

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 736 959**

51 Int. Cl.:

G06Q 10/08 (2012.01)

G06F 9/50 (2006.01)

H04L 12/24 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **19.06.2012 PCT/FR2012/051373**

87 Fecha y número de publicación internacional: **03.01.2013 WO13001204**

96 Fecha de presentación y número de la solicitud europea: **19.06.2012 E 12734969 (4)**

97 Fecha y número de publicación de la concesión europea: **24.04.2019 EP 2727057**

54 Título: **Método y programa de ordenador para identificar de forma dinámica los componentes de un clúster y automatizar las operaciones de gestión optimizada del clúster**

30 Prioridad:

28.06.2011 FR 1155752

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

09.01.2020

73 Titular/es:

**BULL SAS (100.0%)
Rue Jean Jaurès
78340 Les Clayes-sous-Bois, FR**

72 Inventor/es:

FICET, ALINE

74 Agente/Representante:

ELZABURU, S.L.P

ES 2 736 959 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 y programa de ordenador para identificar de forma dinámica los componentes de un clúster y automatizar las operaciones de gestión optimizada del clúster

5 La presente invención se refiere a la gestión de componentes en clústeres y más particularmente a un método y un programa de ordenador para identificar de forma dinámica componentes de un clúster y automatizar operaciones de gestión optimizada del clúster.

10 La computación de alto rendimiento, también conocida como HPC (High Performance Computing en terminología anglosajona), se desarrolla tanto para la investigación universitaria como para la industria, especialmente en campos técnicos tales como la automoción, la aeronáutica, la energía, la climatología y las ciencias de la vida. En particular, la modelización y la simulación permiten en particular reducir los costes de desarrollo y acelerar el lanzamiento de productos innovadores, más fiables y que consumen menos energía. Para los investigadores, la computación de alto rendimiento se ha convertido en un medio de investigación indispensable.

15 Estos cálculos se implementan generalmente en sistemas de procesamiento de datos llamados clústeres. Un clúster consiste normalmente en un conjunto de nodos interconectados. Determinados nodos se utilizan para realizar tareas de cálculo (nodos de cálculo), otros para almacenar datos (nodos de almacenamiento) y uno o varios otros para gestionar el clúster (nodos de administración). Cada nodo es, por ejemplo, un servidor que implementa un sistema operativo tal como Linux (Linux es una marca). La conexión entre los nodos se realiza, por ejemplo, mediante enlaces de comunicación Ethernet o Infiniband (Ethernet e Infiniband son marcas).

20 La figura 1 ilustra esquemáticamente un ejemplo de una topología 100 de un clúster, del tipo *fat-tree*. Este último comprende un conjunto de nodos referenciados de forma general 105. Los nodos que pertenecen al conjunto 110 son aquí nodos de cálculo mientras que los nodos del conjunto 115 son nodos de servicio (nodos de almacenamiento y nodos de administración). Los nodos de cálculo se pueden reagrupar en subconjuntos 120 llamados islas de cálculo, siendo llamado el conjunto 115 isla de servicio.

25 Los nodos se unen entre sí mediante interruptores (llamados *switch* en la terminología anglosajona), por ejemplo, de forma jerárquica. En el ejemplo ilustrado en la figura 1, los nodos se conectan a interruptores de primer nivel 125 que a su vez se conectan a interruptores de segundo nivel 130 que a su vez se conectan a interruptores de tercer nivel 135.

30 Según se muestra en la figura 2, cada nodo generalmente comprende uno o más microprocesadores, memorias locales, así como una interfaz de comunicación. Más concretamente, el nodo 200 incluye en la presente memoria un bus de comunicación 202 al que se conectan:

- unidades centrales de procesamiento o microprocesadores 204 (o CPU, acrónimo de *Central Processing Unit* en terminología anglosajona);
- componentes de memoria de acceso aleatorio 206 (RAM, acrónimo de *Random Access Memory* en terminología anglosajona) que incluye registros adaptados para registrar variables y parámetros creados y modificados durante la ejecución de los programas (según se ilustra, cada componente de memoria de acceso aleatorio se asocia a un microprocesador); y,
- interfaces de comunicación 208 adecuadas para la transmisión y recepción de datos.

El nodo 200 dispone además en la presente memoria de medios de almacenamiento interno 210, tales como discos duros, que pueden incluir en particular el código ejecutable de programas.

