

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 635 719**

51 Int. Cl.:

G06F 17/40 (2006.01)

G06F 17/00 (2006.01)

G06F 17/30 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **19.01.2007 PCT/US2007/001394**

87 Fecha y número de publicación internacional: **30.08.2007 WO07097846**

96 Fecha de presentación y número de la solicitud europea: **19.01.2007 E 07748983 (9)**

97 Fecha y número de publicación de la concesión europea: **10.05.2017 EP 1989646**

54 Título: **Modelo de sincronización de participante de pares**

30 Prioridad:

15.02.2006 US 354677

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

04.10.2017

73 Titular/es:

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

72 Inventor/es:

**KHOSRAVY, MOE;
PFENNING, JORG-THOMAS;
NOVIK, LEV;
BECKEMAN, MICHAEL, S.;
THOMAS, MYRON, C.;
SADOVSKY, VLADIMIR y
LEVY, MARC**

74 Agente/Representante:

CARPINTERO LÓPEZ, Mario

ES 2 635 719 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

Modelo de sincronización de participante de pares

5 En el mundo de la tecnología y el manejo de la información digital de hoy en día, los individuos pueden almacenar información o datos en una diversidad de diferentes dispositivos y localizaciones. A menudo el usuario almacena la misma información en más de un dispositivo y/o localización. El usuario podría desear que todos los diversos almacenes de datos tuvieran la misma información sin tener que introducir manualmente los mismos cambios en cada almacén de datos. La replicación es un proceso usado para asegurar que cada almacén de datos tiene la misma información.

10 Por ejemplo, un usuario puede mantener una libreta de direcciones electrónica en una diversidad de diferentes dispositivos o localizaciones. El usuario puede mantener la libreta de direcciones, por ejemplo, en un gestor de información personal almacenado en su ordenador de sobremesa, en su ordenador portátil, en un asistente digital personal (PDA), en un gestor de contactos en línea y similares. El usuario puede modificar las libretas de direcciones electrónicas en cada localización, por ejemplo, añadiendo un contacto, borrando un contacto o cambiando información de contacto. La replicación se usa para asegurar que el cambio realizado en un dispositivo particular se refleja finalmente en los almacenes de datos de los otros dispositivos del usuario.

15 La replicación cada vez se hace más complicada ya que aumenta el número de dispositivos y/o servicios que usa un usuario particular y/o el tamaño o capacidades de procesamiento de estos dispositivos disminuyen. Por ejemplo, muchos usuarios tienen unidades de almacenamiento en miniatura, memoria extraíble, PDA, teléfonos, dispositivos de música portátiles y así sucesivamente. La información tal como los registros de contactos sería útil tenerla en muchos de estos dispositivos y/o sincronizarse entre estos dispositivos, sin embargo muchos de estos tipos de dispositivos ni siquiera tienen un procesador informático que a menudo se requiere para participar en un proceso de sincronización típico. Estos problemas de sincronización pueden agravarse adicionalmente cuando se ve implicada la compartición de datos, tal como que múltiples usuarios compartan un calendario de grupo.

20 El documento EP 1 369 791 A2 se refiere a un aparato de soporte de terminal portátil, un procedimiento de sincronización de datos y un aparato de terminal portátil que aumentan la frecuencia de sincronismo de datos mediante operación sencilla y que puede potenciar la consistencia de datos objetivo de sincronización. El comunicador de base es un procesador que transmite y recibe datos actualizados a y desde la base en respuesta a una solicitud desde el proceso de sincronización de datos. El comunicador de Pc es un procesador que comunica con el PC para sincronismo de datos, y la sección de almacenamiento de datos objetivo de sincronización es una sección de almacenamiento que almacena datos objetivo de sincronización. La sección de almacenamiento de estado síncrono es una sección de almacenamiento que almacena los estados síncronos de datos objetivo de sincronización. Los estados síncronos incluyen los estados "actualizado", "no procesado", "completado" y "en conflicto". El procesador de sincronización de datos solicita que un usuario designe un procedimiento de resolución para los datos para los que tiene lugar un conflicto y resuelva el conflicto basándose en el procedimiento designado por el usuario. La sección de almacenamiento de datos actualizados almacena los datos objetivo de sincronización actualizados por el PDA o el PC junto con información tal como el nombre de los datos y actualiza el tiempo de los mismos. El PDA y la base están conectados entre sí mediante una comunicación alámbrica o inalámbrica. Si el PDA está instalado en la base mientras el PDA está activo, los datos pueden transmitirse o recibirse entre el PDA y la base. Si el PC y la base están conectados entre sí mediante una comunicación alámbrica o inalámbrica, y si el PC está activo, entonces los datos pueden transmitirse o recibirse siempre entre el PC y la base.

Sumario

Es el objeto de la presente invención mejorar la sincronización de datos entre dispositivos que tienen diferentes capacidades.

Este objeto se resuelve mediante la materia objeto de las reivindicaciones independientes.

45 Se definen realizaciones preferidas en las reivindicaciones dependientes.

Se desvelan diversas tecnologías y técnicas que mejoran la sincronización de datos entre tipos de dispositivos y/o servicios variables. Un dispositivo o servicio participante completo recibe una solicitud desde otro dispositivo o servicio para realizar una operación de sincronización usando un motor de sincronización. El otro dispositivo o servicio comunica con el motor de sincronización del participante completo a través de una implementación de una interfaz de controlador. El motor de sincronización organiza la comunicación entre los diversos controladores instalados en un sistema para controlar la sincronización entre puntos finales. Una vez conectado al motor de sincronización, se inspeccionan los controladores para determinar los escenarios de sincronización en los que pueden tomar parte, o simplemente, sus niveles de participación según se definen en un modelo de participante de pares.

55 En una implementación, el motor de sincronización determina si el dispositivo o servicio es un participante completo, un participante parcial o un participante sencillo. El dispositivo o servicio es un participante sencillo si tiene un almacén de datos para datos sincronizados y no tiene almacén de conocimientos. El participante sencillo no es

responsable de rastrear qué cambios realiza a los datos. El dispositivo o servicio es un participante parcial si tiene un almacén de datos para datos sincronizados y un almacén de conocimientos, pero no puede entender los conocimientos. El participante parcial es responsable de rastrear qué cambios realiza a los datos. El dispositivo o servicio es un tipo de participante completo si tiene un almacén de datos para datos sincronizados y un almacén de conocimientos y entiende los conocimientos y alguna o todas las operaciones en ellos. Los conocimientos se refieren a “metadatos de sincronización”. El motor de sincronización realiza la operación de sincronización con el dispositivo usando un conjunto de lógica que es apropiada para el tipo de dispositivo o servicio. Una implementación de esta arquitectura proporciona una comunidad de sincronización de 2 sentidos de múltiples maestros, y permite que los dispositivos y/o servicios con capacidades de procesamiento y/o de almacenamiento limitadas (tales como unidades de almacenamiento en miniatura, algunos asistentes digitales personales y/o teléfonos, etc.) participen en algún nivel en el proceso de sincronización. Sincronización de múltiples maestros significa permitir a dos o más participantes teniendo cada uno réplicas escribibles de los mismos datos que converjan y se sincronicen se hayan comunicado o no antes.

Como un ejemplo no limitante, un dispositivo o servicio participante parcial puede participar en una operación de sincronización de dos sentidos de múltiples maestros en una implementación debido a los conocimientos almacenados en el participante parcial, incluso aunque el participante parcial ni siquiera entienda los conocimientos. Como otro ejemplo no limitante, un participante sencillo que tiene un almacén de datos para almacenar datos replicados pero no conocimientos (tal como una unidad de almacenamiento en miniatura) puede participar en un proceso de sincronización con un participante completo en una implementación.

Este resumen se proporcionó para introducir una selección de conceptos en una forma simplificada que se describen adicionalmente a continuación en la descripción detallada. Este resumen no se pretende para identificar características clave o características esenciales de la materia objeto reivindicada, ni se pretende que se use como una ayuda para determinar el alcance de la materia objeto reivindicada.

Breve descripción de los dibujos

La Figura 1 es una vista esquemática de un modelo de participante de pares de sincronización de una implementación que muestra una representación gráfica de un participante completo, participante parcial y participante sencillo.

La Figura 2 es una vista esquemática de un modelo de participante de pares de sincronización de una implementación que muestra una representación tabular de un participante completo, participante parcial y participante sencillo.

La Figura 3 es una vista esquemática de un sistema de sincronización de una implementación con controladores para interconectar con dispositivos participantes.

La Figura 4 ilustra un sistema informático ejemplar que es un entorno de operación adecuado para una o más implementaciones, tal como para operar una aplicación de sincronización en un dispositivo de participante completo.

La Figura 5 es una vista esquemática de una aplicación de sincronización de una implementación.

La Figura 6 es un diagrama de flujo de proceso de alto nivel para una implementación del sistema.

La Figura 7 es un diagrama de flujo de proceso para una implementación que ilustra las etapas implicadas en actualizar y sincronizar datos usando un dispositivo participante parcial.

La Figura 8 es un diagrama de flujo de proceso para una implementación que ilustra las etapas implicadas en tener el dispositivo participante parcial que rastrear cambios a los datos actualizando el registro con un nuevo recuento de casilla de marca u otro identificador.

La Figura 9 es un diagrama de flujo de proceso para una implementación que ilustra las etapas implicadas en tener el dispositivo participante parcial que rastrear cambios que realizó a los datos almacenando de manera separada un identificador de registro y fecha/hora para el cambio.

Las Figuras 10-11 ilustran un registro de ejemplo en un dispositivo participante parcial antes de la modificación y después de la modificación por el dispositivo participante parcial.

La Figura 12 ilustra un registro de ejemplo en un dispositivo participante parcial antes de la modificación.

La Figura 13 ilustra un registro de rastreo de cambio de ejemplo en un dispositivo participante parcial para rastrear cambios realizados al registro de la Figura 12.

La Figura 14 ilustra el registro de ejemplo de la Figura 12 que se actualiza por un dispositivo participante completo después de determinar que el participante parcial modificó los datos como se describe en el registro de rastreo de cambio de la Figura 13.

La Figura 15 es un ejemplo de un registro de conocimientos almacenado en un dispositivo participante parcial o dispositivo participante completo para una implementación.

La Figura 16 es un diagrama de flujo de proceso para una implementación que ilustra las etapas implicadas al actualizar y sincronizar datos usando un dispositivo participante sencillo.

La Figura 17 es una vista esquemática de una comunidad de sincronización ejemplar para una implementación que tiene múltiples dispositivos y controladores.

La Figura 18 ilustra un ejemplo de una comunidad de sincronización para una implementación.

La Figura 19 ilustra una ilustración de participante y temporal de una implementación que muestra un cambio que se añade al participante y los conocimientos del participante que se actualizan para incluir el cambio.

La Figura 20 ilustra una implementación de un escenario de replicación temporal entre dos participantes.

La Figura 21 ilustra una implementación de un escenario de detección de conflicto temporal.

La Figura 22 ilustra un ejemplo de asignar los ID de cambio a cambios en un participante en una implementación.

La Figura 23 ilustra una implementación de un escenario de replicación temporal usando vectores de conocimientos.

5 La Figura 24A ilustra una implementación de actualización de conocimiento en un participante posterior a una replicación usando una lista de excepciones.

La Figura 24B ilustra una implementación de actualización de conocimiento en un participante posterior a una replicación usando un máximo por pares de vectores de conocimientos.

10 La Figura 24C ilustra una implementación de actualización de conocimiento en un participante posterior a una replicación donde existen excepciones en los conocimientos actualizados.

La Figura 25 ilustra una topología en estrella de una implementación para implementar replicación incluyendo replicación de sustituto.

La Figura 26A ilustra ejemplos de escenarios de resolución de conflictos en una implementación.

La Figura 26B ilustra otros escenarios de resolución de conflictos en una implementación.

15 **Descripción detallada**

Para los fines de fomentar un entendimiento de los principios de la invención, se hará ahora referencia a las realizaciones ilustradas en los dibujos y se usará lenguaje específico para describir las mismas. Sin embargo, se entenderá que de esta manera no se pretende limitación al alcance. Cualquier alteración y modificaciones adicionales en las realizaciones descritas, y cualquier aplicación adicional de los principios como se describe en el presente documento se contempla como se le ocurriría normalmente al experto en la materia.

20 El sistema puede describirse en el contexto general como una o más técnicas que mejoran la sincronización de datos entre diversos dispositivos que tienen diversas capacidades, pero el sistema también sirve otros fines además de estos. En una implementación, una o más de las técnicas descritas en el presente documento pueden implementarse como características en un programa de sincronización tal como MICROSOFT® ACTIVESYNC®, o a partir de cualquier otro tipo de programa o servicio que participe en un proceso de sincronización entre dispositivos. En otra implementación, una o más de las técnicas descritas en el presente documento se implementan como características con otras aplicaciones que tratan de la sincronización de datos a través de dispositivos y/o servicios. El término dispositivo móvil como se usa en el presente documento se pretende que incluya teléfonos celulares, asistentes digitales personales, reproductores de medios portátiles, teléfonos de voz sobre IP y numerosos otros tipos de dispositivos móviles de niveles de capacidades variables.

25 La Figura 1 es una vista esquemática de un modelo de participante de pares de sincronización de una implementación que muestra una representación gráfica de un participante 10 completo, un participante 20 parcial y un participante 30 sencillo. El término participante se denomina también en el presente documento como "réplica". Participante y réplica se refieren a dispositivos y/o servicios que participan en una comunidad de sincronización. El participante 10 completo tiene un almacén 12 de datos de conocimientos y una capacidad para entender los conocimientos 14. Tiene también un almacén 16 de datos de sincronización para almacenar los datos reales que se sincronizaron, tal como información de contacto u otra información que se sincroniza entre dispositivos. Unos pocos ejemplos no limitantes de participantes completos incluyen ordenadores personales, algunos PDA, algunos teléfonos, algunos otros dispositivos móviles y/u otros dispositivos que pueden almacenar y entender los conocimientos.

30 Cada participante completo mantiene los "conocimientos" en un almacén de datos de conocimientos que facilita la replicación eficaz y mejorada. En una implementación, los conocimientos son metadatos que describen los cambios que son conocidos para un participante dado. Los conocimientos pueden representarse como un vector de parejas o ID de cambio donde cada pareja o ID de cambio representa un ID de réplica y una versión máxima (ID de réplica, versión máxima). El número de parejas en un vector de conocimientos particular puede cambiar a medida que se añaden o eliminan participantes de la comunidad de sincronización. Aunque el vector de conocimientos puede expresarse también de manera diferente, es ventajoso representar de manera concisa los cambios de los cuales tiene conocimiento un participante particular. No hay requisitos de que los conocimientos particulares contengan específicamente un ID de cambio para cada participante en la comunidad de sincronización. Se exige a los participantes de rastrear lo que otros participantes ya conocen, ya que esta información se representa de manera eficaz por los conocimientos del participante.

35 De manera similar al participante 10 completo, el participante 20 parcial también contiene un almacén 22 de datos de conocimientos. A diferencia del participante 10 completo, sin embargo, el participante 20 parcial no tiene capacidad (o una capacidad limitada) para entender los conocimientos 24. El participante parcial incluye un almacén 26 de datos de sincronización para almacenar los datos sincronizados. El participante parcial incluye un almacén 25 de datos de versión para almacenar información relacionada con cambios que realiza para el almacén 26 de datos de sincronización. Ejemplos no limitantes de participantes parciales pueden incluir algunos asistentes digitales personales, teléfonos, algunos otros dispositivos móviles y/u otros tipos de dispositivos que pueden operar un programa sencillo que rastrea cambios realizados en el almacén 26 de datos de sincronización.

40 El participante 30 sencillo tiene un almacén 36 de datos de sincronización, y no almacén de conocimientos.

Ejemplos de participantes sencillos pueden incluir, pero sin limitación, algunas unidades de almacenamiento en miniatura, algunas tarjetas de memoria y/u otros dispositivos que no son capaces de operar un programa sencillo que rastrea cambios realizados en el almacén 36 de datos de sincronización.

5 Volviendo ahora a la Figura 2, se muestra una representación 50 tabular del modelo de participante de pares. El participante 52 sencillo, el participante 54 parcial y el participante 56 completo tienen una o más características. Algunas de estas características se describieron en el análisis de la Figura 1. El participante 52 sencillo, por ejemplo, no puede almacenar los conocimientos 58. El participante sencillo puede sincronizar 60 con un único participante completo a través de un controlador. El participante 54 parcial almacena pero no entiende los conocimientos 62. El participante 54 parcial puede participar en una sincronización 64 de 2 sentidos de múltiples maestros. Como alternativa o adicionalmente, el participante 54 parcial puede sincronizar 66 a través de un controlador en un participante completo. De esta manera, el participante 54 parcial puede participar en un escenario entre pares gestionado a través de uno o más dispositivos participantes completos. Por lo tanto, los participantes parciales pueden sincronizarse entre sí a través del uso de un participante completo. El participante 56 completo entiende y almacena los conocimientos 68, puede participar en un objetivo 70 de sincronización de 2 sentidos de múltiples maestros y puede realizar sincronizaciones 72 entre pares.

15 La Figura 3 es una vista esquemática de un sistema 80 de sincronización de una implementación con controladores para interconectar con uno o más dispositivos participantes. La aplicación 82 de sincronización incluye un motor 84 de organización de sincronización que es responsable de completar el bucle de sincronización entre participantes y transferir cambios actualizados entre otros participantes conectados. Se usan diversos controladores 86, 88 y 90 para permitir a los otros participantes en la comunidad de sincronización comunicarse con el motor 84 de sincronización. El motor 84 de sincronización comunica con los controladores 86, 88 y 90 a través de las interfaces 91, 92 y 97 de controlador, respectivamente. Los controladores 86, 88 y 90 a continuación comunican con los almacenes 93 y 95 de conocimientos, el almacén 94 de datos de sincronización local y el almacén 96 de datos de sincronización remoto para acceder a datos, si fuera aplicable. En una implementación, uno o más de los almacenes de datos mostrados en la Figura 1, tal como el almacén 96 de datos de sincronización remoto, están localizados en uno o más ordenadores o dispositivos separados.

20 Como se muestra en la Figura 4, un sistema informático ejemplar para uso para implementar una o más partes del sistema incluye un dispositivo informático, tal como el dispositivo 100 informático personal. En su configuración más básica, el dispositivo 100 informático típicamente incluye al menos una unidad 102 de procesamiento y memoria 104. Dependiendo de la configuración exacta y tipo de dispositivo informático, la memoria 104 puede ser volátil (tal como RAM), no volátil (tal como ROM, memoria flash, etc.) o alguna combinación de las dos. Esta configuración más básica se ilustra en la Figura 4 mediante la línea 106 discontinua.

