

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 665 781**

51 Int. Cl.:

**G06F 17/30** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **08.11.2011 PCT/US2011/059700**

87 Fecha y número de publicación internacional: **24.05.2012 WO12067890**

96 Fecha de presentación y número de la solicitud europea: **08.11.2011 E 11785544 (5)**

97 Fecha y número de publicación de la concesión europea: **28.02.2018 EP 2641191**

54 Título: **Sistema de cálculo informático basado en rejilla asociado a una arquitectura de base de datos distribuida**

30 Prioridad:

**15.11.2010 US 946079**

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

**27.04.2018**

73 Titular/es:

**SAS INSTITUTE INC. (100.0%)  
Sas Campus Drive  
Cary, NC 27513, US**

72 Inventor/es:

**SCHABENBERGER, OLIVER y  
KRUEGER, STEVE**

74 Agente/Representante:

**LEHMANN NOVO, María Isabel**

**ES 2 665 781 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**

Sistema de cálculo informático basado en rejilla asociado a una arquitectura de base de datos distribuida

## 5 CAMPO DE LA INVENCION

La tecnología aquí descrita se refiere, en general, al procesamiento de datos distribuidos y más concretamente, al procesamiento de datos distribuidos utilizando sistemas de cálculo informático basados en rejilla y sistemas de base de datos distribuidas.

10

## ANTECEDENTES DE LA INVENCION

Las soluciones analíticas de alto rendimiento implican la localización conjunta de datos y código analítico. Lo anterior puede reducir la sobrecarga de I/O (entrada/salida) puesto que se pueden cargar grandes cantidades de datos en la memoria a través de una rejilla de nodos y puede tener lugar un cálculo informático en paralelo. Una técnica para la localización conjunta de datos y código analítico se refiere como el modelo en la base de datos. Otra técnica se conoce como el modelo fuera de la base de datos.

15

20

Con el modelo en la base de datos, el código analítico se ejecuta en los nodos de un sistema de base de datos distribuida en un entorno sin compartición. El proceso suele iniciarse mediante la llamada desde una consulta SQL a una función definida por el usuario (UDF) que fue instalada previamente en el sistema de gestión de la base de datos. Los datos son ya locales a los nodos o se desplazan a los nodos según lo demandado por la consulta SQL.

25

El modelo en la base de datos es un entorno centrado en SQL, sin compartición, en el que las funciones UDFs se ejecutan bajo el control del sistema de gestión de base de datos. Es decir, los nodos no pueden comunicarse entre sí, por lo general la información no puede persistir entre consultas en los nodos de la base de datos a menos que esté escrita en forma de tablas para el sistema de gestión de base de datos, y el sistema de gestión de base de datos controla los recursos consumidos por la función UDF. En este modelo, la base de datos puede proporcionar tolerancia a fallos, replicación y soporte para transacciones.

30

Con el modelo de fuera de la base de datos, se utiliza un entorno informático basado en rejilla en donde los datos residen en la memoria en los nodos de cálculo, y el código analítico, en lugar del sistema de gestión de la base de datos, controla todo el proceso. Los datos se ubican conjuntamente mediante su distribución previa en los nodos de la rejilla en donde el código analítico carga los datos locales en la memoria.

35

40

El modelo de fuera de la base de datos puede ser un entorno de todo compartido y también podría llamarse el modelo "sin la base de datos" puesto que no existe una conexión inmediata con una base de datos distribuida. Los datos pueden provenir, a modo de ejemplo, de una base de datos de Teradata®, pero se distribuyen en los nodos de cálculo de rejilla antes del análisis, y la conexión con la base de datos distribuida, en donde se originaron los datos se separa. En este entorno, el código analítico tiene control completo sobre la comunicación de nodo a nodo mediante la adición de un protocolo de paso de mensajes.

45

La técnica anterior más próxima es la del documento US 2007/0078960 A1: este es un sistema de gestión contable de cálculo informático basado en rejillas; los datos de rendimiento se capturan en un entorno de cálculo informático basado en rejillas y se memorizan en un depósito de datos direccionables por contexto con el fin de que sean accesibles para consultas relacionadas con su empleo

## SUMARIO DE LA INVENCION

50

La presente invención se refiere a un entorno de cálculo informático según se estipula en la reivindicación 1 y un método según se estipula en la reivindicación 8.

55

De conformidad con las enseñanzas proporcionadas en este documento, se dan a conocer métodos y sistemas para un sistema de cálculo informático basado en rejilla que realiza cálculos analíticos sobre datos memorizados en un sistema de base de datos distribuida. Esta técnica para la localización conjunta de datos y código analítico, se refiere aquí como el modelo de base de datos asociado, que permite al código analítico ejercer el control completo del proceso analítico y el control completo del acceso a datos, pero utiliza una base de datos distribuida para alimentar datos a los nodos de cálculo locales. Realiza la localización conjunta de datos, el código analítico y el hardware necesario para memorizar los datos y realizar el código analítico. El código analítico permanece bajo control y permite la comunicación de nodo a nodo. Los datos se recuperan directamente a partir de una base de datos distribuida co-ubicada para su carga en la memoria en los nodos de rejilla, que permite la flexibilidad necesaria para realizar los cálculos analíticos tal como en el modelo de fuera de base de datos y utilizar las capacidades de base de datos distribuida para proporcionar tolerancia a fallos, replicación y soporte para transacciones.

60

65

A modo de ejemplo, un componente de software basado en rejilla se ejecuta en un nodo de control y proporciona órdenes para una pluralidad de componentes de software basados en rejilla de nodo de operador. Cada uno de los

componentes de software basados en rejilla de nodo de operador está asociado con, y se ejecuta, en un nodo separado. El nodo ejecuta, además, software de gestión de base de datos (DBMS) que gestiona datos sobre los que pueden realizarse cálculos analíticos, o consultas, o actividad de carga. Al estar el sistema DBMS situado en cada nodo en comparación con el modelo en la base de datos antes descrito y/o el modelo fuera de la base de datos, se pueden desplazar los datos lo menos posible. Al procesar los datos en el nodo de la base de datos en donde residen, se minimiza el movimiento. Al enviar los datos fuera de la base de datos para el proceso de DBMS, es posible un mayor control. A modo de ejemplo, si fuere necesario, resulta posible una comunicación con otros nodos. El procesamiento de los datos en el interior de la base de datos no proporciona la oposición de comunicación entre unidades de trabajo (las funciones UDFs se ejecutan en los nodos DBMS). Con la capacidad de comunicarse de este modo, se pueden realizar tareas analíticas mucho más complejas, que son difíciles o imposibles de realizar en un modelo en la base de datos. Otra ventaja de desplazar los datos del sistema DBMS a un proceso de localización conjunta es que es posible memorizar en memoria caché los datos; los algoritmos que tienen que pasar a través de los datos varias veces ahora pueden aprovechar una representación en memoria de los datos, en lugar de tener que volver a realizar su lectura desde el disco. El componente de software basado en rejilla, en el nodo de control, está configurado para solicitar al DBMS en el nodo de control, con el fin de que el sistema DBMS, en una pluralidad de nodos de operador, haga disponible una parte de los datos para el componente de software basado en rejilla local, de modo que su nodo proporcione instrucciones a los componentes de software basados en rejilla, entre la pluralidad de nodos de operador, para realizar un cálculo analítico sobre los datos recibidos y envíe los resultados del análisis de datos al componente de software basado en rejilla en el nodo de control; y reúna los resultados del análisis de datos puesto en práctica por los componentes de software basados en rejilla, en la pluralidad de nodos de operador.

Más concretamente, la invención se dirige a un sistema de cálculo informático basado en rejilla para realizar cálculos analíticos sobre datos memorizados en un sistema de base de datos distribuida, comprendiendo el sistema de cálculo informático basado en rejilla:

una pluralidad de nodos de cálculo informático basados en rejilla que incluyen un nodo de control y una pluralidad de nodos de operador, comprendiendo cada nodo:

un procesador para realizar un software de gestión de base de datos (DBMS) y para ejecutar un componente de software basado en rejilla,

una memoria intermedia local accesible por el componente de software basado en rejilla, y

un soporte de memorización local accesible por el sistema DBMS para la memorización de datos;

en donde el componente de software basado en rejilla está configurado para realizar los cálculos analíticos sobre una parte de los datos memorizados por el sistema de gestión de base de datos;

estando el componente de software basado en rejilla, en el nodo de control, configurado para:

solicitar el sistema DBMS en el nodo de control para hacer que DBMS, en una pluralidad de nodos de operador, haga disponible una parte de los datos para el componente de software basado en rejilla local, para su propio nodo;

proporcionar instrucciones a los componentes de software basados en rejilla, en la pluralidad de nodos de operador, para realizar un cálculo analítico sobre los nodos recibidos y para enviar los resultados del análisis de datos al componente de software basado en rejilla en el nodo de control; y

reunir los resultados del análisis de los datos, que se realiza por los componentes de software basados en rejilla, en la pluralidad de nodos de operador.

Una forma de realización preferida es que el sistema DBMS, en una pluralidad de nodos de operador, esté configurado para hacer que los datos estén disponibles para su componente de software basado en rejilla local, guardando los datos en una memoria compartida, accesible a nivel local por, a la vez, el sistema DBMS y el componente de software basado en rejilla.

Con la invención se propone, además, que el sistema DBMS, en la pluralidad de nodos de operador, está configurado para hacer disponibles los datos para su propio componente de software basado en rejilla local, mediante la comunicación a través de una conexión eléctrica enchufable.

En la forma de realización anteriormente descrita, resulta especialmente preferido que los componentes de software basados en rejilla, en la pluralidad de nodos de operador, estén configurados para abrir conexiones enchufables utilizando una dirección de puerto común.

Con la invención se propone, además, para la forma de realización anteriormente mencionada, que el componente de software basado en rejilla, en el nodo de control, esté configurado para proporcionar la dirección de puerto común

al DBMS local, para su propio nodo, y el sistema DBMS, en el nodo de control, esté configurado para proporcionar la dirección de puerto común al DBMS en la pluralidad de nodos de operador.

5 En un sistema de la invención puede estar caracterizado, además, por el hecho de que los componentes de software basados en rejilla, en la pluralidad de nodos de operador, estén configurados para memorizar los datos recibidos y para realizar el análisis de datos utilizando su memoria intermedia local.

10 Se propone, además, que la pluralidad de nodos de operador estén configurados para transmitir los resultados procedentes del análisis de datos a su propio DBMS local, para su memorización en la base de datos local.

El sistema puede estar caracterizado, además, por cuanto que los resultados del análisis recibidos desde la pluralidad de componentes de software basados en rejilla del nodo de operador, se reciben a través de una red.

15 Una forma de realización ventajosa, a modo de ejemplo, se prefiere que una pluralidad de componentes de software basados en rejilla del nodo de operador transmita sus resultados utilizando la interfaz MPI a través de la red.

La invención propone, además, que los resultados reunidos por el componente de software basado en rejilla del nodo de control, se memoricen en el sistema de base de datos distribuida.

20 Es también preferido que los resultados reunidos por el componente de software basado en rejilla del nodo de control se transmitan a un componente de software basado en rejilla, en un nodo de cliente.

El sistema de conformidad con la invención puede estar caracterizado, además, por cuanto que a la detección de un fallo operativo en uno de los nodos de operador:

25 el sistema DBMS, en el nodo de control, está configurado para hacer que el sistema DBMS, en uno de los nodos de operador restantes, redistribuya la parte de los datos anteriormente distribuidos para el componente de software basado en rejilla, en el nodo en condición de fallo, al componente de software basado en rejilla local, para ese nodo;

30 el componente de software basado en rejilla que recibe los datos re-distribuidos está configurado para realizar un cálculo analítico sobre los datos redistribuidos y proporcionar los resultados al componente de software basado en rejilla en el nodo de control; y

35 el componente de software basado en rejilla, en el nodo de control, está configurado para reunir los resultados recibidos a partir de los componentes de software basados en rejilla, en los nodos de operador restantes.

Además, el sistema puede incluir un componente de software basado en rejilla en un nodo de cliente: estando el componente de software basado en rejilla, en el nodo de cliente, configurado con un primer modo de ejecución y un segundo modo de ejecución;

40 estando el componente de software basado en rejilla del nodo de cliente, en el primer modo de ejecución, configurado para realizar un análisis de datos local para proporcionar los resultados al software del cliente;

45 el componente de software basado en rejilla del nodo de cliente, en el segundo modo de ejecución, está configurado para solicitar el componente de software basado en rejilla, en el nodo de control, para realizar el análisis de datos en un entorno de cálculo informático basado en rejilla; y

50 el componente de software basado en rejilla, en el nodo de cliente, está configurado para funcionar en el primer modo o en el segundo modo, sobre la base de una indicación que se proporciona por el software del cliente.

55 En la forma de realización anteriormente descrita, resulta especialmente preferido que el componente de software basado en rejilla del nodo de cliente esté configurado para recibir cuestiones *ad hoc* procedentes de una aplicación de cliente y para proporcionar respuestas a las cuestiones *ad hoc* proporcionando los resultados reunidos, por el componente de software basado en rejilla a la aplicación del cliente.

60 La invención da a conocer, además, que el sistema incluye un componente de software basado en rejilla de nodo de cliente que está configurado para recibir cuestiones *ad hoc* procedentes de una aplicación del cliente y para proporcionar respuestas a las cuestiones *ad hoc* proporcionando los resultados reunidos por el componente de software basado en rejilla del nodo de control, a la aplicación del cliente.

Además, se propone que el componente de software basado en rejilla, en el nodo de control, esté configurado para reunir los resultados mediante la agregación y concatenación de los resultados del análisis realizado en los procesadores de datos del nodo de operador.

65 Además, un sistema puede estar caracterizado por cuanto que el componente de software basado en rejilla, en una pluralidad de nodos de cálculo informático basado en rejilla, se pone en práctica en un entorno de máquina virtual en

cada nodo.

Otro ejemplo puede estar caracterizado por cuanto que el software de gestión de base de datos, en una pluralidad de nodos de cálculo informático basado en rejilla, se pone en práctica en un entorno de máquina virtual en cada

5 nodo.  
Además, resulta preferible que el componente de software basado en rejilla, en la pluralidad de nodos de cálculo informático basado en rejilla, se ponga en práctica en un entorno de máquina virtual separado en cada nodo.

10 Por último, se propone para el sistema secundario que el componente de software basado en rejilla, en la pluralidad de nodos de cálculo basados en rejilla, se ponga en práctica en un entorno de máquina virtual común, con el software de gestión de base de datos en cada nodo.

15 La invención da a conocer, además, un método en un sistema de cálculo informático basado en rejilla, para la puesta en práctica de cálculos analíticos sobre datos memorizados en un sistema de base de datos distribuida, comprendiendo dicho método:

20 el suministro de una pluralidad de nodos de cálculo basados en rejilla que incluyen un nodo de control y una pluralidad de nodos de operador, comprendiendo cada nodo un procesador para la puesta en práctica de software de gestión de base de datos (DBMS) y para ejecutar un componente de software basado en rejilla, siendo la memoria intermedia local accesible para el componente de software basado en rejilla, y siendo un soporte de memorización local accesible para el sistema DBMS para la memorización de datos;

25 la aplicación del software DBMS, en el nodo de control, para hacer que el sistema DBMS, en una pluralidad de nodos de operador, haga disponible una parte de los datos para el componente de software basado en rejilla local, en su propio nodo;

30 proporcionando instrucciones a los componentes de software basados en rejilla, en la pluralidad de nodos de operador, para realizar un cálculo analítico sobre los datos recibidos, y para enviar los resultados del análisis de datos, al componente de software basado en rejilla en el nodo de control; y para reunir los resultados del análisis de datos realizado por los componentes de software basados en rejilla en la pluralidad de nodos de operador.

