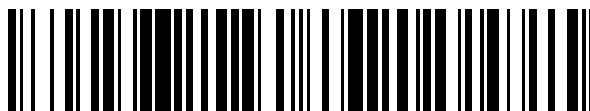


19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 374 332**

51 Int. Cl.:  
**H04M 3/42** (2006.01)  
**H04Q 3/00** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 96 Número de solicitud europea: **08715316 .9**  
96 Fecha de presentación: **25.03.2008**  
97 Número de publicación de la solicitud: **2109335**  
97 Fecha de publicación de la solicitud: **14.10.2009**

54 Título: **MÉTODO, DISPOSITIVO Y SISTEMA PARA EJECUTAR UNA ORDEN DE PROVISIÓN DE SERVICIO.**

30 Prioridad:  
**03.08.2007 CN 200710075566**

45 Fecha de publicación de la mención BOPI:  
**15.02.2012**

45 Fecha de la publicación del folleto de la patente:  
**15.02.2012**

73 Titular/es:  
**Huawei Technologies Co., Ltd.  
Huawei Administration Building Bantian  
Longgang District, Shenzhen  
Guangdong 518129 , CN**

72 Inventor/es:  
**SUN, Wei**

74 Agente: **Lehmann Novo, Isabel**

**ES 2 374 332 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**

Método, dispositivo y sistema para ejecutar una orden de provisión de servicio

**Campo de la invención**

5 La presente invención está relacionada con el campo de las comunicaciones, en particular con un método, un dispositivo y un sistema para ejecutar una orden de provisión de servicio.

**Antecedentes de la invención**

10 Para proporcionar servicios, un Sistema de Soporte de Negocios, BSS, p.e., Gestión de Relación con el Cliente, sistema CRM, sistema de Atención al Usuario y Sistema de Facturación, u otros Sistemas de Soporte de Operaciones, OSS, es necesario tener interfaces a Elementos de Red, NEs, en la parte de red, p.e., Registro de Localización Local, HLR, o Centro de Autenticación, AuC, Red Inalámbrica Inteligente, WIN, Plataforma Móvil de Servicio de Datos, MDSP, Centro de Servicio de Mensajes Cortos, SMSC, Sistema de Correo de Voz, VMS, Centro de Servicio de Mensajes Multimedia, MMSC, Sistema de Mensajería Unificada, UM, Red de Próxima generación, NGN, Red de Telefonía Pública Conmutada, PSTN, de modo que se pueda llevar a cabo la provisión del servicio. Como se muestra en la Figura 1, el Sistema de Provisión del Servicio es equivalente a una interfaz entre el BSS u OSS y el NE, y también se denomina sistema de órdenes en línea, el cual recibe las peticiones de servicio del BSS o del OSS, envía peticiones de órdenes a los NEs a través de la red de equipos, recibe respuestas de órdenes de los NEs, y devuelve al BSS o al OSS respuestas a los servicios. Se puede ver un ejemplo en WO98/41002.

15 En el caso de las interfaces de provisión de servicios existentes, en general cada interfaz de NE se proporciona con un programa o proceso independiente, las interfaces son independientes entre sí, y cada interfaz tiene una lógica de proceso interna independiente. Como se muestra en la Figura 2, una base de datos de fichas de trabajo almacena todas las peticiones y órdenes de servicio para su envío, un proceso del NE recupera de la base de datos de fichas de trabajo la orden para su envío, envía al NE la orden, y escribe después en la base de datos de fichas de trabajo el resultado final de la ejecución del NE. Cada proceso del NE interactúa directamente con la base de datos de fichas de trabajo y recupera periódicamente la orden para su envío, y por lo tanto hace uso de algunos recursos de conexión a la base de datos de fichas de trabajo.

20 En la solución descrita más arriba, el consumo de tiempo necesario para enviar una orden consta de cuatro partes.

A. Intervalo de tiempo entre las operaciones de recuperación de la orden: el proceso del NE busca en la base de datos de fichas de trabajo y recupera las órdenes para su envío en un cierto intervalo de tiempo.

25 B. Tiempo necesario para recuperar una orden: el proceso del NE accede a la base de datos de fichas de trabajo para recuperar la orden y, al mismo tiempo, puede ser necesario algún trabajo de procesamiento previo para recuperar la orden.

C. Tiempo para enviar una orden, esperar la respuesta e implementar el análisis: después de la obtención de una orden, el proceso del NE envía la orden al NE correspondiente, espera la respuesta del NE y analiza el mensaje de respuesta después de la obtención de la respuesta.

30 D. Tiempo necesario para escribir posteriormente el resultado de la ejecución de la orden: el proceso del NE escribe posteriormente en la base de datos de fichas de trabajo el resultado de la ejecución de la orden.

35 Cuando el proceso del NE accede a la base de datos de fichas de trabajo para recuperar una orden, puede recuperar una única orden cada vez para su envío o recuperar un lote de órdenes cada vez para su envío. La recuperación de una única orden cada vez es por la necesidad para soportar lógicas complejas del servicio, como por ejemplo lógicas de prioridad, retrocesión, omisión de fallo, procesado en paralelo y procesado en serie, en los que la ejecución de una orden se asocia con la ejecución de otra orden. La recuperación de un lote de órdenes cada vez para su envío se utiliza normalmente en el caso de que no exista ninguna lógica compleja y únicamente interesa si las órdenes son correctas o no. Si es necesario soportar una pluralidad de conexiones con el mismo NE, cuando se insertan fichas de trabajo en la base de datos de fichas de trabajo normalmente se realiza una división con un algoritmo fijo (p.e., distribuir en un número de segmentos u otros) con el fin de asignar las órdenes a conexiones distintas. Sin embargo, normalmente es difícil llevar la cuenta del estado ocupado/no ocupado de una conexión en el algoritmo de asignación para llevar a cabo una asignación dinámica de forma razonable.

40 En la solución actual, en la interfaz de la base de datos de fichas de trabajo la interacción se realiza con alta frecuencia, los NEs respectivos necesitan interactuar con la base de datos de fichas de trabajo, lo que ocupa algunos recursos de la base de datos de fichas de trabajo. El acceso frecuente provoca una degradación del rendimiento de la base de datos de fichas de trabajo y afecta al rendimiento global del sistema. Además, debido a que la obtención, proceso y escritura posterior de todas las órdenes es demasiado dependiente de la base de datos de fichas de trabajo, los accesos excesivos a la base de datos de fichas de trabajo degrada la eficacia global del

proceso de órdenes individuales y órdenes por lotes.

### Resumen de la invención

5 El objeto de la presente invención es proporcionar un método para ejecutar órdenes de provisión de servicio de modo que se resuelva el problema actual de un rendimiento degradado de la base de datos de fichas de trabajo y un rendimiento degradado en el proceso de órdenes debido a que Elementos de Red, NEs, están conectados respectivamente a la base de datos de fichas de trabajo a través de procesos independientes.

10 Un método para ejecutar órdenes de provisión de servicio proporcionado en la presente invención incluye: recuperar al menos dos órdenes para su envío; insertar órdenes que tienen la misma característica de dependencia en una cola correspondiente de órdenes en memoria de acuerdo con la relación lógica entre órdenes del servicio; y asignar a la conexión correspondiente del NE una orden de la cola de órdenes en memoria para que un NE la ejecute en el momento.

15 Un dispositivo de provisión del servicio proporcionado en la presente invención incluye: una unidad de memoria intermedia para recuperación de órdenes y de escritura posterior adaptada para recuperar al menos dos órdenes para su envío; una unidad de gestión de colas de órdenes adaptada para insertar órdenes que tengan la misma característica de dependencia en una correspondiente cola de órdenes en memoria de acuerdo con la relación lógica entre órdenes del servicio; y una unidad de distribución de órdenes a los NE adaptada para asignar a la conexión correspondiente del NE la orden de la cola de órdenes de memoria para que un NE la ejecute en el momento.