25 Adicionalmente, el dispositivo 100 puede tener también características/funcionalidad adicionales. Por ejemplo, el dispositivo 100 puede incluir también almacenamiento adicional (extraíble y/o no extraíble) incluyendo, pero sin limitación, discos o cinta magnética u óptica. Tal almacenamiento adicional se ilustra en la Figura 4 mediante el almacenamiento 108 extraíble y el almacenamiento 110 no extraíble. El medio de almacenamiento informático incluye memoria extraíble y no extraíble, volátil y no volátil, implementada en cualquier procedimiento o tecnología para almacenamiento de información tal como instrucciones legibles por ordenador, estructuras de datos, módulos de programa u otros datos. La memoria 104, el almacenamiento 108 extraíble y el almacenamiento 110 no extraíble son todos ejemplos de medio de almacenamiento informático. Medio de almacenamiento informático incluye, pero sin limitación, RAM, ROM, EEPROM, memoria flash u otra tecnología de memoria, CD-ROM, discos versátiles digitales (DVD) u otro almacenamiento óptico, cassetes magnéticos, cinta magnética, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, o cualquier otro medio que pueda usarse para almacenar información deseada y que pueda accederse mediante el dispositivo 100. Cualquier medio de almacenamiento informático de este tipo puede ser parte del dispositivo 100.

30 El dispositivo 100 informático incluye una o más conexiones 114 de configuración que permiten que el dispositivo 100 informático comunique con otros dispositivos 115, tal como participantes completos, participantes parciales y/o participantes sencillos. El dispositivo 100 informático puede comunicarse también con otros ordenadores y/o aplicaciones 113. El dispositivo 100 puede tener también dispositivo o dispositivos 112 de entrada tal como teclado, ratón, lápiz óptico, dispositivo de entrada de voz, dispositivo de entrada táctil, etc. Pueden incluirse también dispositivo o dispositivos 111 de salida tal como una pantalla, altavoces, impresora, etc. Estos dispositivos son bien conocidos en la técnica y no necesitan analizarse en profundidad en este punto.

35 En una implementación, el dispositivo 100 informático sirve como un dispositivo participante completo para implementar una o más de las técnicas analizadas en el presente documento. En una implementación de este tipo, el dispositivo 100 informático contiene una aplicación 122 de sincronización con un motor 124 de organización de sincronización, así como un almacén 125 de datos de conocimientos y un almacén 126 de datos de sincronización. En una implementación, el almacén 125 de datos de conocimientos y/o el almacén 126 de datos de sincronización se incluyen como parte del medio de almacenamiento informático como se describe en el presente documento, tal como la memoria 104, el almacenamiento 108 extraíble, el almacenamiento 110 no extraíble y/u otro medio de almacenamiento informático. En una implementación, la aplicación 122 de sincronización es la misma que la aplicación 82 de sincronización mostrada en la Figura 3.

Volviendo ahora a la Figura 5 con referencia continuada a la Figura 4, se ilustra una aplicación 200 de sincronización de una implementación. En una implementación, la aplicación 200 de sincronización es uno de los programas de aplicación que residen en el dispositivo 100 informático. Como alternativa o adicionalmente, una o más partes de la aplicación 200 de sincronización pueden ser parte de la memoria 104 de sistema, en otros ordenadores y/o aplicaciones 113, u otras variaciones tales como pudieran ocurrírsele al experto en la técnica de software informático.

La aplicación 200 de sincronización incluye lógica 204 de programa, que es responsable de llevar a cabo alguna o todas las técnicas descritas en el presente documento. La lógica 204 de programa incluye lógica para registrar controladores para cada dispositivo y/o servicio para participar en el proceso 206 de sincronización; lógica para detectar el tipo de dispositivo y/o servicio conectado (participante sencillo, parcial o completo) 208; lógica para recibir conocimientos desde un participante parcial, realizar modificaciones en caso de excepciones, y transferir conocimientos de vuelta al participante 210 parcial; lógica para detectar cambios en participantes sencillos y almacenar conflictos en su propio almacén 212 de datos local; lógica para funcionar con otro participante o participantes completos para sincronizar conjuntos de datos cambiados usando los conocimientos 214; lógica para realizar organización para completar el bucle de sincronización y extraer cambios actualizados entre otros dispositivos y/o servicios 216 conectados; y otra lógica para operar la aplicación 220.

En una implementación, la lógica 204 de programa es operable para ser solicitada mediante programación desde otro programa, tal como usando una única llamada a un procedimiento en la lógica 204 de programa. Como alternativa o adicionalmente, se entenderá que la lógica 204 de programa puede realizarse, como alternativa o adicionalmente, como instrucciones ejecutables por ordenador en uno o más componentes y/o en diferentes variaciones a las mostradas en la Figura 4.

La Figura 6 es un diagrama de flujo de proceso de alto nivel para la aplicación 200 de sincronización. En una forma, el proceso de la Figura 6 se implementa al menos parcialmente en la lógica de operación del dispositivo 100 informático, otros ordenadores/aplicaciones 113, y/u otros dispositivos 115 participantes. El procedimiento comienza en el punto 240 de inicio conectándose el dispositivo o servicio participante a un participante completo (tal como el dispositivo 100 informático o algún dispositivo móvil) con el motor de sincronización (etapa 242). El dispositivo o servicio registra un controlador o comunica de otra manera con el motor de sincronización en el participante completo (etapa 244). El motor de sincronización detecta el tipo de dispositivo o servicio (etapa 246), y ejecuta la lógica de sincronización apropiada basándose en el tipo de participante: sencillo (punto 248 de decisión), parcial (punto 250 de decisión), o completo (punto 252 de decisión). Por ejemplo, si el dispositivo o servicio es un participante sencillo que tiene un almacén de datos de sincronización pero no conocimientos (punto 248 de decisión), el motor de sincronización detecta cambios en el participante sencillo y almacena cualquier conflicto en su propio almacén de datos local en el participante completo (etapa 254).

Si el dispositivo o servicio es un participante parcial que tiene un almacén de datos de sincronización y conocimientos almacenados pero no entendidos (punto 250 de decisión), entonces el dispositivo o servicio proporciona su almacén de conocimientos al motor de sincronización en el participante completo (etapa 258). El participante completo realiza modificaciones a los conocimientos, tal como en el caso de excepciones (etapa 260), y el participante completo transfiere los conocimientos cambiados de vuelta al participante parcial (etapa 262). Los conocimientos llevados en el dispositivo o servicio participante parcial permiten sincronizar de manera segura con múltiples maestros, incluso cuando tengan lugar excepciones (etapa 264). Si el dispositivo o servicio es un participante completo que tiene un almacén de datos de sincronización y también almacena y entiende los conocimientos (punto 252 de decisión), entonces ambos participantes conocen cómo sincronizar de manera óptima conjuntos de datos cambiados usando los conocimientos (etapa 266). Una o más implementaciones para sincronizar datos entre participantes completos se describen en detalle en el análisis en las Figuras 18-26.

Después de determinar el tipo de participante que es el dispositivo o servicio y manejar cambios y conflictos en consecuencia, el motor de sincronización a continuación sigue la organización para completar el bucle de sincronización y extraer cambios actualizados entre otros dispositivos y/o servicios participantes conectados (etapa 256). El proceso finaliza en el punto 268 final.

La Figura 7 es un diagrama de flujo de proceso que ilustra las etapas implicadas en actualizar y sincronizar datos usando un dispositivo participante parcial. En una forma, el proceso de la Figura 7 se implementa al menos parcialmente en la lógica de operación del dispositivo 100 informático, otros ordenadores/aplicaciones 113 y/u otros dispositivos 115 participantes. El procedimiento comienza en el punto 280 de inicio proporcionando a un dispositivo o servicio participante parcial con un almacén de datos para almacenar datos recuperados durante un proceso de sincronización y un almacén de datos de versión para almacenar información con respecto a cambios realizados a los datos (por ejemplo en un vector, cadena y/u otra manera para rastrear cambios que ha observado el participante/maestro o participantes/maestros completos) (etapa 282). Unos pocos ejemplos no limitantes de participantes parciales incluyen algunos servicios web, algunas unidades de almacenamiento en miniatura, algunos dispositivos móviles tales como algunos PDA y/o algunos teléfonos y/u otros dispositivos o servicios que pueden almacenar conocimientos pero no entenderlos.

El participante parcial no sabe cómo entender los conocimientos, sino que simplemente los almacena (etapa 284). El

participante parcial recibe una solicitud desde un usuario para cambiar alguno o todos los datos en el almacén de datos de sincronización (etapa 286). El participante parcial cambia los datos en el almacén de datos según se solicita por el usuario (etapa 288). El participante parcial es responsable de rastrear qué cambios realizó a los datos, y por lo tanto almacena información relacionada con los cambios que realizó basándose en la solicitud del usuario (etapa 290). En una implementación, cuando el participante parcial sincroniza con un participante completo, la sincronización tiene lugar de completa a parcial y de parcial a completa (etapa 292). En una implementación, existen dos sincronizaciones de un sentido separadas. En otras implementaciones, son posibles también otros escenarios de orden y/o sincronización, tal como una sincronización de un sentido única. Durante la sincronización, cualquier cambio que se realizara por el participante parcial se recupera por el participante completo (etapa 294). El participante completo actualiza el almacén de datos y los conocimientos en el participante parcial después de resolver cualquier conflicto (etapa 296). El proceso finaliza en el punto 298 final.

La Figura 8 es un diagrama de flujo de proceso para una implementación que ilustra las etapas implicadas en tener el participante parcial que rastrear cambios a los datos actualizando el registro con un nuevo recuento de casilla de marca u otro identificador. En una forma, el proceso de la Figura 8 se implementa al menos parcialmente en la lógica de operación del dispositivo 100 informático, otros ordenadores/aplicaciones 113 y/u otros dispositivos 115 participantes. El procedimiento comienza en el punto 300 de inicio siendo responsable el participante parcial de rastrear qué cambios realiza a los datos (etapa 302). El participante parcial actualiza el registro de almacén de datos de versión para indicar que tuvo lugar un cambio (etapa 304). Como unos pocos ejemplos no limitantes, el almacén de datos de versión puede actualizarse con un nuevo recuento de casilla de marca, versión u otro identificador que identifique que el registro cambió y/o que identifique el último dispositivo o servicio que realizó el cambio. El participante completo lee el recuento de casilla de marca cambiado u otro identificador para conocer que el registro particular se cambió por el participante parcial (etapa 306). El proceso finaliza en el punto final 308.

La Figura 9 es un diagrama de flujo de proceso que ilustra las etapas implicadas en una implementación donde el participante parcial rastrea cambios que realizó a los datos almacenando por separado un identificador de registro y fecha/hora para el cambio. En una forma, el proceso de la Figura 9 se implementa al menos parcialmente en la lógica de operación del dispositivo 100 informático, otros ordenadores/aplicaciones 113 y/u otros dispositivos 115 participantes. El procedimiento comienza en el punto 310 de inicio siendo responsable el participante parcial de rastrear qué cambios realiza a los datos (etapa 312). El participante parcial rastrea un identificador único que identifica el registro de réplica y una indicación de fecha/hora que indica la fecha/hora de cuándo se cambió el registro (etapa 314). El participante completo lee el identificador único y la indicación de fecha/hora para conocer qué el registro particular cambió (etapa 316). El participante completo actualiza el recuento de casilla de marca u otro identificador que identifica el último dispositivo o servicio para cambiar el registro, que en este caso es el participante parcial (etapa 318). El proceso finaliza en el punto 320 final.

Las Figuras 10-11 ilustran un registro de ejemplo en un dispositivo participante parcial antes de la modificación y después de la modificación por el dispositivo participante parcial de acuerdo con las etapas descritas en la Figura 8. En una implementación, el registro 330 de la Figura 10 incluye un campo 332 de ID de réplica, un campo 334 de recuento de casilla de marca del último participante que cambió el registro, y un campo de recuento de casilla de marca local para cuando el registro proviene del participante 336 completo. Los respectivos valores 338, 340 y 342 en los campos 332, 334 y 336, pueden almacenarse respectivamente en una cadena, vector y/o cualquier otro tipo de representación adecuada para almacenamiento en dispositivos o servicios que tienen capacidad y/o recursos de almacenamiento limitados. Como se ha descrito anteriormente, pueden usarse también numerosas otras variaciones para indicar que un registro particular en el participante parcial ha sido cambiado.

Volviendo ahora a la Figura 11, el registro 343 muestra los datos después de que se han revisado por y en el participante parcial. El campo 332 de ID de réplica ha mantenido un valor de "X1" (350), puesto que es un identificador único para el registro. El campo 334 de recuento de casilla de marca del último ordenador que cambió el registro se ha modificado de un valor de "G66" (340 en la Figura 10) a un valor de "G67" (352). La "G" representa el participante que realizó el cambio, y el "67" es el siguiente número superior disponible en la secuencia de contador de casilla de marca. El recuento de casilla de marca local cuando proviene desde el campo 336 de participante completo mantiene el mismo valor de "34" (354).

La Figura 12 ilustra un registro de ejemplo en un dispositivo participante parcial antes de la modificación de acuerdo con las etapas descritas en la Figura 9. Similar al ejemplo de las Figuras 10 y 11, el registro 360 incluye un valor 368 para el campo 362 de ID de réplica, un valor 370 para el campo 364 de recuento de casilla de marca del último participante que cambió el registro, y un valor 372 para el campo 366 de recuento de casilla de marca local desde el que proviene el participante completo. En esta implementación, el participante parcial actualiza un registro de rastreo de cambio separado en lugar del registro con el recuento de casilla de marca. El registro 360 muestra el registro con el recuento de casilla de marca antes de la modificación de los datos subyacentes por el participante parcial. El registro 375 de la Figura 13 ilustra un registro de rastreo de cambio de ejemplo en un dispositivo participante parcial que es para rastrear cambios realizados al registro de la Figura 12 de acuerdo con las etapas descritas en la Figura 9. El campo 374 de ID de réplica y el campo 376 de fecha/hora para cuando el registro cambió se almacenan en el participante parcial. En este ejemplo, el valor "X1" 378 se almacena para el ID 374 de réplica, y "26-01-06-12:32PM" 380 se almacena para el campo 376 de fecha/hora. Cuando el participante parcial se conecta después a un participante completo, el participante completo recupera e interpreta el registro 375 para determinar que el

participante parcial realizó cambios a los datos subyacentes en el almacén de datos de sincronización. Volviendo ahora a la Figura 14, el participante completo a continuación actualiza el campo 390 de recuento de casilla de marca del registro 381 del último participante que cambió el registro, que en este caso es “G67” para representar el participante parcial y el siguiente número superior de recuento de casilla de marca. El campo 366 de recuento de casilla de marca local se revisa a un valor 392 actualizado, si fuera apropiado. El valor 388 para el campo 362 de ID de réplica se mantiene igual.

La Figura 15 es un ejemplo de un registro de conocimiento almacenado en un dispositivo participante parcial o dispositivo participante completo para una implementación. En el ejemplo mostrado, el registro 396 de conocimientos se representa como un vector de cadena como se describe en el presente documento, indicando los valores 398, 400, 402 y 404 el identificador de participante y número de secuencia para los últimos cambios que se han observado para ese dispositivo particular. Por ejemplo, el valor “G100” (398) significa que este participante ha observado todos los cambios para el dispositivo G a través del registro 100. Estos vectores de conocimientos se describen en mayor detalle en el análisis en las Figuras 18-26.

Volviendo ahora a la Figura 16, se muestra un diagrama de flujo de proceso que ilustra las etapas implicadas en actualizar y sincronizar datos usando un dispositivo participante sencillo. En una forma, el proceso de la Figura 16 se implementa al menos parcialmente en la lógica de operación del dispositivo 100 informático, otros ordenadores/aplicaciones 113 y/u otros dispositivos 115 participantes. El procedimiento comienza en el punto 420 de inicio proporcionando un dispositivo o servicio participante sencillo que tiene un almacén de datos para almacenar datos sincronizados pero no tiene conocimientos (etapa 424). Unos pocos ejemplos no limitantes de dispositivos sencillos incluyen algunas unidades de almacenamiento en miniatura, algunos lápices/tarjetas de memoria, algunos PDA, algunos teléfonos celulares y/u otros dispositivos o servicios que no pueden almacenar y entender los conocimientos. El participante sencillo no puede almacenar o entender los conocimientos (etapa 426), debido a las limitaciones o ajustes de usuario del dispositivo o servicio. El participante sencillo recibe una solicitud desde un usuario para cambiar algunos o todos los datos en el almacén de fecha de sincronización (etapa 428).

El participante sencillo cambia los datos en el almacén de datos de sincronización según se solicita por el usuario (etapa 430). Un ejemplo no limitante de cómo el usuario puede cambiar el almacén de datos de sincronización incluye modificar los datos en un explorador de ficheros desde otro dispositivo tal como un ordenador personal, tal como insertando una unidad de almacenamiento en miniatura en un ordenador personal y a continuación cambiar los contenidos de la unidad de almacenamiento en miniatura. El participante sencillo no es responsable de rastrear qué cambios realizó a los datos, puesto que se supone que no sabe nada (etapa 432). Cuando el participante sencillo sincroniza con el participante completo, la sincronización tiene lugar desde el participante completo al sencillo y a continuación del sencillo al completo (etapa 434). Durante la sincronización, cualquier cambio realizado por el participante sencillo al almacén de datos de sincronización se recupera por el participante completo (etapa 436). El participante completo actualiza el almacén de datos de sincronización en el participante sencillo después de resolver cualquier conflicto (etapa 438). El proceso finaliza en el punto 440 final.

La Figura 17 es una vista esquemática de una comunidad de sincronización ejemplar para una implementación que tiene múltiples dispositivos y controladores. La Figura 17 ilustra un participante 500 completo denominado ordenador personal 1, o “PC1” (506); un participante sencillo denominado dispositivo 1 (514); un participante 502 parcial denominado “Servicio 1”; y un segundo participante 504 completo denominado ordenador personal 2, o “PC2”. Por motivos de ilustración, se supone que el dispositivo 1 (514) es una unidad de almacenamiento en miniatura u otra tarjeta de memoria, el servicio 1 es un servicio de música localizado en un servidor web, y PC1 y PC2 son ordenadores personales, similares al dispositivo 100 informático. El participante 500 completo tiene los controladores 508, 510 y 512, y el participante 504 completo tiene los controladores 516, 518 y 520. Estos controladores son responsables de interactuar con los diversos participantes que son parte de la comunidad de sincronización. Cuando el dispositivo 1 (514) se conecta al PC1 (506), se ejecuta el proceso de sincronización descrito en la Figura 6. Se determina el tipo de participante, que en este caso es sencillo, y a continuación tiene lugar la sincronización entre el dispositivo 1 (514), y el PC1 (506) del participante 500 completo. Una vez que se completa la sincronización entre estos dos participantes (500 y 514), la organización provocará que los otros participantes (502 y 504) se actualicen si están conectados y/o la próxima vez que se conecten al PC1 (506) o al dispositivo 1 (514).

