se desactiven PNR y la unidad de administración de actualizaciones actualice las páginas con los localizadores de registros correspondientes.
- 40 Los punteros también se comprueban para determinar si la siguiente relación es verdadera: $W + 1 = R$. En el caso de que la página a la que apunta el puntero W sea la página inmediatamente anterior a la que apunta el puntero R, esto significa que todas las páginas están actualizadas. Por lo tanto se puede determinar que no es necesario que la unidad **140** de actualización realice una actualización.
- 45 La **Figura 6** es un flujo de datos que ilustra las etapas para asignar un localizador de registros a un dispositivo cliente.
- En una primera etapa **600**, un dispositivo cliente **130** solicita un localizador de registros a una unidad **120** de gestión de identificadores únicos.
- A continuación, en la etapa **601**, la unidad **120** de gestión de identificadores únicos carga el valor de puntero del puntero **505** ("R") desde la unidad **103** de punteros de control. La unidad **120** de gestión de identificadores únicos incrementa entonces el valor de puntero en la etapa **602**.
- 50 El valor de puntero en realidad no se carga directamente desde la unidad de punteros de control. Una solicitud de "lectura" se dirige al sistema de base de datos, que realiza el procesamiento necesario para leer los valores de la

unidad de punteros de control y enviarlos de vuelta a la unidad de gestión de identificadores únicos. El flujo de datos está simplificado para una mayor claridad y concisión.

La unidad **120** de gestión de identificadores únicos almacena el valor incrementado en la unidad **103** de punteros de control en la etapa **605**.

- 5 En la etapa **604**, la unidad **120** de gestión de identificadores únicos carga la página en la unidad **102** de índice correspondiente al puntero **505** (después del incremento).

10 Los sistemas de bases de datos implementan reglas de control de concurrencia que permiten garantizar que la carga, el incremento y el almacenamiento del valor del puntero R son atómicos, es decir, dos unidades diferentes de gestión de identificadores únicos no pueden leer la misma página. A modo de ejemplo, estas reglas de control están garantizadas por las propiedades "ACID" (acrónimo de "*Atomicity, Consistency, Isolation, Durability*" - "Atomicidad, Consistencia, Aislamiento, Durabilidad") de las transacciones de bases de datos relacionales.

En caso de que no haya ninguna página disponible, el sistema devuelve un error (no representado).

15 La primera página puede ser cargada por la unidad **120** de gestión de identificadores únicos al inicio. Alternativamente, la primera carga de una página se puede llevar a cabo con la primera solicitud recibida. La unidad de gestión de identificadores mantendrá la página cargada en la memoria durante un período de tiempo determinado o hasta que la página ya no tenga más localizadores de registros disponibles.

Por ejemplo, en un sistema de gestión de viajes, la vida útil de una página cargada puede ser de 12 minutos.

20 La unidad **120** de gestión de identificadores únicos comprueba si puede mantener una página cargada en memoria cada vez que accede a la propia página (para encontrar un localizador de registros disponible para el dispositivo cliente).

Cuando transcurre una duración llamada "vida útil de la página" correspondiente a un período de consulta de la página o cuando ésta ya no tiene más localizadores de registros disponibles (etapa **606**, NO), la unidad **120** de gestión de identificadores únicos recarga inmediatamente otra página (etapas **601** a **605**).

25 La unidad **120** de gestión de identificadores únicos también puede recargar una página (después de comprobar las condiciones arriba indicadas) tras haber asignado un localizador de registros a un dispositivo cliente.

La carga de una página completa, en lugar de un único localizador de registros, permite garantizar que la mayoría de las solicitudes se puedan satisfacer sin acceder a la base de datos.

30 Al probar el estado de la página después de responder al último dispositivo cliente, pero antes de aceptar la siguiente solicitud de un dispositivo cliente, el tiempo de respuesta se mantiene corto (la carga de la página se realiza "entre" en lugar de "durante" el procesamiento de solicitudes).

Una vez que la unidad **120** de gestión de identificadores únicos tiene una página válida en la memoria, busca un localizador de registros disponible durante la etapa **606** y lo devuelve (Sí) al dispositivo cliente **130** durante la etapa **607**.

35 La unidad **120** de gestión de identificadores únicos actualiza el estado de disponibilidad del localizador de registros en la página que ha cargado durante la etapa **608**. La unidad **120** de gestión de identificadores únicos no actualiza directamente la unidad **102** de índice (para evitar accesos a la base de datos y, por tanto, para aumentar el rendimiento).

40 El sistema permite garantizar que ninguna otra unidad de gestión de identificadores únicos pueda leer la página (ahora desactualizada) en la base de datos antes de que la unidad **140** de actualización haya regenerado la información contenida en la misma.