20 Un sistema de provisión de servicio proporcionado en la presente invención incluye: una base de datos de fichas de trabajo adaptada para almacenar órdenes para su envío; una unidad de memoria intermedia de recuperación de órdenes y de escritura posterior adaptada para recuperar al menos dos órdenes para su envío en la base de datos de fichas de trabajo; una unidad de gestión de colas de órdenes adaptada para insertar órdenes que se van a enviar, con la misma característica de dependencia en una correspondiente cola de órdenes en memoria de acuerdo con la relación lógica entre órdenes del servicio; y una unidad de distribución de órdenes a los NE adaptada para asignar a la conexión correspondiente del NE la orden de la cola de órdenes de memoria para que un NE la ejecute en el momento.

30 En el modo de realización de la presente invención, las órdenes para ser enviadas en la base de datos de fichas de trabajo se recuperan uniformemente y se reenvían rápidamente a las conexiones correspondientes del NE mediante la cola de órdenes de la memoria para su ejecución y, a continuación, se escribe después el resultado de la ejecución de las órdenes en la base de datos de fichas de trabajo; por lo tanto, es útil para reducir la ocupación del recurso de la base de datos de fichas de trabajo y mejorar la eficacia del proceso de órdenes.

### Breve descripción de los dibujos

La Figura 1 es un diagrama esquemático que ilustra la situación de un sistema de provisión de servicio en una red en la técnica anterior;

35 La Figura 2 es un diagrama esquemático que ilustra el principio del proceso de instrucciones en la provisión de servicio en la técnica anterior;

La Figura 3 es una representación de la estructura de un árbol binario de órdenes en un ejemplo de acuerdo con un modo de realización de la presente invención;

40 La Figura 4 es un diagrama de flujo que ilustra la implementación de un método para ejecutar una orden de provisión de servicio de acuerdo con un modo de realización de la presente invención;

La Figura 5 es un diagrama de la estructura de un sistema de provisión de servicio de acuerdo con un modo de realización de la presente invención;

La Figura 6 es un diagrama de la estructura de una unidad de gestión de colas en el sistema de provisión de servicio de acuerdo con un modo de realización de la presente invención;

45 La Figura 7 es un diagrama de la estructura de una unidad de distribución de instrucciones en el sistema de provisión de servicio de acuerdo con un modo de realización de la presente invención; y

La Figura 8 es un diagrama esquemático de la ejecución de una orden en una cola de órdenes de memoria en un NE de acuerdo con un modo de realización de la presente invención.

### Descripción detallada de los modos de realización

50 A continuación se describe la presente invención en detalle con referencia a los modos de realización y los dibujos

adjuntos para hacer entender mejor el objeto, la solución técnica y las ventajas de la presente invención. Se debe entender que aquí se describen los modos de realización únicamente como ilustración de la presente invención en lugar de constituir cualquier limitación a la presente invención.

5 En el modo de realización de la presente invención, las órdenes de la base de datos de fichas de trabajo que se van a enviar se recuperan y se reenvían rápidamente al Elemento de Red correspondiente, las conexiones del NE mediante la cola de órdenes de memoria para su ejecución y, después, el resultado de la ejecución de las órdenes se escribe posteriormente en la base de datos de fichas de trabajo; por lo tanto, es útil para reducir la ocupación del recurso de la base de datos de fichas de trabajo, mejora la eficacia del procesamiento de las órdenes y soporta varias lógicas de servicio complejas.

10 En un modo de realización de la presente invención, en la base de datos de fichas de trabajo se pueden almacenar múltiples lotes de órdenes de fichas de trabajo, y los distintos lotes de fichas de trabajo se identifican mediante el BATCH IDs (identificación de lote) de las fichas de trabajo. Un lote de fichas de trabajo puede incluir una pluralidad de fichas de trabajo, como por ejemplo la provisión de una cuenta general, la provisión del servicio de mensajes cortos, SMS, y la provisión del Servicio de Paquetes de Radio General, GPRS. La pluralidad de fichas de trabajo en un lote de fichas de trabajo tiene, en general, cierta relación de negocio, por ejemplo, si falla la provisión de una  
15 cuenta, no se ejecutarán las siguientes fichas de trabajo. En un lote de fichas de trabajo, la secuencia de cada ficha de trabajo se identifica por un número de secuencia interno, SEQ, del lote de fichas de trabajo; y si en un lote de fichas de trabajo existe únicamente una ficha de trabajo, el SEQ de la ficha de trabajo puede ser "1".

20 Una ficha de trabajo puede incluir una pluralidad de órdenes para que distintos NEs las ejecuten y los NEs pueden ejecutar las órdenes en una secuencia de ejecución en paralelo y ejecución en serie. En un modo de realización de la presente invención, las secuencias de ejecución de órdenes de la misma ficha de trabajo que van a ejecutar NEs diferentes se identifican por el número de secuencia de ejecución del NE, NESEQ, por ejemplo, si tanto el NESEQ para el NE1 como el NESEQ para el NE2 son "1", indica que las órdenes de la ficha de trabajo para el NE1 y NE2 se pueden enviar en paralelo; y si el NESEQ para el NE1 es "1" y el NESEQs para el NE2 es "2", indica que la orden  
25 para el NE2 únicamente se puede enviar después de enviar la orden para el NE1. Si es necesario enviar las órdenes de una ficha de trabajo a un único NE, el NESEQ correspondiente al NE puede ser "1".

Incluso en el caso de que una ficha de trabajo se corresponda con un único NE, puede ser necesario enviar varias órdenes para completar el servicio. Por ejemplo, para algunos NEs tienen que enviarse dos órdenes para proporcionar el Servicio de Mensajes Cortos, SMS, i.e., se debe enviar un SMS Terminado en el Móvil, MT, y un SMS Originado en el Móvil, MO. Para el mismo NE puede existir ejecución en serie o en paralelo entre distintas órdenes. En un modo de realización de la presente invención, la secuencia de ejecución en serie y paralelo de distintas órdenes para el mismo NE de una ficha de trabajo se identifica mediante números de secuencia de órdenes internos, CMDSEQ, del NE. Por ejemplo, si no existe dependencia, limitación o relación de ejecución en serie entre la orden A y la orden B y, por lo tanto, la orden A y la orden B se pueden enviar en paralelo, los CMDSEQs de la orden A y de la orden B pueden ser los mismos; y si la orden A necesita enviarse antes que la transmisión de la orden B, el CMDSEQ de la orden A puede ser "1", y el CMDSEQ de la orden B puede ser "2". Si únicamente existe una orden en el caso de que una ficha de trabajo se corresponda con un único NE, el CMDSEQ de la orden puede ser "1".  
30

35 En un modo de realización de la presente invención, si en un momento determinado se puede ejecutar o no una orden se identifica por el número de órdenes (COUNT) (cuenta) adyacentes que necesitan ejecutarse antes de la ejecución de esta orden. Si el COUNT de una orden es "0", indica que esta orden ya se puede ejecutar; si el COUNT de una orden es "1", indica que existe una orden adyacente que se debe ejecutar antes que esta orden; si el COUNT de una orden es "2", indica que existen dos órdenes adyacentes que se deben ejecutar antes que esta orden; si existe una relación paralela entre las dos órdenes, se pueden enviar las dos órdenes en paralelo.  
40