Volviendo ahora a la Figura 18-26, se describe una o más implementaciones para sincronizar datos entre participantes completos (por ejemplo dos ordenadores personales tales como el dispositivo 100). Uno o más de los ejemplos analizados en las Figuras 18-26 podrían aplicarse también al menos en alguna parte a un escenario de participante parcial u otros escenarios descritos en las figuras anteriores. Como alternativa o adicionalmente, una o más de las técnicas analizadas en las Figuras 18-26 pueden implementarse en un dispositivo tal como el dispositivo 100 informático de la Figura 4. El término “réplica” como se usa en el siguiente análisis significa también “participante”.

Las réplicas/participantes en una comunidad de sincronización se replican proporcionando sus propios conocimientos con la réplica con la que se replican. Para reducir la cantidad de datos que representan los conocimientos que debe enviarse entre réplicas de replicación, los conocimientos pueden expresarse como un vector de conocimientos como se ha descrito anteriormente. Por lo tanto, los conocimientos que se envían entre las réplicas no necesitan incluir cada ID de cambio, sino que pueden ser en forma de un vector que representa un

número de ID de cambios. Por ejemplo, si una réplica tiene conocimiento de todos los cambios realizados por una réplica A desde un primer cambio a un décimo cambio, y todos los cambios realizados por una réplica etiquetada B desde un primer cambio a un quinto cambio, la réplica puede enviar un vector de conocimientos A10B5 que indica que la réplica tiene conocimiento de todos los cambios que corresponden a los ID de cambio A1 a A10 y todos los cambios que corresponden a los ID de cambio B1 a B5. Aunque los conocimientos pueden expresarse como un vector de conocimientos, otras implementaciones de la invención contemplan otras expresiones de conocimientos también. Por ejemplo, algunas implementaciones de la invención expresan los conocimientos usando cualquier expresión de conocimientos en la que se puede (1) añadir un cambio a la expresión de conocimientos, (2) comprobar si un cambio está incluido en la expresión de conocimientos, y (3) unir dos expresiones de conocimientos juntas.

La Figura 18 ilustra un ejemplo de una comunidad 1100 de sincronización con la topología ilustrada. La comunidad 1100 de sincronización incluye un número de réplicas y es un ejemplo de un entorno para implementar implementaciones de la presente invención. Las réplicas en la comunidad 1100 de sincronización representan diversos almacenes de datos o dispositivos que pueden incluir, pero sin limitación, ordenadores, ordenadores portátiles, asistentes digitales personales, teléfonos celulares, otros dispositivos inalámbricos, ordenadores de servidor, servicios en línea y similares o cualquier combinación de los mismos.

En la Figura 18, una réplica A 1102 puede acoplarse electrónicamente a una réplica B 1104 a través de un enlace 1106 de comunicación. La réplica A 1102 puede conectarse a través de un enlace 1108 de comunicación a una réplica C 1110. La réplica C 1110 puede conectarse a la réplica B 1104 a través de un enlace 1112 de comunicación. La réplica C 1110 puede conectarse adicionalmente a una réplica D 1114 a través de un enlace 1116 de comunicación. En esta comunidad 1100 de sincronización, aunque no todas las réplicas se conectan directamente a través de enlaces de comunicación, los cambios en cualquiera de las réplicas pueden replicarse a cualquiera de las otras réplicas dentro la comunidad 1100 de sincronización.

Por ejemplo, para la réplica A 1102 a replicarse con la réplica D 1114, las réplicas A 1102 y C 1110 pueden replicarse a través del enlace 1108 de comunicación. Por lo tanto, la réplica C 1110 incluye cambios realizados en la réplica A 1102. Las réplicas C y D a continuación replican a través del enlace 1116 de comunicación, y como tal la réplica D 1114 incluye cambios de la réplica A 1102. De esta manera, la réplica A 1102 puede replicar con la réplica D 1114 sin ninguna clase de enlace directo. De hecho, las réplicas A 1102 y D 1114 pueden incluso no tener conocimiento de la existencia unas de las otras dentro de la comunidad 1100 de sincronización. Los enlaces de comunicación ilustrados pueden ser enlaces alámbricos y/o inalámbricos.

Haciendo referencia ahora a la Figura 19, una implementación de la invención ilustra cómo se gestionan los cambios en una réplica. La Figura 19 muestra una progresión temporal de una réplica A 1200. La réplica A 1200 incluye los conocimientos 1202, en este caso etiquetados K_A , y los cambios 1204 en este caso etiquetados Δ_A . Cada cambio en los cambios 1204 es el contenido de datos actual de un elemento. Un cambio puede ser un nuevo elemento añadido a una réplica incluso aunque no se cambiara el elemento per sé, el borrado de un elemento y similares. Cada uno de los cambios 1204 está asociado con una versión que en una implementación de la invención es un ID de cambio. En particular, un aspecto ventajoso de la invención es que no hay necesidad de mantener un registro de cambios que incluya información acerca de cambios anteriores. En su lugar, cada réplica incluye conocimientos y una base de datos de cambios (es decir elementos actuales) donde cada cambio tiene una versión correspondiente. En el tiempo (1), la réplica A 1200 está en un estado fijo. En el tiempo (2), un usuario introduce un cambio etiquetado X en la réplica A 1200. La Figura 19 muestra el cambio X añadiéndose como un miembro de los cambios 1204. Los conocimientos 1202 se actualizan para incluir un ID de cambio, $\text{ChangelD}(X)$, que está asociado con el cambio X e identifican la adición del cambio X a los cambios 1204. Esta implementación ilustra una manera en la que los cambios a la réplica están asociados con los ID de cambio específicos. Los conocimientos 1202 pueden ser un vector de conocimientos y representan los cambios de los que tiene conocimiento la réplica A 1200. En una implementación de la presente invención, las versiones o los ID de cambio se mantienen para los elementos u objetos en una base de datos y las versiones pueden usarse para identificar lo que necesita replicarse. Como alternativa, puede mantenerse también un registro de cambios.

La Figura 20 ilustra el uso de conocimientos para enumerar cambios durante la replicación. La Figura 20 muestra dos réplicas, en concreto la réplica A 1302 y una réplica B 1304. La réplica A 1302 incluye un conjunto de cambios 1306 en este ejemplo etiquetados Δ_A . La réplica A 1302 incluye adicionalmente los conocimientos 1308, en este ejemplo etiquetados K_A . Los conocimientos 1308 incluyen una lista de los ID de cambio tales como aquellos anteriormente descritos. De manera similar, la réplica B 1304 incluye un conjunto de cambios 1310 cada uno asociado con una versión que es un ID de cambio. Para comenzar la replicación, en el tiempo (1) la réplica A 1302 envía una solicitud de sincronización a la réplica B 1304 que incluye los conocimientos 1308. La réplica B 1304, comparando los conocimientos 1308 a las versiones asociadas con cada uno de los cambios en el conjunto de cambios 1310, puede tomar decisiones con respecto a cuáles de los cambios de la réplica B 1310 ya tiene la réplica A 1302 en sus cambios 1306 y cambios acerca de los que la réplica A tiene conocimiento. Como alternativa, la réplica B 1304 compara los conocimientos 1308 a cada versión del elemento. Por lo tanto, la réplica B 1304 envía a la réplica A 1302 en el tiempo (2) únicamente esa porción de los cambios de la réplica B 1310 que están asociados con versiones que no están incluidas en los conocimientos 1308 de la réplica A 1302 como se ilustra mediante los cambios 1314. Por ejemplo, si el vector de conocimientos de la réplica A era A3B12 y la réplica B tiene cambios

actuales asociados con las versiones que son los ID de cambio B13 y B14, entonces los cambios enviados a la réplica A incluirían aquellos asociados con los ID de cambio B13 y B14. En una implementación, únicamente se envía B14 si B13 y B14 se realizaron al mismo elemento.

5 Además, la réplica B 1304 también envía los conocimientos de la réplica B 1312 a la réplica A 1302. Puesto que la
 10 réplica B 1304 ha enviado todos los cambios 1310 disponibles en la réplica B 1304 todavía no en la réplica A 1302 a
 la réplica A 1302, la réplica A 1302 ahora tiene todos los cambios 1306 que estaban originalmente en la réplica A
 1302, hasta ahora como los cambios 1310 que no se han sustituido por los cambios enviados por la réplica B 1304,
 además de los cambios 1310 que estaban originalmente en la réplica B 1304. La réplica A 1302 tiene
 15 adicionalmente información acerca de todos los cambios de los que tenía conocimiento la réplica B 1304. Por lo
 tanto, la réplica A 1302 puede actualizar sus conocimientos 1308 para reflejar la adición de los cambios 1310. Esto
 se hace de manera sencilla añadiendo los conocimientos de la réplica A 1308 a los conocimientos de la réplica B
 1312 y definiendo ese valor como los conocimientos de la réplica A 1308 tal como se muestra en el tiempo (3) en la
 Figura 20.

15 Como tal, se realiza una replicación eficaz en la que únicamente se replican los cambios necesarios y en la que las
 réplicas individuales replican únicamente para mantener la información con respecto a los cambios que residen
 dentro de la réplica particular y cambios anteriores acerca de los que tienen conocimiento. Aunque este ejemplo
 muestra una replicación completa de todos los cambios en la réplica B a la réplica A, existen casos donde
 únicamente se replican porciones de los cambios. Como tal, únicamente se añaden los ID de cambio que
 corresponden a cambios que se replican a los conocimientos de la réplica que recibe actualizaciones.

20 Además de enumerar cambios, los conocimientos de una réplica pueden usarse también en detección de conflictos.
 Haciendo referencia ahora a la Figura 21, una implementación de la presente invención ilustra cómo puede
 conseguirse la detección de conflictos. La Figura 21 muestra dos réplicas conectadas mediante un enlace
 electrónico (inalámbrico y/o alámbrico) para comunicación y replicación. La réplica A 1402 incluye los conocimientos
 25 1408 y un conjunto de cambios 1406. Como con el ejemplo en la Figura 20, los conocimientos 1408 incluye una
 colección de los ID de cambio asociados con los cambios 1406 y asociados con cambios anteriores. La réplica A
 1402 incluye además, para fines de este ejemplo, un cambio a un elemento realizado en la réplica A 1402. El
 cambios se etiqueta X y X es un miembro de los cambios 1406. De manera similar, la réplica B 1404 incluye los
 conocimientos 1412, una colección de elementos 1410, cada uno con su versión actual (ID de cambio). De manera
 ilustrativa, en el tiempo (1) la réplica A 1402 envía el cambio X a la réplica B 1404.

30 Asociado y enviado con el cambio X existen otros dos valores, en concreto el ID de cambio asociado con el cambio
 X, etiquetado $ChangeID(X)$, y un valor realizado con conocimientos, etiquetado $K_A(X)$. El valor realizado con
 conocimientos son los conocimientos que existieron en la réplica A 1402 en el tiempo que se realizó el cambio X a la
 réplica A 1402. Como alternativa, en algunas implementaciones de la invención lo realizado con conocimientos
 pueden ser los conocimientos que existieron en una réplica cuando se envía un cambio. Los conocimientos actuales
 35 de la réplica A 1408 pueden enviarse también a la réplica B 1404. Como se muestra en el tiempo (2), la réplica B
 1404 compara el elemento cambiado por el cambio X con el elemento cambiado por el cambio Y para determinar si
 el cambio de A (X) puede entrar en conflicto con el estado de B.

Si los cambios hacen referencia a diferentes versiones del mismo elemento, entonces se requiere análisis adicional.
 La réplica B 1404 a continuación comprueba para observar si el cambio de X era conocido para la réplica B 1404
 40 cuando se realizó el cambio Y en la réplica B 1404. El cambio Y tiene un ID de cambio, $ChangeID(Y)$ y un valor
 realizado con conocimientos, $K_B(Y)$, asociado con él. Si $ChangeID(X)$ es un miembro del cambio realizado con
 conocimientos de Y, $K_B(Y)$, entonces no hay conflicto. En otras palabras, el cambio Y se realizó en la réplica B 1404
 con los conocimientos del cambio X realizado en la réplica A 1402. Como tal, el cambio Y ahora representa los datos
 45 más actuales y válidos para las réplicas A y B. Aunque no se muestra en el ejemplo ilustrado mediante la Figura 21,
 en un tiempo posterior, el cambio Y probablemente se enviará a la réplica A 1402 y el elemento asociado con los
 cambios X e Y actualizado para cambiar Y en la réplica A 1402 de una manera descrita en la Figura 20.

Si los cambios X e Y son para el mismo elemento, y $ChangeID(X)$ no aparece en $K_B(Y)$, entonces como se muestra
 en el tiempo (4), se realiza una comprobación para observar si el cambio Y era conocido por la réplica A 1402
 cuando se realizó el cambio X. Esto se hace típicamente comprobado para observar si la enumeración de cambio
 50 para el cambio Y, ilustrada como $ChangeID(Y)$, se incluye en los conocimientos de la réplica A 1408 en el tiempo
 que se realizó el cambio X, $K_A(X)$. Si $ChangeID(Y)$ es un miembro de $K_A(X)$, entonces el cambio X se realizó con
 conocimiento del cambio Y y no hay conflicto. El cambio X es el cambio más actual y válido para el elemento
 particular. Como tal, la réplica B 1404 probablemente se actualizará con el cambio X de una manera como se
 describe en la Figura 20.

55

Si los cambios X e Y son para el mismo elemento, el $\text{ChangelD}(Y)$ no aparece en $K_A(X)$ y $\text{ChangelD}(X)$ no aparece en $K_B(Y)$, entonces existe un conflicto verdadero. En otras palabras, el cambio X y el cambio Y se realizaron independientes entre sí. En este caso, se informará un conflicto y pueden aplicarse diversas reglas de resolución de conflictos para determinar qué cambio, X o Y, es el cambio más actual y válido. Tales reglas pueden incluir

5 comprobar indicaciones de tiempo para determinar qué cambio se realizó más recientemente, resolver siempre conflictos en favor de cierto tipo de réplicas (tal como aquellas almacenadas en servidores) y/o cualquier otra resolución de conflictos adecuada. Como alternativa, en una forma de resolución de conflictos, un elemento con cambios en conflicto puede actualizarse de manera que los cambios en conflicto se unen para formar un nuevo cambio.

10 Haciendo referencia ahora a la Figura 22, se muestra una implementación ejemplar de los ID de cambio y rastreo de conocimientos. La Figura 22 muestra una réplica 1502. La réplica 1502 incluye una colección de cambios 1506 y conocimientos 1508. La colección de cambios 1506 incluye varios cambios 1510 individuales en este ejemplo ilustrados como X, Y y Z. En el ejemplo mostrado en la Figura 22, el estado actual de los conocimientos de la réplica se indica mediante un vector 1512 de conocimientos que en este caso es A4. El vector 1512 de conocimientos

15 representa todos los conocimientos de la réplica A 1508.

También representado en la Figura 22 se encuentra un número de los ID de cambio 1514. En el ejemplo de la Figura 22, la réplica A 1502 incluye tres elementos cambiados 1516, I_X , I_Y e I_Z , que corresponden a los cambios 1510. Usando el ID de cambios, se puede discernir que el elemento I_X , con el ID de cambio A1, se realizó en la réplica A 1502 en un primer tiempo. El cambio I_Y , con el ID de cambio A2, se realizó en la réplica A 1502 en un tiempo posterior al elemento I_X . Y el elemento I_Z , con el ID de cambio A4, se realizó en la réplica A 1502 en un tiempo posterior cuando se realizó el elemento I_Y . A3, aunque no se ilustra directamente en la Figura 22, puede corresponder a un cambio anterior tal como en un ejemplo, un cambio que se sustituye por el cambio al elemento I_Z etiquetado A4.

20

Existe una diferencia entre el ID de cambio A4 y el vector de conocimientos de la réplica A 1512 que también se etiqueta A4. En este ejemplo, el vector de conocimientos A4 significa que los conocimientos de la réplica A 1508 incluyen los cambios que corresponden a los ID de cambios etiquetados A4, A3, A2 y A1. Dicho de manera diferente, un vector de conocimientos incluye el cambio representado por el ID de cambio 1518 que es igual al vector de conocimientos así como todos los cambios con el mismo ID de réplica que se realizaron antes del ID de cambio 1518 representado en el vector de conocimientos. Por otra parte, en el presente ejemplo el ID de cambio 1518 etiquetado A4 únicamente representa el cambio Z realizado al elemento I_Z .

25

Haciendo referencia ahora a la Figura 23, se muestra un ejemplo de dos réplicas que replican en una topología que contiene un número de réplicas. La réplica A 1602 contiene un conjunto de cambios 1604, los conocimientos 1606 y un vector 1608 de conocimientos que es una representación corta de los conocimientos 1606. De manera ilustrativa, el vector 1608 de conocimientos de la réplica A 1602, A5B3C1D10, muestra que los conocimientos 1606 de la réplica A incluyen cambios realizados hasta un quinto cambio en la réplica A 1602, los conocimientos hasta un tercer cambio en una réplica B 1610, los conocimientos hasta un primer cambio en una réplica C y los conocimientos hasta un décimo cambio en una réplica D. La réplica B 1610, en el ejemplo de la Figura 23, incluye un conjunto de cambios 1612, los conocimientos 1614 y un vector 1616 de conocimientos que es una representación corta de los conocimientos de la réplica B 1614. El vector 1616 de conocimientos de la réplica B, A3B3C5D8, ilustra que la réplica B tiene conocimientos que incluyen los conocimientos hasta un tercer cambio realizado por la réplica A 1602, los conocimientos hasta un tercer cambio realizado por la réplica B 1610, los conocimientos hasta un quinto cambio realizado por la réplica C y los conocimientos hasta un octavo cambio realizado por la réplica D. Los vectores de conocimientos anteriormente expuestos incluye una representación continua de enumeraciones de cambio realizadas por una réplica desde un primer cambio a algún cambio posterior. Como se explicará en más detalle más adelante en el presente documento, un vector de conocimientos puede incluir también un punto de comienzo que es alguna otra enumeración de cambio distinta a la primera enumeración de cambio realizada por una réplica.