El proceso no finaliza, a menos que un operador apague las unidades de gestión de identificadores únicos. Éstas siguen respondiendo consultas por dispositivos cliente.

Si se completa la creación del PNR, sus datos se almacenan en la unidad **101** de memoria durante la etapa **610**.

45 El dispositivo cliente **130** confirma la creación del PNR con el localizador de registros asociado que ha recuperado de la única gestión de identificadores únicos y envía los datos correspondientes a la unidad **110** de gestión de transacciones (etapa **609**). Después, la unidad de gestión de transacciones envía los datos a la base de datos durante la etapa **611** y el almacenamiento se realiza durante la etapa **610**.

50 En el proceso arriba descrito, las páginas a las que apuntan los punteros R y W son sin duda alguna diferentes, por lo tanto, la escritura de la base de datos no se puede realizar en un localizador de registros que actualmente está siendo leído por la unidad **140** de actualización. Esto garantiza la exactitud del acceso a la base de datos por la unidad **140** de actualización, incluso si no bloquea los PNR.

La **Figura 7** es un flujo de datos que ilustra las etapas para actualizar el sistema de base de datos de acuerdo con la asignación de localizadores de registros a PNR.

La unidad **140** de actualización realiza la actualización independientemente de las operaciones de las unidades **120**, **121**, **122** de gestión de identificadores únicos.

5 La unidad **150** de control activa la actualización durante etapa **701**.

Después, la unidad **140** de actualización lee, durante la etapa **702**, los valores de los punteros **503** a **506** (B, R, W, E) de la unidad **103** de punteros de control.

10 La solicitud de lectura en realidad no se dirige directamente a la unidad de punteros de control. La solicitud de lectura va al sistema de base de datos, que realiza el procesamiento necesario para leer los valores de la unidad de punteros de control y enviarlos de vuelta a la unidad de actualización. El flujo de datos está simplificado para una mayor claridad y concisión.

15 Después del inicio, la unidad de actualización almacena el valor inicial del puntero R durante la etapa **703**. Después espera un tiempo denominado "vida útil de la página" (correspondiente a un período de consulta de la página) durante la etapa **704**. El tiempo de duración de la página es igual al tiempo durante el cual la unidad de administración de identificadores únicos mantiene las páginas.

La etapa **704** permite garantizar que cualquier página cargada en la memoria por una unidad de gestión de identificadores únicos ha sido realmente utilizada o descartada al final de la etapa.

Después, la unidad de actualización carga el valor del puntero W desde la unidad de punteros de control durante la etapa **705** y calcula el bloque de localizadores de registros contenidos en la página correspondiente en la etapa **706**.

20 La unidad de actualización consulta después al sistema de base de datos sobre el estado de disponibilidad de estos localizadores de registros durante la etapa **707**. La unidad de actualización comprueba si los localizadores de registros están almacenados en la unidad **101** de memoria que comprende los PNR activos.

Al utilizar una consulta por página, la unidad de actualización carga información para un gran número de localizadores de registros a la vez, lo que reduce el tráfico de la base de datos y el impacto de la latencia de la red.

25 Una vez que el sistema de base de datos devuelve los datos, la unidad de actualización los codifica en la página (etapa **708**) y los almacena en la unidad **102** de índice de la base de datos (etapa **709**). Finalmente, la unidad de actualización incrementa el valor del puntero W en la etapa **710**. El valor incrementado del puntero W se almacena entonces en la unidad **103** de punteros de control en la etapa **711**. El proceso anterior se repite después para procesar otra página.

30 La unidad de actualización comprueba regularmente, en cada implementación del proceso de la Figura 7, si el valor W alcanza el valor del puntero R como está almacenado (en la etapa **703**) antes de la etapa **704** de espera porque las páginas después de este punto todavía pueden estar siendo utilizadas por nodos en línea.

La **Figura 8** es un diagrama de bloques esquemático de un dispositivo **800** para implementar una o más realizaciones de la invención. Las diversas partes del sistema arriba descritas pueden tener la misma estructura que el dispositivo **800**.