35 Resulta especialmente preferido que el sistema DBMS, en una pluralidad de nodos de operador, esté configurado para hacer disponibles los datos para su propio componente de software basado en rejilla, guardando los datos en una memoria compartida, accesible a nivel local por, a la vez, el sistema DBMS y el componente de software basado en rejilla.

40 El método de la invención puede estar caracterizado por cuanto que el sistema DBMS, en la pluralidad de nodos de operador, esté configurado para hacer disponibles los datos para su propio componente de software basado en rejilla, mediante una comunicación a través de una conexión enchufable.

45 Para esta realización ejemplo, resulta especialmente preferible que los componentes de software basados en rejilla, en la pluralidad de nodos de operador, estén configurados para abrir conexiones enchufables utilizando una dirección de puerto común.

50 La realización, a modo de ejemplo, antes descrita puede estar caracterizada, además, por cuanto que el componente de software basado en rejilla, en el nodo de control, esté configurado para proporcionar la dirección de puerto común al DBMS local para su propio nodo, y el sistema DBMS, en el nodo de control, esté configurado para proporcionar la dirección de puerto común al DBMS en la pluralidad de nodos de operador.

55 Resulta también preferible que los componentes de software basados en rejilla, en la pluralidad de nodos de operador, estén configurados para memorizar los datos recibidos y para realizar el análisis de los datos utilizando su propia memoria intermedia local.

La invención propone, además, que la pluralidad de nodos de operador estén configurados para transmitir los resultados procedentes del análisis de datos, a su propio DBMS local, para su memorización en la base de datos distribuida.

60 Un método de la invención puede estar caracterizado por cuanto que los resultados del análisis, recibidos a partir de la pluralidad de componentes de software basados en rejilla del nodo de operador, se reciban a través de una red.

65 En el ejemplo anteriormente descrito, resulta especialmente preferido que una pluralidad de componentes de software basados en rejilla de nodo de operador, transmita sus propios resultados utilizando la interfaz MPI a través de la red.

Además, la invención propone que el método incluya la memorización de los resultados reunidos por el componente

de software basado en rejilla en el sistema de base de datos distribuida.

5 Con la invención se propone, además, que el método incluya la transmisión de los resultados reunidos, por el componente de software basado en rejilla del nodo de control, a un componente de software basado en rejilla en un nodo de cliente.

Un método de la invención puede estar caracterizado, además, por cuanto que a la detección de un fallo operativo en uno de los nodos de operador:

10 el sistema DBMS, en el nodo de control, esté configurado para hacer que el sistema DBMS, en uno de los nodos de operador restantes, redistribuya la parte de los datos previamente distribuidos al componente de software basado en rejilla, en el nodo en condición de fallo, al componente de software basado en rejilla local para ese nodo;

15 el componente de software basado en rejilla, que recibe los datos redistribuidos, está configurado para realizar un cálculo analítico sobre los datos redistribuidos, y proporcionar los resultados al componente de software basado en rejilla, en el nodo de control; y

20 el componente de software basado en rejilla, en el nodo de control, está configurado para reunir los resultados recibidos desde los componentes de software basados en rejilla, en los nodos de operador restantes.

Se propone, además, que el método comprenda, además, la etapa de proporcionar un componente de software basado en rejilla, en un nodo de cliente cuando:

25 el componente de software basado en rejilla, en el nodo de cliente, esté configurado con un primer modo de ejecución y un segundo modo de ejecución;

el componente de software basado en rejilla del nodo de cliente, en el primer modo de ejecución, esté configurado para realizar un análisis de datos local para proporcionar los resultados al software del cliente;

30 estando el componente de software basado en rejilla del nodo de cliente configurado para solicitar el componente de software basado en rejilla, en el nodo de control, para realizar el análisis de datos en un entorno de cálculo informático basado en rejilla; y

35 estando el componente de software basado en rejilla, en el nodo de cliente, configurado para funcionar en el primer modo o el segundo modo, sobre la base de una indicación proporcionada por el software del cliente.

40 El método de la invención puede estar caracterizado, además, por cuanto que el componente de software basado en rejilla del nodo de cliente esté configurado para recibir cuestiones *ad hoc* procedentes de una aplicación del cliente, y para proporcionar respuestas a las cuestiones *ad hoc* proporcionando los resultados reunidos por el componente de software basado en rejilla del nodo de control, a la aplicación del cliente.

45 La invención propone, además, que el método comprenda proporcionar un componente de software basado en rejilla de nodo de cliente que esté configurado para recibir cuestiones *ad hoc* desde una aplicación del cliente y para proporcionar respuestas a las cuestiones *ad hoc* proporcionando los resultados reunidos por el componente de software basado en rejilla del nodo de control a la aplicación del cliente.

Se propone que la etapa de realizar la reunión comprenda la agregación y concatenación de los resultados del análisis que se realiza en los procesadores de datos del nodo de operador.

## 50 BREVE DESCRIPCIÓN DE LOS DIBUJOS

La Figura 1 es un diagrama de bloques que ilustra un entorno en donde usuarios pueden interactuar con un entorno de cálculo informático.

55 La Figura 2 es un diagrama de bloques que ilustra un entorno de cálculo informático que contiene un sistema de cálculo informático basado en rejilla y un sistema de gestión de base de datos.

La Figura 3 es un diagrama de bloques que ilustra componentes de hardware y software de un sistema de cálculo informático basado en rejilla.

60 La Figura 4 es un diagrama de bloques que ilustra componentes de hardware y software de un sistema de gestión de base de datos distribuida.

65 Las Figuras 5 y 6 son diagramas de bloque que ilustran componentes de hardware y software de un sistema de cálculo informático basado en rejilla, ubicado, de forma conjunta, con un sistema de base de datos distribuida.

Las Figuras 7-11 son diagramas de flujo de procesos que ilustran un flujo de proceso de un sistema de cálculo informático basado en rejilla para el análisis de datos en un entorno basado en rejillas o en un entorno denominado solo.

- 5 Las Figuras 12-16 son diagramas de flujo de procesos que ilustran un flujo de proceso de componentes de software para realizar un análisis de datos en un entorno de cálculo informático basado en rejilla.

#### DESCRIPCIÓN DETALLADA DE LA INVENCION

- 10 La Figura 1 ilustra en la referencia 30 un entorno de cálculo informático para el procesamiento de grandes cantidades de datos para numerosos tipos de aplicaciones diferentes, tales como para aplicaciones científicas, técnicas o comerciales que requieren un gran número de ciclos de procesamiento de cálculo informático. Ordenadores de usuario 32 pueden interactuar con el entorno de cálculo informático 30 a través de numerosas maneras, tal como a través de una red 34.

- 15 Se utilizan uno o más soportes de memorización de datos 36 para memorizar datos que han de procesarse por el entorno de cálculo informático 30 así como cualesquiera datos intermedios o finales generados por el sistema informático en una memoria no volátil. Sin embargo, en algunas realizaciones ejemplo, la configuración del entorno de cálculo informático 30 permite que se realicen esas operaciones de modo que los resultados de datos intermedios o finales se puedan memorizar exclusivamente en una memoria volátil (p.ej., memoria RAM) sin la necesidad de que los resultados de datos intermedios o finales se memoricen en memoria de tipo no volátil (p.ej., disco).

- 20 Lo que antecede puede ser útil en determinadas situaciones, tales como cuando el entorno de cálculo informático 30 recibe cuestiones *ad hoc* procedentes de un usuario y cuando las respuestas, que se generan mediante el procesamiento de grandes cantidades de datos, necesitan generarse sobre la marcha. En esta situación no limitadora, el entorno de cálculo informático 30 está configurado para retener la información procesada dentro de la memoria de modo que la respuesta se pueda generar por el usuario, en diferentes niveles de detalle, así como para permitir a un usuario consultar, de forma interactiva, de nuevo esta información.

- 25 La Figura 2 ilustra un entorno de cálculo informático a modo de ejemplo 30. El entorno de cálculo informático a modo de ejemplo 30 incluye un sistema de cálculo informático basado en rejilla 38 para el procesamiento de grandes cantidades de datos y un sistema de gestión de base de datos 40 para la gestión, memorización y recuperación de grandes cantidades de datos que se distribuyen para, y se memorizan, en los múltiples soportes de memorización de datos 36 que están situados en diferentes localizaciones dentro de un sistema de base de datos distribuida. Las líneas de trazo en la Figura 2 se proporcionan para indicar que los nodos de cálculo, en el sistema de cálculo informático basado en rejilla 38, y el sistema de gestión de base de datos 40 comparten el mismo hardware de software.

- 30 La Figura 3 ilustra componentes de hardware y software para un sistema de cálculo informático basado en rejilla 38. El sistema de cálculo informático basado en rejilla 38 incluye una cantidad de nodos de procesamiento de datos 42, 44, que comprenden procesadores de datos de múltiples núcleos. Uno de los nodos está diseñado como un nodo de procesamiento de datos de control 42, y una pluralidad de los nodos están diseñados como nodos de procesamiento de datos de trabajo 44. Cada nodo de procesamiento de datos 42, 44, incluye un componente de software basado en rejilla (GESC) 46 que se ejecuta sobre el procesador de datos asociado con ese nodo y se interrelaciona con una memoria intermedia 48, también asociada con ese nodo.

- 35 El software GESC 46, en los diversos nodos 42, 44, está conectado a través de una red 50 y pueden comunicarse entre sí utilizando un protocolo de comunicación predeterminado tal como una Interfaz de Paso de Mensaje (MPI). Cada GESC 46 puede participar en una comunicación punto a punto con el GESC en otro nodo, o participar en una comunicación colectiva con múltiples GESCs a través de la red 50. El GESC 46, en cada nodo en esta realización a modo de ejemplo, contiene instrucciones de software idénticas, y cada nodo de procesamiento de datos es capaz de funcionar como un nodo de control 42 o un nodo de operador 44. El GESC en el nodo de control 42, se puede comunicar con una aplicación del cliente 52 a través de una ruta de comunicación 54 para recibir cuestiones *ad hoc* procedentes de un usuario y para responder a dichas cuestiones *ad hoc* después de procesar una gran cantidad de datos.

- 40 La Figura 4 ilustra componentes de hardware y software para un sistema de gestión de base de datos distribuida 40. El sistema de gestión de base de datos distribuida 40 incluye un software de gestión de base de datos (DBMS) 60 que se ejecuta en un servidor de base de datos, en un nodo de base de datos de control 56, y en un servidor de base de datos en cada uno de una pluralidad de nodos de base de datos de operador 58. Cada nodo de base de datos 56, 58 incluye servidor de base de datos y uno o más soportes de memorización de datos 36 que están unidos al servidor de base de datos. El servidor de base de datos ejecuta el software de gestión de base de datos (DBMS) 60, que controla la creación, el mantenimiento y la utilización de una base de datos (es decir, una estructura de datos) que organiza la memorización de datos en el soporte de memorización de datos 36. El sistema DBMS, en el nodo de control 56, del sistema de gestión de base de datos distribuida 40, se comunica con una aplicación del

cliente 62 para aceptar demandas para datos y proporcionar instrucciones al sistema de gestión de base de datos para transmitir los datos adecuados. Los sistemas DBMS 60, en cada nodo de base de datos, pueden comunicarse entre sí a través de una red 64.

5 Con el sistema de gestión de base de datos distribuida, la recopilación de datos se distribuye a través de múltiples localizaciones físicas. En esta realización ejemplo, cada nodo de base de datos 56, 58, en el sistema distribuido, realiza la memorización en su soporte de memorización de datos asociado 36 de una parte de la totalidad de los datos gestionados por el sistema DBMS. Un proceso que suele utilizarse por el sistema DBMS para protegerse  
10 contra la pérdida de datos es la replicación. La replicación suele implicar proporcionar una copia de seguridad de los datos memorizados en un nodo a uno o más otros nodos. Cuando se utiliza la replicación, si falla un nodo, entonces se pueden recuperar los datos procedentes del nodo en condición de fallo desde una copia de seguridad situada en otro nodo.

15 Tal como se ilustra en la Figura 4, cada nodo de base de datos 56, 58, en el sistema secundario de base de datos distribuida, tiene distintos soportes de memorización de datos 36, tal como discos u otros dispositivos de memoria, así como distintas memorias principales. En sistemas de base de datos distribuida, en donde los datos se mantienen actualizados en todas las localizaciones, puede tener lugar un procesamiento de consulta distribuida. Lo que antecede significa que los sistemas DBMS, en cada nodo, pueden procesar una consulta para los datos.

20 La Figura 5 ilustra un sistema de cálculo informático basado en rejilla que está ubicado, de forma conjunta, con un sistema de gestión de base de datos distribuida. En este ejemplo, cada uno de varios nodos de procesador de datos 100, 102, incluye un procesador de datos que puede compartirse por un sistema de gestión de base de datos distribuida y un sistema de cálculo informático basado en rejilla. Cada nodo de procesamiento de datos incluye un componente de software basado en rejilla (GESC) 120, que tiene acceso a la memoria intermedia 122, y un software  
25 de gestión de base de datos (DBMS) 130, que está conectado a un soporte de memorización de datos 136. Cada nodo puede ejecutar, a la vez, el GESC 120 y el sistema DBMS 130. La puesta en práctica del GESC 120 en varios nodos, permite el funcionamiento de un entorno de cálculo informático basado en rejilla para realizar análisis de datos complejos. De modo similar, la puesta en práctica del sistema DBMS 130, en los nodos, permite el funcionamiento de un sistema de gestión de base de datos distribuida para la memorización de datos que han de  
30 utilizarse en el análisis de datos.

De modo similar al sistema de gestión de base de datos distribuida que se ilustra en la Figura 4, el sistema DBMS 130, en cada nodo 100, 102, se puede comunicar con DBMS en otros nodos a través de una red 138, y de modo similar al sistema de cálculo informático basado en rejilla, ilustrado en la Figura 3, el GESC 120, en varios nodos  
35 100, 102, está conectado a través de una red 124 y pueden comunicar entre sí utilizando un protocolo de comunicación predeterminado tal como la Interfaz de Paso de Mensaje (MPI). Codificada componente GESC 120 puede participaren una comunicación punto a punto con el componente GESC en otro nodo, o en una comunicación colectiva con múltiples GESCs a graves de la red 124. En el ejemplo ilustrado, el nodo de control 100 tiene la misma configuración de hardware que los nodos de operador 102.

40 En este ejemplo, el componente GESC 120, en cada nodo, contiene instrucciones de software idénticas. El GESC, sin embargo, tiene múltiples modos de puesta en práctica. Debido a los múltiples modos de puesta en práctica, cada nodo de procesamiento de datos 100, 102 es capaz de funcionar como un nodo de control 100 o como un nodo de operador 102.

45 Con cada DBMS 130, en cada nodo, se proporciona una función definida por usuario (UDF) 138. La función UDF da a conocer un mecanismo para un programa de aplicación para la transmisión de datos para, o la recepción de datos desde, la base de datos en los soportes de memorización de datos 136 que se gestionan por el sistema DBMS 130. A modo de ejemplo, en los nodos de procesamiento de datos 100, 102, una función UDF 138 se puede solicitar por  
50 el sistema DBMS para proporcionar datos al componente GESC. En el ejemplo ilustrado, la función UDF 138 establece una conexión enchufable con el GESC para la transmisión de datos. como alternativa, según se ilustra por la fecha con marcas de trazos en la Figura 5, la función UDF 138 puede transmitir datos al GESC mediante la escritura de datos en la memoria compartida accesible por, a la vez, la función UDF y el componente GESC.

55 La Figura 6 ilustra un ejemplo en donde un nodo de aplicación del cliente 150 está acoplado, de forma comunicativa, con el nodo de control 100 de un sistema de cálculo informático basado en rejilla que está ubicado, de forma conjunta, con un sistema de gestión de base de datos distribuida. El nodo de aplicación del cliente 150 incluye un componente GESC cliente 152 y una aplicación del cliente 154, que se ejecuta en un procesador de datos compartido y una memoria 156. La aplicación del cliente 154 ilustrada puede demandar que se realice un análisis de  
60 datos por el sistema de cálculo informático basado en rejilla. El componente GESC cliente ilustrado 152, interrelaciona con la aplicación de cliente 154 para recibir la demanda, procedente de la aplicación del cliente, para el análisis de datos y para proporcionar los resultados del análisis de datos a la aplicación del cliente.