30

Una ilustración temporal de la replicación de la réplica A 1602 con la réplica B 1610 se ilustra en la Figura 23. En el tiempo (1), la réplica A 1602 envía una solicitud 1618 de sincronización junto con los conocimientos 1606 de la réplica A, que puede representarse mediante el vector 1608 de conocimientos de la réplica A, a la réplica B 1610. La réplica B 1610 en el tiempo (2) examina los conocimientos 1606 de la réplica A comparándolos para cambiar los ID asociados con los cambios en la réplica B. La réplica B 1610 descubre que la réplica A no tiene conocimiento de los cambios realizados por la réplica C que se etiquetan con los ID de cambio C2, C3, C4 y C5. Por lo tanto, la réplica B envía los cambios 1612 de la réplica B que corresponden a estos ID de cambio siempre que los cambios etiquetados con estos ID de cambio sean los cambios actuales aplicables a los elementos en la réplica B 1610. Si un ID de cambio corresponde a un cambio desactualizado anterior, no se envía cambio que corresponda a ese ID. Por ejemplo, si se actualizó un elemento que tenía una versión C3 y se asignó una nueva versión, el cambio asociado con C3 ya no existe en la réplica B 1610 y no se envía a la réplica A. Posteriormente o de manera simultánea como se ilustra en el tiempo (3) la réplica B 1610 envía a la réplica A 1602 los conocimientos de la réplica B 1614 que pueden representarse como un vector 1616 de conocimientos.

35

En el tiempo (4) la réplica A 1602 examina los conocimientos 1614 enviados por la réplica B comparándolos a los ID de cambio que corresponden a cambios en la réplica A 1602. La réplica A 1602 descubre que la réplica B no tiene

40

ninguno de los cambios representados por los ID de cambio A4, A5, D9 y D10, o conocimiento acerca de estos cambios. Por lo tanto, la réplica A 1602 envía cualquier cambio actual existente en los cambios 1604 de la réplica A que corresponden a estos ID de cambio (excepto cuando el ID de cambio representa un cambio desactualizado de manera que no se envía el cambio). La réplica A 1602 puede enviar posteriormente un mensaje a la réplica B 1610 que indica que todos los cambios se han enviado de manera que la réplica A 1602 y la réplica B 1610 pueden ahora actualizar sus vectores 1608 y 1616 de conocimientos respectivamente para incluir los cambios replicados recientemente. Como se muestra en la figura 23 en el tiempo (5), el vector de conocimientos de la réplica A, A5B3C5D10, es igual al vector de conocimientos de la réplica B que incluye todos los cambios realizados por la réplica A hasta una quinta enumeración de cambio, todos los cambios realizados por la réplica B hasta una tercera enumeración de cambio, todos los cambios realizados por la réplica C hasta una quinta enumeración de cambio y todos los cambios realizados por la réplica D hasta una décima enumeración de cambio.

Haciendo referencia ahora a las Figuras 24A y 24B, se muestran dos procedimientos de actualización de los vectores de conocimientos siguiendo una replicación completa tal como la representada en la Figura 23. Específicamente, la Figura 24A ilustra un procedimiento para actualizar los vectores de conocimientos usando una lista 1702 de excepciones almacenada en una réplica. Para crear una lista 1702 de excepciones, como los cambios se envían entre réplicas, los cambios se envían con un ID de cambio asociado con el cambio. Cuando el cambio se añade a una réplica, el ID de cambio se añade como una excepción a una lista 1702 de excepciones. Examinando ahora los conocimientos para la réplica A en la Figura 24A; los conocimientos incluyen un vector 1608 de conocimientos y una lista 1702 de excepciones que incluye las excepciones C2, C3, C4 y C5. Una examinación de la lista 1702 de excepciones en conjunto con el vector 1608 de conocimientos revela que incluir los ID de cambio de la lista 1702 de excepciones, los conocimientos de la réplica A incluyen todos los cambios hasta un quinto cambio realizado por la réplica C. Por lo tanto, las excepciones pueden eliminarse de los conocimientos de la réplica A 1602 y el vector de conocimientos actualizarse para incluir un elemento C5 como se muestra en el vector 1704 de conocimientos actualizado. Puede realizarse un análisis similar en los conocimientos 1614 de la réplica B 1610. El vector 1616 de conocimientos original combinado con las excepciones A4, A5, D9 y D10 en la lista 1703 de excepciones permite que el vector 1616 de conocimientos se actualice hasta un vector 1706 de conocimientos actualizado.

En particular, si únicamente se realizó una replicación parcial, tal como por ejemplo si los cambios que corresponden a los ID de cambio A4 y D9 no se enviaron en una replicación tal como la representada mediante la Figura 23, entonces los conocimientos 1614 de la réplica B 1610 necesitaría mantener las excepciones A5 y D10 hasta una replicación posterior con otra réplica que transfiera los cambios representados por los ID de cambio A4 y D9 a la réplica B 1610.

La Figura 24B ilustra otro procedimiento de actualización de los vectores 1608 y 1616 de conocimientos para reflejar la replicación mostrada en la Figura 23. En este ejemplo, los vectores de conocimientos se actualizan usando un máximo a nivel de elementos para cada uno de los elementos en los vectores 1608 y 1616 de conocimientos originales para formar un vector 1708 de conocimientos actualizado. El primer elemento de cada uno de los vectores 1608 y 1616 de conocimientos corresponde a un conjunto de los ID de cambio que etiqueta cambios realizados en la réplica A. Puesto que A5 es el elemento máximo a nivel de elementos de los dos vectores 1608 y 1616 de conocimientos, el vector 1708 de conocimientos actualizado incluye un elemento A5. Análogamente, los elementos del vector B3, C5 y D10 cada uno representan un elemento máximo a nivel de elementos que corresponde a los cambios en las réplicas particulares a las que corresponden cada uno de los elementos. La examinación de cada uno de los vectores 1704, 1706 y 1708 de conocimientos actualizados revela que mediante cualquier procedimiento, se obtiene el mismo vector de conocimientos actualizado. El procedimiento de máximo a nivel de elementos de actualización de vector de conocimientos se usa típicamente cuando se ha realizado una replicación completa mientras que un procedimiento de lista de excepciones de actualización del vector de conocimientos puede ser útil cuando no es seguro que haya tenido lugar una replicación completa (un usuario puede cancelar la replicación, un dispositivo puede bloquearse, etc.). En concreto, el procedimiento de lista de excepciones puede ser necesario que se use de manera que las excepciones puedan continuar para comprender una porción de los conocimientos de una réplica particular cuando los conocimientos completos de la réplica no pueden representarse en forma vectorial sencilla.

Haciendo referencia ahora a la Figura 24C, se muestra un ejemplo de actualización de conocimientos para una réplica que tiene información desde una replicación incompleta. La Figura 24C incluye un vector 1710 de conocimientos original, una lista 1712 de excepciones original, un vector 1714 de conocimientos actualizado y una lista 1716 de excepciones actualizada. Con respecto a la réplica mostrada, después de la replicación parcial, la réplica tiene todos los ID de cambio etiquetados A1 a A5, representados por el elemento de vector A5, y todos los ID de cambio etiquetados A7 a A10 representados por la lista de excepciones que incluye A7, A8, A9 y A10. Como se muestra en la Figura 24C, en una versión actualizada de los conocimientos, la lista 1716 de excepciones actualizada puede acortarse para indicar la inclusión de todos los elementos de A7 a A10 tal como mediante la expresión (A7:A10) mostrada en la Figura 24C. Esta expresión es simplemente un vector tal como aquellos que se han analizado previamente en el presente documento excepto que el punto de inicio del vector es algún punto distinto a la primera enumeración de cambio para la réplica A. Por lo tanto, la representación de los conocimientos de la réplica en lo que se refiere A se representa mediante el elemento de vector A5 y el vector de excepción (A7:A10).

En el caso de los conocimientos de la réplica con respecto a la réplica B, el vector 1710 de conocimientos puede actualizarse para incluir los ID de cambio continuo posteriores a los ID de cambio incluidos en el elemento de vector para la réplica B. El elemento de vector B1 incluye únicamente el ID de cambio B1. Puesto que los ID de cambio B2, B3 y B4 existen en la lista 1712 de excepciones, y son continuos con el ID de cambio B1 incluido en el vector 1710 de conocimientos, el elemento de vector para la réplica B puede actualizarse a B4 en el vector 1714 de conocimientos actualizado que representa la inclusión de los elementos B1 a B4. Puesto que el ID de cambio B5 está ausente de la lista de excepciones, la excepción B6 debe permanecer en la lista 1716 de excepciones en los conocimientos actualizados.

Puede realizarse un análisis similar con respecto a los conocimientos de la réplica de la Figura 24C con respecto a cambios realizados por la réplica C. El vector 1710 de conocimientos original incluye C5. La lista de excepciones original incluye C6, C7 y C8. Puesto que el elemento de vector de conocimientos original C5 incluye los ID de cambio C1 a C5, y C5 es continuo con los ID de cambio en la lista 1712 de excepciones original, el elemento de vector de conocimientos actualizado para la réplica C puede actualizarse a C8.

Un desafío que puede surgir con respecto al tamaño de vectores de conocimientos es especialmente frecuente cuando el número de réplicas en una comunidad de sincronización es enorme. En una topología donde el vector de conocimientos incluye un ID de cambio u otro elemento de vector para todas y cada una de las réplicas en la comunidad de sincronización, el vector de conocimientos aumenta con cada réplica que se añade a la comunidad de sincronización. Una optimización es reconocer que en algunas comunidades de sincronización cada réplica no necesita representarse en el vector de conocimientos. Una ilustración de un caso de este tipo es la comunidad de sincronización mostrada en la Figura 25 que representa una topología de servidor en estrella. La Figura 25 muestra un servidor 1802 conectado a un número de clientes que incluyen la réplica A 1804, la réplica B 1806, la réplica C 1808 y la réplica D 1810. En este ejemplo, todas las trayectorias 1812-1818 de replicación entre los clientes son a través del servidor 1802 y por lo tanto el servidor 1802 puede asignar un ID de cambio que incluye el servidor 1802 como el ID de réplica. Todos los cambios realizados en los clientes 1804 a 1810 individuales permanecen en el respectivo cliente en el que se realizó el cambio sin la asignación de un ID de cambio hasta que se realice una replicación. Por lo tanto, en este ejemplo, el vector de conocimientos incluye un único elemento que comprende el ID de réplica y el ID de cambio del servidor 1802. De manera ilustrativa, si se realiza un cambio en la réplica A 1804 y se replica con el servidor 1802 en un primer tiempo, el servidor 1802 asigna una enumeración de cambio de S1 al cambio. En un tiempo posterior, un cambio realizado en la réplica B 1806 se replica con el servidor 1802. Este cambio se asigna a una enumeración de cambio por el servidor de S2. En particular, aunque en este ejemplo, el servidor 1802 asigna todas las enumeraciones de cambio, pueden existir otras implementaciones donde el servidor 1802 asigna algunas enumeraciones de cambio y otras réplicas asignan otras enumeraciones de cambio.

Las implementaciones de la invención son adaptables para optimizar el vector de conocimientos en otras topologías también. Por ejemplo, en la Figura 18, la réplica D 1114 únicamente replica con la réplica C 1110. Por lo tanto, los cambios realizados por C y D pueden enumerarse usando enumeraciones de cambio que tienen un único ID de réplica. En un ejemplo, si el ID de réplica de la réplica C se elige para que sea parte de la enumeración de cambio para todos los cambios por cualquiera de la réplica C 1110 o la réplica D 1114, un primer cambio en la réplica C se etiquetaría con la enumeración de cambio C1. Un cambio posterior en la réplica D 1114 se etiqueta C2, y así sucesivamente. Cuando una réplica crea un ID de cambio para cambios realizados en una réplica diferente, la réplica que crea el ID de cambio puede denominarse como un autor sustituto.

Optimizando el vector de conocimientos para la topología particular o comunidad de sincronización, los recursos usados para almacenar el vector de conocimientos pueden conservarse en topologías que se aproximan a las topologías de servidor-cliente en estrella tal como las mostradas en la Figura 25. En topologías más similares a redes entre pares, se requiere un vector de conocimientos mayor, pero las réplicas individuales pueden replicar de manera eficaz e independiente con un mayor número de otras réplicas mientras se evitan problemas tales como bucles de sincronización, conflictos falsos y similares.

Cuando se permite que diferentes réplicas hagan cambios a elementos independientemente unas de las otras, pueden resultar conflictos entre los cambios realizados de manera independiente que deberían resolverse. La resolución de conflictos típicamente requiere que haya ciertas reglas para determinar qué versión de elemento debería elegirse como el elemento válido. Ejemplos de algunas de estas reglas incluyen seleccionar el cambio de elemento que se realizó en último lugar o seleccionar cambios de elemento que se realizaron por tipos particulares de réplicas tal como prefiriendo cambios realizados por servidores sobre cambios realizados por otros tipos de réplicas. Como alternativa, todos los conflictos podrían registrarse para resolución manual. La resolución manual se consigue proporcionando a un usuario un nuevo valor para el elemento en conflicto que sustituirá los cambios que entran en conflicto.

Si todas las réplicas en una comunidad de sincronización o topología resuelven conflictos de la misma manera, típicamente no se requieren otras reglas de resolución o sistemas de resolución ya que todas las réplicas en el sistema migrarán a una resolución replicada de cualquier conflicto. Aunque las réplicas en la comunidad de sincronización pueden no estar específicamente diseñadas para resolver conflictos exactamente de la misma manera, las réplicas en una comunidad de sincronización pueden, sin embargo, resolver conflictos exactamente de la misma manera. Un ejemplo de este tipo se muestra en la Figura 26A. La Figura 26A muestra una réplica D 1902.

La réplica D 1902 recibe un ID de cambio que corresponde a un cambio en un elemento I_x en el que el ID de cambio es A4. Posteriormente la réplica D 1902 recibe un ID de cambio para el mismo elemento I_x en el que el ID de cambio es B5. La réplica D 1902 tiene reglas de resolución de conflictos para elegir cuál de los cambios al elemento I_x es el cambio preferido. En este caso replica D elige el cambio al elemento I_x etiquetado por el ID de cambio A4. Para

5 indicar que un conflicto se resolvió mediante la réplica D 1902 y cómo se resolvió el conflicto, se asigna un nuevo ID de cambio al elemento I_x que incluye tanto los resultados de la resolución de conflictos y un nuevo ID de cambio asignado por la réplica particular que realizó la resolución de conflictos. El nuevo ID de cambio incluye la siguiente enumeración de cambio secuencial para la réplica que realizó la resolución de conflictos. En este caso, el nuevo ID de cambio se etiqueta A4 (D7) para indicar que el cambio etiquetado A4 se eligió en la resolución de conflictos y que

10 el conflicto se resolvió por la réplica D 1902. Como se muestra en la figura 26A, tiene lugar un proceso similar cuando se detecta un conflicto en cambios por una réplica C 1904. La réplica C 1904 resuelve el conflicto de la misma manera que la réplica D 1902. Por lo tanto se asigna un nuevo ID de cambio etiquetado A4 (C3) al cambio del elemento I_x . En este caso, el conflicto entre los cambios al elemento I_x etiquetado con los ID de cambio A4 y B5 se resolverá eventualmente de la misma manera que en todas las réplicas dentro de la topología.

15 La Figura 26B ilustra un ejemplo donde se resuelven conflictos de manera diferente mediante diferentes réplicas en una topología. En la Figura 26B, en el tiempo (1) la réplica D 1902 resuelve el conflicto de una manera y asigna un nuevo ID de cambio a los elementos que ilustran la resolución del conflicto, B5, y la réplica que realizó el cambio, (D7). En el tiempo (2) la réplica C 1904 resuelve el mismo conflicto de una manera diferente mostrada mediante el nuevo ID de cambio asignado por la réplica C 1904, A4 (C3). En el tiempo (3), la réplica D 1902 recibe la resolución del conflicto de la réplica C. La réplica D 1902 en este punto reconoce que este conflicto particular se ha resuelto de

20 dos maneras diferentes. Algunas implementaciones de la presente invención por lo tanto especifican que puede realizarse una resolución determinística entre los cambios que entran en conflicto para el elemento I_x . La resolución determinística particular ilustrada mediante la Figura 26B provoca que se seleccione el cambio con el ID de réplica más bajo como el resultado determinístico. Por lo tanto, puesto que A es un ID de réplica de valor inferior que la réplica B, se selecciona la resolución determinística del conflicto para que sea el cambio etiquetado por el ID de cambio A4. La réplica D 1902 por lo tanto cambia el ID de cambio asociado con el cambio al elemento I para que sea A4 (D7). Obsérvese que para evitar bucles de replicación u otros problemas de conflicto a la enumeración de cambio (es decir D7) asociada con la réplica que realiza el cambio es la misma en el resultado 1906 determinístico que en la resolución original del conflicto 1908.

25 Aunque se ha descrito la materia objeto en lenguaje específico a características estructurales y/o actos metodológicos, se ha de entender que la materia objeto definida en las reivindicaciones adjuntas no está necesariamente limitada a las características específicas o actos anteriormente descritos. En su lugar, las características específicas y actos anteriormente descritos se desvelan como formas de ejemplo para implementar las reivindicaciones. Se desea que se protejan todos los equivalentes, cambios y modificaciones que entran dentro

30 del alcance de las reivindicaciones.

35 Por ejemplo, un experto en la materia en la técnica de software informático reconocerá que las disposiciones cliente y/o servidor, contenido de pantalla de interfaz de usuario y/o distribuciones de datos como se describen en los ejemplos analizados en el presente documento podrían organizarse de manera diferente en uno o más componentes para incluir opciones adicionales o menos que las representadas en los ejemplos.

40

REIVINDICACIONES

1. Un procedimiento de sincronización de datos que comprende las etapas de:

5 proporcionar un participante parcial, teniendo el participante parcial un almacén de datos y un almacén de conocimientos, siendo el almacén de datos operable para almacenar un conjunto de datos recuperados durante un proceso de sincronización con un primer participante completo, siendo operable el almacén de conocimientos para almacenar un conjunto de conocimientos acerca de los datos en el almacén de datos, en el que el conjunto de conocimientos representa cambios en los datos de los que tiene conocimiento (282) el primer participante completo, en el que el participante parcial no entiende el conjunto de conocimientos (284), y en el que el participante parcial es responsable de rastrear qué cambios realiza el participante parcial en el conjunto de datos en el almacén (290) de datos;
 10 recibir una solicitud de un usuario del participante parcial para cambiar un registro particular en el conjunto de datos en el almacén (286) de datos; y
 15 actualizar el registro particular en el almacén de datos después de recibir la solicitud del usuario (288), en el que la actualización incluye almacenar información que identifica un origen del cambio como el participante (290) parcial.