35 El dispositivo **800** comprende un bus de comunicación conectado a:

- una unidad central **801** de procesamiento, tal como un microprocesador, denominada CPU;
- una memoria **802** de acceso aleatorio, denominada RAM, para almacenar el código ejecutable del método de realizaciones de la invención, así como los registros adaptados para registrar variables y parámetros necesarios para implementar un método de acuerdo con realizaciones, la capacidad de memoria de la misma se puede ampliar mediante una RAM opcional conectada a un puerto de expansión, por ejemplo;
- 40 - una memoria **803** de solo lectura, denominada ROM, para almacenar programas informáticos para implementar realizaciones de la invención;
- una interfaz **804** de red, que normalmente está conectada a una red de comunicaciones a través de la cual se transmiten o reciben datos digitales que han de ser procesados; la interfaz **804** de red puede consistir en una
- 45 única interfaz de red o estar compuesta por un conjunto de diferentes interfaces de red (por ejemplo, interfaces cableadas e inalámbricas, o diferentes tipos de interfaces cableadas o inalámbricas); los datos se escriben en la interfaz de red para su transmisión o se leen desde la interfaz de red para su recepción bajo el control de la aplicación de *software* que se ejecuta en la CPU **801**;
- una interfaz **805** de usuario para recibir entradas de un usuario o para mostrar información a un usuario;
- 50 - un disco duro **806**, denominado HD;

- y un módulo **807** de E/S para recibir/enviar datos desde/hacia dispositivos externos tales como una fuente o pantalla de video.

5 El código ejecutable se puede almacenar bien en la memoria **803** de solo lectura, bien en el disco duro **806** o en un medio digital extraíble tal como, por ejemplo, un disco. Según una variante, el código ejecutable de los programas se puede recibir por medio de una red de comunicaciones, a través de la interfaz **804** de red, para almacenarlo en uno de los medios de almacenamiento del dispositivo **800** de comunicación, tal como el disco duro **806**, antes de ser ejecutado.

10 La unidad central **801** de procesamiento está adaptada para controlar y dirigir la ejecución de las instrucciones o partes de código de *software* del programa o los programas de acuerdo con realizaciones de la invención, cuyas instrucciones se almacenan en uno de los medios de almacenamiento arriba mencionados. Una vez encendida, la CPU **801** puede ejecutar instrucciones desde la memoria RAM **802** principal relacionadas con una aplicación de *software* después de que esas instrucciones se hayan cargado desde la ROM **803** de programa o el disco duro (HD) **806**, por ejemplo. Dicha aplicación de *software*, cuando es ejecutada por la CPU **801**, hace que se lleven a cabo las etapas de un método de acuerdo con realizaciones.

15 Si bien la invención se ha ilustrado y descrito en detalle en los dibujos y en la anterior descripción, dicha ilustración y dicha descripción han de ser consideradas como ilustrativas o ejemplares y no restrictivas, no estando limitada la invención a la realización descrita. Los expertos en la técnica pueden entender y efectuar otras variaciones de la realización descrita al poner en práctica la invención reivindicada, a partir de un estudio de los dibujos, la descripción y las reivindicaciones adjuntas.

20 En las reivindicaciones adjuntas, la expresión "que comprende" no excluye otros elementos o etapas, y el artículo indeterminado "un" o "una" no excluye una pluralidad. El simple hecho de que diferentes características se enumeren en reivindicaciones subordinadas diferentes entre sí no indica que no se pueda utilizar de forma ventajosa una combinación de estas características. Cualquier signo de referencia en las reivindicaciones no ha de ser interpretado como limitativo del alcance de la invención.

25

REIVINDICACIONES

1. Un sistema para asociar identificadores con objetos representados en el sistema, en el que, en un momento dado, un objeto dado representado en una base de datos del sistema se asocia con un identificador único,

comprendiendo el sistema:

- 5 - una base de datos (100), comprendiendo dicha base de datos:
 - una unidad (101) de memoria configurada para almacenar dichos objetos y sus respectivas asociaciones con dichos identificadores,
 - una unidad (102) de índice configurada para almacenar un estado de disponibilidad de los identificadores del sistema, y
 - 10 • una unidad (103) de punteros de control configurada para almacenar punteros (503, 504, 505, 506) para datos en la unidad de índice;
- al menos una unidad (120, 121, 122) de gestión de identificadores configurada para acceder a dicha base de datos (100); y
- al menos una unidad (140) de actualización configurada para actualizar el estado de disponibilidad de los 15 identificadores del sistema,

en donde:

- dichos identificadores pertenecen a un conjunto finito de identificadores organizados, dentro de dicha unidad (102) de índice, de acuerdo con páginas (P1, P2, P3, P4, P5) de identificadores,
 - estando representada cada página por un mapa de bits,
 - 20 • en donde cada bit del mapa de bits corresponde a un valor de identificador respectivo, y
 - cada valor de bit corresponde a la disponibilidad del identificador correspondiente,
- dichos punteros para datos en el índice comprenden al menos
 - un primer puntero (505) para una última página cargada por una unidad (120, 121, 122) de gestión de identificadores del sistema, y
 - 25 • un segundo puntero (506) para una última página actualizada por una unidad (140) de actualización del sistema, y

en donde:

- dicha al menos una unidad (120, 121, 122) de gestión de identificadores está configurada además para:
 - cargar (604) una primera página de identificadores basada en dicho primer puntero (505) para asignar (607) 30 un identificador de dicha página de identificadores a un objeto que ha de ser almacenado (610) en la base de datos, sobre la base de un mapa de bits correspondiente a dicha primera página cargada, y
- dicha al menos una unidad (140) de actualización está configurada además para:
 - determinar (706) identificadores correspondientes a una segunda página de identificadores basada en dicho 35 segundo puntero (506),
 - acceder (707) a dicha unidad de memoria para comprobar la disponibilidad de los identificadores determinados, y
 - actualizar (708) un mapa de bits que representa dicha segunda página cargada sobre la base de la disponibilidad de dichos identificadores determinados, y
- dicha base de datos está configurada además para:
 - 40 • almacenar (610) en dicha unidad (101) de memoria un objeto y su identificador asignado tras su activación en el sistema, y
 - borrar de dicha unidad (101) de memoria un objeto y su identificador asignado tras su desactivación en el sistema.

2. Un sistema según la reivindicación 1, en el que dicha al menos una unidad (120, 121, 122) de gestión de identificadores está configurada además para descartar dicha primera página de identificadores cargada, después de un tiempo que no supera una duración máxima de la vida útil de la página.
- 5 3. Un sistema según la reivindicación 2, en el que dicha al menos una unidad (140) de actualización está configurada además para esperar (704) dicha duración máxima de la vida útil de la página antes de determinar un identificador correspondiente a dicha segunda página de identificadores.
4. Un sistema según una cualquiera de las reivindicaciones precedentes, en el que al menos una unidad de gestión de identificadores o una unidad de actualización, o ambas, están configuradas además para comprobar si dicho primer puntero y dicho segundo puntero apuntan simultáneamente a una misma página de identificadores.
- 10 5. Un sistema según una cualquiera de las reivindicaciones precedentes, en el que dichas páginas de identificadores se ordenan de forma cíclica.
6. Un sistema según la reivindicación 5, en el que, antes de asignar un identificador a un objeto que ha de ser almacenado en la base de datos, se comprueban los punteros para verificar que el segundo puntero no esté apuntando a la página de identificadores situada junto a la página a la que apunta el primer puntero.
- 15 7. Un sistema según la reivindicación 5 o 6, en el que, antes de actualizar un mapa de bits de una página de identificadores, se comprueban los punteros para verificar que el primer puntero no esté apuntando a la página de identificadores situada junto a la página a la que apunta el segundo puntero.
8. Un método para asociar identificadores con objetos representados en un sistema, en el que, en un momento dado, un objeto dado representado en una base de datos del sistema se asocia con un identificador único,
- 20 comprendiendo el sistema:
- una base de datos (100), comprendiendo dicha base de datos:
 - una unidad (101) de memoria configurada para almacenar dichos objetos y sus respectivas asociaciones con dichos identificadores,
 - una unidad (102) de índice configurada para almacenar un estado de disponibilidad de los identificadores del sistema, y
 - una unidad (103) de punteros de control configurada para almacenar punteros (503, 504, 505, 506) para datos en la unidad de índice,
 - al menos una unidad (120, 121, 122) de gestión de identificadores configurada para acceder a dicha base de datos (100), y
 - al menos una unidad (140) de actualización configurada para actualizar el estado de disponibilidad de los identificadores en el sistema,
- 25
- en donde:
- dichos identificadores pertenecen a un conjunto finito de identificadores organizados, dentro de dicha unidad (102) de índice, de acuerdo con páginas (P1, P2, P3, P4, P5) de identificadores,
 - estando representada cada página por un mapa de bits,
 - en donde cada bit del mapa de bits corresponde a un valor de identificador respectivo, y
 - cada valor de bit corresponde a la disponibilidad del identificador correspondiente, y
 - dichos punteros para datos en el índice incluyen al menos
 - un primer puntero (505) para una última página cargada por una unidad (120, 121, 122) de gestión de identificadores del sistema, y
 - un segundo puntero (506) para una última página actualizada por una unidad (140) de actualización del sistema,
- 35
- comprendiendo el método las siguientes etapas, realizadas por dicha al menos una unidad (120, 121, 122) de gestión de identificadores:
- cargar (604), mediante dicha al menos una unidad (120, 121, 122) de gestión de identificadores, una primera página de identificadores basada en dicho primer puntero (505),
- 40
- 45

- asignar (607) un identificador de dicha página de identificadores a un objeto que ha de ser almacenado (610) en la base de datos, sobre la base de un mapa de bits correspondiente a dicha primera página cargada,

comprendiendo el método las siguientes etapas, realizadas por dicha al menos una unidad (140) de actualización:

- 5 - determinar (706) identificadores correspondientes a una segunda página de identificadores sobre la base de dicho segundo puntero (506),
- acceder (707) a dicha unidad de memoria para comprobar la disponibilidad de los identificadores determinados, y
- actualizar (708) un mapa de bits que representa dicha segunda página cargada sobre la base de la disponibilidad de dichos identificadores determinados, y

10 comprendiendo el método las siguientes etapas, realizadas por dicha base de datos:

- almacenar (610) en dicha unidad (101) de memoria un objeto y su identificador asignado tras su activación en el sistema, y
- borrar de dicha unidad (101) de memoria un objeto y su identificador asignado tras su desactivación en el sistema.

15 9. Un método según la reivindicación 8, en el que dicha al menos una unidad (120, 121, 122) de gestión de identificadores descarta además dicha primera página de identificadores cargada, después de un tiempo que no supera una duración máxima de la vida útil de la página.

20 10. Un método según la reivindicación 9, en el que dicha al menos una unidad (140) de actualización espera (704) además dicha duración máxima de la vida útil de la página antes de determinar un identificador correspondiente a dicha segunda página de identificadores.

11. Un método según una cualquiera de las reivindicaciones 8 a 10, en el que al menos una unidad de gestión de identificadores o una unidad de actualización, o ambas, comprueban además si dicho primer puntero y dicho segundo puntero apuntan simultáneamente a una misma página de identificadores.

25 12. Un método según una cualquiera de las reivindicaciones 8 a 11, en el que dichas páginas de identificadores se ordenan de forma cíclica.

13. Un método según la reivindicación 12, en el que, antes de asignar un identificador a un objeto que ha de ser almacenado en la base de datos, se comprueban los punteros para verificar que el segundo puntero no esté apuntando a la página de identificadores situada junto a la página a la que apunta el primer puntero.

30 14. Un método según la reivindicación 12 o 13, en el que, antes de actualizar un mapa de bits de una página de identificadores, se comprueban los punteros para verificar que el primer puntero no esté apuntando a la página de identificadores situada junto a la página a la que apunta el segundo puntero.

15. Un producto de programa informático que comprende instrucciones para implementar un método según una cualquiera de las reivindicaciones 8 a 14 cuando el programa se carga y ejecuta mediante un aparato programable.