45 En un modo de realización de la presente invención, para las órdenes que tienen la misma característica de dependencia, p.e., el mismo BATCH ID, el mismo Número de Identificación de Abonado Móvil Internacional, IMSI, o el mismo ISDN Internacional de Estación Móvil, MSISDN, la estructura de datos es un árbol binario entrelazado, y cada nodo del árbol tiene dos nodos hijos, i.e., un nodo de orden en paralelo y un nodo de orden en serie. Los nodos de órdenes en serie son iguales para una pluralidad de nodos, por ejemplo, las órdenes A, B, C y D se pueden ejecutar en paralelo mientras que la orden E se debe ejecutar después de la ejecución de las órdenes A, B, C y D, por lo tanto el nodo de orden en serie de las órdenes A, B, C y D es la orden E. El nodo de orden en paralelo correspondiente a la orden A es la orden B, el nodo de orden en paralelo correspondiente a la orden B es la orden C, y el nodo de orden en paralelo correspondiente a la orden C es la orden D. Por lo tanto, en la estructura de árbol binario de órdenes, el valor COUNT es el número de nodos padre de órdenes que tienen relación directa de limitación en serie con el nodo actual.  
50  
55

De acuerdo con este documento la estructura de árbol binario de órdenes se describe mediante un ejemplo. Se supone que el usuario A solicita ser provisto de dos lotes de servicios, y los BATCH IDs correspondientes a los dos lotes son BATCH1 y BATCH2. BATCH1 es para la provisión de cuentas generales, mientras que BATCH2 es para la

provisión del servicio de red GPRS. BATCH1 tiene dos fichas de trabajo SEQ1 y SEQ2, siendo los valores de SEQ "1" y "2". BATCH2 tiene una ficha de trabajo SEQ1, siendo el valor de SEQ "1".

5 En BATCH1, la ficha de trabajo SEQ1 tiene que enviar órdenes a tres NEs, i.e., NE1, NE2 y NE3. Para NE1 y NE2, las órdenes se pueden enviar en paralelo y, después de enviar las órdenes a NE1 y NE2, se pueden enviar a NE3  
 10 otras órdenes en serie. Por lo tanto, los NESEQs para NE1, NE2 y NE3 son "1", "1" y "2". Las órdenes a enviar a NE2 incluyen dos órdenes A y B que necesitan enviarse en serie, i.e., la orden B se puede enviar únicamente después de enviar la orden A; por lo tanto, los CMDSEQs para las órdenes A y B son "1" y "2". Al NE1 sólo se le va a enviar la orden C, y la CMDSEQ correspondiente a la orden C es "1". Al NE3 sólo se le va a enviar la orden D, y el  
 15 CMDSEQ correspondiente a la orden D es "1". Debido a que no hay ninguna orden que se tenga que ejecutar antes que la orden A, el valor COUNT de la orden A es "0", lo que también indica que la orden A se puede enviar de forma inmediata. Debido a que la orden B únicamente se puede ejecutar después de la ejecución de la orden A, el valor  
 20 COUNT de la orden B es "1". La orden D se puede ejecutar únicamente después de la ejecución de las órdenes A, B y C. Pero únicamente las órdenes B y C tienen relación de adyacencia con la orden D y las órdenes B y C se pueden ejecutar en paralelo por lo que el valor COUNT de la orden D es "2". Después de la ejecución de la orden B, el valor COUNT de la orden D se decrementa en 1; y después de la ejecución de la orden C, el valor COUNT de la orden de la orden D se decrementa en 1. Por lo tanto, después de que se han ejecutado ambas órdenes, el valor  
 25 COUNT de la orden D es "0", lo que indica que la orden D ya se puede enviar directamente al NE.

La ficha de trabajo SEQ2 tiene que enviar órdenes en paralelo a NE1, NE2 y NE3; por lo tanto, los NESEQ correspondientes son "1", "1" y "1". La orden E se corresponde con NE1, las órdenes F y G se corresponden con  
 20 NE2 y se envían en paralelo, y la orden H se corresponde con NE3. Por lo tanto, los CMDSEQs correspondientes a las órdenes E, F, G y H son, respectivamente, "1", "1", "1" y "1". Debido a que las órdenes E, F, G y H únicamente se pueden enviar después de la ejecución de la orden D, los valores COUNT correspondientes a las órdenes E, F, G y H son, respectivamente, "1", "1", "1" y "1".

En BATCH2, la ficha de trabajo SEQ1 tiene que enviar órdenes a tres NEs, NE1, NE2 y NE3. Las órdenes se envían  
 25 en paralelo a NE1 y NE2, y después de que se puedan enviar otras órdenes en serie a NE3. Por lo tanto, los NESEQs correspondientes a NE1, NE2 y NE3 son, respectivamente, "1", "1" y "2". Para NE2 existen dos órdenes I y J y las dos órdenes se tienen que enviar en serie, i.e., la orden J se puede enviar únicamente después de que se haya enviado la orden I; por lo tanto los CMDSEQs de las instrucciones I y J son, respectivamente, "1" y "2". Existe una orden K para NE1 y el CMDSEQ de la orden K es 1, lo que indica que la orden K se puede ejecutar en paralelo  
 30 con la orden I. Existe una orden L para NE3 y el CMDSEQ de la orden L es 1, lo que indica que la orden L se puede ejecutar únicamente después de que se hayan ejecutado las órdenes J y K. Los valores COUNT de las órdenes I y K son, respectivamente, "4" debido a que las órdenes E, F, G y H se tienen que enviar antes que las órdenes I y K. Debido a que la orden I se tiene que enviar antes que la orden J, el valor COUNT de la orden J es "1". Debido a que las órdenes J y K se tienen que enviar antes que la orden L, el valor COUNT de la orden L es "2".

35 La estructura de datos de las órdenes descritas más arriba se muestran en la tabla siguiente, y en la Figura 3 se muestra la estructura del árbol binario correspondiente a las órdenes.

BATCH ID	SEQ	NESEQ	NE	CMDSEQ	Orden	Orden Adyacente		
BATCH 1	1	1	NE 1	1	Orden A	Ninguna		
				2	Orden B	Orden A		
		2	1	NE 2	1	Orden C	Ninguna	
					1	Orden D	Orden B Orden C	
	2	1	NE 1	1	Orden E	Orden D		
				1	NE 2	1	Orden F	Orden D
						1	Orden G	Orden D
				1	NE 3	1	Orden H	Orden D
BATCH 2	1	1	NE 2	1	Orden I	Orden E Orden F Orden G Orden H		
				2	Orden J	Orden I		
		1	NE 1	1	1	Orden K	Orden E Orden F Orden G Orden H	
					1	NE 3	1	Orden L

5 Además, en un modo de realización de la presente invención, diferentes órdenes tienen diferentes prioridades. Se ejecutan primero las órdenes con las prioridades más altas. Sin embargo, para órdenes que tienen el mismo BATCH ID, el mismo IMSI o el mismo MSISDN no existe orden de prioridad para su ejecución entre ellas, y las órdenes se pueden ejecutar entre ellas en secuencia temporal.

La Figura 4 es un diagrama de flujo que ilustra la implementación de un método para ejecutar la orden de provisión de servicio de acuerdo con un modo de realización de la presente invención, lo que se describe a continuación en detalle:

10 En el paso S401 se recuperan las órdenes que se van a enviar.

En un modo de realización de la presente invención los interfaces de interacción entre los NEs y la base de datos de fichas de trabajo u otros almacenamientos de datos o entidades de grabación, p.e., un motor de flujos de trabajo, están unificados, a través de los cuales se envían y recuperan todas las órdenes de la base de datos de fichas de trabajo.

15 En el paso S402, las órdenes que tienen la misma característica de dependencia se insertan en la correspondiente cola de órdenes en memoria de acuerdo con la relación lógica entre órdenes del servicio a enviar.

20 En un modo de realización de la presente invención, las órdenes que tienen el mismo BATCH, el mismo IMSI o el mismo MSISDN se insertan en la cola de órdenes correspondiente en memoria, de acuerdo con la relación lógica entre las órdenes del servicio, p.e., las relaciones de dependencia, retrocesión, omisión de fallos, proceso en paralelo, proceso en serie o interacción. Las colas de órdenes en memoria son paralelas entre sí, mientras que las órdenes en una cola de órdenes en memoria tienen, generalmente, una relación en serie entre sí.