2. El procedimiento de la reivindicación 1, en el que el participante parcial es operable para sincronizar con un segundo participante (66) completo o en el que el participante parcial está sincronizado con el primer participante completo a través de un controlador en el primer participante (66) completo o
 20 en el que el primer participante completo es un ordenador (242) personal o en el que el primer participante completo es un dispositivo (242) móvil o en el que el participante parcial es un dispositivo (282) móvil o en el que el participante parcial es un servicio (282) web.

3. El procedimiento de la reivindicación 1, en el que la información que identifica el origen del cambio es una versión (304).

4. El procedimiento de la reivindicación 3, en el que una primera porción de la versión incluye un identificador que identifica de manera inequívoca el participante parcial y en el que una segunda porción de la versión incluye un número que indica una versión (304) de registro.

5. El procedimiento de la reivindicación 1, en el que la información que identifica el origen del cambio es un identificador único para el registro y un identificador de fecha y hora para indicar cuándo cambió (314) el registro.

6. El procedimiento de la reivindicación 1, que comprende adicionalmente:
 recibir un conjunto actualizado de conocimientos desde el primer participante completo después de que el primer participante completo actualice el conjunto de conocimientos después del proceso (296) de sincronización.

7. El procedimiento de la reivindicación 1, que comprende adicionalmente:
 35 proporcionar un participante sencillo, teniendo el participante sencillo un almacén de datos y no almacén de conocimientos, siendo el almacén de datos operable para almacenar un conjunto de datos proporcionado durante un proceso de sincronización con el participante completo, y en el que el participante sencillo no es responsable de rastrear qué cambios realiza el participante sencillo en el conjunto de datos en el almacén (424) de datos;
 40 recibir una solicitud de un usuario del participante sencillo para cambiar un registro particular en el conjunto de datos en el almacén (428) de datos; y
 actualizar el registro particular en el almacén de datos después de recibir la solicitud del usuario (430).

8. Un sistema que comprende un participante (20) parcial que contiene un almacén (22) de datos de conocimientos, un almacén (26) de datos de sincronización y un almacén (25) de datos de versión, y un participante (10) completo que tiene un almacén (12) de datos de conocimientos y un almacén (16) de datos de sincronización, estando adaptado el sistema para realizar las etapas indicadas en una de las reivindicaciones 1 a 6.

9. El sistema de la reivindicación 8, que comprende adicionalmente un participante (30) sencillo que tiene un almacén (36) de datos de sincronización, estando adaptado el sistema adicionalmente para realizar las etapas indicadas en la reivindicación 7 (200).

10. Un medio legible por ordenador que tiene instrucciones ejecutables por ordenador para provocar que un ordenador realice las etapas que comprenden:

proporcionar un participante parcial, teniendo el participante parcial un almacén de datos y un almacén de conocimientos, siendo el almacén de datos operable para almacenar un conjunto de datos recuperados durante un proceso de sincronización con un primer participante completo, siendo operable el almacén de conocimientos

- 5 para almacenar un conjunto de conocimientos acerca de los datos en el almacén de datos, en el que el conjunto de conocimientos representa cambios en los datos de los que tiene conocimiento (282) el primer participante completo, en el que el participante parcial no entiende el conjunto de conocimientos (284), y en el que el participante parcial es responsable de rastrear qué cambios realiza el participante parcial en el conjunto de datos en el almacén (290) de datos;
- 10 recibir una solicitud de un usuario del participante parcial para cambiar un registro particular en el conjunto de datos en el almacén (286) de datos; y
actualizar el registro particular en el almacén de datos después de recibir la solicitud del usuario (288), en el que la actualización incluye almacenar información que identifica un origen del cambio como el participante (290) parcial.
11. El medio legible por ordenador de la reivindicación 10, en el que la información que identifica el origen del cambio es una versión.
- 15 12. El medio legible por ordenador de la reivindicación 11, en el que una primera porción de la versión incluye un identificador que identifica de manera inequívoca el participante parcial y en el que una segunda porción de la versión incluye un número que indica una versión de registro.
13. El medio legible por ordenador de la reivindicación 10, en el que la información que identifica el origen del cambio es un identificador único para el registro y un identificador de fecha y hora para indicar cuándo cambió el registro.
- 20 14. El medio legible por ordenador de la reivindicación 10, que comprende adicionalmente recibir un conjunto actualizado de conocimientos desde el primer participante completo después de que el primer participante completo actualice el conjunto de conocimientos después del proceso de sincronización.

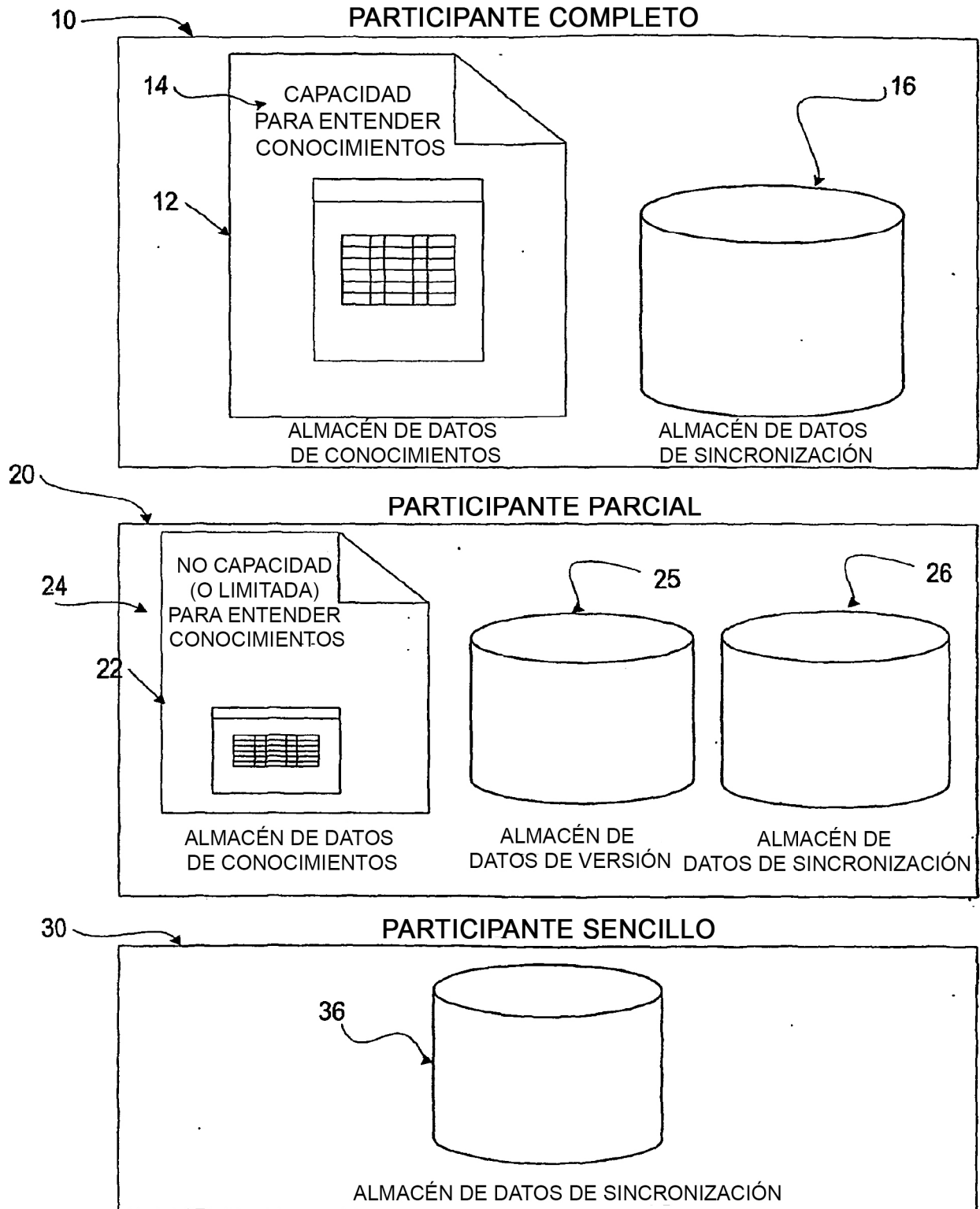


FIG. 1

50	52	54	56	68
	PARTICIPANTE SENCILLO	PARTICIPANTE PARCIAL	PARTICIPANTE COMPLETO	
ENTIENDE Y ALMACENA CONOCIMIENTOS			X	68
CONOCIMIENTOS SIMPLEMENTE ALMACENADOS; NO ENTENDIDOS		62 → X	NA	
NO PUEDE ALMACENAR CONOCIMIENTOS	58 → X	NA	NA	
OBJETIVO DE SINCRONIZACIÓN DE 2 SENTIDOS DE MÚLTIPLES MAESTROS		64 → X	X	70
SINCRONIZACIÓN	60 A TRAVÉS DE UN CONTROLADOR EN UN ÚNICO PARTICIPANTE SENCILLO	66 A TRAVÉS DE UN CONTROLADOR EN UN PARTICIPANTE COMPLETO	72 ALL	

FIG. 2

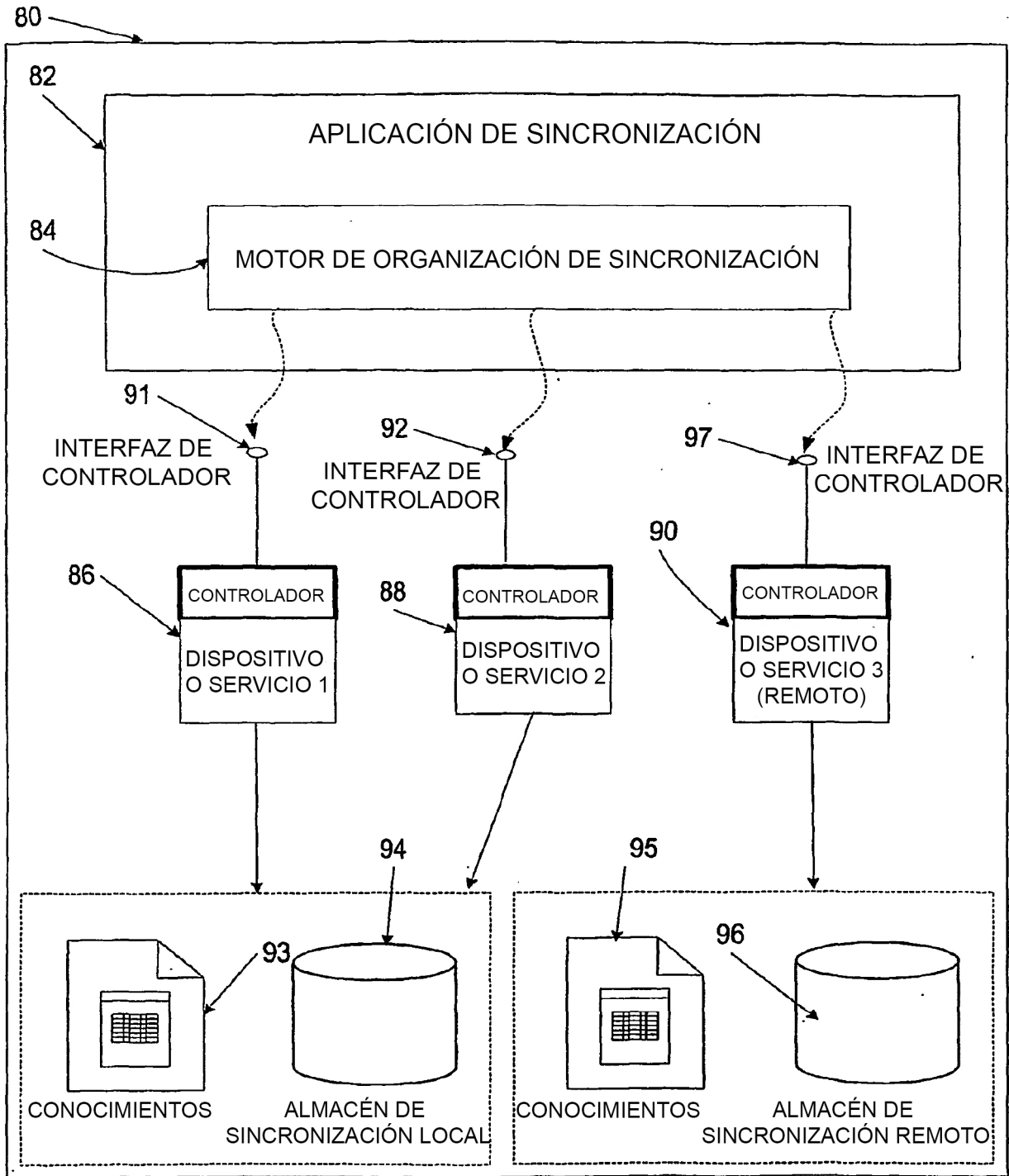


FIG. 3

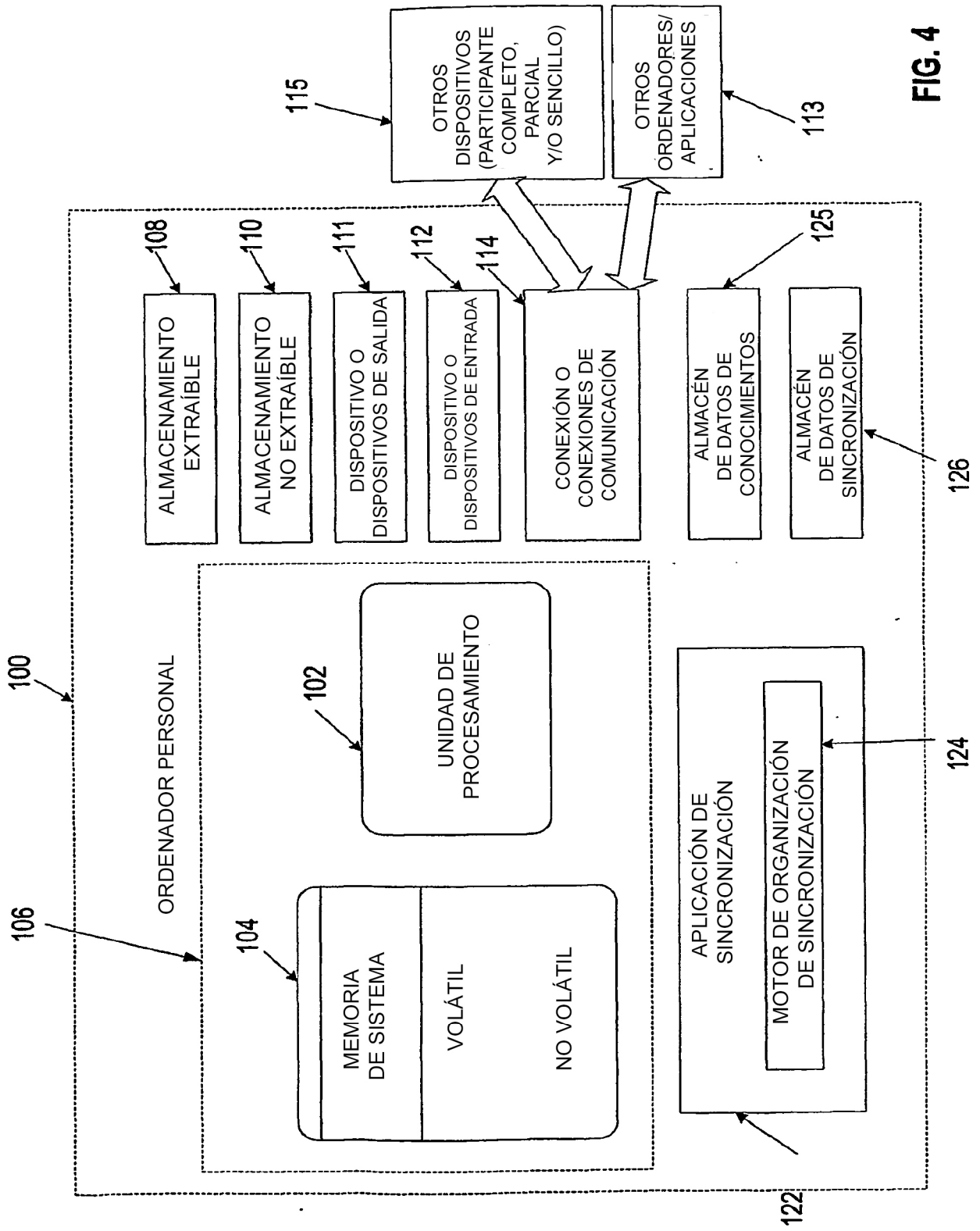


FIG. 4

APLICACIÓN DE SINCRONIZACIÓN	
<u>200</u>	
LÓGICA DE PROGRAMA	
<u>204</u>	
<u>206</u>	LÓGICA PARA REGISTRAR CONTROLADORES PARA CADA DISPOSITIVO/SERVICIO PARA PARTICIPAR EN PROCESO DE SINCRONIZACIÓN
<u>208</u>	LÓGICA PARA DETECTAR TIPO DE DISPOSITIVO/SERVICIO CONECTADO (PARTICIPANTE SENCILLO, PARCIAL O COMPLETO)
<u>210</u>	LÓGICA PARA RECIBIR CONOCIMIENTOS DESDE PARTICIPANTE PARCIAL, REALIZAR MODIFICACIONES EN CASO DE EXCEPCIONES, Y TRANSFERIR CONOCIMIENTOS DE VUELTA A PARTICIPANTE PARCIAL
<u>212</u>	LÓGICA PARA DETECTAR CAMBIOS EN PARTICIPANTES SENCILLOS Y ALMACENAR CONFLICTOS EN SU PROPIO ALMACÉN DE DATOS LOCAL
<u>214</u>	LÓGICA PARA FUNCIONAR CON OTRO PARTICIPANTE COMPLETO PARA SINCRONIZAR CONJUNTOS CAMBIADOS DE DATOS USANDO CONOCIMIENTOS
<u>216</u>	LÓGICA PARA REALIZAR ORGANIZACIÓN PARA COMPLETAR EL BUCLE DE SINCRONIZACIÓN Y EXTRAER CAMBIOS ACTUALIZADOS ENTRE OTROS DISPOSITIVOS/SERVICIOS CONECTADOS
<u>220</u>	OTRA LÓGICA PARA OPERAR LA APLICACIÓN