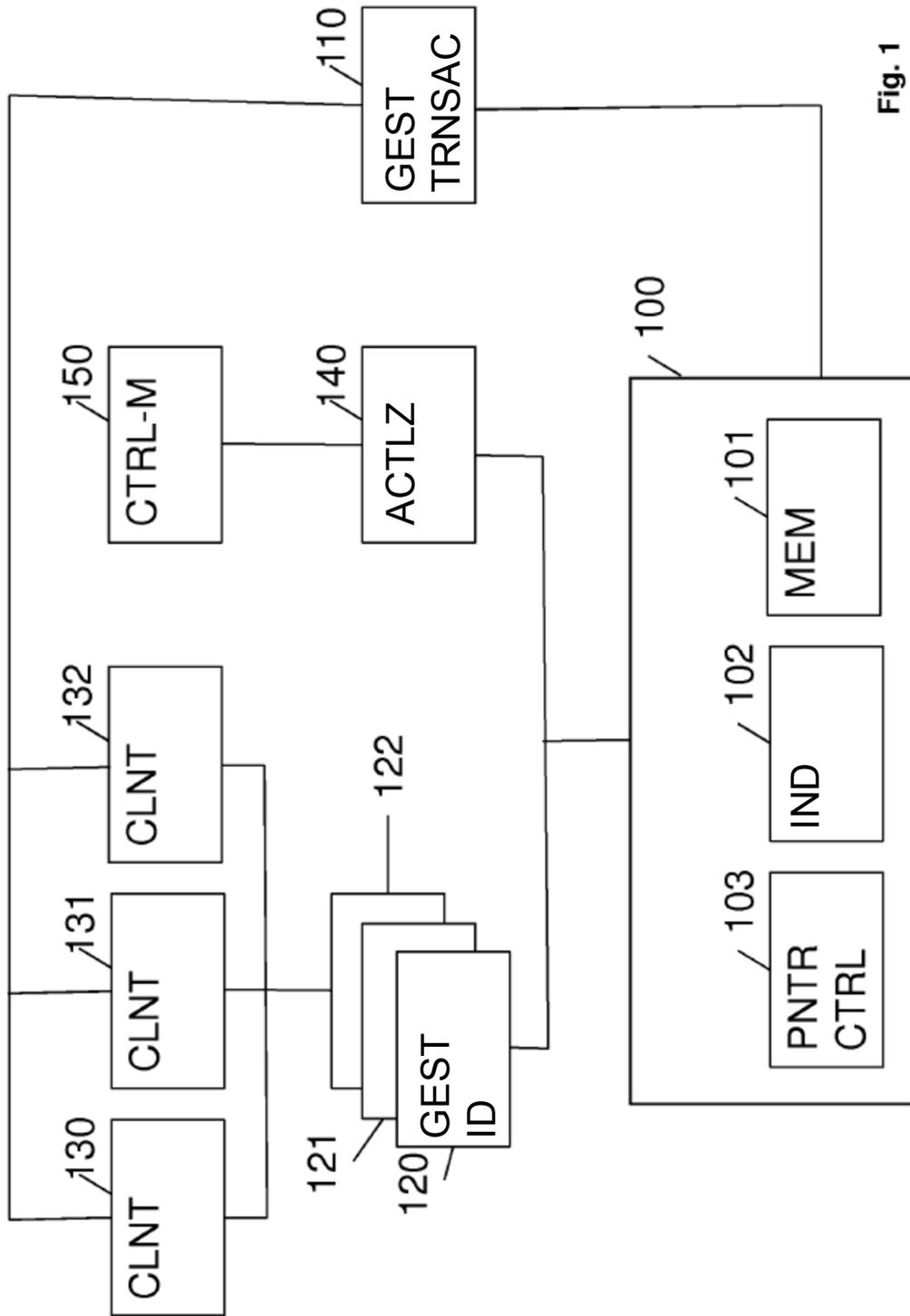
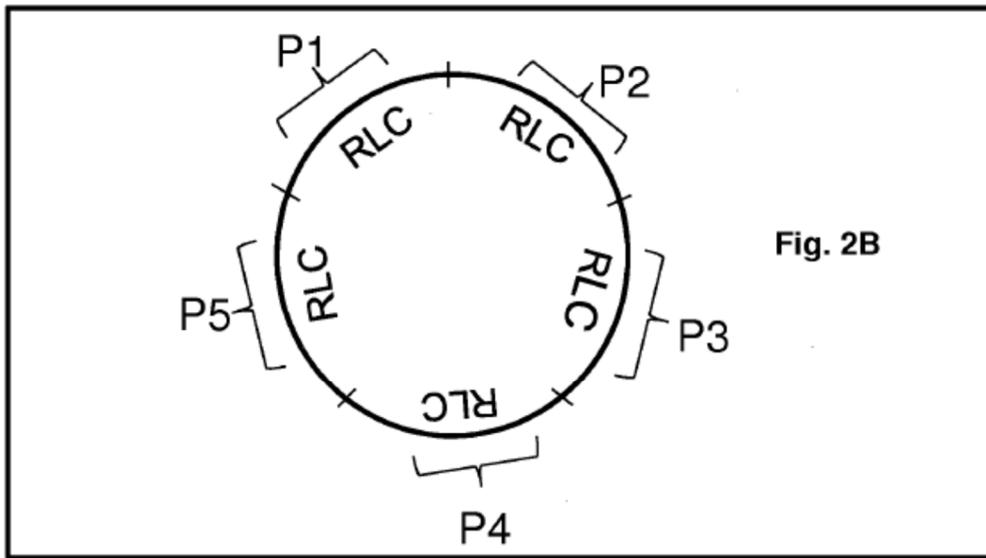
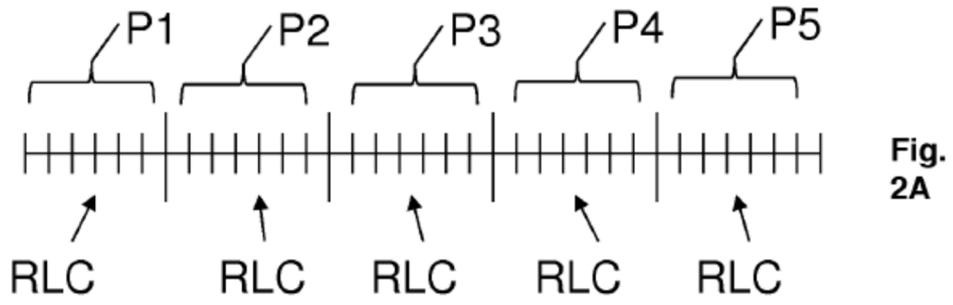