65 En este ejemplo, el GESC cliente 152 contiene instrucciones de software que son idénticas a las instrucciones de software del componente GESC 120 en los nodos de control y de operador. El GESC, en cada uno de estos nodos, en este ejemplo, tiene un modo operativo adicional, un modo cliente. En el modo cliente, el GESC 152 se puede

comunicar con el GESC 120 en el nodo de control, a modo de ejemplo, a través de una conexión enchufable tal como una conexión TCP, con el fin de proporcionar instrucciones al entorno de cálculo informático basado en rejilla para realizar un análisis de datos, y para recibir los resultados del análisis de datos para su suministro a la aplicación del cliente que los demanda.

5 En otro ejemplo, el GESC cliente 152 puede funcionar en un primer modo de ejecución o un modo denominado solo, en donde el GESC cliente 152 puede realizar el análisis de datos demandado sin la utilización del entorno de cálculo informático basado en rejilla. En este ejemplo, la aplicación del cliente 154 indica al GESC cliente 152 si el análisis de datos solicitado debe realizarse, o no, utilizando el entorno de cálculo informático basado en rejilla, en un  
10 segundo modo de ejecución o en el modo denominado solo o primer modo de ejecución en el nodo de aplicación del cliente. Utilizando estos dos modos de ejecución, el sistema se hace compatible para diferentes tareas. Cuando la magnitud del problema es pequeña, el sistema se puede adaptar para ejecutarse en un nodo único para la optimización de recursos y velocidad. Otro efecto positivo de organizar la función para permitir múltiples funciones es el hecho de que un producto de software basado en rejilla puede desarrollarse, potencialmente, como un producto  
15 no basado en rejilla, con una pequeña modificación de las funciones.

Según se ilustra en la Figura 6, el componente del sistema de cálculo informático basado en rejilla que está situado en cada nodo de procesamiento 100, 102, que se ilustra como un componente de sistema de cálculo informático basado en rejilla 125, puede ponerse en práctica como una máquina virtual que funciona en el procesador de datos  
20 situado en cada nodo. Además, o de forma alternativa, el componente del sistema secundario de gestión de base de datos distribuida, que está situado en cada nodo de procesamiento 100, 102, que se ilustra como un componente de base de datos distribuida 135, se puede realizar como una máquina virtual que funciona en el procesador de datos situado en ese nodo. El funcionamiento de uno o ambos de los componentes 125, 135, como una máquina virtual, permite compartir los recursos subyacentes de una máquina física que se proporcionan por el procesador de datos  
25 situado en ese nodo al mismo tiempo que se mantiene un fuerte aislamiento entre el componente del sistema de cálculo informático basado en rejilla 125 y el componente de base de datos distribuida 135. Los efectos positivos y las ventajas técnicas de la puesta en práctica de entornos de máquina virtual es que el usuario podría no necesitar una cuenta en la misma máquina virtual en la que se ejecuta la base de datos. Lo que antecede permite entornos de múltiples ocupantes, seguros, en donde múltiples entidades pueden utilizar los mismos recursos de forma aislada y  
30 acelerar, además, el proceso de cálculo y, asimismo, disminuir la cantidad de datos que han de transmitirse. Como alternativa, el componente de sistema de cálculo informático basado en rejilla 125 y el componente de base de datos distribuida 135 podrían funcionar como una máquina virtual común proporcionando aislamiento para dichos componentes 125, 135 respecto a otros procesos que puede compartir los recursos subyacentes de máquina física. Por lo tanto, un sistema de cálculo informático basado en rejilla que está ubicado, de forma conjunta, con un sistema  
35 de gestión de base de datos distribuida, podría funcionar bajo cualquiera de las configuraciones siguientes: (i) una configuración en donde no se utilizan máquinas virtual con respecto al componente de sistema de cálculo informático basado en rejilla 125 o componente de base de datos distribuida 135; (ii) una configuración en donde se utiliza una máquina virtual para un componente de sistema de cálculo informático basado en rejilla 125 pero no para un componente de base de datos distribuida 135; (iii) una configuración en donde se utiliza una máquina virtual para  
40 un componente de base de datos distribuida 135, pero no para un componente de sistema de cálculo informático basado en rejilla 125; (iv) una configuración en donde máquinas virtual separadas se utilizan para cada uno de entre un componente de sistema de cálculo informático basado en rejilla 125 y un componente de base de datos distribuida 135; y (v) una configuración en donde se utiliza una máquina virtual común para, a la vez, un componente de sistema de cálculo informático basado en rejilla 125 y un componente de base de datos distribuida 135.

45 La Figura 7 ilustra un proceso, a modo de ejemplo, para realizar un análisis de datos. El componente GESC, en el nodo de cliente, se comunica con una aplicación del cliente en el nodo de cliente, para recibir cuestiones *ad hoc* procedentes de un usuario y para responder a dichas consultadas *ad hoc* después de que se hayan procesado grandes cantidades de datos. Tal como se ilustra en la etapa 200, la aplicación del cliente presenta una consulta *ad hoc* al componente GESC cliente que demanda que se realice un análisis de datos. La consulta *ad hoc* puede  
50 incluir instrucciones sobre el tipo de análisis de datos que ha de realizarse y si el análisis debe realizarse en un segundo modo de ejecución utilizando el entorno de cálculo informático basado en rejilla. En la etapa 210, el GESC cliente inicia el análisis de datos demandado y obtiene los resultados del análisis de datos. En la etapa 220, el GESC cliente proporciona los resultados del análisis a la aplicación del cliente.

55 En el ejemplo ilustrado, la iniciación del análisis de datos demandado y la obtención de los resultados (etapa 210) requiere la puesta en práctica de varias etapas. Según se ilustra, en la etapa 212, el GESC cliente determina si la consulta *ad hoc* requiere la utilización del entorno de cálculo informático basado en rejilla para la puesta en práctica del análisis de datos. Si la solución a esa determinación es negativa, entonces el GESC cliente, en la etapa 214, pone en práctica el análisis de datos en el nodo de cliente. Si la solución a la determinación realizada en la etapa  
60 212 es positiva, entonces el GESC cliente, en la etapa 216, hace que el componente GESC sea instanciado en el entorno de cálculo informático basado en rejilla y hace que el sistema de cálculo informático basado en rejilla realice el análisis de datos. Esta determinación puede implicar, además, la determinación de la configuración del entorno de cálculo informático basado en rejilla. A modo de ejemplo, esta determinación puede incluir el número de nodos de operador y el número de conexiones por nodo de operador que han de utilizarse para realizar el análisis de datos. En la etapa 218, el componente GESC, en el entorno de cálculo informático basado en rejilla, proporciona los

resultados del análisis al GESC cliente.

La transmisión de los resultados desde el entorno de cálculo informático basado en rejilla al nodo de cliente puede demandar el desplazamiento de datos debido a diferencias entre el sistema operativo, en el entorno de cálculo informático basado en rejilla, y el sistema operativo que se ejecuta en el nodo de cliente. A modo de ejemplo, el procesador de datos que ejecuta el GESC de nodo de control puede tener un sistema operativo Linux de 64 bits, y el procesador de datos que ejecuta el GESC cliente puede tener un sistema operativo de 32 bits, tal como un sistema operativo PC Windows de 32 bits. La representación de caracteres amplios es diferente en los dos sistemas operativos y los caracteres en el formato Linux deben traducirse a la representación adecuada en el cliente Windows.

En el ejemplo ilustrado en la Figura 8, se dan a conocer elementos, a modo de ejemplo, para ilustrar componentes de una consulta *ad hoc* desde una aplicación del cliente al GESC cliente. Según se ilustra en 302, la aplicación del cliente proporciona instrucciones sobre el tipo de análisis de datos a realizar. Según se ilustra en 304, la aplicación del cliente identifica la localización origen de los datos que han de utilizarse en el análisis de datos. Los datos que han de utilizarse en el análisis de datos, en este ejemplo, se memorizan en el sistema de base de datos distribuida que está ubicado, conjuntamente, con el entorno de cálculo informático basado en rejilla. Por último, en 306, la aplicación del cliente proporciona instrucciones sobre si ha de utilizarse el entorno de cálculo informático basado en rejilla o el entorno denominado solo para realizar el análisis de datos demandado.

En el ejemplo ilustrado en la Figura 9, se proporcionan etapas a modo de ejemplo para la puesta en práctica de GESC en el entorno de cálculo informático basado en rejilla para realizar el análisis de datos demandado. En la etapa 310, el GESC de nodo de cliente hace que el GESC sea instanciado en el nodo de control y en una pluralidad de nodos de operador. En la etapa 312, el GESC de nodo de control hace que los datos que han de utilizarse en el análisis, pasen al GESC en los nodos de operador. En la etapa 314, el GESC, en el nodo de control, inicia el análisis de datos en el entorno de cálculo informático basado en rejilla. Por último, en la etapa 316, el GESC en el nodo de control reúne los resultados procedentes del análisis realizado por el GESC en los diversos nodos de operador.

En el ejemplo ilustrado en la Figura 10, se dan a conocer etapas, a modo de ejemplo, para ilustrar un método para su uso por el GESC de nodo de control para proporcionar los resultados del análisis de datos para el GESC de nodo de cliente. En la etapa 320, el GESC de nodo de control abre una conexión enchufable TCP. En la etapa 322, el GESC de nodo de control serializa los resultados del análisis de datos. En la etapa 324, el nodo de control proporciona los resultados puestos en serie del análisis de datos al GESC de nodo de cliente a través de la conexión enchufable TCP.

En el ejemplo ilustrado en la Figura 11, se dan a conocer etapas, a modo de ejemplo, para su uso por el GESC de nodo de cliente para proporcionar los resultados del análisis de datos a la aplicación del cliente que los demanda. Al utilizar el nodo de control para reunir los resultados, se mejora la presentación de los resultados del análisis al usuario/cliente. Es el nodo al que se conecta el cliente, y el flujo del análisis en el entorno de cálculo informático distribuido se controla por dicho nodo. Si los resultados llegasen desde otros nodos al cliente, entonces, el entorno podría necesitar proporcionar conexiones entre el cliente y todos los nodos a los que podría comunicarse de forma potencial. Por lo tanto, el hecho de utilizar el nodo para reunir los datos disminuye, además, la cantidad de datos que han de transmitirse. Además, si la rejilla está detrás de un 'cortafuegos', en tal caso, la gestión de la comunicación entre la rejilla y el cliente conduce a un aumento en la eficiencia operativa y en la seguridad. En la etapa 330, el GESC de nodo de cliente traduce los resultados del análisis de datos recibido desde el GESC del nodo de control, en un formato que se pueda reconocer por el sistema operativo de la aplicación del cliente. En la etapa 332, el GESC del nodo de cliente memoriza los resultados del análisis de datos traducidos en la memoria local compartida entre el GESC del nodo de cliente y la aplicación del cliente. Por último, en la etapa 334, el GESC de nodo de cliente proporciona, a la aplicación del cliente, la dirección, en la memoria compartida, de los resultados del análisis de datos traducidos.

En el ejemplo ilustrado en la Figura 12, se dan a conocer etapas, a modo de ejemplo, para ilustrar un método para la instanciación del GESC en el entorno de cálculo informático basado en rejilla. En la etapa 400, el GESC, en el nodo de control, es objeto de instanciación. En la etapa 402, sobre la base del número de nodos de operador potenciales disponibles y el número de nodos de operador deseados para realizar el análisis de datos, el GESC del nodo de control hace que el GESC, en el nodo de operador, sea instanciado. En la etapa 404, el GESC en los nodos de control y de operador intentan abrir un puerto común para la comunicación de conexión enchufable. Utilizando conexiones enchufables, concretamente, utilizando una dirección de puerto común, en comparación con otras técnicas de comunicación, se consiguen ventajas respectivas. Permite proporcionar instrucciones a la función UDF, en todos los nodos, para enviar sus datos a otro proceso en la misma máquina. Concretamente, utilizando un número de puerto común, es posible agrupar las transmisiones para un análisis particular en un sistema concurrente en donde múltiples usuarios están operativos o en donde una aplicación de usuario único desea realizar operaciones de lectura o escritura simultánea sobre múltiples fuentes de datos – un número de puerto común se asocia entonces con una fuente de datos particular. Por último, en la etapa 406, la dirección de puerto común, desde los nodos de operador se comunica al GESC del nodo de control, que abre, además, un puerto en la misma dirección para una comunicación de conexión enchufable.

En la Figura 13 se ilustran etapas, a modo de ejemplo, de un método para hacer que el GESC del nodo de control haga que los datos que han de utilizarse en el análisis, se proporcionen a los GESC del nodo de operador. En la etapa 410, el GESC del nodo de control emite una consulta de SQL al DBMS en el nodo de control y proporciona el número de puerto común de los puertos de conexión enchufable del nodo de operador. En la etapa 412, el sistema DBMS del nodo de control aplica una función definida por usuario (UDF) en todos los nodos y pasa el número de puerto a la función UDF en todos los nodos. En la etapa 414, cada función UDF de nodo de operador recupera los datos que han de utilizarse en el análisis de datos y pasa los datos al componente GESC en su nodo local a través de una conexión enchufable en la dirección de puerto proporcionada. En este ejemplo, cada GESC de nodo de operador recibe una parte del conjunto completo de datos que han de utilizarse en el análisis de datos. Como alternativa, cada uno o alguno de los GESC del nodo de operador podrían recibir el conjunto completo de datos que han de utilizarse para el análisis. En la etapa 416, los GESC del nodo de operador memoriza los datos recibidos en su propia memoria intermedia local. La ventaja técnica de la memorización en la memoria intermedia local en contraste con la memorización en otras ubicaciones es concretamente que disminuye la necesidad de anchos de banda de transmisión de datos y se puede aumentar la velocidad de procesamiento. Numerosos algoritmos analíticos avanzados requieren múltiples pasos a través de los datos. La memorización de datos en la memoria intermedia local permite un procesamiento más rápido para pasos posteriores después de la recuperación inicial de los datos. Permite, además, la memorización de formas procesadas de los datos en bruto, a modo de ejemplo, eliminando observaciones que no se utilizarán en el análisis debido a campos nulos u otros defectos. Los componentes analíticos especialmente diferentes, que suele utilizarse en un único análisis, pueden compartir punteros para los datos en lugar de recuperarlos múltiples veces desde el disco. Como una alternativa a las etapas 414 y 416, cada función UDF podría recuperar los datos que han de utilizarse en el análisis de datos y pasar los datos al GESC en su nodo local mediante la memorización de los datos en una memoria común accesible, a la vez, por la función UDF y el componente GESC. El GESC podría entonces procesar los datos procedentes de la memoria compartida o desplazar los datos a la memoria dedicada para su procesamiento.

En la Figura 14 se ilustran etapas, a modo de ejemplo, que pueden ponerse en práctica cuando se inicia el análisis de datos en el entorno de cálculo informático basado en rejilla. En la etapa 420, el GESC de conocimiento de control se comunica con los GESC de nodo de operador a través de una conexión de red utilizando la Interfaz de Paso de Paquete (MPI). En la etapa 422, el GESC del nodo de control asigna una parte del análisis a los GESC del nodo de operador. En este ejemplo, todos los GESC del nodo de operador realizan los mismos cálculos analíticos, pero sobre diferentes partes del conjunto completo de los datos que se han de utilizar en el análisis. Como alternativa, el GESC del nodo de control podría dar instrucciones a una pluralidad de GESC de nodo de operador para realizar diferentes etapas analíticas.