FIG. 5

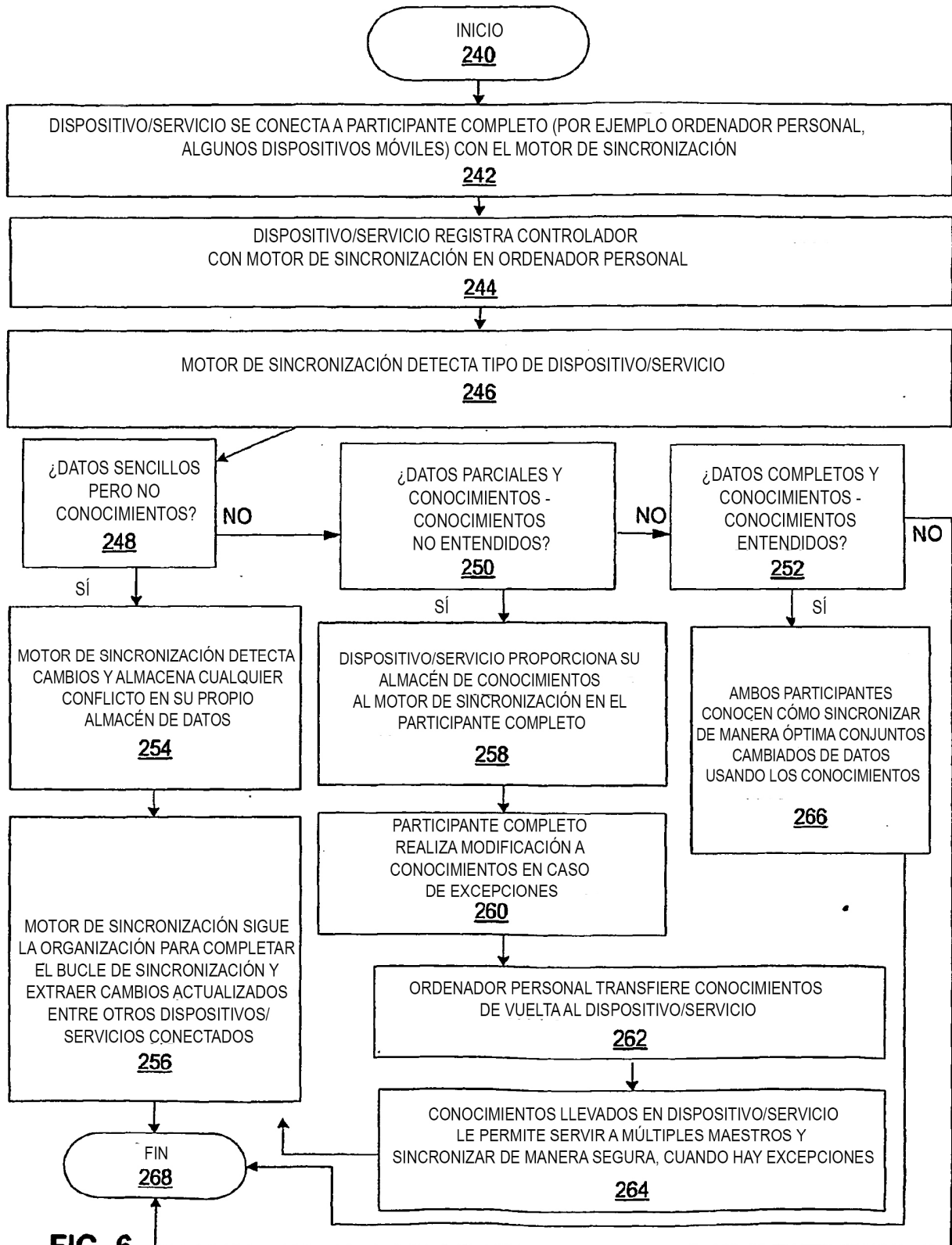


FIG. 6

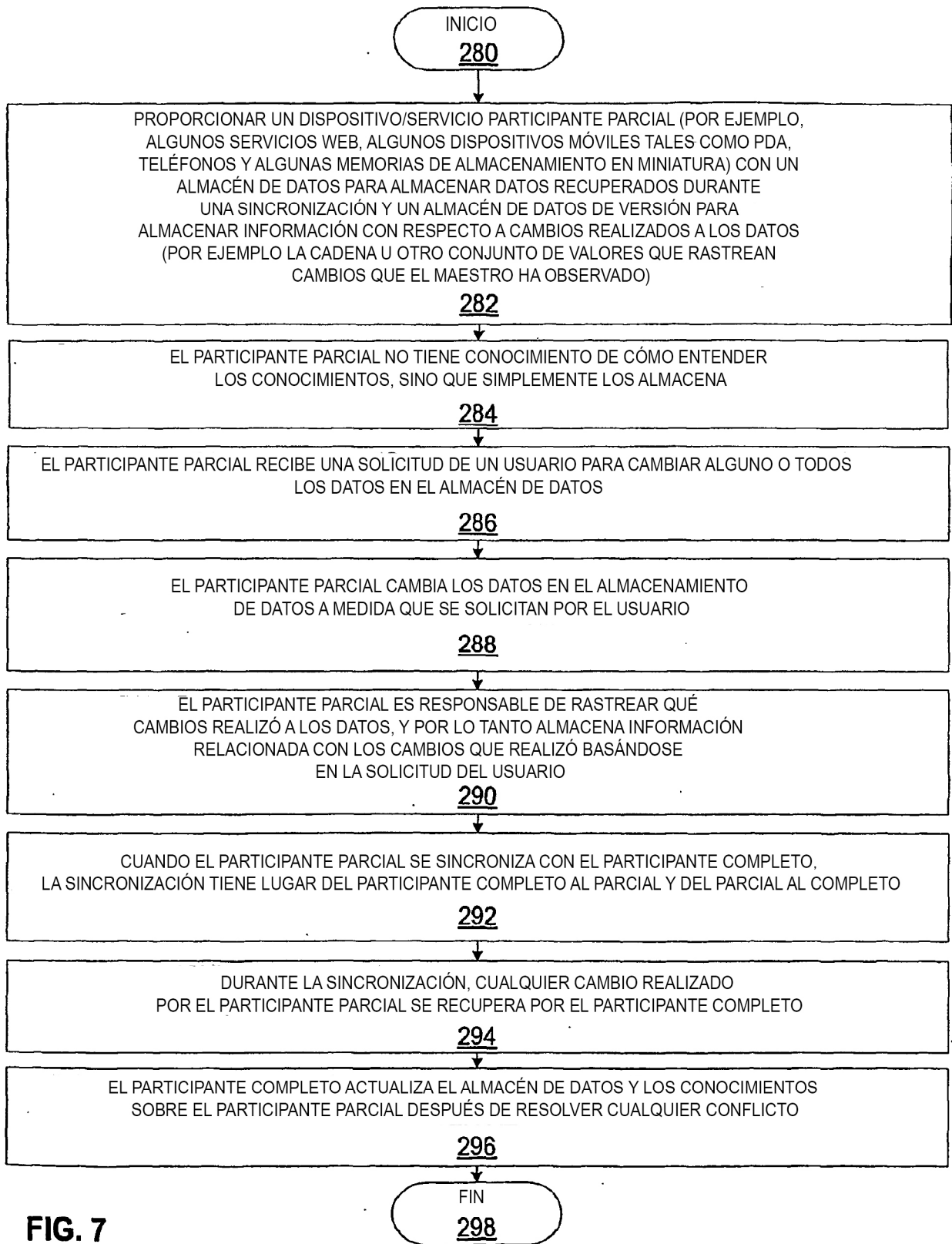


FIG. 7

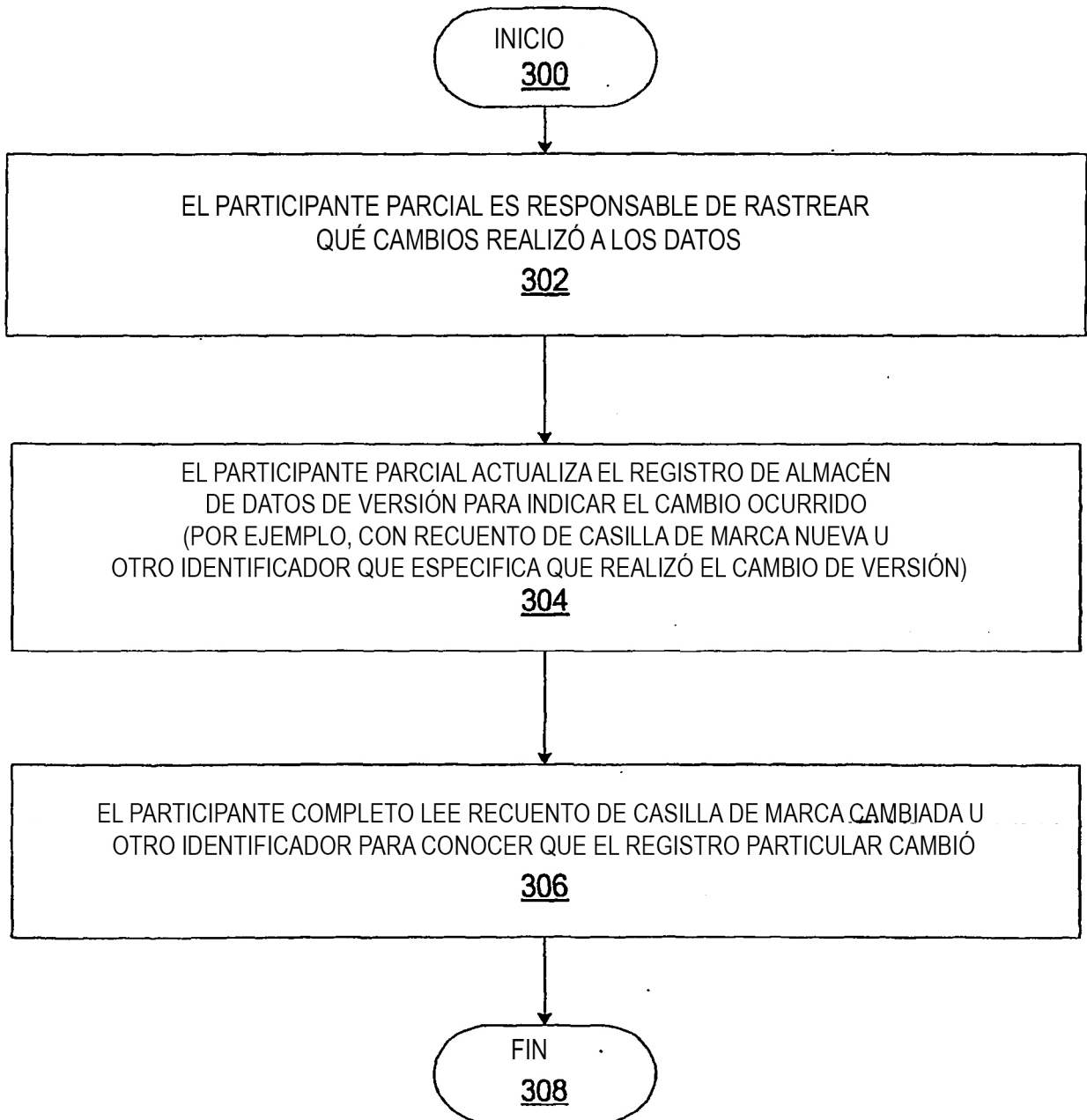


FIG. 8

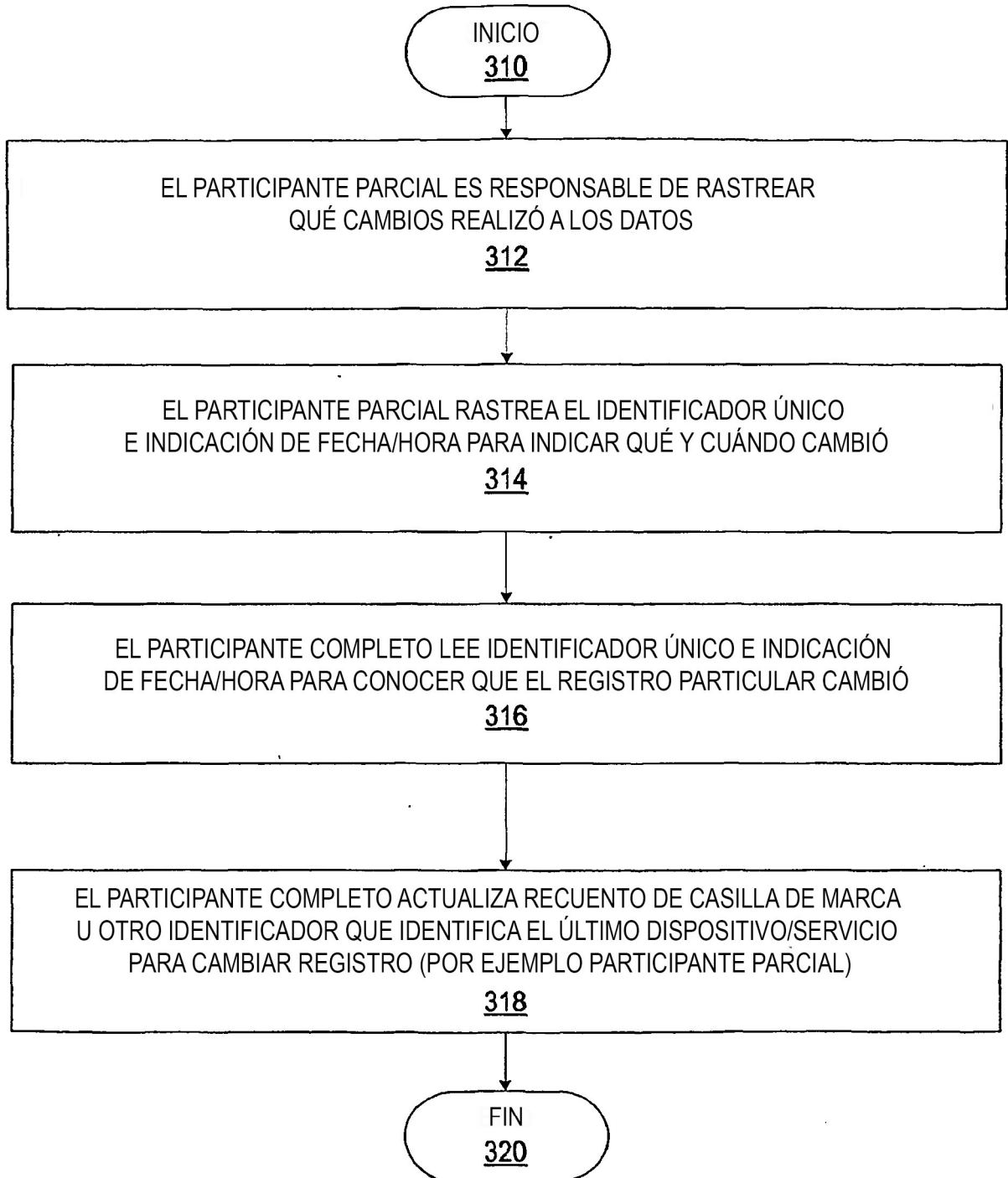


FIG. 9

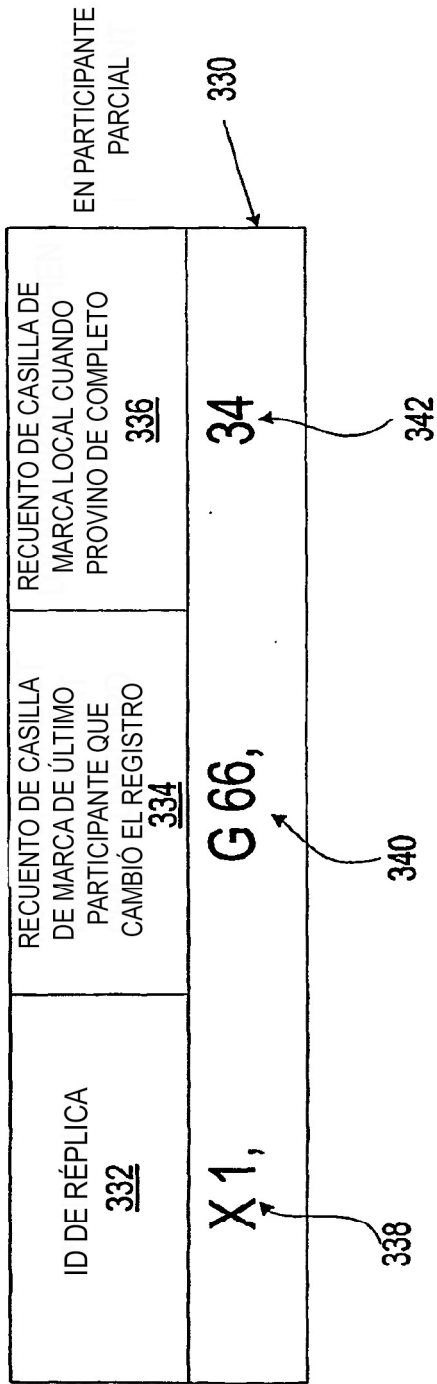


FIG. 10

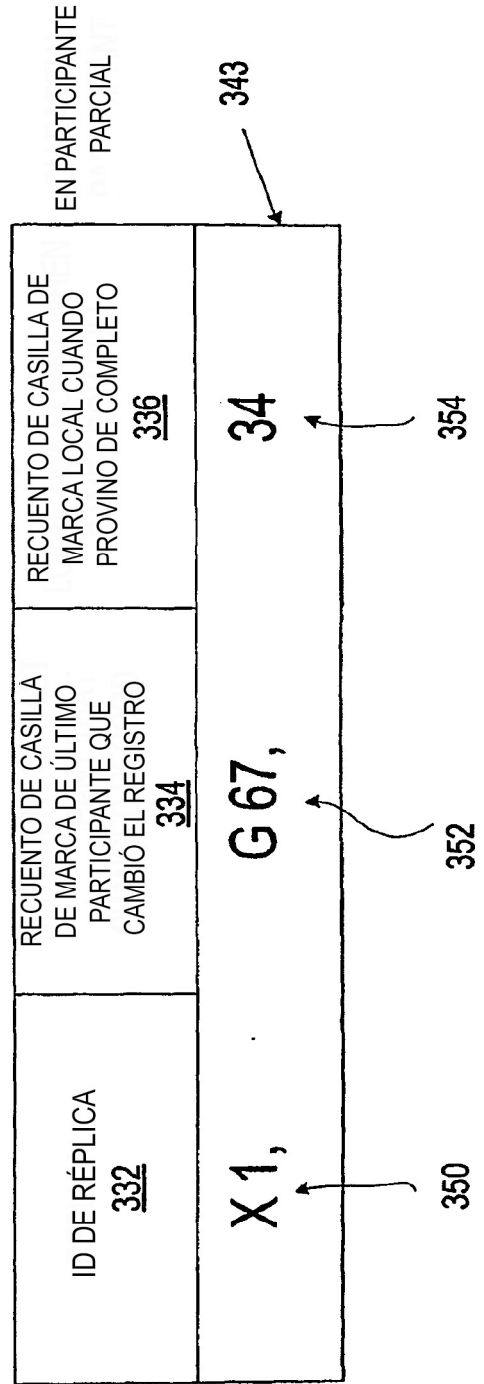


FIG. 11

ID DE RÉPLICA <u>362</u>	RECUENTO DE CASILLA DE MARCA DE ÚLTIMO PARTICIPANTE QUE CAMBIÓ EL REGISTRO <u>364</u>	RECUENTO DE CASILLA DE MARCA LOCAL CUANDO PROVINO DE COMPLETO <u>366</u>
X 1,	G 66,	34