25 En un modo de realización de la presente invención, la estructura de datos de las órdenes en una cola de órdenes en memoria es un árbol binario entrelazado, lo que incluye un árbol binario de una cola de órdenes normales a ejecutar, un árbol binario de una cola de órdenes de retrocesión y un árbol binario de una cola de órdenes de retrocesión a ejecutar. Cada nodo del árbol tiene dos nodos hijo: un nodo de órdenes en paralelo y un nodo de órdenes en serie.

Cuando se insertan órdenes en la correspondiente cola de órdenes en memoria, se determina si existe la

correspondiente cola de órdenes en memoria de acuerdo con las características de dependencia de las órdenes, como por ejemplo BATCH ID, IMSI o MSISDN. Si existe la correspondiente cola de órdenes en memoria, las órdenes se insertan en la cola de órdenes en memoria que se corresponde con las características de dependencia de BATCH ID, IMSI o MSISDN de las órdenes de acuerdo con la relación lógica entre órdenes del servicio y, si no existe la correspondiente cola de órdenes en memoria, se crea una nueva cola de órdenes en memoria. Al mismo tiempo, el orden de ejecución de las órdenes, i.e., los valores COUNT de las órdenes, se ajusta en función de la ejecución de la orden y la relación lógica entre órdenes en la cola de órdenes en memoria.

En el paso S403, las órdenes de la cola de órdenes en memoria que van a ejecutar los NEs, están asignadas en este momento a las conexiones correspondientes de los NE.

En un modo de realización de la presente invención, las órdenes de la cola de órdenes en memoria que van a ejecutar los NE se pueden asignar en este momento a las conexiones correspondientes de los NE en modo paralelo, o en modo serie en función de las prioridades de las órdenes.

En el paso S404, los resultados de la ejecución de las órdenes en los NEs se escriben posteriormente en la base de datos de fichas de trabajo o en otro almacenamiento de datos o entidad de grabación, p.e., un motor de flujos de trabajo.

La Figura 5 muestra la estructura del sistema de provisión de servicio de acuerdo con un modo de realización de la presente invención. Para comodidad de la descripción, en la Figura 5 únicamente se muestran las partes relacionadas con la presente invención.

La base de datos 51 de fichas de trabajo almacena la información de todas las órdenes a enviar y los resultados de la ejecución de las órdenes son escritos posteriormente en la base de datos de fichas de trabajo. La información de las órdenes incluye el BATCH ID del lote de fichas de trabajo al que pertenece la orden, el SEQ de la ficha de trabajo en un lote de fichas de trabajo, el NESEQ y el CMDSEQ, y puede incluir la prioridad de la orden.

En un modo de realización de la presente invención, la base de datos 51 de fichas de trabajo también puede ser una unidad independiente del sistema de provisión de servicio. Por ejemplo, la base de datos 51 de fichas de trabajo se puede colocar en otra entidad de red y para su comunicación se puede conectar con el sistema de provisión de servicio. Alternativamente, la base de datos 51 de fichas de trabajo puede ser otro almacenamiento de datos o entidad de grabación, p.e., un motor de flujos de trabajo.

La unidad 52 de memoria intermedia para recuperación de órdenes y escritura posterior es una interfaz unificada para la interacción entre los NEs y la base de datos 51 de fichas de trabajo. La unidad 52 de memoria intermedia para recuperación de órdenes y escritura posterior recupera todas las órdenes que se van a enviar desde la base de datos 51 de fichas de trabajo a lo largo de un cierto intervalo de tiempo y escribe posteriormente en la base de datos 51 de fichas de trabajo los resultados de la ejecución de todas las órdenes. De ese modo, Las órdenes almacenadas en la base de datos 51 de fichas de trabajo se pueden cargar por lotes en segundo plano. En el proceso en el que la unidad 52 de memoria intermedia para recuperación de órdenes y escritura posterior recupera las órdenes por lotes, se puede configurar el intervalo de tiempo de recuperación de órdenes, el número de órdenes a recuperar cada vez y el número de órdenes que se pueden almacenar temporalmente en la memoria, para así evitar una ocupación excesiva del recurso de memoria debido al periodo de tiempo demasiado largo para recuperar una orden cada vez o demasiadas órdenes. La unidad 52 de memoria intermedia para recuperación de órdenes y escritura posterior recupera órdenes de la base de datos 51 de fichas de trabajo en la secuencia BATCH, SEQ, NESEQ y CMDSEQ, y almacena las órdenes en memoria en secuencia BATCH.

Para mejorar más aún el rendimiento de lectura/escritura del sistema, en un modo de realización de la presente invención, la unidad 52 de memoria intermedia para recuperación de órdenes y escritura posterior se puede implementar como una cache (memoria de almacenamiento de alta velocidad).

Después de recuperar las órdenes de la base de datos 51 de fichas de trabajo, la unidad 52 de memoria intermedia para recuperación de órdenes y escritura posterior envía todas las órdenes recuperadas a la unidad 53 de gestión de las colas de órdenes. La unidad 53 de gestión de las colas de órdenes distribuye de forma centralizada a las correspondientes colas de órdenes en memoria las órdenes enviadas desde la unidad 52 de memoria intermedia para recuperación de órdenes y escritura posterior. En un modo de realización de la presente invención, la unidad 52 de memoria intermedia para recuperación de órdenes y escritura posterior recupera órdenes de la base de datos 51 de fichas de trabajo en la unidad de BATCH, y la unidad 53 de gestión de colas de órdenes también distribuye las órdenes en la unidad de BATCH.

Una cola de órdenes en memoria es una cola formada con todas las órdenes que tienen el mismo BATCH ID, el mismo IMSI o el mismo MSISDN. Se debería observar que las colas de órdenes en memoria no se clasifican estrictamente por BATCH. Una cola de órdenes en memoria puede contener diferentes lotes de órdenes, las cuales tienen una característica común, i.e., los BATCH ID, IMSI o MSISDN son idénticos, como por ejemplo una cola de órdenes que tienen el mismo número (p.e. IMSI o MSISDN).

Cuando se distribuyen las órdenes, la unidad 53 de gestión de colas de órdenes asigna las órdenes a los nodos de órdenes correspondientes de acuerdo con el BATCH ID del lote de fichas de trabajo, el SEQ de la ficha de trabajo dentro de un lote de fichas de trabajo, el NESEQ y el CMDSEQ.

5 Las órdenes que tienen la misma característica de dependencia necesitan enviarse en secuencia temporal en lugar de en orden de prioridad, por ejemplo, primero se crea una cuenta para el usuario; después se provee el Servicio SMS, el Servicio de Tonos de Llamada, CRBT, y el Servicio GPRS; a continuación, se cambia el número de la cuenta de forma separada; después, se cambia la tarjeta; y por último, se cancela la cuenta. Para dichas operaciones, es necesario ejecutar en secuencia temporal estricta las órdenes correspondientes en los lotes. Por lo tanto, todas las órdenes que están asociadas entre sí con cierta relación de ejecución en serie se almacenan en una cola de órdenes en memoria. Se debería observar que las colas de órdenes en memoria tienen una relación de ejecución en paralelo entre sí y, por lo tanto, las órdenes se pueden enviar en paralelo.