La Figura 15 ilustra etapas, a modo de ejemplo, para reunir los resultados del análisis de datos en el nodo de control. En la etapa 430, los GESC del nodo de operador envía los resultados de sus cálculos analíticos al GESC del nodo de control. Una forma en la que se pueden transmitir los resultados generados por los GESC del nodo de operador, al GESC del nodo de control, es a través de MPI. Otra forma en la que podrían transmitirse los resultados es mediante GESC del nodo de operador que memorizan sus propios resultados del análisis de datos en la base de datos distribuida y proporcionan la localización de los resultados en la base de datos distribuida al GESC del nodo de control a través de la interfaz MPI. Mediante la memorización de los resultados en la base de datos distribuida, se aumenta, además, la velocidad de procesamiento. Dependiendo del tamaño del conjunto de resultados, el usuario podría desear llevarlos, de nuevo, a la máquina cliente, o memorizarlos en la base de datos. Si los conjuntos de resultados son pequeños, a modo de ejemplo, cuando consisten en una serie de pequeñas tablas que resumen los resultados analíticos, el envío de los resultados a través de la red es un modo conveniente para presentar los resultados a un motor de representación del lado de cliente. Si los resultados son grandes, o si el conjunto de resultados tiene una forma similar a los datos de entrada distribuidos, entonces es posible evitar una transmisión sobre la red a través de un nodo único (el nodo de control). La situación en la que los conjuntos de resultados tienen la forma de los datos de entrada se produce cuando el proceso analítico puntúa los datos de entrada, a modo de ejemplo, para calcular medidas estadísticas para cada observación procesada. La colocación del conjunto de resultados en la base de datos y el mantenimiento de las mismas características de distribución que en los datos de entrada, proporciona uniones de SQL convenientes de datos de entrada y de salida, aumentando la velocidad general de procesamiento. En la etapa 432, el GESC del nodo de control agrega y reúne los resultados procedentes de los cálculos analíticos puestos en práctica por los GESC del nodo de operador. Los resultados podrían memorizarse en la memoria local o en la base de datos distribuida. El GESC del nodo de control puede realizar, además, en la etapa 432, operaciones matemáticas adicionales (p.ej., cálculos estadísticos descriptivos) sobre los datos agregados antes del agrupamiento final de los resultados.

En caso de fallo operativo de un nodo mientras se realizan los cálculos, el GESC del nodo de control puede reiniciar el análisis de datos en el entorno de cálculo informático basado en rejilla sin tener que redistribuir los datos que han de utilizarse en el análisis de datos. Puesto que varios GESC de nodos de operador están ubicados, de forma conjunta, en nodos con DBMS, los mecanismos de protección del sistema de gestión de base de datos distribuida pueden proteger contra pérdida de datos durante un fallo de un nodo. El sistema DBMS local tiene acceso a una copia replicada de los datos que han de utilizarse en el análisis de datos y proporciona los datos a los diversos GESC del nodo de operador. Cuando es necesario volver a realizar los cálculos debido a un fallo en un nodo, el

GESC del nodo de control puede reasignar la parte del análisis, asignada previamente al nodo objeto de fallo para el GESC en uno o más de los nodos de operador estantes (o un nuevo nodo de operador). Lo que antecede puede implicar proporcionar instrucciones a uno o más GESCs de nodo de operador para realizar sus cálculos analíticos sobre datos adicionales.

5 La Figura 16 ilustra etapas, a modo de ejemplo, para la recuperación desde un nodo objeto de fallo. El GESC del nodo de cliente detecta un fallo en el entorno de cálculo informático basado en rejilla (etapa 500). En la etapa 502, el GESC del nodo de cliente proporciona instrucciones al GESC del nodo de control para reiniciar el análisis de datos. En la etapa 504, el GESC del nodo de control emite una nueva consulta de SQL para el sistema DBMS del nodo de control. Si existe un nodo en condición de fallo, el sistema DBMS hace que los datos que están situados en el nodo objeto de fallo se sustituyan por su copia replicada que están situadas en uno o más nodos sin condición de fallo. Los datos proporcionados con anterioridad al GESC, en el nodo en condición de fallo, se proporcionan por una función UDF de trabajo al GESC en un nodo distinto (etapa 506). Lo anterior puede dar lugar a que el GESC, en uno de los nodos reciba los datos inicialmente proporcionados junto con los datos originalmente proporcionados al GESC en el nodo en condición de fallo. Como alternativa, lo que antecede puede dar lugar a que al GESC, en un nuevo nodo, se le proporcionen los datos proporcionados originalmente al GESC en el nodo en condición de fallo. En la etapa 508, el GESC en el nodo de control reinicia el análisis de datos. Lo anterior implica, en la etapa 510, que el GESC del nodo de control reinicia el análisis de datos sin el nodo en condición de fallo. En la etapa 514, los GESCs del nodo de operador envían los resultados procedentes de sus cálculos analíticos al GESC del nodo de control. En la etapa 516, el GESC del nodo de control agrega y reúne los resultados procedentes del análisis de datos. En la etapa 518, el GESC del nodo de control transmite los resultados del análisis de datos al GESC del nodo de cliente.

En la Figura 2 se ilustra un entorno de múltiples usuarios que implica al entorno de cálculo informático basado en rejilla. En dicho entorno, cada usuario 32 recibirá su propio componente GESC cliente para manipular sus consultas y para la comunicación con un GESC de nodo de control seleccionado.

Esta descripción escrita utiliza ejemplos para dar a conocer la invención, incluyendo el mejor modo de ejecución, y permite, además, a un experto en esta técnica, realizar y utilizar la invención.

Además, los métodos y sistemas aquí dados a conocer pueden ponerse en práctica en numerosos tipos diferentes de dispositivos de procesamiento mediante código de programa que incluye instrucciones de programa, que son ejecutables por el subsistema de procesamiento de dispositivo. Las instrucciones del programa de software pueden incluir un código fuente, un código objeto, un código de máquina o cualquier otro dato memorizado que sea utilizable para hacer que un sistema de procesamiento realice los métodos y operaciones aquí dadas a conocer.

Los datos de sistemas y métodos (p.ej., asociaciones, mapeados, entrada de datos, salida de datos, resultados de datos intermedios, resultados de datos finales, etc.) se pueden memorizar y realizar en uno o más tipos diferentes de soportes de memorización de datos puestos en práctica por ordenador, tal como tipos distintos de dispositivos de memorización y sistemas de programación (p.ej., memoria RAM, memoria ROM, memoria instantánea, archivos planos, bases de datos, estructuras de datos de programación, variables de programación, expresiones de programación de tipo IF-THEN (o tipo similar), etc.). Conviene señalar que las estructuras de datos describen formatos para uso en la organización y memorización de datos en bases de datos, programas, memorias u otros soportes legibles por ordenador para uso por un programa informático.

Los componentes informáticos, módulos de software, funciones, soportes de memorización de datos y estructuras de datos, aquí dados a conocer pueden conectarse de forma directa o indirecta, entre sí, con el fin de permitir el flujo de datos necesario para sus operaciones. Conviene señalar que un módulo o procesador incluye, pero no está limitado a, una unidad de código que realiza una operación de software, y puede ponerse en práctica, a modo de ejemplo, como una unidad de código de sub-rutina, o como una unidad de código de función de software, o como un objeto (tal como en un paradigma orientado al objeto), o como una aplicación informática applet o en un lenguaje de texto informático, o como otro tipo de código informático. Los componentes de software y/o la funcionalidad pueden situarse en un ordenador único o distribuirse a través de múltiples ordenadores, dependiendo de la situación en cuestión.

**REIVINDICACIONES**

1. Un entorno de cálculo informático (30), que comprende:
- 5 un sistema de gestión de base de datos distribuida (40) y  
un sistema de cálculo informático basado en rejilla (38) ambos de conformidad con la reivindicación 8.
- 10 **2.** El entorno según una de las reivindicaciones precedentes, en donde los componentes de software basados en rejilla, en la pluralidad de nodos de operador, están configurados para memorizar los datos recibidos y para realizar el análisis de datos utilizando su propia memoria intermedia local y/o en donde la pluralidad de nodos de operador están configurados para transmitir los resultados procedentes del análisis de datos a su DBMS local para su memorización en la base de datos distribuida.
- 15 **3.** El entorno según una de las reivindicaciones precedentes, en donde el nodo de control (100) está configurado, además, para:
- la determinación de un número de nodos de operador disponibles y un número de nodos de operador deseados para realizar el análisis de datos, y
- 20 sobre la base de los números determinados, proporcionar instrucciones a los componentes de software basados en rejilla (120), en la pluralidad de nodos de operador (102) para realizar el cálculo analítico.
- 25 **4.** El entorno según una de las reivindicaciones precedes, en donde los resultados reunidos por el componente de software basado en rejilla del nodo de control, están memorizados en el sistema de base de datos distribuida, y/o en donde los resultados reunidos por el componente de software basado en rejilla del nodo de control se transmiten a un componente de software basado en rejilla en un nodo de cliente.
- 30 **5.** El entorno según una de las reivindicaciones precedentes, en donde a la detección de un fallo operativo en uno de los nodos de operador:
- el sistema DBMS, en el nodo de control, está configurado para hacer que el sistema DBMS, en uno de los nodos de operador restantes, redistribuya la parte de los datos previamente distribuidos al componente de software basado en rejilla en el nodo en condición de fallo, al componente de software basado en rejilla local para ese nodo;
- 35 el componente de software basado en rejilla, que recibe los datos redistribuidos, está configurado para realizar un cálculo analítico sobre los datos redistribuidos y para proporcionar los resultados al componente de software basado en rejilla en el nodo de control; y
- 40 el componente de software basado en rejilla, en el nodo de control, está configurado para reunir los resultados recibidos procedentes de los componentes de software basados en rejilla en los nodos de operador restantes.
- 45 **6.** El entorno según una de las reivindicaciones precedentes, que comprende, además, un componente de software basado en rejilla en un nodo de cliente:
- estando el componente de software basado en rejilla, en el nodo de cliente, configurado con un primer modo de ejecución y un segundo modo de ejecución;
- 50 el componente de software basado en rejilla del nodo de cliente, en el primer modo de ejecución, está configurado para realizar un análisis de datos local con el fin de proporcionar los resultados al software del cliente;
- estando el componente de software basado en rejilla del nodo de cliente, en el segundo modo de ejecución, configurado para solicitar al componente de software basado en rejilla, en el nodo de control, para realizar el análisis de datos en un entorno de cálculo informático basado en rejilla; y
- 55 estando el componente de software basado en rejilla, en el nodo de cliente, configurado para funcionar en el primer modo o en el segundo modo, sobre la base de una indicación proporcionada por el software del cliente, en donde, especialmente, el componente de software basado en rejilla del nodo de cliente está configurado para recibir cuestiones *ad hoc* procedentes de una aplicación del cliente y para proporcionar respuestas a las cuestiones *ad hoc* proporcionando los resultados reunidos por el componente de software basado en rejilla del nodo de control, a la aplicación del cliente.
- 60 **7.** El entorno según una de las reivindicaciones precedentes que comprende, además, un componente de software basado en rejilla de nodo de cliente que está configurado para recibir cuestiones *ad hoc* procedentes de una aplicación del cliente y para proporcionar respuestas a las cuestiones *ad hoc* proporcionando los resultados reunidos por el componente de software basado en rejilla del nodo de control, a la aplicación del cliente, y/o en
- 65

donde el componente de software basado en rejilla, en el nodo de control, está configurado para reunir los resultados mediante agregación y concatenación de los resultados del análisis realizado en los procesadores de datos del nodo de operador, en donde el componente de software basado en rejilla, en una pluralidad de nodos de cálculo informático basado en rejilla, se pone en práctica en un entorno de máquina virtual en cada nodo, y/o en donde el software de gestión de base de datos, en una pluralidad de nodos de cálculo informático basado en rejilla, se pone en práctica en un entorno de máquina virtual en cada nodo, en donde el componente de software basado en rejilla, en la pluralidad de nodos de cálculo informático basados en rejilla, se pone en práctica en un entorno de máquina virtual separado en cada nodo, o en donde, preferentemente, el componente de software basado en rejilla, en la pluralidad de nodos de cálculo informático basado en rejilla, se pone en práctica en un entorno de máquina virtual común con el software de gestión de base de datos en cada nodo.

**8.** Un método realizado en un entorno de cálculo informático (30) que incluye un sistema de gestión de base de datos distribuida (40) y un sistema de cálculo informático basado en rejilla (38) para realizar cálculos analíticos en datos memorizados en el sistema de gestión de base de datos distribuida (40), comprendiendo dicho método:

proporcionar una pluralidad de nodos de cálculo informático basado en rejilla (100, 102) incluyendo un nodo de control (100) y una pluralidad de nodos de operador (102), comprendiendo cada nodo (100, 102):

un procesador para ejecutar software de gestión de base de datos, DBMS (130) y para ejecutar un componente de software basado en rejilla (120), una memoria intermedia local (122) accesible por el componente de software basado en rejilla, y

un soporte de memorización local (136) accesible para que el sistema DBMS memorice una copia duplicada de datos procedente del sistema de gestión de base de datos distribuida (40), en donde el componente de software basado en rejilla (120) está configurado para realizar los cálculos analíticos sobre una parte de los datos memorizados;

en donde el procesador de cada uno de los nodos (100, 102) se comparte por el sistema de cálculo informático basado en rejilla (38) y el sistema de gestión de base de datos distribuida (40); y

solicitar al DBMS en el nodo de control (100) que haga que el sistema DBMS, en una pluralidad de nodos de operador (102), ponga disponible una parte de los datos para el componente de software basado en rejilla local para su nodo (102);

proporcionar instrucciones a los componentes de software basados en rejilla, en la pluralidad de nodos de operador (102), para realizar un cálculo analítico sobre los datos recibidos y para enviar los resultados del análisis de datos al componente de software basado en rejilla en el nodo de control (100); y

reunir los resultados del análisis de datos, realizado por los componentes de software habilitados por rejilla, en la pluralidad de nodos de operador (102);

en donde el sistema DBMS, en una pluralidad de nodos de operador, está configurado para hacer disponibles datos para su componente de software basado en rejilla local, guardando los datos en una memoria compartida, localmente accesible, a la vez, por el sistema DBMS y el componente de software basado en rejilla, y para la comunicación a través de una conexión enchufable, en donde los componentes de software basados en rejilla, en la pluralidad de nodos de operador, están configurados para abrir conexiones enchufables utilizando una dirección de puerto común, y el componente de software basado en rejilla, en el nodo de control, está configurado para proporcionar la dirección de puerto común al DBMS local para su nodo, y el sistema DBMS, en el nodo, de control está configurado para proporcionar la dirección de puerto común al DBMS, en la pluralidad de nodos de operador.

**9.** El método según la reivindicación 8, en donde los componentes de software basados en rejilla, en la pluralidad de nodos de operador, están configurados para memorizar los datos recibidos y para realizar el análisis de datos utilizando su propia memoria intermedia local, en donde la pluralidad de nodos de operador están configurados para transmitir los resultados procedentes del análisis de datos a su DBMS local para su memorización en la base de datos distribuida, en donde los resultados del análisis, recibidos desde la pluralidad de componentes de software basados en rejilla del nodo de operador, se reciben a través de una red, en donde, concretamente, una pluralidad de componentes de software basados en rejilla del nodo de operador transmiten sus resultados utilizando la interfaz MPI a través de la red, y/o en donde el método comprende, además, la memorización de los resultados reunidos por el componente de software basado en rejilla del nodo de control, en el sistema de base de datos distribuida y/o la transmisión de los resultados reunidos por el componente de software basado en rejilla del nodo de control a un componente de software basado en rejilla en un nodo de cliente.