Fig. 1

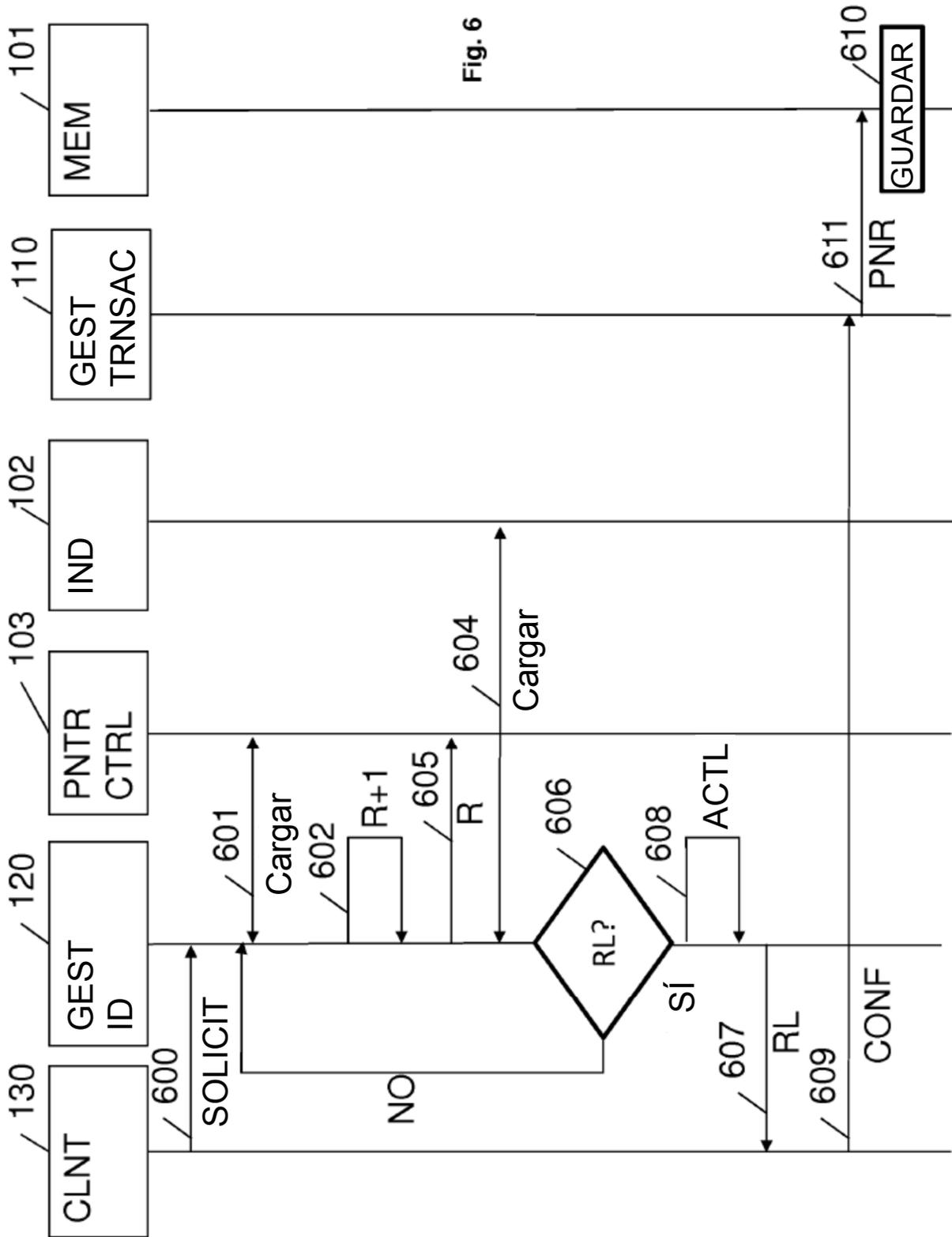


300	301	302
PNR	RLC	
303		
PNR _i	RLC _i	
304		
PNR _k	RLC _k	

Fig. 3

400	401	402
	Pg	BM
	P1	01001.....101