10 En un modo de realización de la presente invención, la estructura de datos de las órdenes en una cola de órdenes en memoria es un árbol binario entrelazado. Se pueden gestionar diferentes órdenes mediante los árboles binarios correspondientes de la cola, incluyendo un árbol binario de la cola de órdenes normales a ejecutar, un árbol binario de una cola de órdenes de retrocesión y un árbol binario de la cola de órdenes de retrocesión a ejecutar. El árbol binario de la cola de órdenes normales a enviar almacena órdenes que no se pueden deshacer en la cola de órdenes en memoria y estas órdenes se envían de acuerdo con cierta lógica del servicio. De acuerdo con la lógica correspondiente, es necesario evaluar si se envía una parte de estas órdenes. Por ejemplo, una orden puede ser ejecutada únicamente cuando se determina que un servicio está en estado cerrado mediante la consulta en una orden de consulta anterior. El árbol binario de la cola de órdenes de retrocesión almacena todas las órdenes de retrocesión que no es necesario ejecutar temporalmente. Normalmente, estas órdenes no se ejecutarán, pero en algunos casos especiales se pueden activar. El árbol binario de la cola de órdenes de retrocesión a enviar almacena las órdenes de retrocesión a enviar. En el caso de que el resultado de la ejecución de una orden normal dispare la retrocesión, la orden de retrocesión correspondiente de la cola de órdenes de retrocesión se inserta en la cola de órdenes de retrocesión para ser enviada y espera su envío.

Cuando se han ejecutado todas las órdenes en la cola de órdenes en memoria, la unidad 53 de gestión de colas de órdenes libera la memoria y los objetos.

30 Como se muestra en la Figura 6, la unidad 53 de gestión de colas de órdenes incluye al menos una cola 61 de órdenes en memoria, y cada cola de órdenes en memoria recibe órdenes que tienen la misma característica de dependencia. Un módulo 62 de equivalencias entre colas y órdenes almacena información o relación de equivalencia entre un lote de órdenes o una IDs de orden y la correspondiente cola de órdenes en memoria. Un módulo 63 de inserción de órdenes busca o crea correspondientes colas de órdenes en memoria de acuerdo con la relación de equivalencia entre los lotes de órdenes o ID de orden y las correspondientes colas de órdenes en memoria, e inserta las órdenes en las respectivas colas de órdenes en memoria de acuerdo con la relación lógica entre órdenes del servicio. Un módulo 64 de ajuste del orden de ejecución ajusta los valores del COUNT de las órdenes de acuerdo con la ejecución de las órdenes en la cola de órdenes y la relación lógica entre órdenes del servicio, y de este modo ajusta el orden de ejecución de las órdenes.

40 Una unidad 54 de distribución de órdenes del NE asigna las órdenes de la cola de órdenes en memoria que van a ejecutar los NEs a las conexiones correspondientes de los NE. Después de la ejecución de las órdenes, la unidad 54 de distribución de órdenes del NE devuelve los resultados de la ejecución de órdenes en los NEs a la unidad 52 de memoria intermedia para recuperación de órdenes y escritura posterior, y a continuación la unidad 52 de memoria intermedia para recuperación de órdenes y escritura posterior escribe los resultados en paralelo en la base de datos 51 de fichas de trabajo.

45 Como se muestra en la Figura 7, la unidad 54 de distribución de órdenes a los NE incluye al menos una cola interna 71 de órdenes del NE adaptada para recibir órdenes para ser ejecutadas inmediatamente por el NE, i.e., órdenes que tienen COUNT=0. Cada cola interna de órdenes del NE se corresponde con un NE y cada NE puede tener una pluralidad de conexiones del NE. Un módulo 72 de distribución de órdenes y escritura posterior de resultados distribuye las órdenes de la cola de órdenes en memoria que va a ejecutar el NE inmediatamente a la correspondiente cola interna 71 de órdenes del NE. Además, otras órdenes que cumplen la condición de envío se asignan a la correspondiente cola interna de órdenes del NE de acuerdo con el resultado de la ejecución de la orden en curso.

55 En un modo de realización de la presente invención, las colas internas 71 de órdenes de los NE se pueden organizar en paralelo, en las que no existe orden de prioridad entre órdenes, y las órdenes en las colas internas 71 de órdenes de los NE también se pueden organizar en serie. Como se muestra en la Figura 8, la cola interna 71 de órdenes del NE organiza las órdenes recibidas en secuencia de prioridad o tiempo de envío. De este modo, se envían primero a las conexiones del NE las órdenes con prioridad mayor. En el caso de que un NE tenga una pluralidad de conexiones del NE, cada conexión del NE recupera mediante la cola interna 71 de órdenes del NE, las órdenes que se pueden enviar. Siempre y cuando una conexión esté inactiva, las órdenes para su envío se recuperan



automáticamente de la cola interna 71 de órdenes del NE.

Como conclusión, comparado con la técnica anterior, la presente invención tiene las siguientes ventajas:

1. Los interfaces de la base de datos de fichas de trabajo están unificados y de este modo se reduce la ocupación de recursos de la base de datos de fichas de trabajo.

5 Todas las interacciones con la base de datos de fichas de trabajo se implementan de forma centralizada a través de la unidad de memoria intermedia para recuperación de órdenes y escritura posterior en lugar de utilizar comunicaciones independientes mediante conexiones de los NE. La utilización de la interfaz unificada puede mejorar la eficiencia de la interacción con la base de datos de fichas de trabajo, reducir la ocupación de recursos de la base de datos de fichas de trabajo y asegurar el rendimiento de la base de datos de fichas de trabajo.

10 2. Se implementa un reenvío rápido a través de colas de órdenes en memoria, lo que mejora la eficacia del procesamiento de las órdenes.

Todas las órdenes se recuperan en memoria de forma centralizada a través de la unidad de memoria intermedia para recuperación de órdenes y escritura posterior para formar las colas de órdenes en memoria, de modo que se consigue una distribución y escritura posterior uniformes de todas las órdenes y un procesamiento único de las lógicas del servicio. Esta estrategia puede incrementar en gran medida la velocidad de reenvío de órdenes y mejorar la eficacia del procesamiento de órdenes.

15

3. Se soportan varias lógicas de servicio complejas.

Se pueden soportar más lógicas de procesamiento de servicios complejas gracias a la distribución de órdenes de forma uniforme, al procesamiento de varias lógicas de órdenes en serie y en paralelo mediante las colas de órdenes en memoria, a la activación de varias lógicas de procesamiento de transacciones complejas incluyendo dependencia, interacción, condición, retrocesión en función del resultado de la ejecución de la orden, y a la extensión de la lógica del servicio de distribución de órdenes de fichas de trabajo en lotes y la escritura posterior de los resultados de la ejecución de forma continua.

20

4. Se mejora el rendimiento del procesamiento concurrente de tareas.

25 En la técnica anterior, la recuperación de una orden y la escritura posterior del resultado de la ejecución en la base de datos de fichas de trabajo necesitan implementarse estrictamente en secuencia antes de enviar otra orden. En un modo de realización de la presente invención, se puede enviar otra orden inmediatamente una vez que se ha recuperado una orden y el resultado de su ejecución se ha escrito en la memoria, en lugar de tener que esperar la implementación estricta de la escritura posterior del resultado de la ejecución en la base de datos de fichas de trabajo. Los resultados de la ejecución de órdenes se pueden escribir en la base de datos de fichas de trabajo en modo concurrente con la técnica de multithreading (conurrencia) y la escritura concurrente puede mejorar el rendimiento del procesamiento de tareas concurrentes.

30

Las órdenes de distintas colas de órdenes en memoria se ejecutan concurrentemente. Esa característica, junto con la característica del aumento/disminución flexible de múltiples conexiones con un NE, produce una capacidad de procesamiento concurrente más robusta y una capacidad de configuración de conexiones múltiples concurrentes más flexible, de modo que las órdenes se pueden procesar más rápidamente. Mediante las múltiples conexiones con un NE, las órdenes se pueden recuperar de las colas de órdenes en el NE, y por lo tanto se puede implementar rápidamente el reenvío de órdenes.