**10.** El método según una de las reivindicaciones 8 o 9, en donde a la detección de un fallo operativo en uno de los nodos de operador:

el sistema DBMS, en el nodo de control, está configurado para hacer que el sistema DBMS, en uno de los nodos de

operador restantes, redistribuya la parte de los datos previamente distribuidos al componente de software basado en rejilla en el nodo en condición de fallo, al componente de software basado en rejilla local para ese nodo;

5 el componente de software basado en rejilla que recibe los datos redistribuidos está configurado para realizar un cálculo analítico sobre los datos redistribuidos y para proporcionar los resultados al componente de software basado en rejilla en el nodo de control; y

10 el componente de software basado en rejilla, en el nodo de control, está configurado para reunir los resultados recibidos procedentes de los componentes de software basados en rejilla, en los nodos de operador restantes.

11. El método según una de las reivindicaciones 8 a 10 que comprende, además, la etapa de proporcionar un componente de software basado en rejilla en un nodo de cliente, en donde:

15 el componente de software basado en rejilla en el modo cliente está configurado con un primer modo de ejecución y un segundo modo de ejecución;

20 estando el componente de software basado en rejilla del nodo de cliente, en el primer modo de ejecución, configurado para realizar un análisis de datos local para proporcionar los resultados al software del cliente; estando el componente de software basado en rejilla del nodo de cliente, en el segundo modo de ejecución, configurado para solicitar al componente de software basado en rejilla en el nodo de control para realizar el análisis de datos en un entorno de cálculo informático basado en rejilla; y

25 estando configurado el componente de software basado en rejilla, en el nodo de cliente, para funcionar en el primer modo o en el segundo modo sobre la base de una indicación proporcionada por el software del cliente, en donde, preferentemente, el componente de software basado en rejilla del nodo de cliente está configurado para recibir cuestiones *ad hoc* procedentes de una aplicación de cliente y para proporcionar respuestas a las cuestiones *ad hoc* proporcionando los resultados reunidos por el componente de software basado en rejilla del nodo de control a la aplicación del cliente.

30 12. El método según una de las reivindicaciones 8 a 11 que comprende, además, proporcionar un componente de software basado en rejilla de nodo de cliente que está configurado para recibir cuestiones *ad hoc* procedentes de una aplicación del cliente y para proporcionar respuestas a las cuestiones *ad hoc* proporcionando los resultados reunidos por el componente de software basado en rejilla del nodo de control a la aplicación del cliente y/o en donde la etapa de reunión comprende una agregación y concatenación de los resultados del análisis realizado en los procesadores de datos del nodo de operador.