EN PARTICIPANTE PARCIAL

360 →

370 →

372 →

368 →

FIG. 12

ID DE RÉPLICA <u>374</u>	CUANDO CAMBIÓ REGISTRO <u>376</u>
X 1,	26-01-06-12:32PM

EN PARTICIPANTE PARCIAL

375 →

380 →

378 →

FIG. 13

ID DE RÉPLICA <u>362</u>	RECUENTO DE CASILLA DE MARCA DE ÚLTIMO PARTICIPANTE QUE CAMBIÓ EL REGISTRO <u>364</u>	RECUENTO DE CASILLA DE MARCA LOCAL CUANDO PROVINO DE COMPLETO <u>366</u>
X 1,	G 67,	34

EN PARTICIPANTE COMPLETO

381 →

390 →

392 →

388 →

FIG. 14

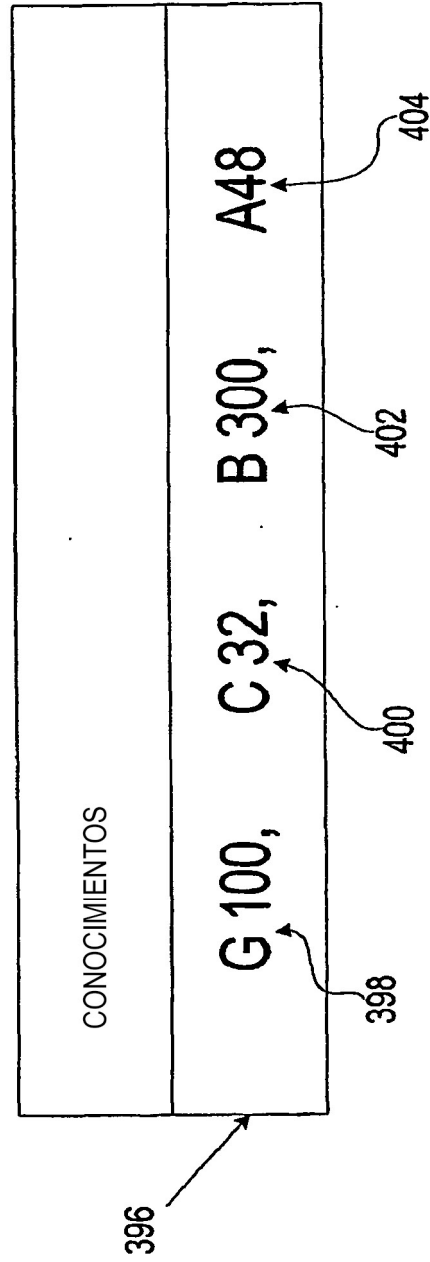
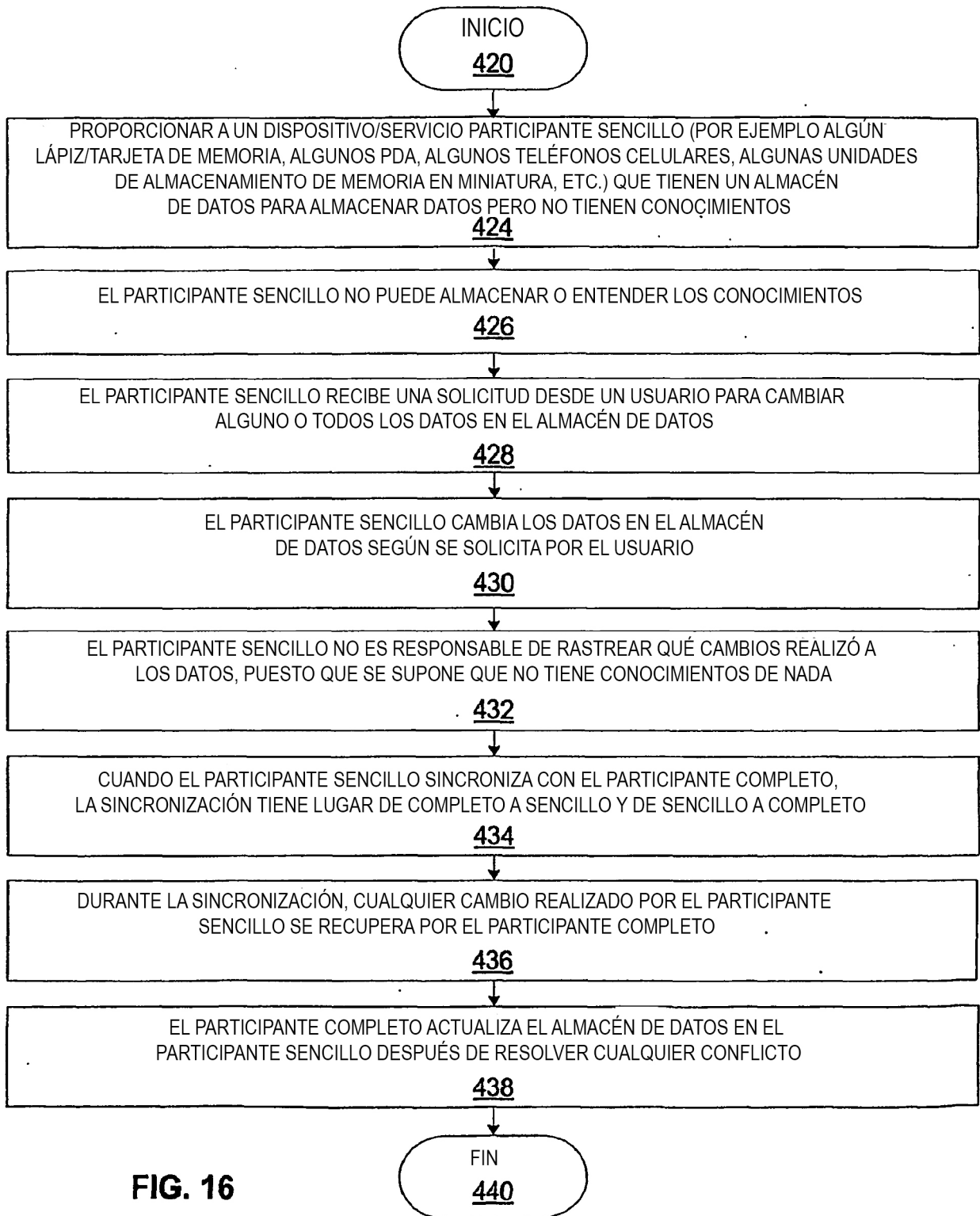
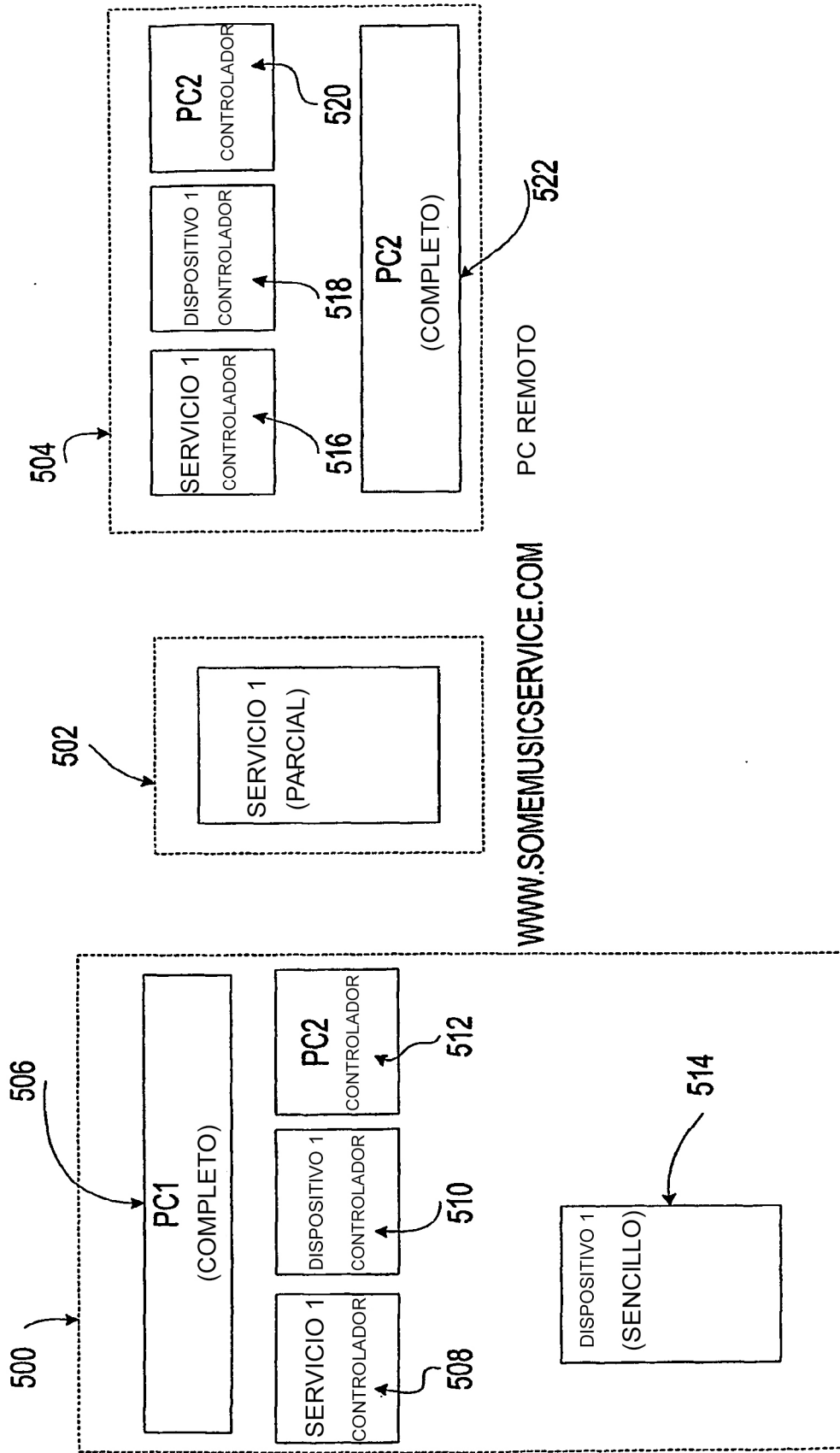