35

5. Se reducen los requisitos del rendimiento del hardware.

40 En un modo de realización de la presente invención, se reducen de forma significativa las interacciones con la base de datos de fichas de trabajo, y se reducen en gran medida las operaciones de Entrada/Salida, I/O; además, se realizan más cálculos en memoria y los gestiona directamente la CPU; por lo tanto se reducen los requisitos del rendimiento del hardware.

6. Se mejora la eficiencia de compartición de carga entre múltiples conexiones de los NE.

45 En un modo de realización de la presente invención, se puede determinar de forma arbitraria el número de conexiones con un NE, y las órdenes en el NEs se pueden procesar concurrentemente en modo de compartición de carga; por lo tanto, se puede eliminar la situación en la que una conexión del NE esté en estado inactivo mientras que otra conexión del NE está ocupada. Además, las órdenes se pueden recuperar y los resultados de la ejecución se pueden escribir posteriormente con rapidez. Se utiliza completamente la capacidad de las conexiones de los NE.

50

**REIVINDICACIONES**

1. Un método para ejecutar órdenes de provisión de servicio, que comprende:  
recuperar al menos dos órdenes para ser enviadas;  
insertar las órdenes que tienen la misma característica de dependencia en una correspondiente cola de órdenes en memoria de acuerdo con la relación lógica entre las órdenes del servicio; y  
asignar una orden de una cola de órdenes en memoria a la conexión correspondiente del Elemento de Red, NE, para que el NE la ejecute inmediatamente.
2. El método de acuerdo con la reivindicación 1, donde el proceso de insertar órdenes que tienen la misma característica de dependencia en la correspondiente cola de órdenes en memoria de acuerdo con la relación lógica entre las órdenes del servicio comprende:  
buscar o crear la correspondiente cola de órdenes en memoria en función de la característica de dependencia de las órdenes; e  
insertar las órdenes a enviar en la cola de órdenes en memoria que se corresponde con la característica de dependencia de las órdenes de acuerdo con la relación lógica entre las órdenes del servicio.
3. El método de acuerdo con la reivindicación 2, donde el proceso de inserción de órdenes que tienen la misma característica de dependencia en la correspondiente cola de órdenes en memoria de acuerdo con la relación lógica entre las órdenes del servicio comprende, además:  
ajustar el orden de ejecución de las órdenes a enviar en función de la ejecución de la órdenes de la cola de órdenes en memoria y la relación lógica entre las órdenes del servicio.
4. El método de acuerdo con la reivindicación 1, 2 ó 3, donde una estructura de datos de órdenes en la cola de órdenes en memoria es un árbol binario entrelazado, y cada nodo del árbol tiene dos nodos hijo: un nodo para órdenes en paralelo y un nodo para órdenes en serie.
5. El método de acuerdo con una de las reivindicaciones 1 a 5, donde el árbol binario entrelazado comprende un árbol binario de la cola de órdenes normales a ejecutar, un árbol binario de la cola de órdenes de retrocesión y un árbol binario de la cola de órdenes de retrocesión a ejecutar.
6. El método de acuerdo con la reivindicación 1, donde las órdenes de la cola de órdenes en memoria a ejecutar inmediatamente por los NEs se asignan a las conexiones correspondientes de los NE en paralelo, o se disponen en las conexiones correspondientes de los NE en serie de acuerdo con las prioridades de las órdenes.
7. El método de acuerdo con cualquiera de las reivindicaciones 1-6, donde la característica de dependencia es el BATCH ID de la ficha de trabajo, o el Número de Identificación del Abonado Móvil Internacional, IMSN, o el ISDN Internacional de Estación Móvil, MSISDN.
8. El método de acuerdo con cualquiera de las reivindicaciones 1-7, que comprende, además: la escritura posterior del resultado de la ejecución de las órdenes desde la conexión actual del NE.
9. Un dispositivo de provisión de servicio que comprende:  
una unidad de memoria intermedia de recuperación de órdenes y escritura posterior adaptada para recuperar al menos dos órdenes para ser enviadas;  
una unidad de gestión de colas de órdenes adaptada para insertar órdenes que tienen la misma característica de dependencia en una correspondiente cola de órdenes en memoria, de acuerdo con la relación lógica entre órdenes del servicio; y  
una unidad de distribución de órdenes de Elementos de Red, NE, adaptada para asignar a las conexiones correspondientes de los NE las órdenes de la cola de órdenes en memoria a ejecutar inmediatamente por los NE.
10. El dispositivo de acuerdo con la reivindicación 9, donde la unidad de gestión de colas de órdenes comprende:  
al menos una cola de órdenes en memoria adaptada para recibir órdenes que tienen la misma característica de dependencia;
- un módulo de equivalencia entre colas y órdenes adaptado para almacenar información sobre la relación de equivalencia entre la característica de dependencia de las órdenes y la correspondiente cola de órdenes en memoria; y

un módulo de inserción de órdenes adaptado para buscar o crear correspondientes colas de órdenes en memoria en función de la relación de equivalencia entre las características de dependencia de las órdenes y las correspondientes colas de órdenes en memoria, y para insertar las órdenes en las respectivas colas de órdenes en memoria de acuerdo con la relación lógica entre las órdenes del servicio.

- 5 11. El dispositivo de acuerdo con la reivindicación 10, donde la unidad de gestión de colas de órdenes comprende, además: un módulo de ajuste del orden de ejecución adaptado para ajustar el orden de ejecución de las órdenes de acuerdo con la ejecución de las órdenes de la cola de órdenes en memoria y la relación lógica entre las órdenes del servicio.
- 10 12. El dispositivo de acuerdo con la reivindicación 9, 10 u 11, donde una estructura de datos de órdenes en la cola de órdenes en memoria es un árbol binario entrelazado, y cada nodo del árbol tiene dos nodos hijo: un nodo para órdenes en paralelo y un nodo para órdenes en serie.
13. El dispositivo de acuerdo con una de las reivindicaciones 9 a 12, donde la unidad de distribución de órdenes del NE comprende:
- 15 al menos una cola interna de órdenes del NE adaptada para recibir las órdenes para ser ejecutadas inmediatamente por el NE correspondiente; y
- un módulo de distribución de órdenes y escritura posterior del resultado adaptado para distribuir las órdenes de la cola de órdenes en memoria para ser ejecutadas inmediatamente por el NE en las correspondientes colas internas de órdenes de los NE, y para devolver el resultado de la ejecución de las órdenes de la conexión del NE a la unidad de memoria intermedia para recuperación de órdenes y escritura posterior; donde
- 20 la unidad de memoria intermedia para recuperación de órdenes y escritura posterior se adapta, además, para escribir posteriormente el resultado de la ejecución de las órdenes de la conexión actual del NE.
14. El dispositivo de acuerdo con una de las reivindicaciones 9 a 13, donde las órdenes de las colas internas de órdenes del NE se organizan en paralelo, o se organizan en serie de acuerdo con las prioridades de las órdenes.
15. Un sistema de provisión de servicio que comprende:
- 25 una base de datos de fichas de trabajo adaptada para almacenar órdenes a ser enviadas;
- y el dispositivo de ejecución de órdenes para la provisión del servicio de acuerdo con cualquiera de las reivindicaciones 9-14.