35

40

45

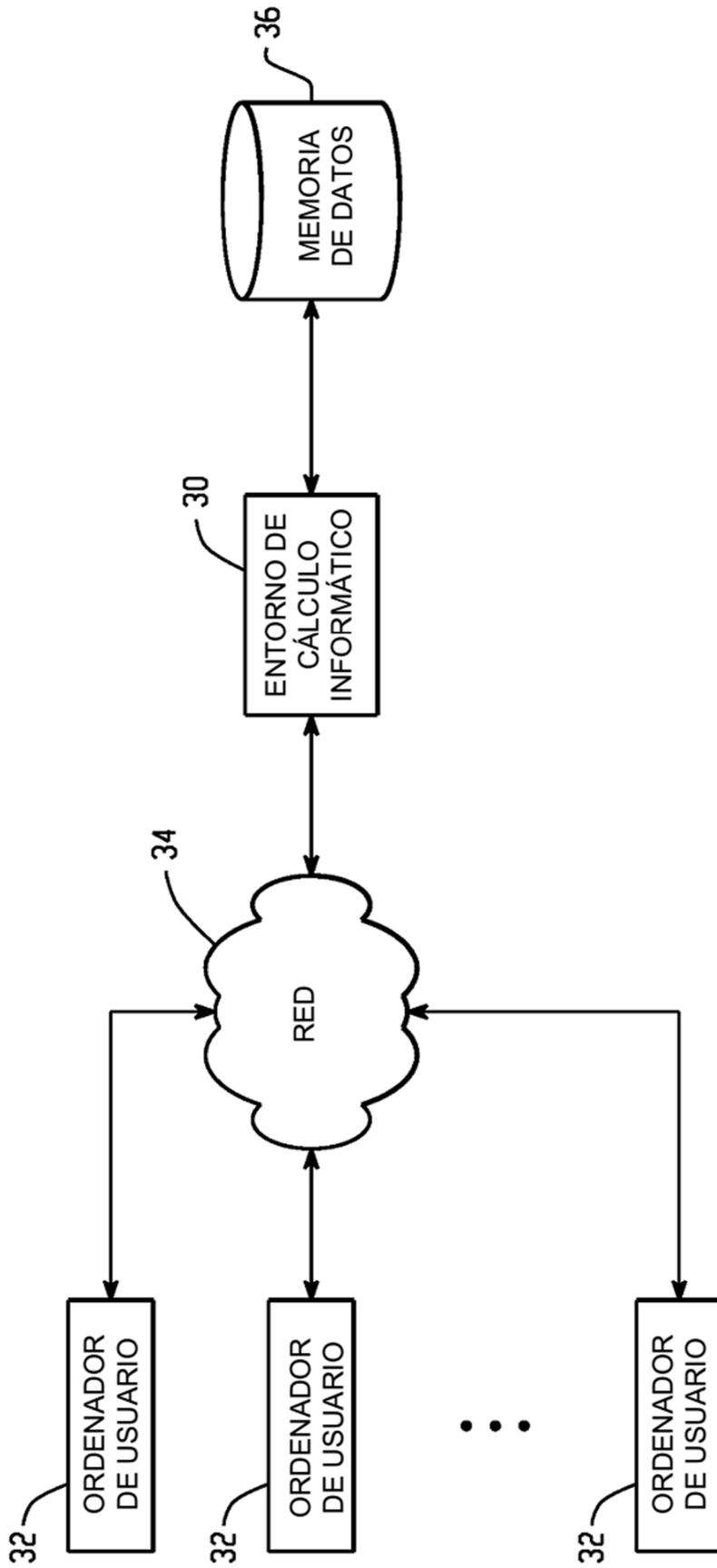


Fig. 1

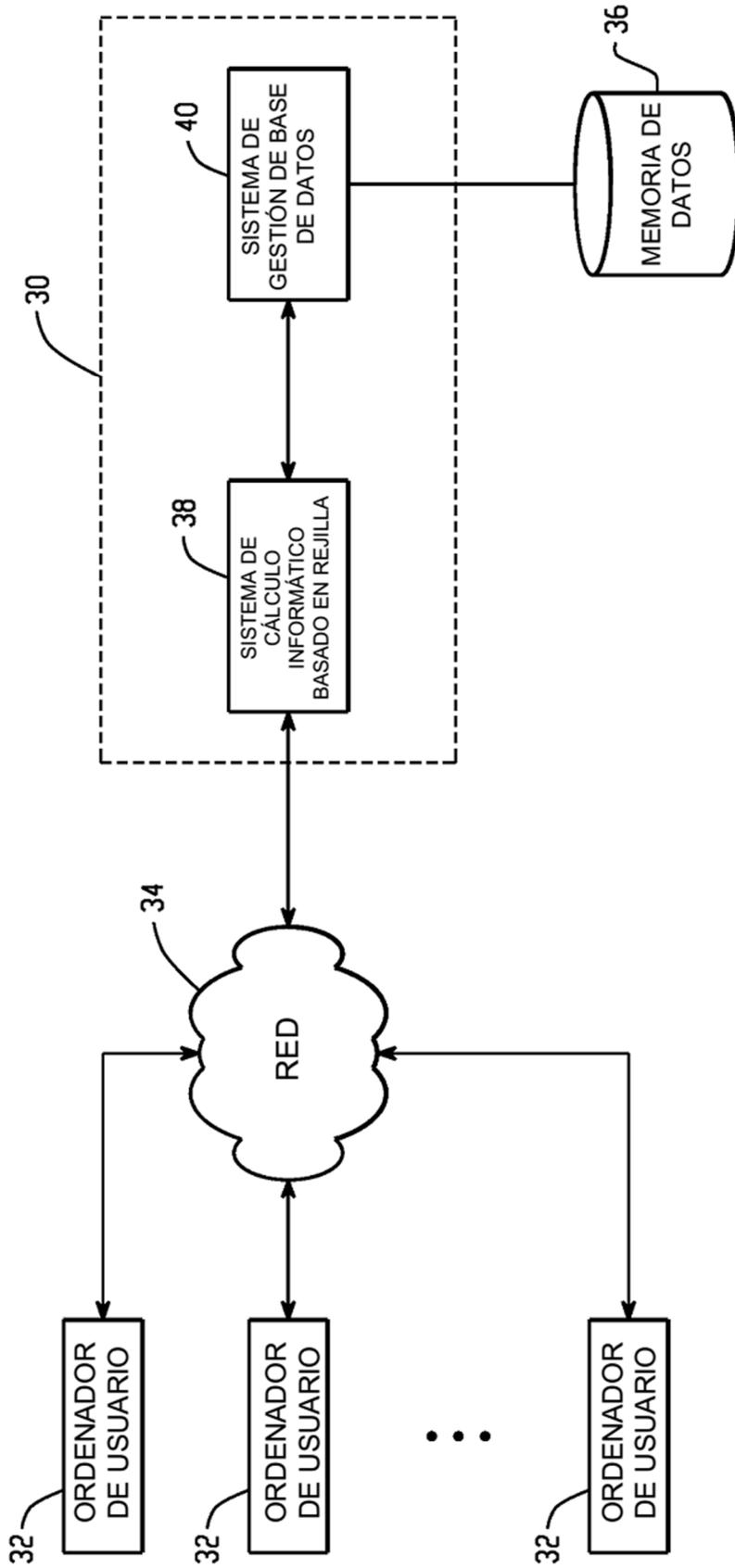


Fig. 2

SISTEMA DE CÁLCULO INFORMÁTICO BASADO EN REJILLA

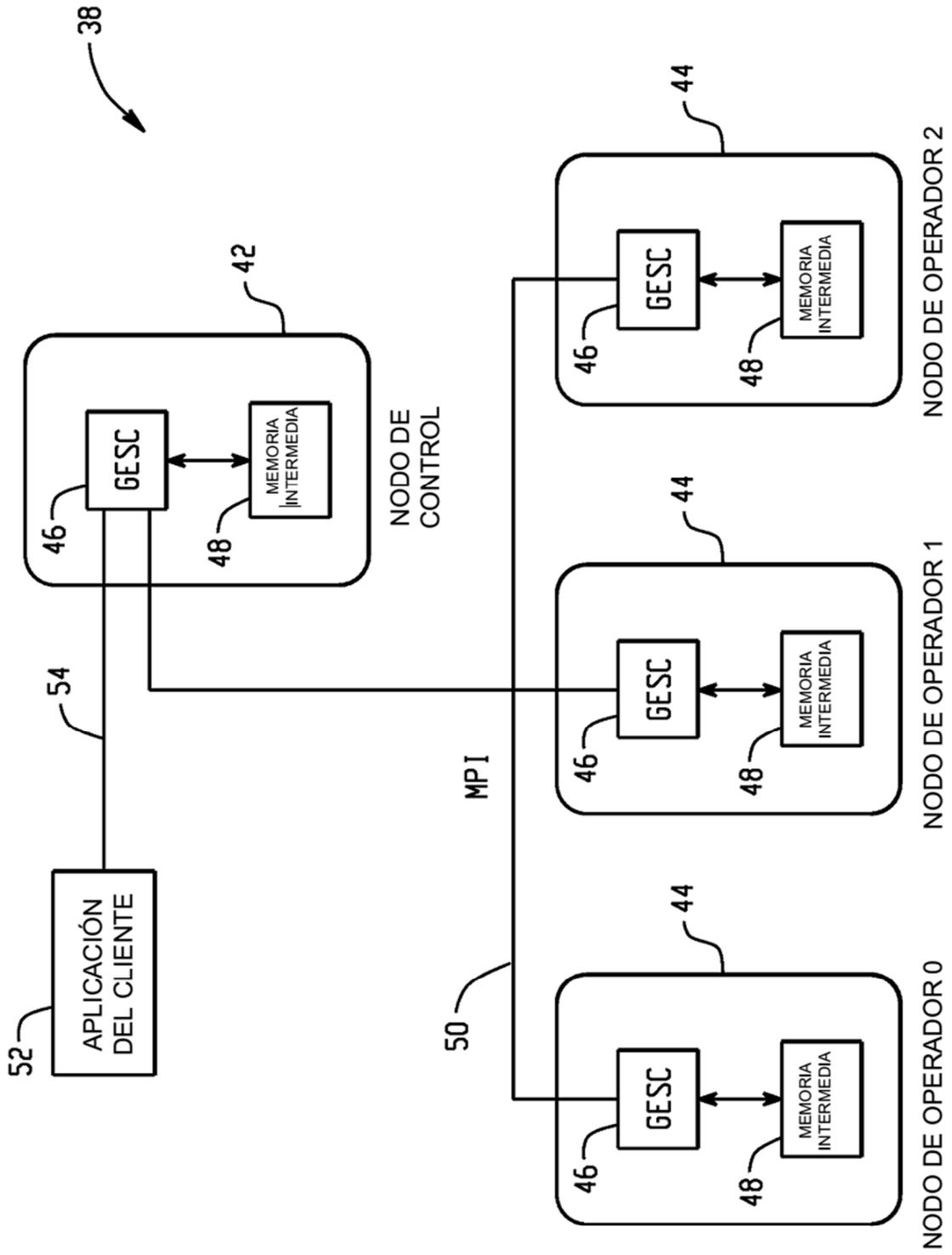


Fig. 3

SISTEMA DE GESTIÓN DE BASE DE DATOS DISTRIBUIDA

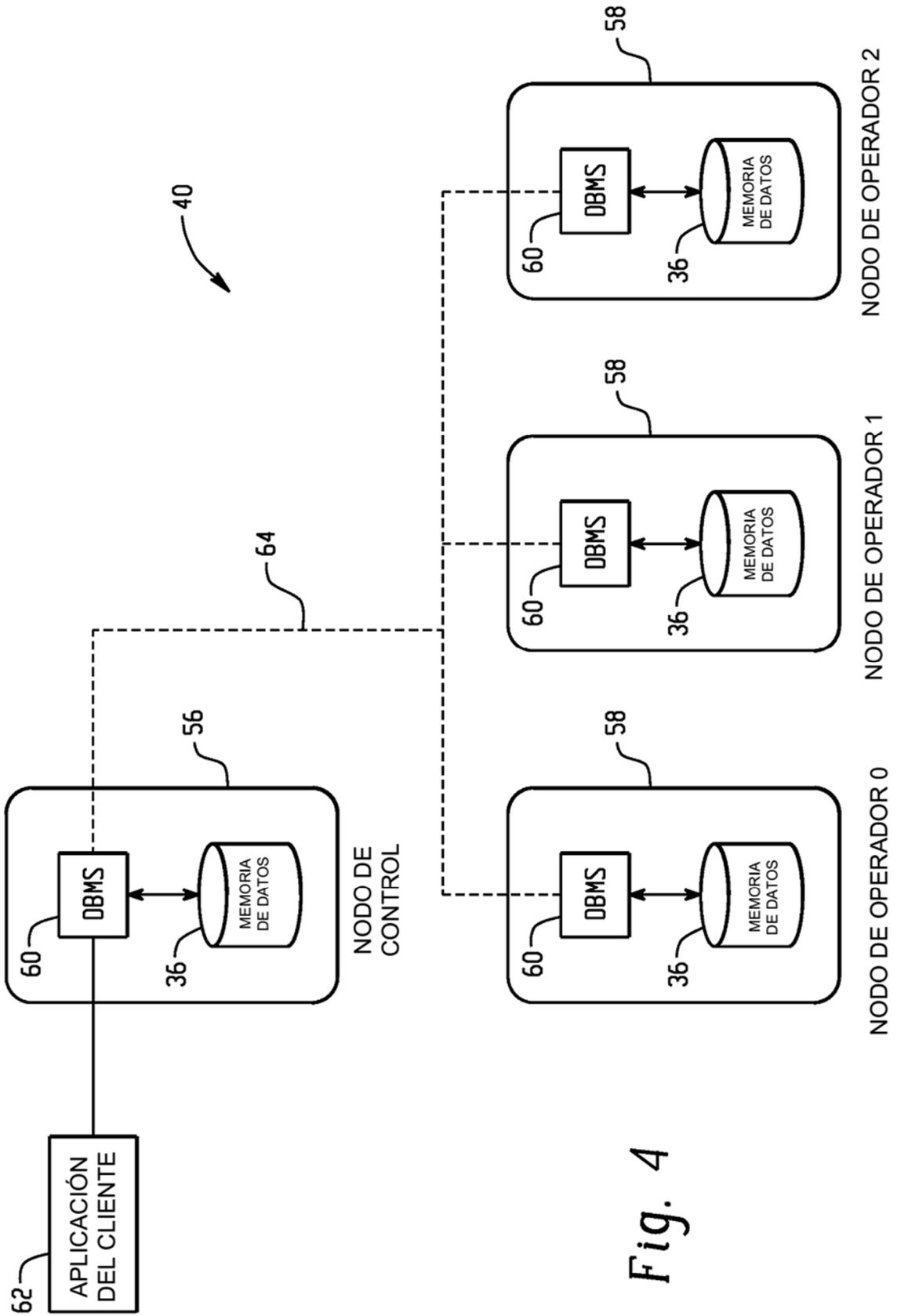


Fig. 4

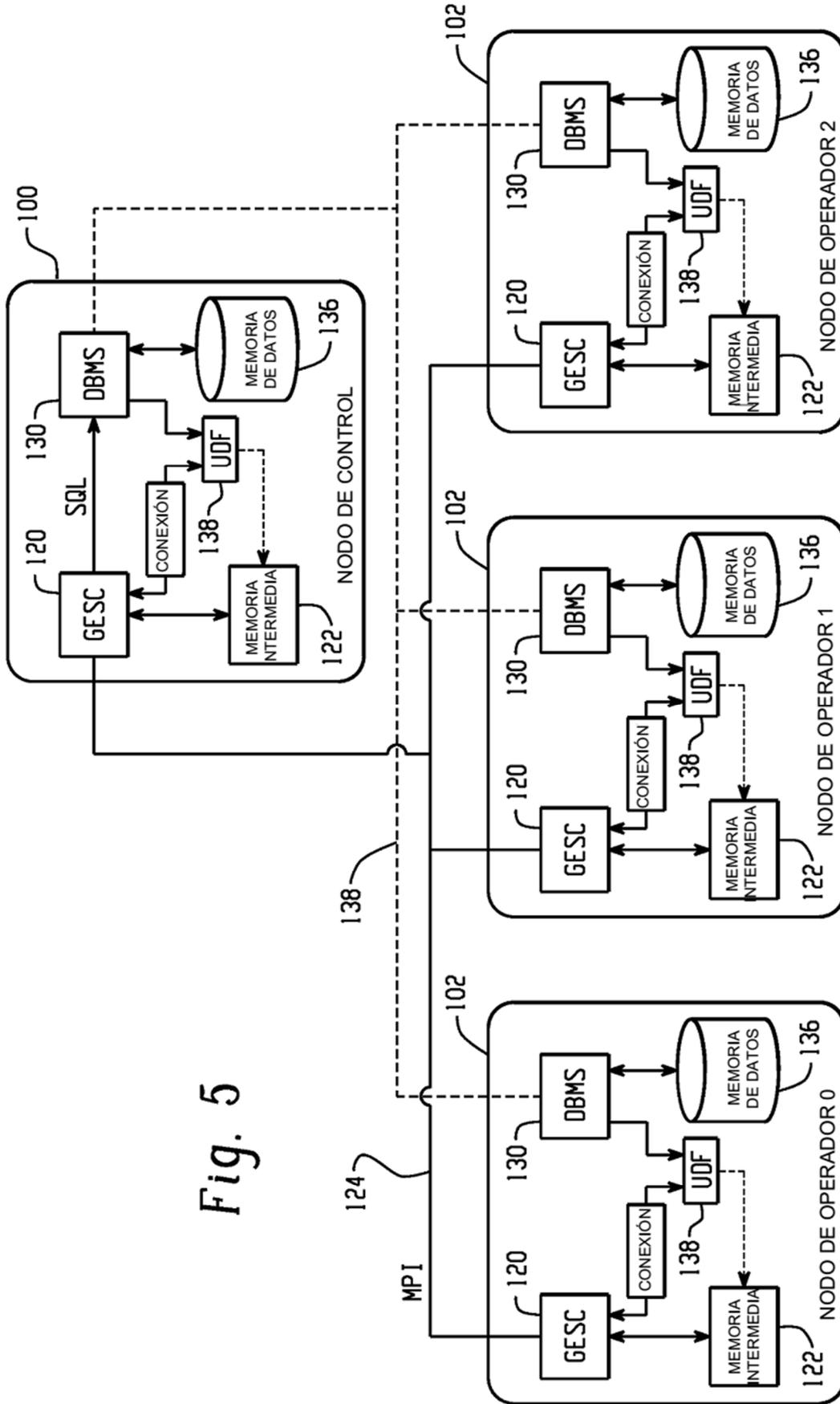


Fig. 5

GESC: COMPONENTE DE SOFTWARE BASADO EN REJILLA  
 DBMS: SOFTWARE DE GESTIÓN DE BASE DE DATOS  
 UDF: FUNCIÓN DEFINIDA POR USUARIO

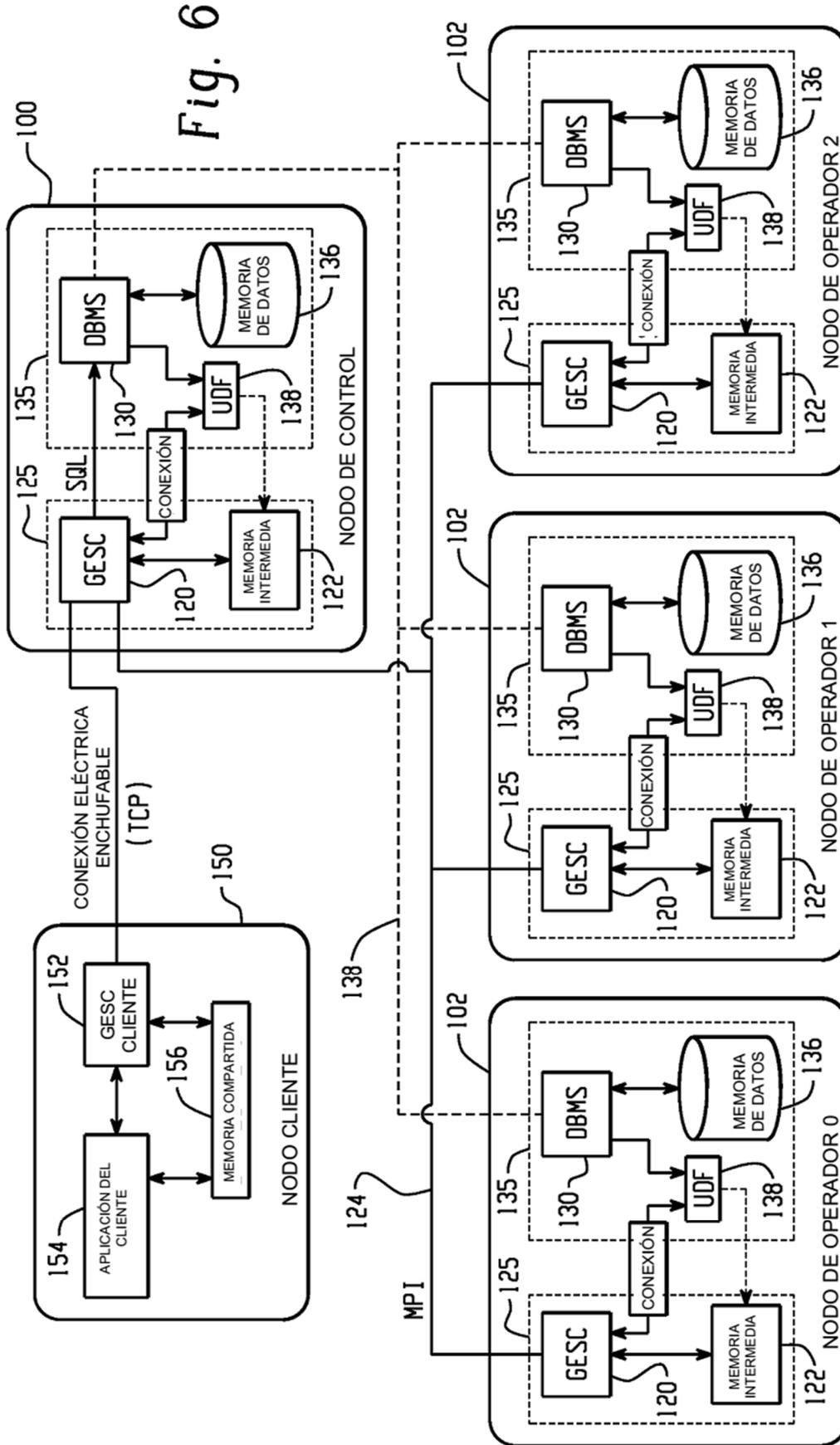


Fig. 6

GESC: COMPONENTE DE SOFTWARE BASADO EN REJILLA  
 DBMS: SOFTWARE DE GESTIÓN DE BASE DE DATOS  
 UDF: FUNCIÓN DEFINIDA POR USUARIO