	P2	11101.....001
	.	.
	.	.
	.	.
	.	.
	.	.
RLC		

Fig. 4



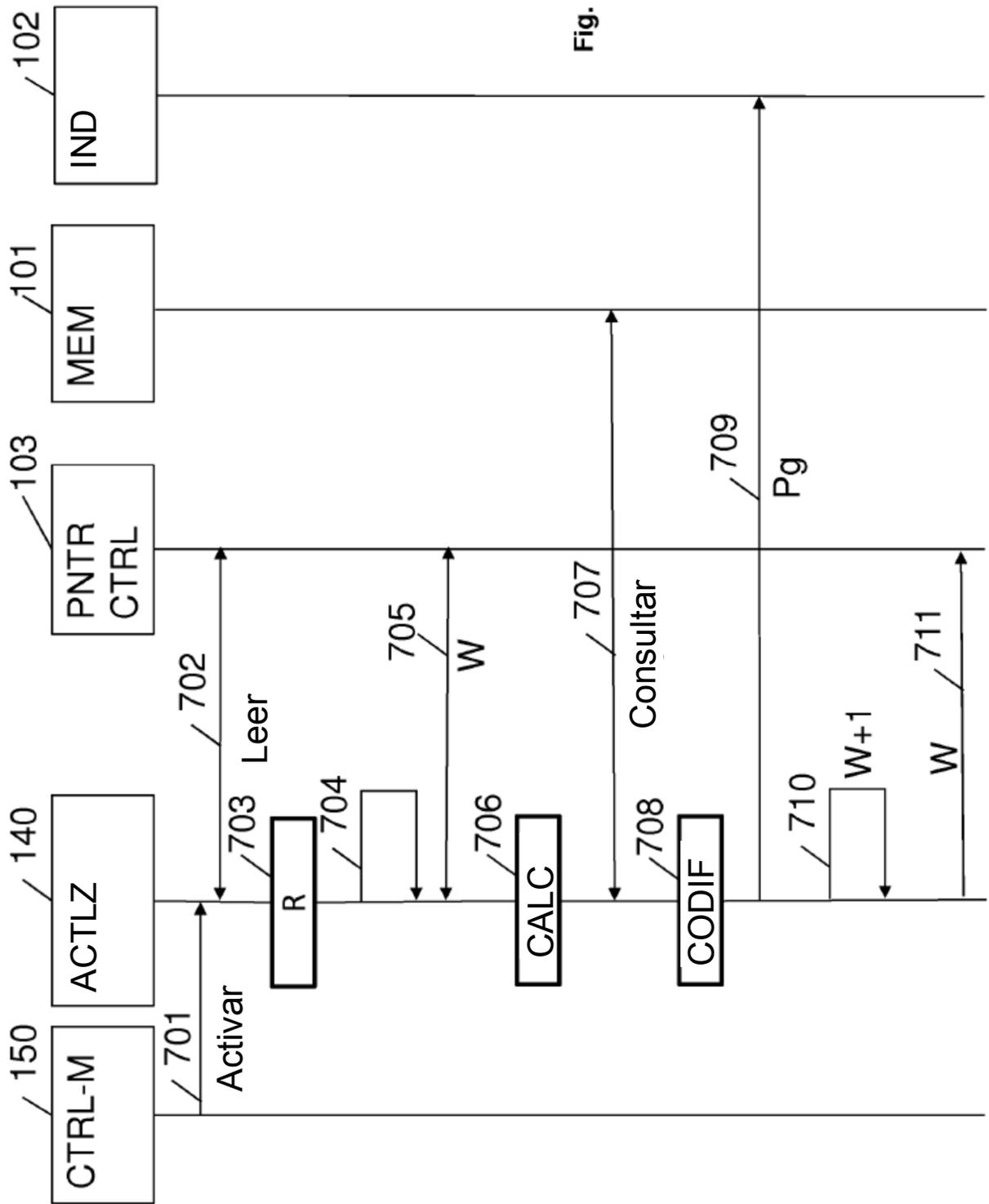


Fig. 7

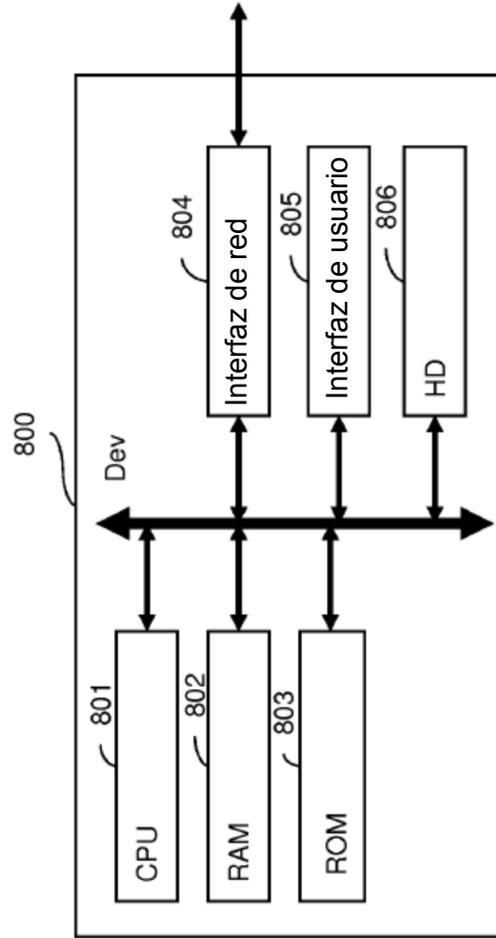


Fig. 8