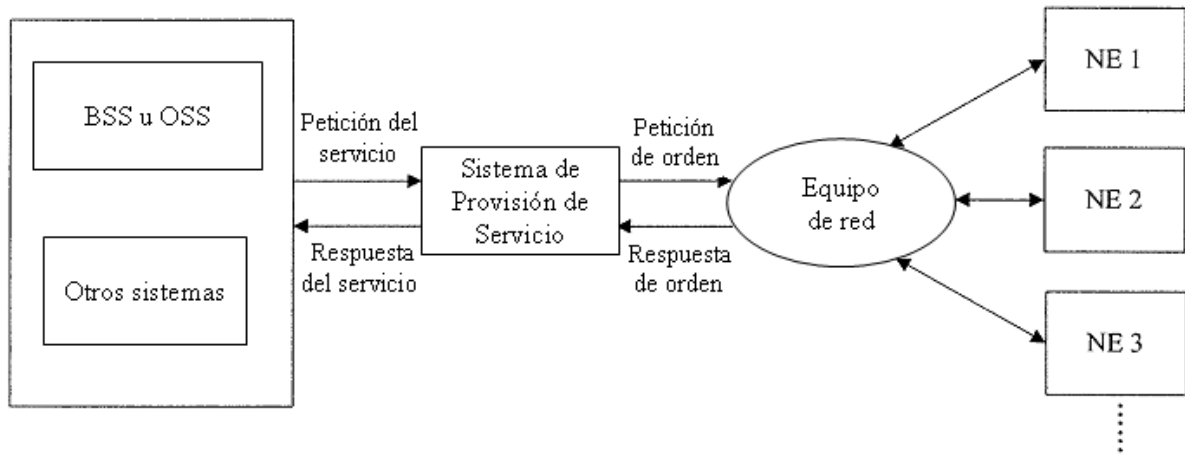


Fig. 1

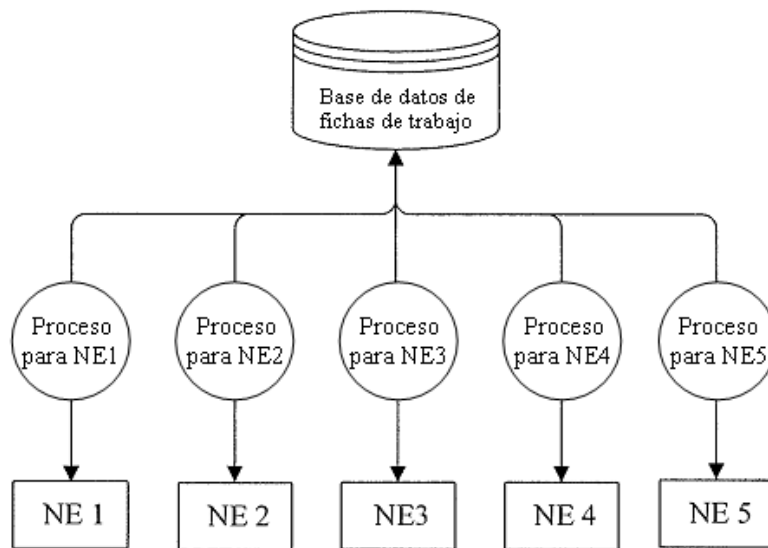


Fig. 2

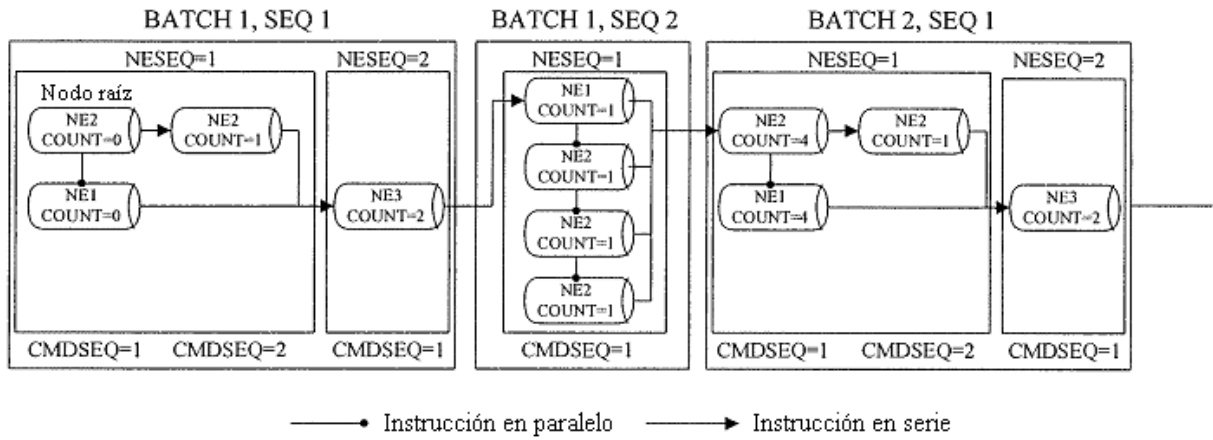


Fig. 3

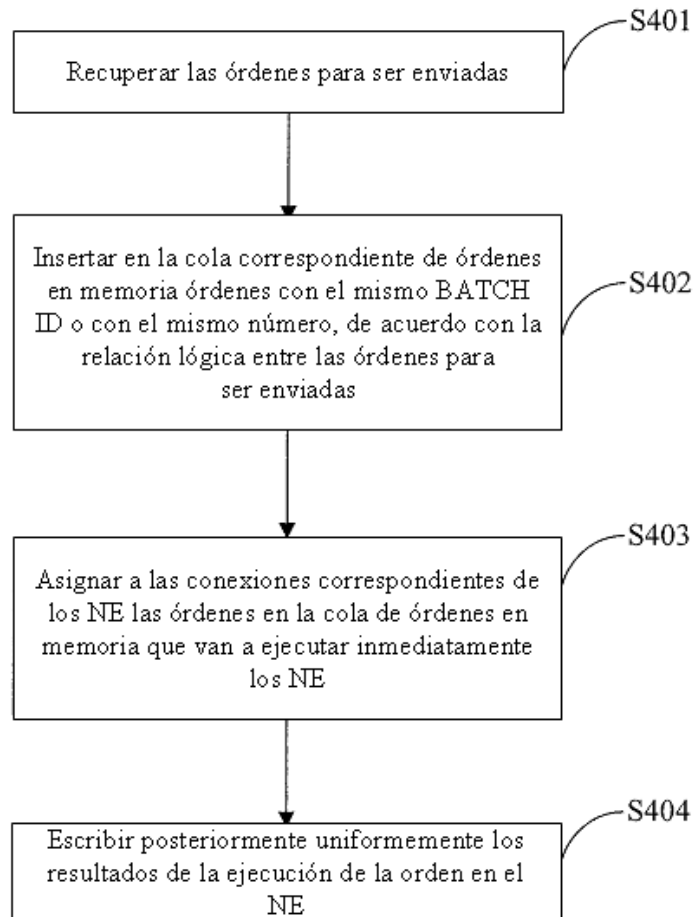


Fig. 4