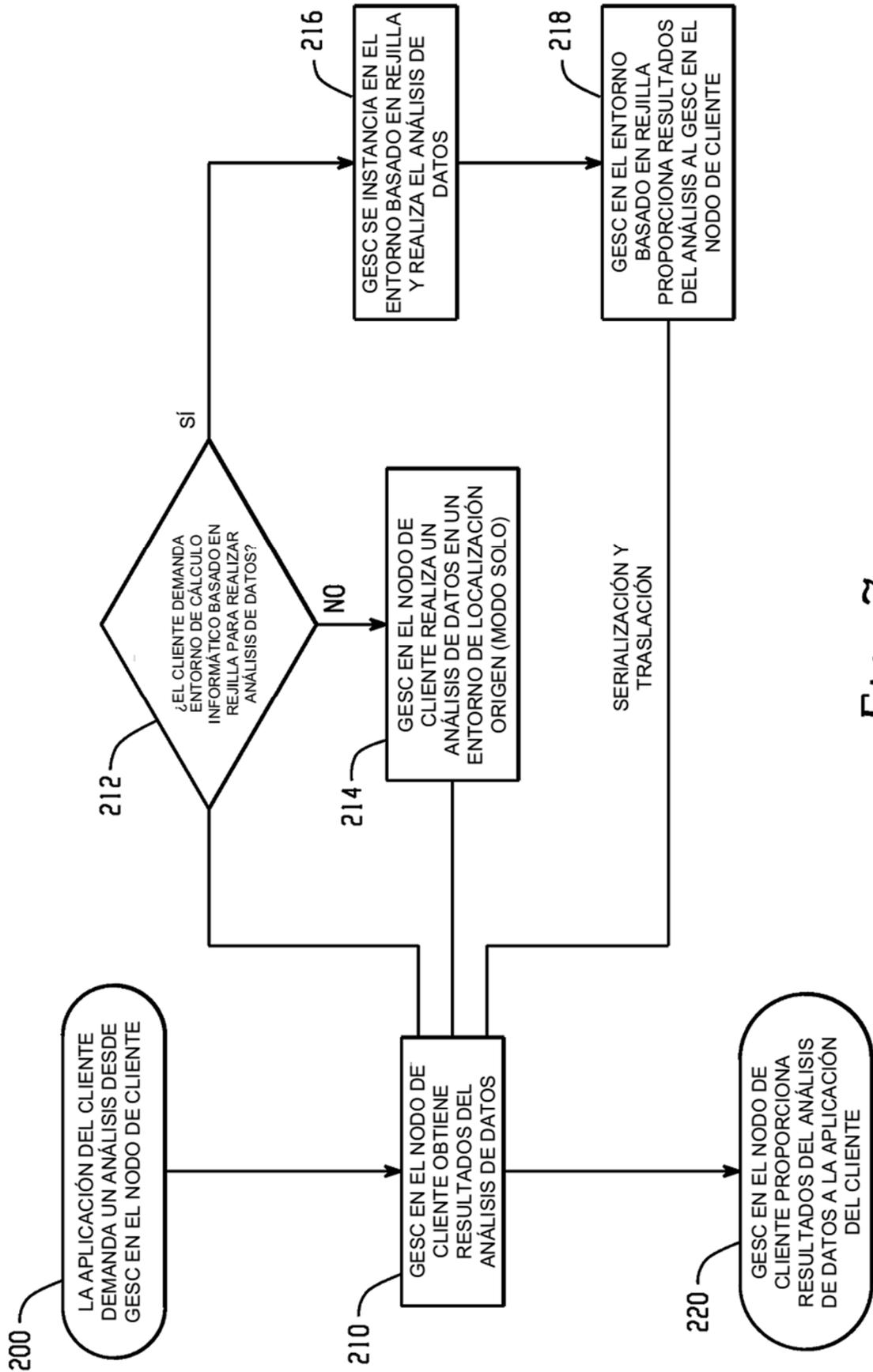


Fig. 7

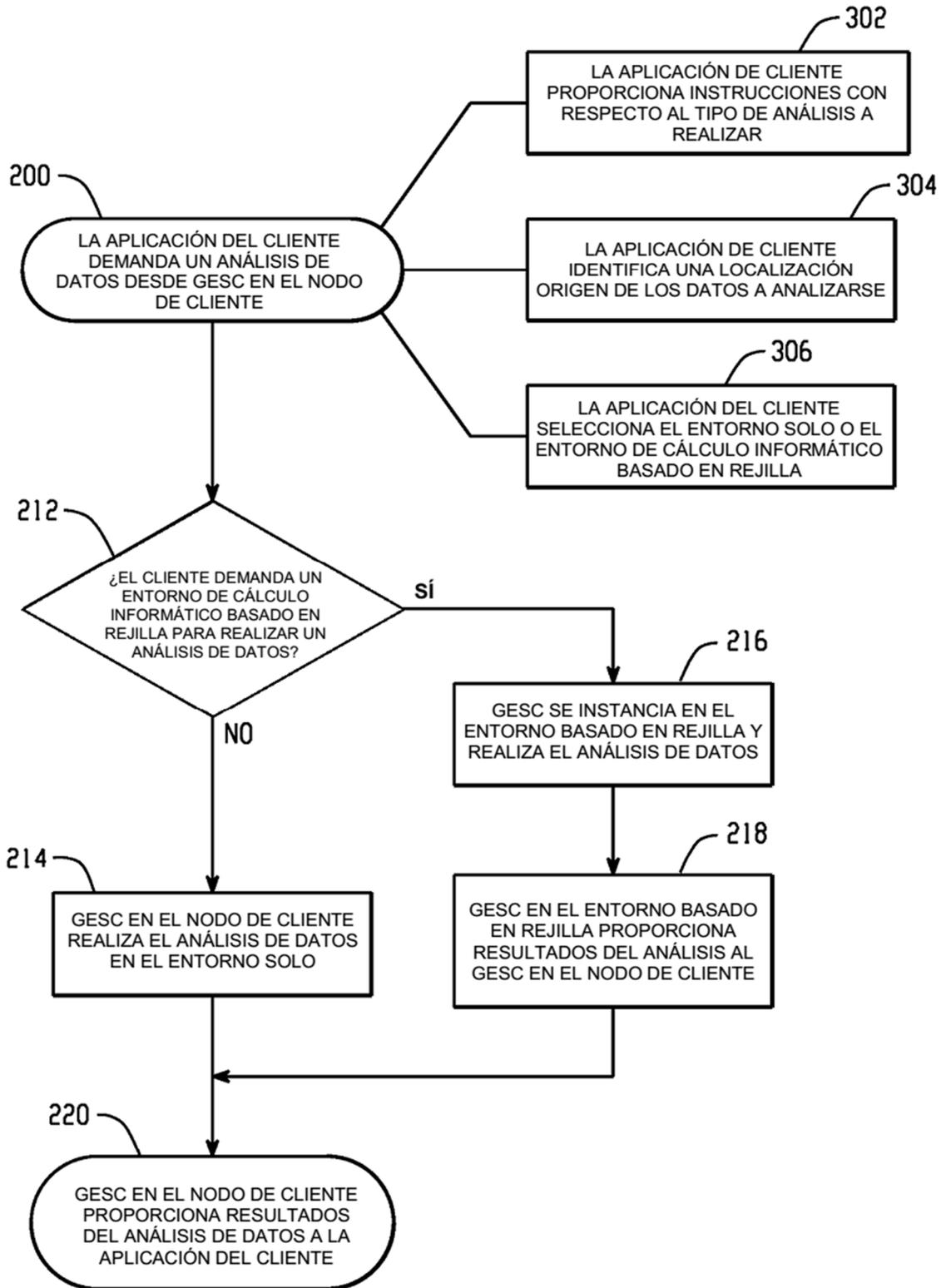


Fig. 8

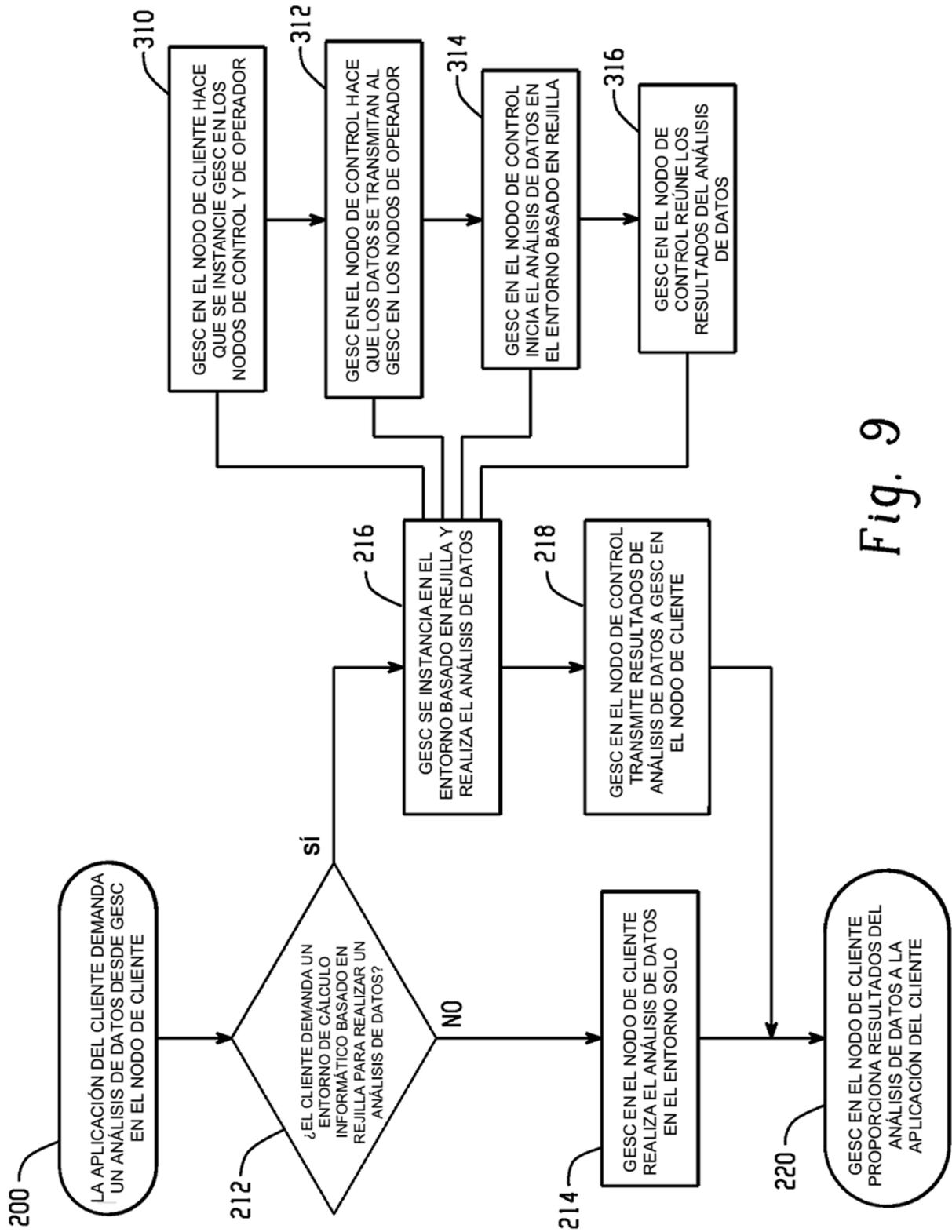


Fig. 9

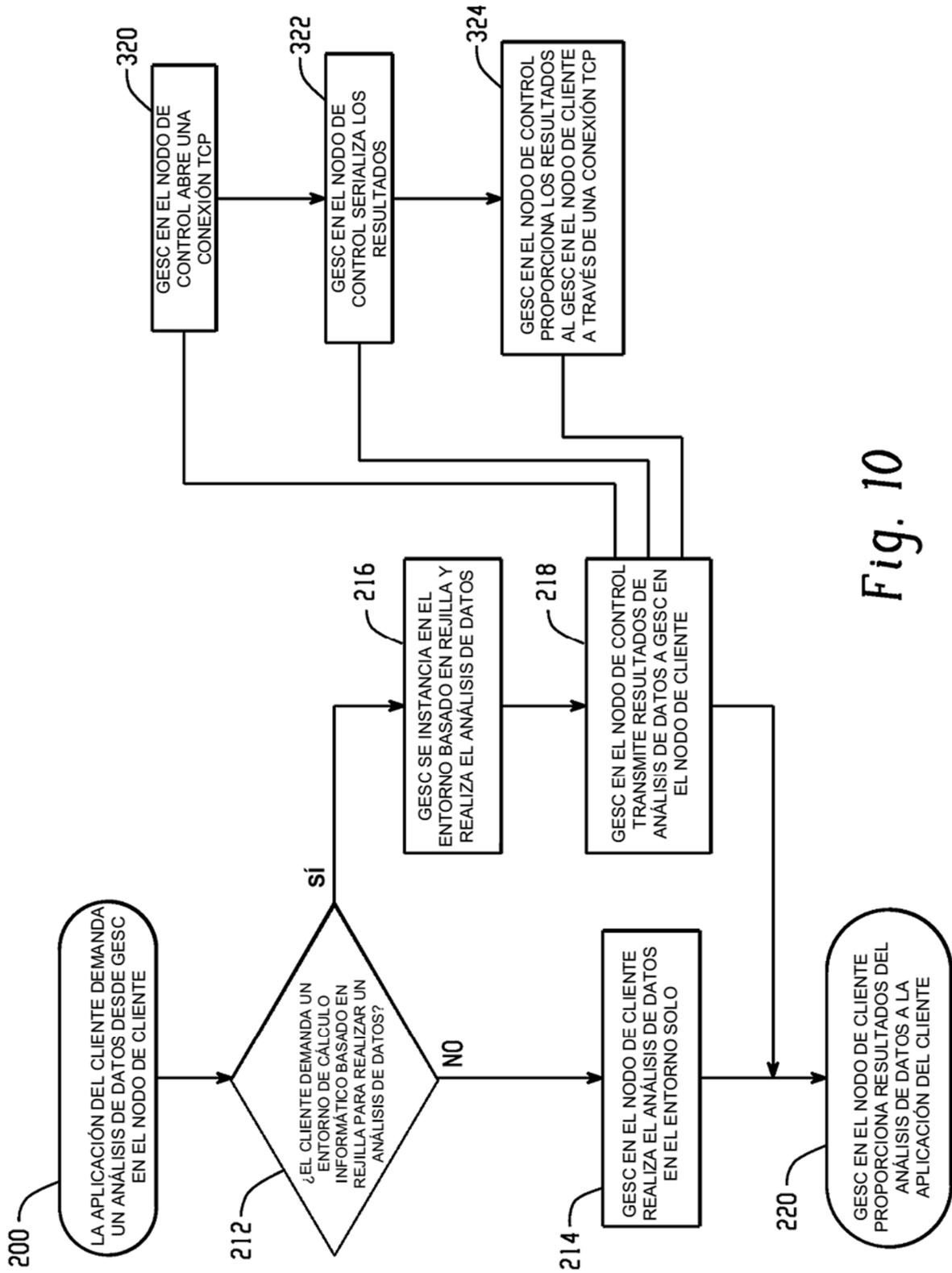


Fig. 10

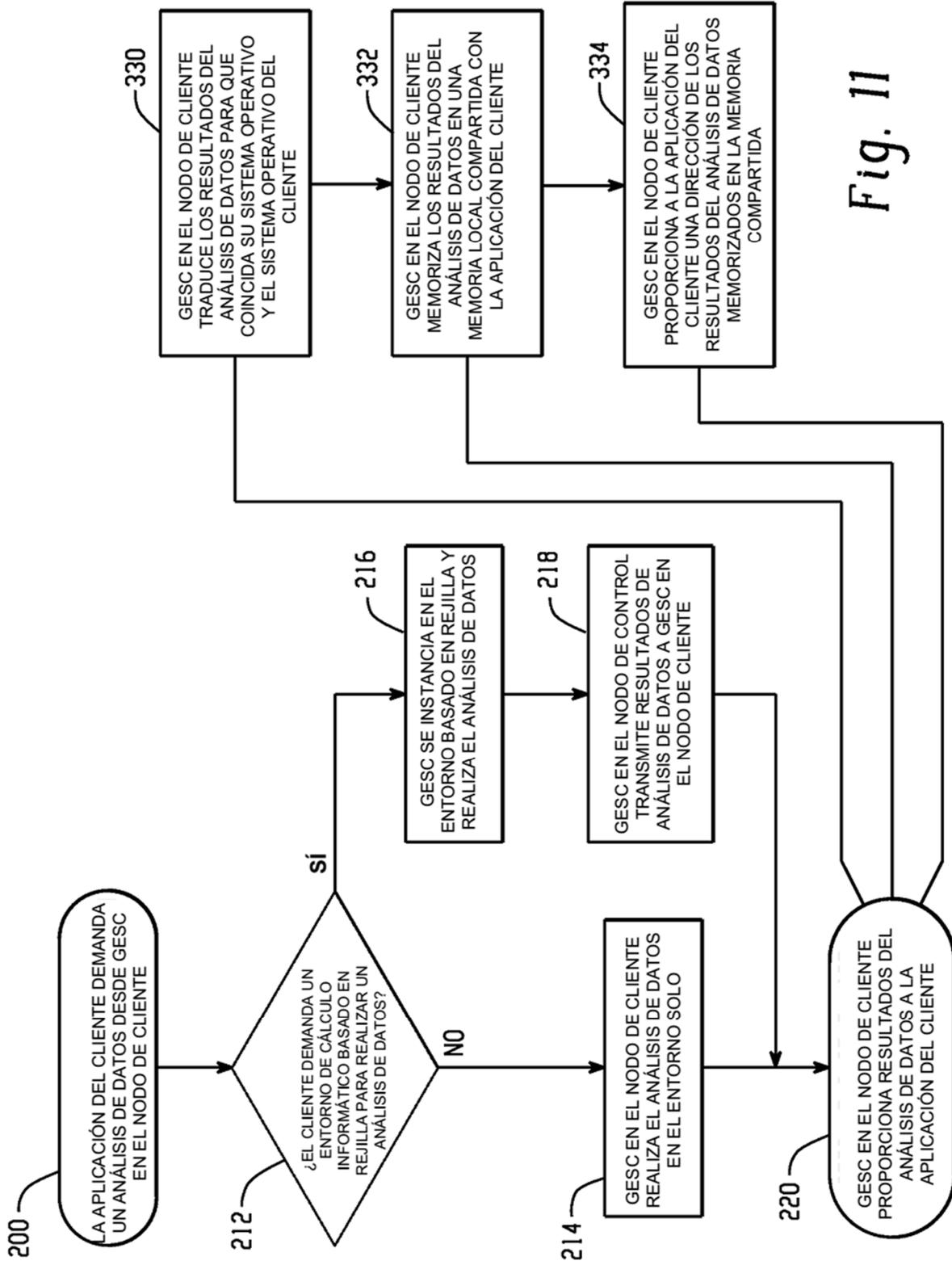


Fig. 11