FIG. 15





PC LOCAL CON DISPOSITIVO 1 CONECTADO

FIG. 17

ES 2 635 719 T3

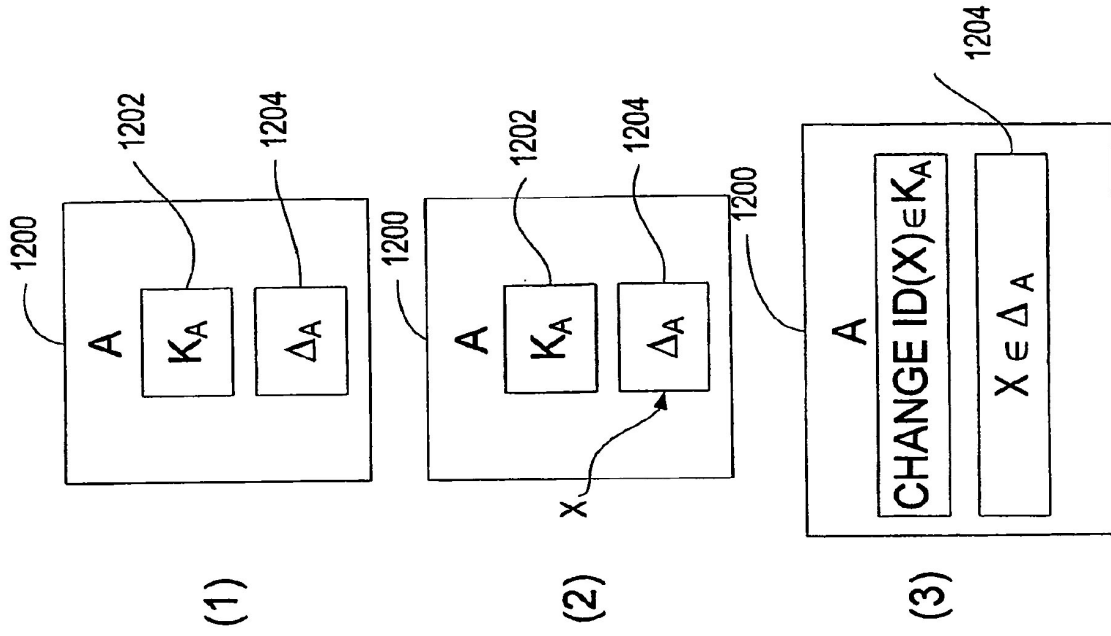


FIG. 19

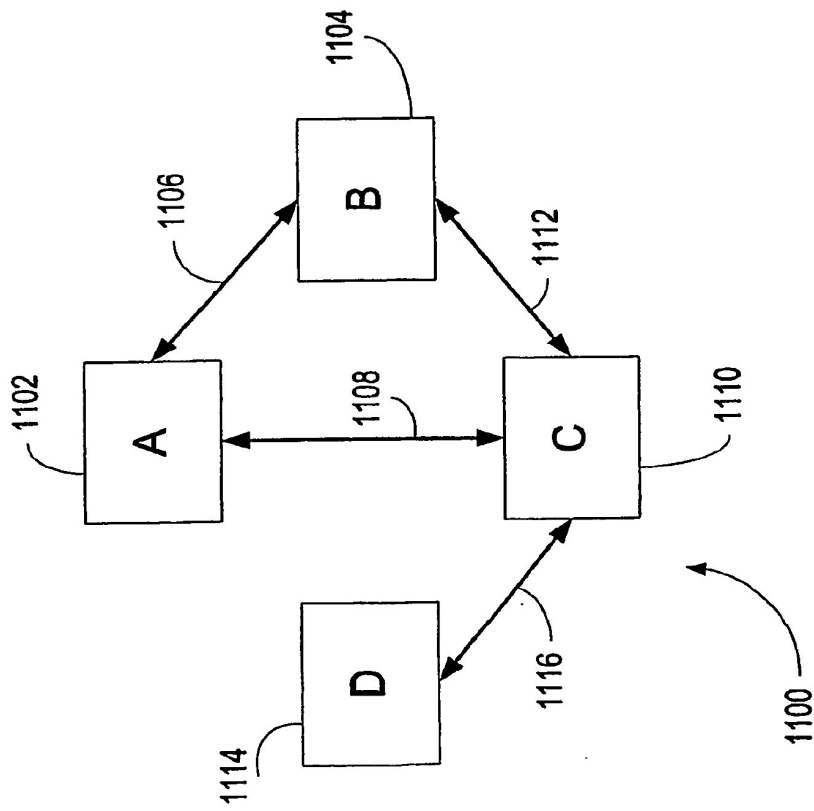


FIG. 18

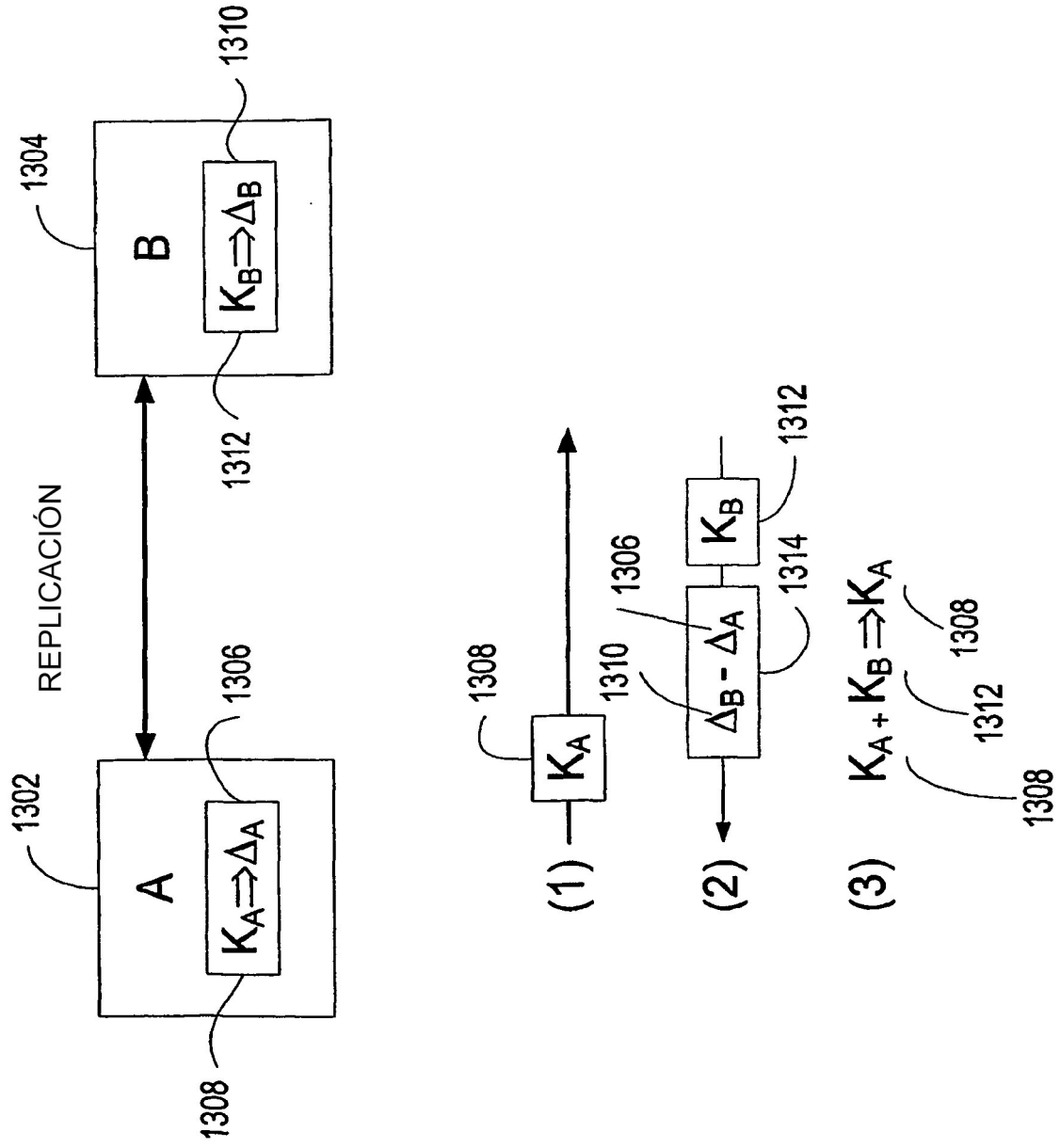


FIG. 20

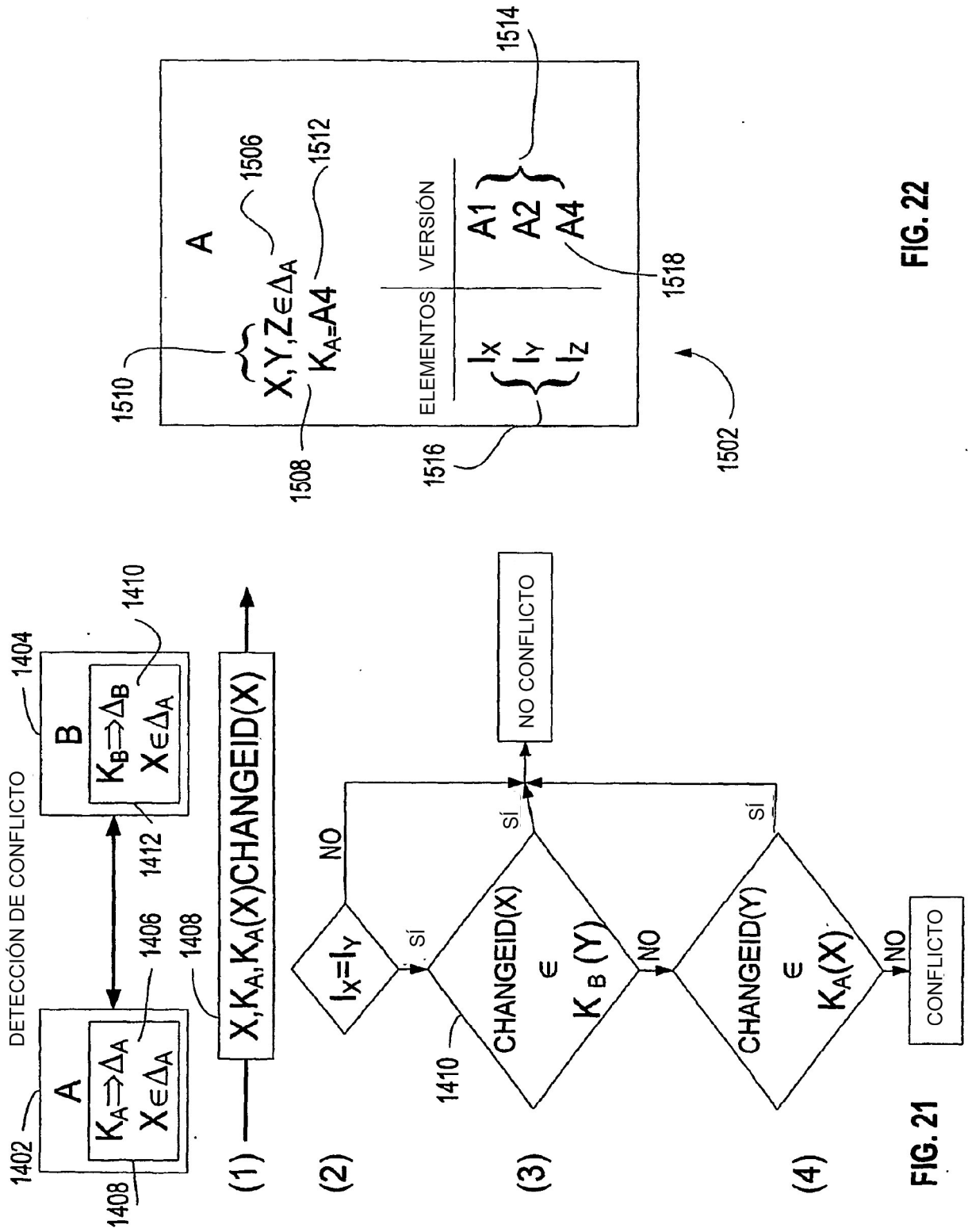


FIG. 22

FIG. 21

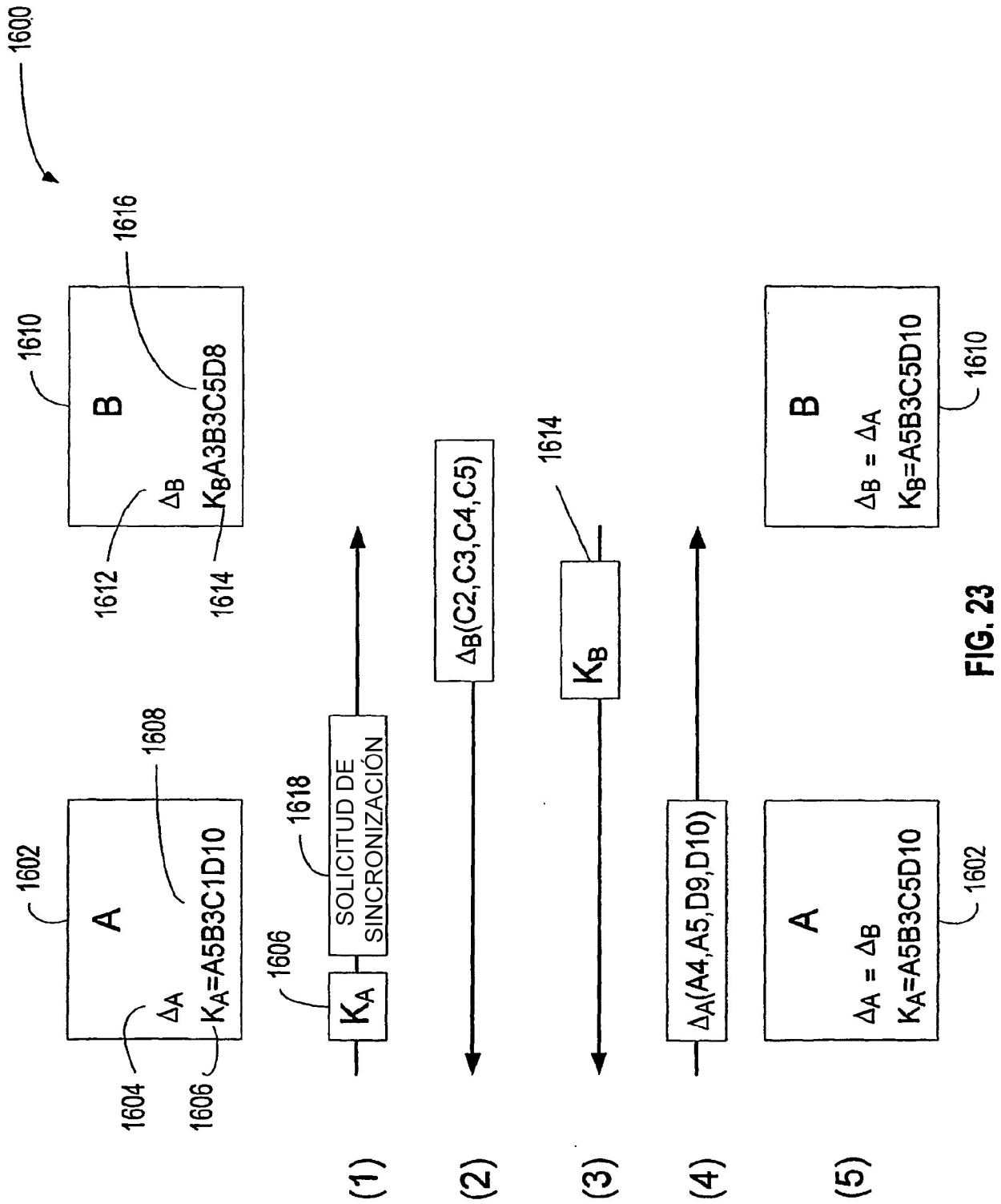


FIG. 23

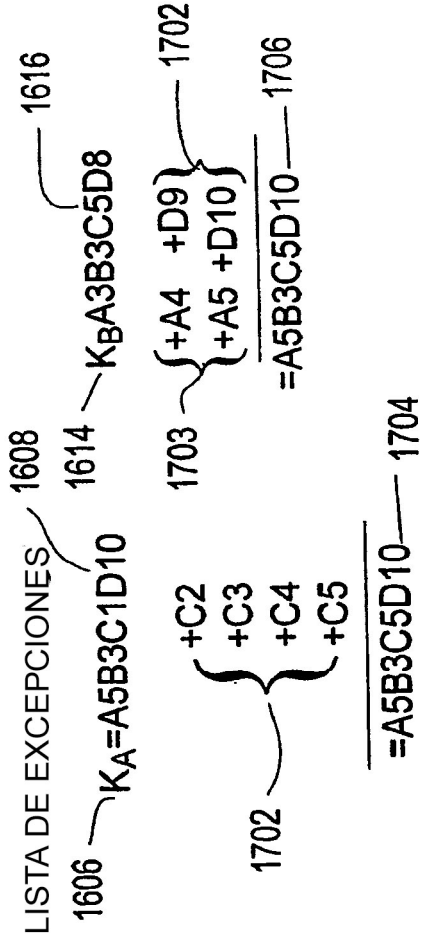


FIG. 24A

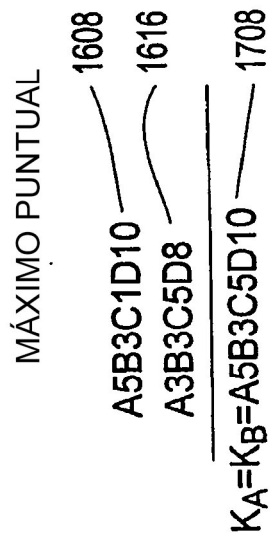


FIG. 24B

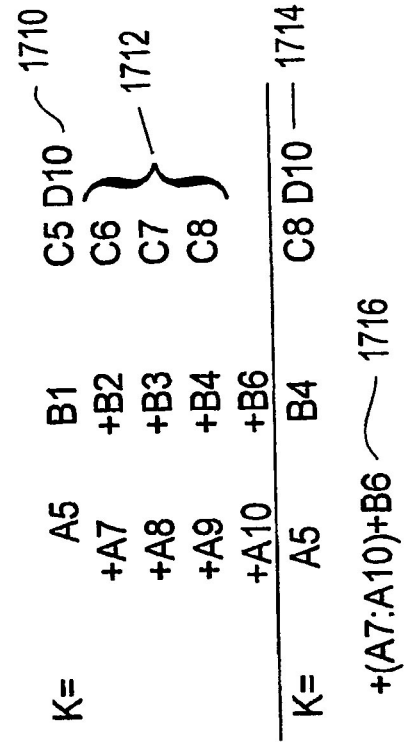


FIG. 24C

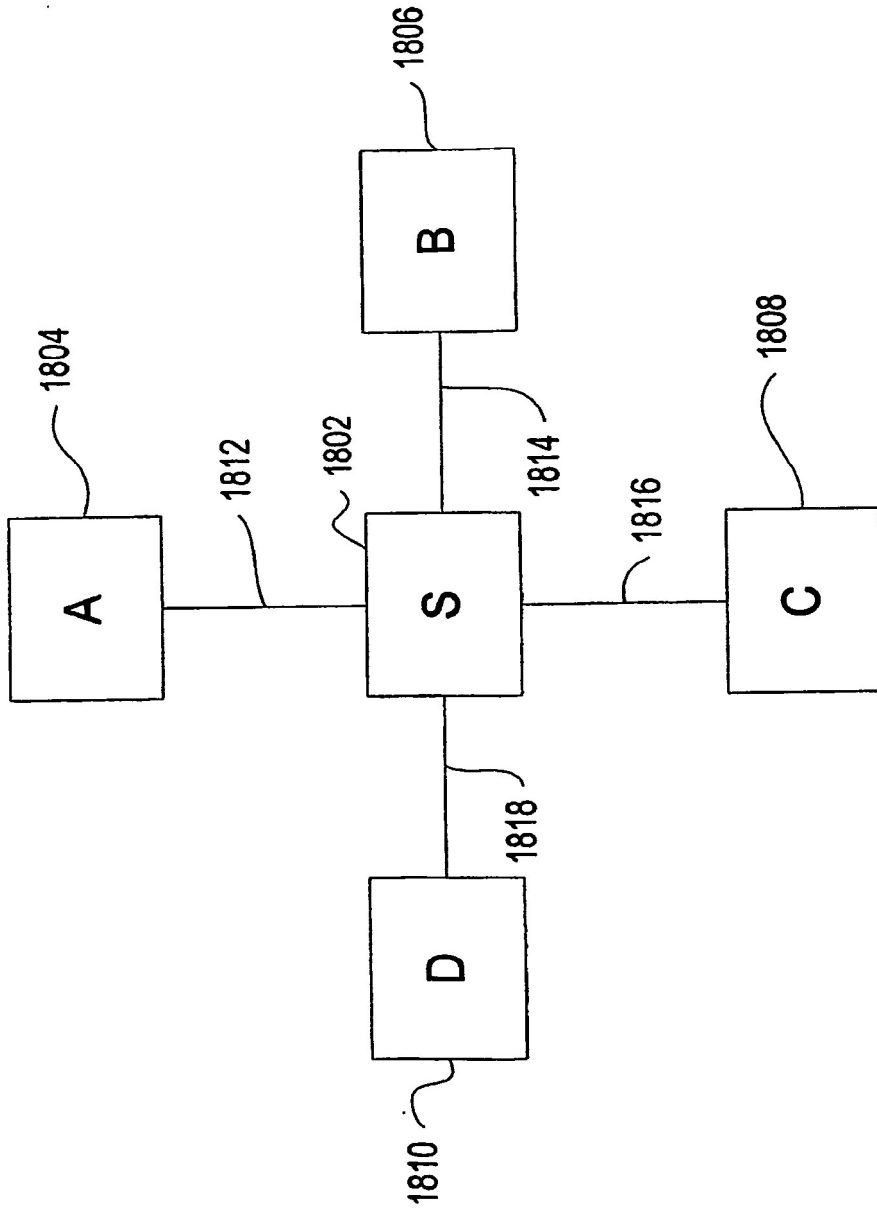


FIG. 25

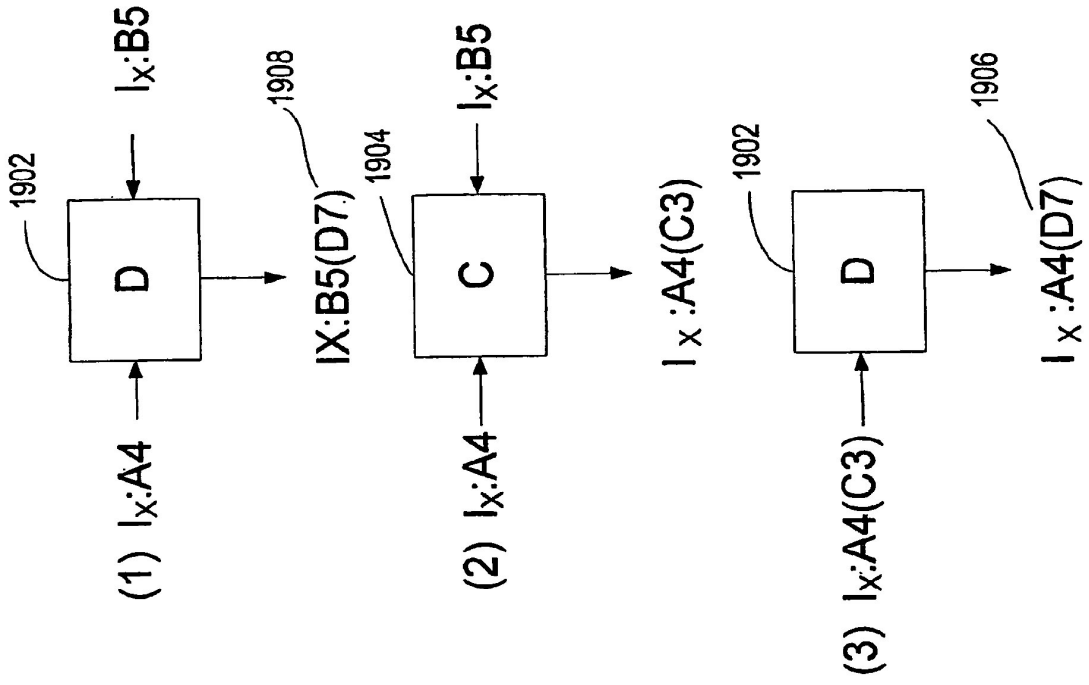


FIG. 26B

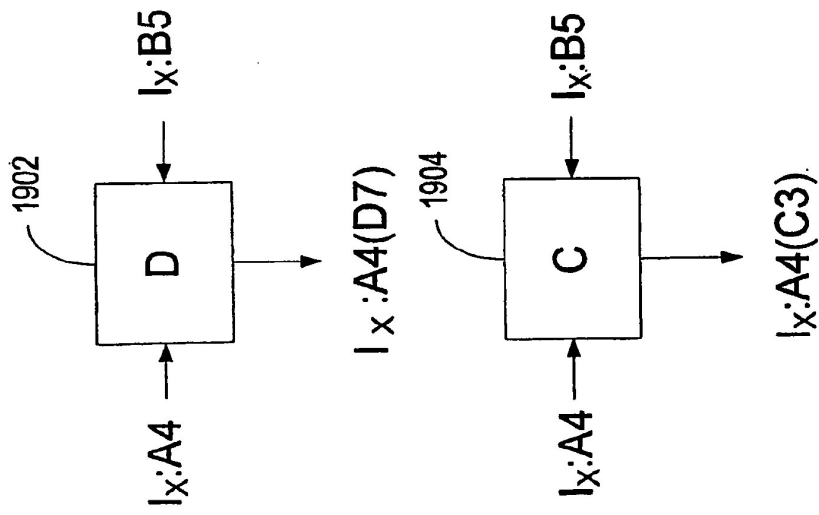


FIG. 26A