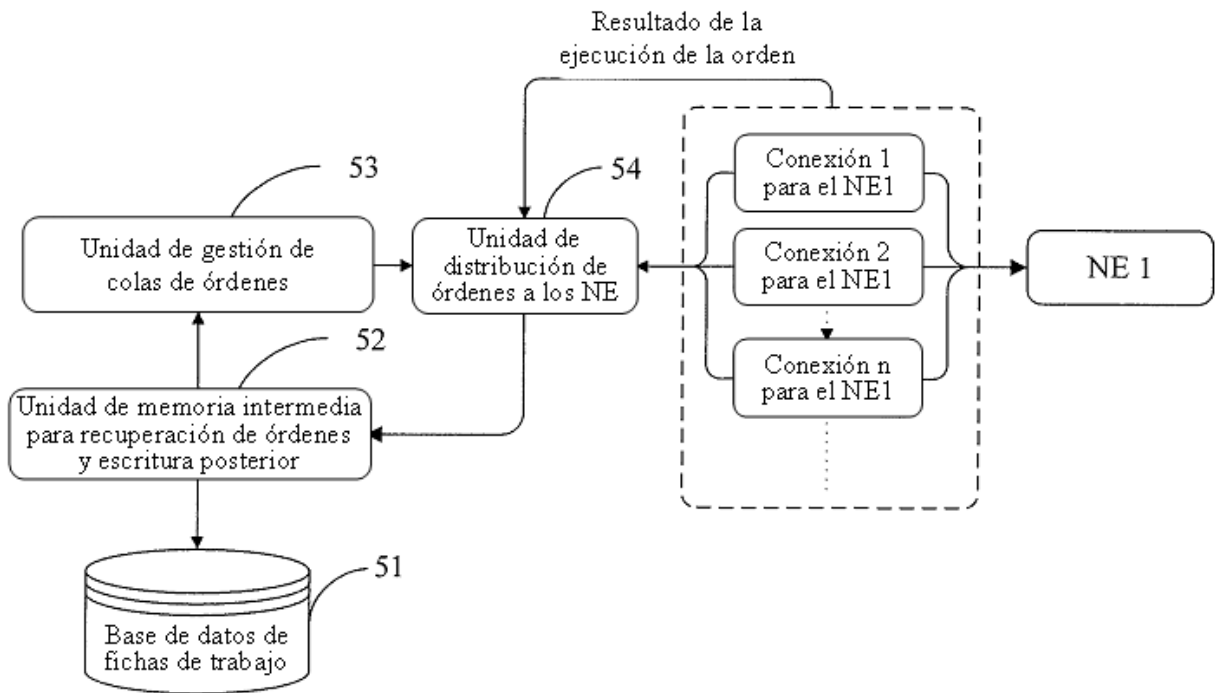


Fig. 5

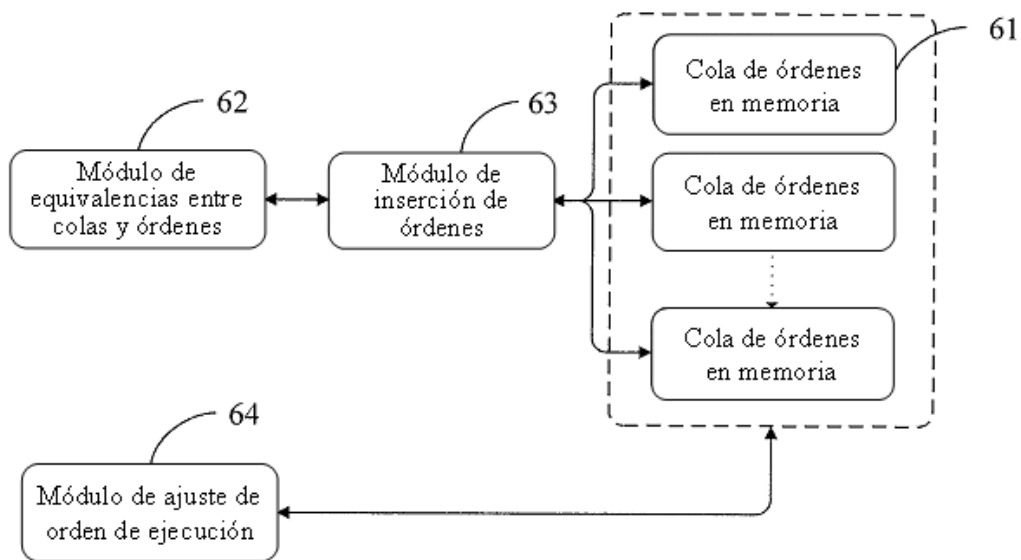


Fig. 6

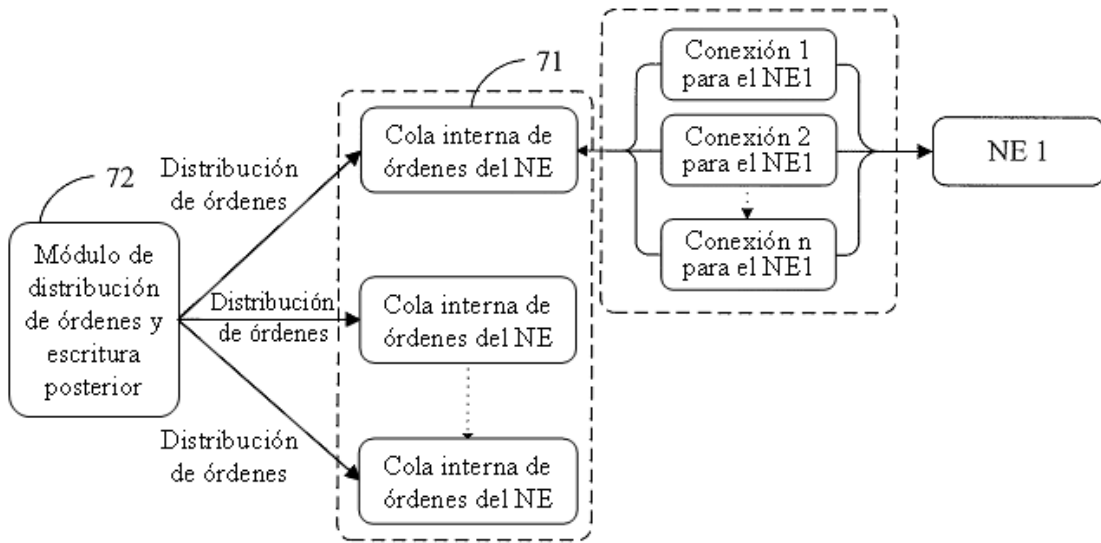


Fig. 7

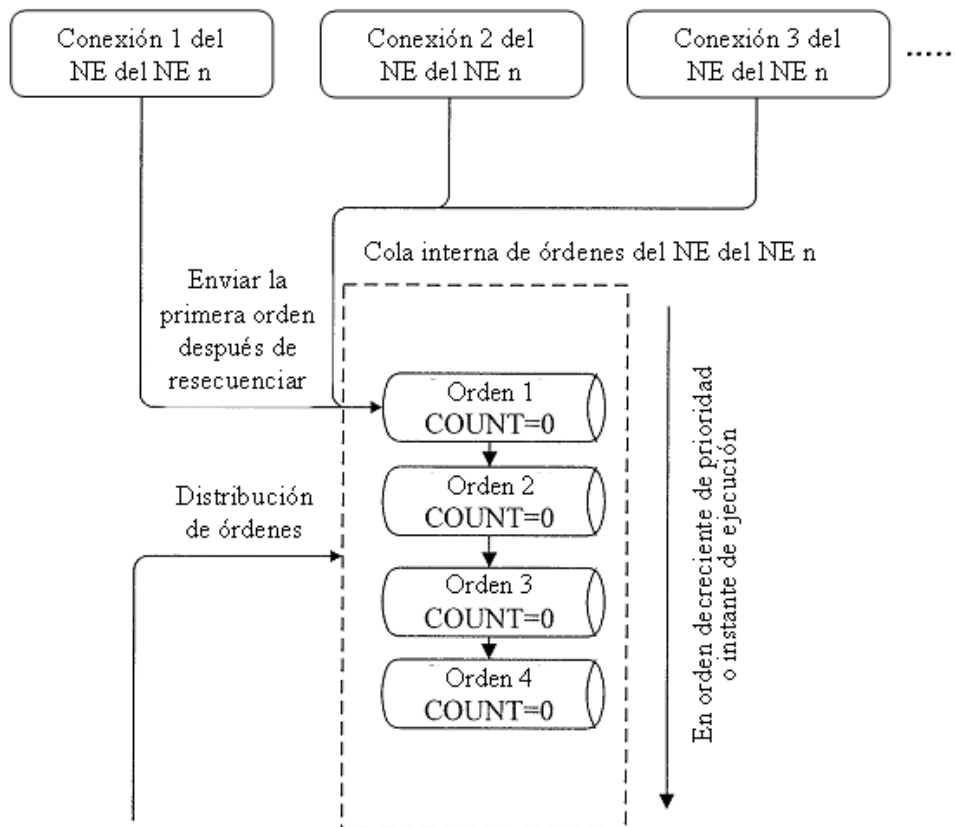


Fig. 8