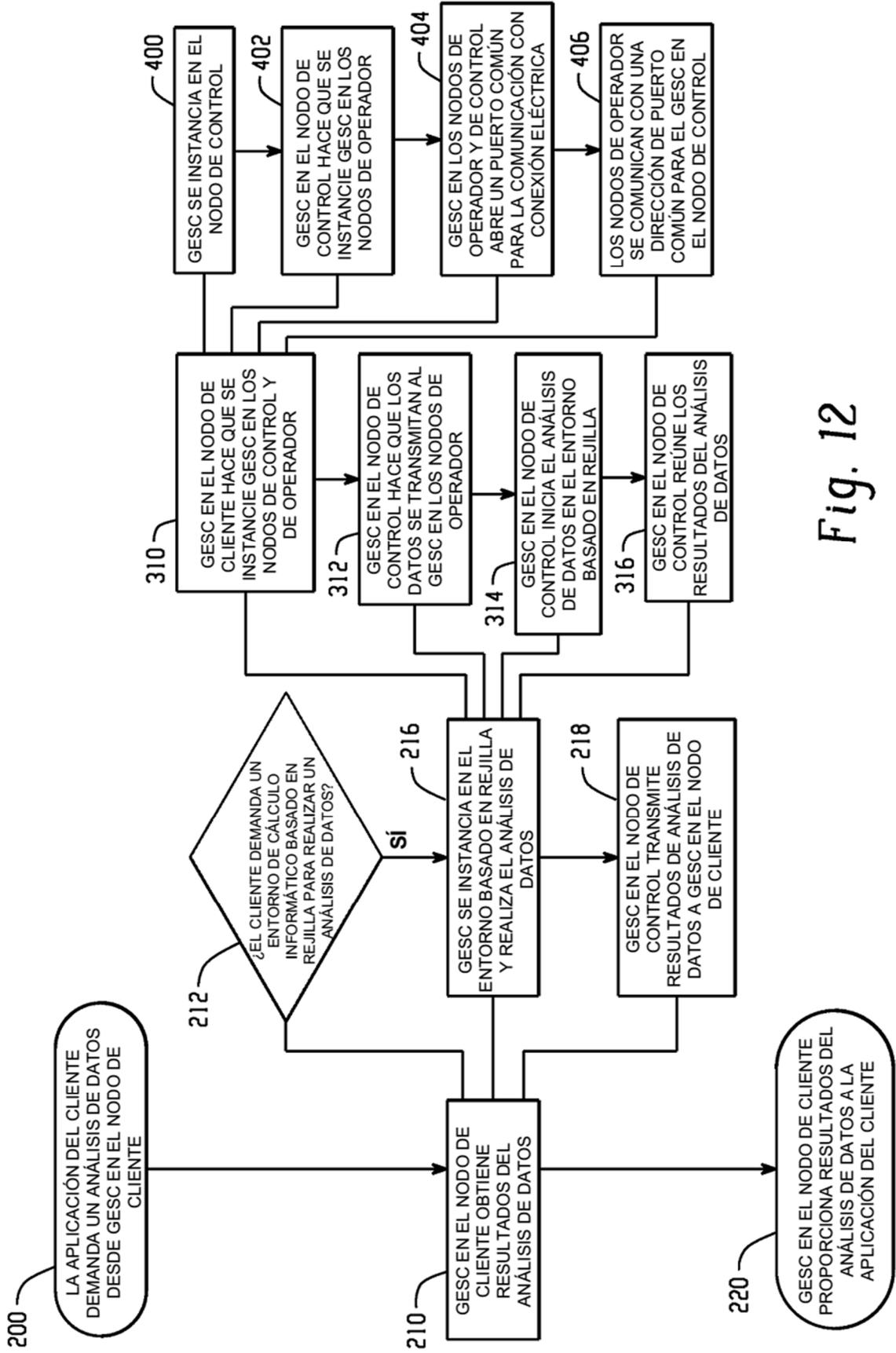


Fig. 12

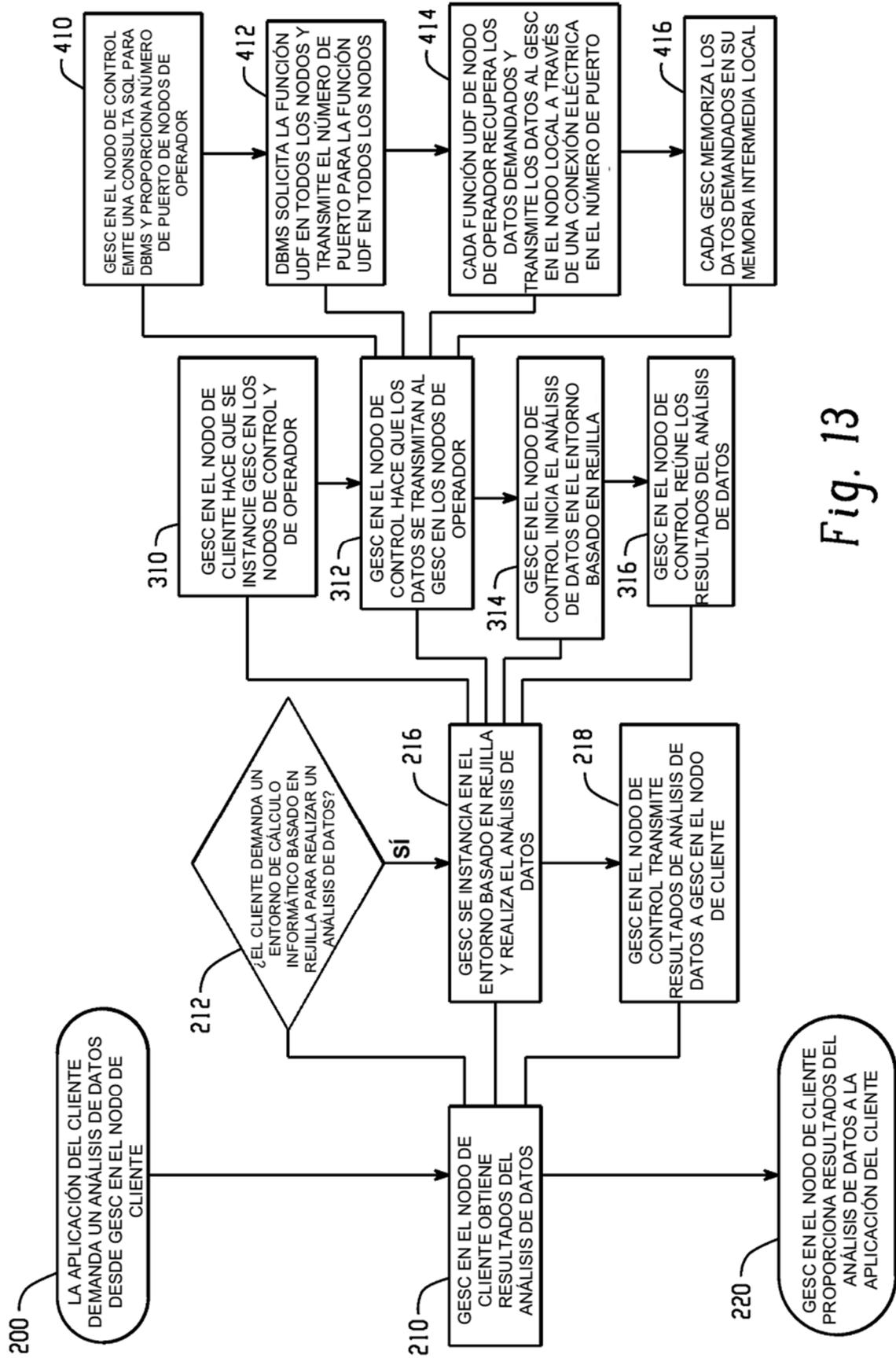


Fig. 13

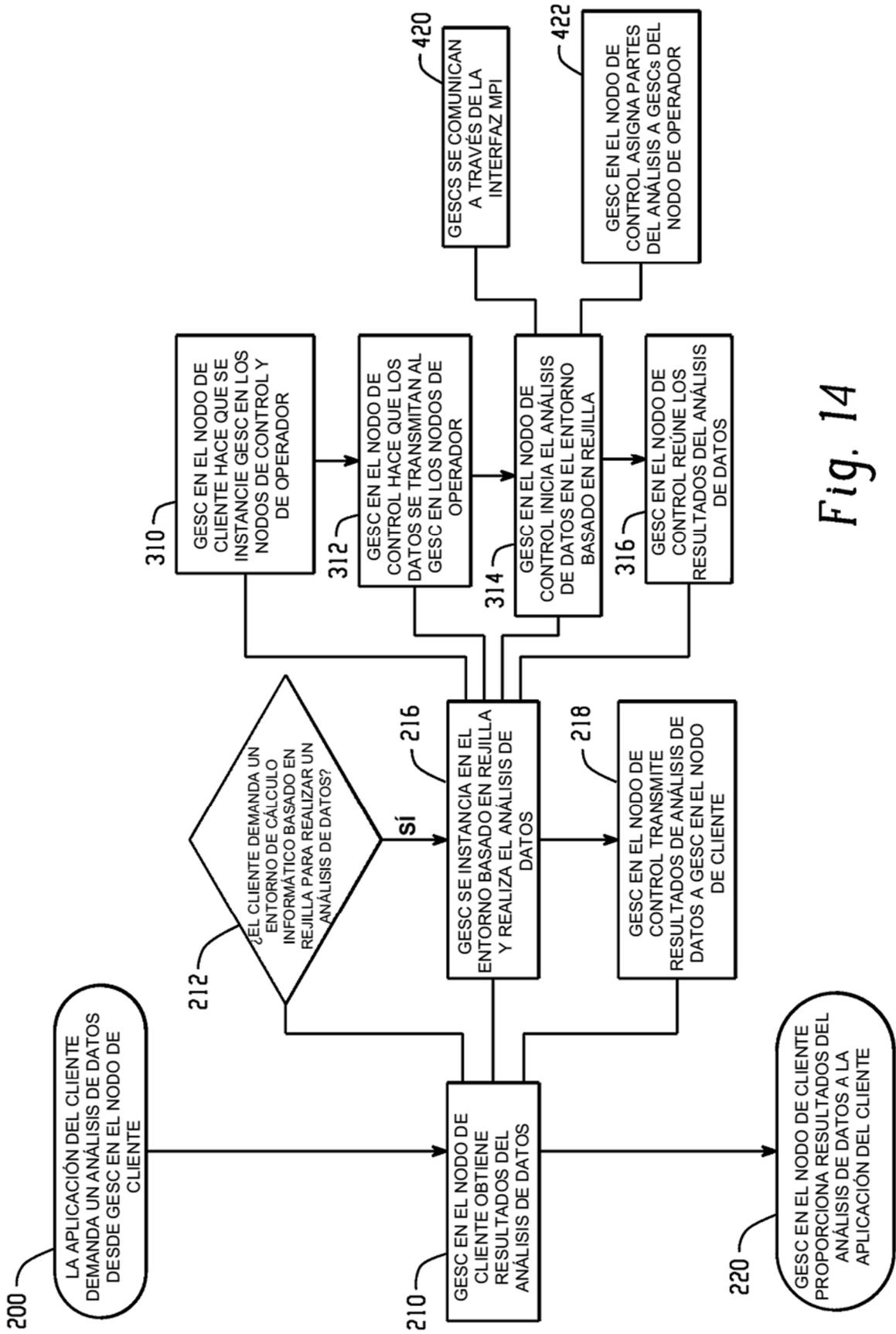


Fig. 14

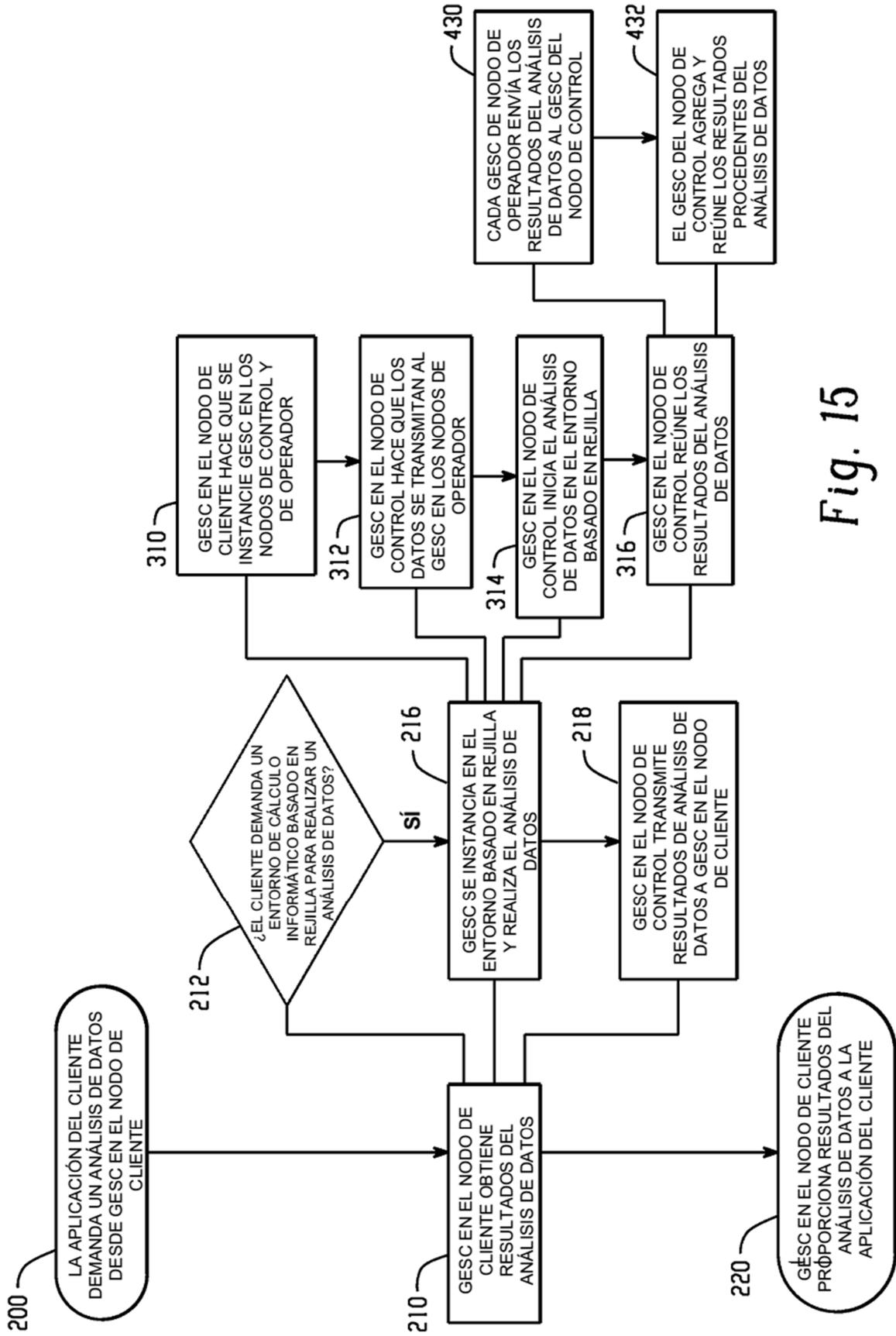


Fig. 15

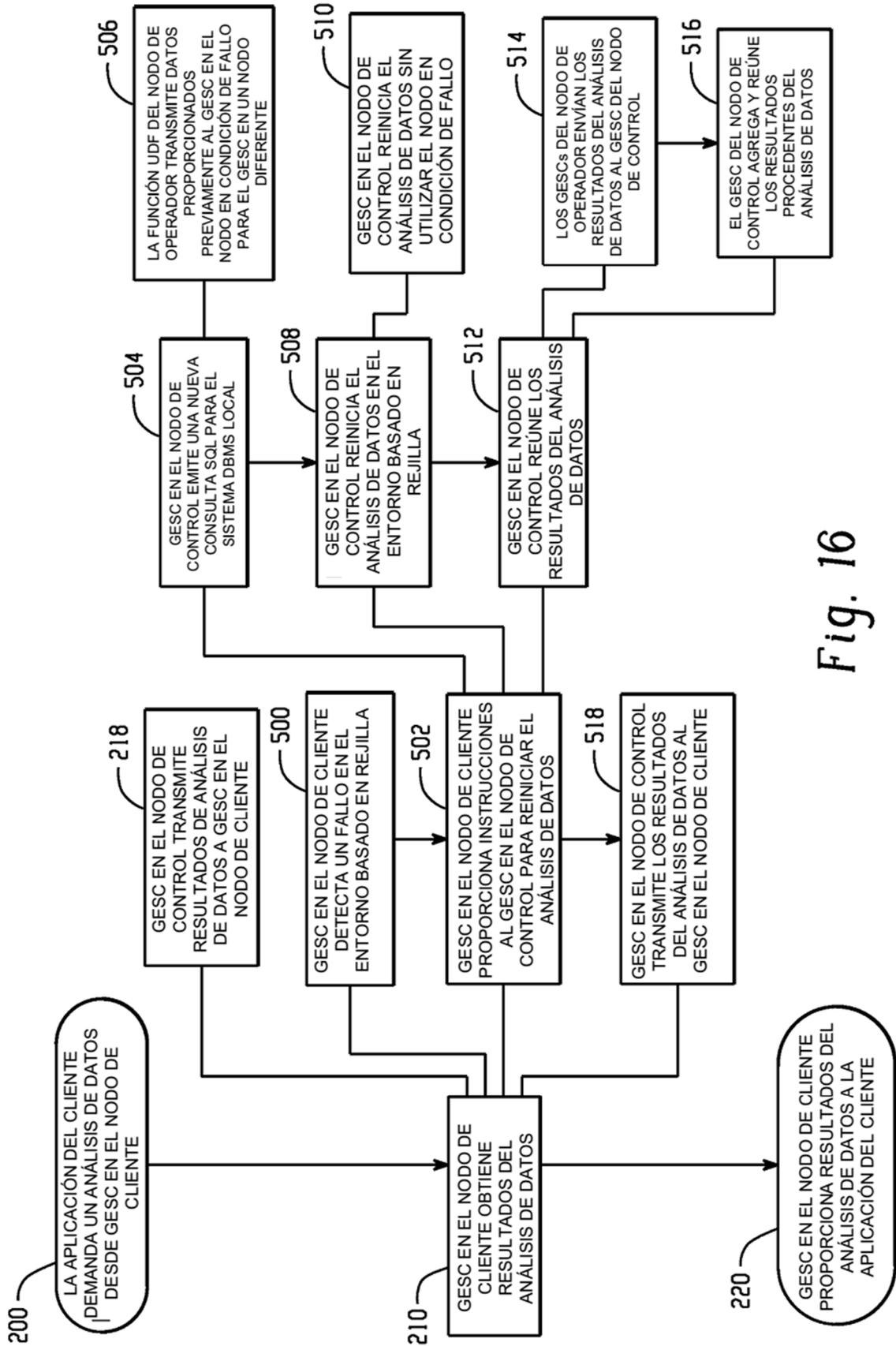


Fig. 16