40 El bus de comunicación permite la comunicación y la interoperabilidad entre los diferentes elementos incluidos o vinculados al nodo 200. Los microprocesadores 204 controlan y dirigen la ejecución de instrucciones o partes de código de software del programa o de los programas. Cuando se enciende el equipo, el programa o los programas almacenados en una memoria no volátil, por ejemplo, un disco duro, se transfieren a la memoria de acceso aleatorio 206.

45 El número de componentes implementados en un clúster, especialmente equipos tales como nodos e interruptores, varía de un clúster a otro. Puede ser relativamente pequeño, por ejemplo, unas pocas decenas de nodos, o muy grande, por ejemplo, unos pocos miles o incluso decenas de miles de nodos. Además, los componentes implementados, de un mismo tipo, pueden ser diferentes. Por lo tanto, por ejemplo, determinados nodos utilizados son nodos que comprenden una sola unidad de cálculo mientras que otros nodos comprenden varias. Del mismo modo, los tamaños de memoria y las versiones de software no son necesariamente las mismas para cada nodo.

50 Para gestionar un clúster de forma eficiente y, en particular, para garantizar su buen funcionamiento, es necesario conocer todos los componentes de hardware y software que lo componen. La lista de componentes de un clúster se define de este modo a menudo en una base de datos. Si bien se puede determinar de forma teórica a partir de los datos de diseño del clúster, normalmente se establece una lista de los componentes realmente implementados para

garantizar la conformidad del clúster. Un inventario de este tipo se suele realizar por un operador que consultará, de forma manual o automática, cada uno de los componentes del clúster para determinar la composición del clúster.

Sin embargo, la creciente complejidad de los componentes de los clústeres y la evolución en el tamaño de éstos últimos requieren recursos de administración y gestión cada vez más eficientes.

5 El documento US2008/0049644 describe un estado actual de la técnica.

La invención permite resolver al menos uno de los problemas descritos anteriormente.

El propósito de la invención, por lo tanto, tiene por objetivo un método de acuerdo con la reivindicación 1.

10 El método de acuerdo con la invención permite, por lo tanto, mejorar la utilización de un clúster a partir del conocimiento de los componentes de sus elementos. Dado que el inventario de componentes se realiza de forma remota, se puede realizar de forma automática en muchas configuraciones y situaciones.

De acuerdo con una forma de realización particular, el método comprende, además, en dicho al menos un primer elemento, una etapa de detección de un evento, dichas etapas de establecimiento y transmisión de dicho al menos un inventario local se efectúa en respuesta a dicha detección. De este modo, es posible actualizar un inventario de acuerdo con eventos predeterminados.

15 También de acuerdo con una forma de realización particular, el método comprende, además, en dicho segundo elemento, una etapa de envío de una solicitud de inventario, comprendiendo dicha etapa de detección de un evento una detección de una recepción de dicha solicitud. De este modo, es posible actualizar un inventario bajo solicitud de un nodo de administración. Una solicitud de este tipo se puede producir de forma automática o manual.

20 De forma alternativa, dicho evento detectado se genera en dicho al menos un primer elemento. De este modo, es posible actualizar un inventario de acuerdo con un evento local, por ejemplo, un reinicio o una actualización.

De forma ventajosa, el método comprende además una etapa de identificación de dicho al menos un componente de hardware y/o software de dicho al menos un primer elemento, comprendiendo dicho al menos un inventario local un identificador de dicho al menos un componente de hardware y/o software de dicho al menos un primer elemento.

25 De acuerdo con formas de realización particulares, dicha operación de gestión de dicho clúster comprende una optimización de la distribución de la carga entre varios componentes de hardware de elementos de dichos varios elementos, una verificación de conformidad de al menos un componente de hardware y/o software de al menos un elemento de dichos varios elementos y/o una verificación de actualización de al menos un componente de software de al menos un elemento de dichos varios elementos.

30 También de acuerdo con una forma de realización particular, dichas etapas de recepción de dicho al menos un inventario y de establecimiento de un inventario parcial o global se repiten para todos los elementos de dichos varios elementos de un tipo predeterminado de tal manera que el inventario parcial o global sea relativo a varios elementos.

35 También de acuerdo con una forma de realización particular, al menos un inventario local de un elemento de dicho clúster que no tiene capacidad de procesamiento de datos es establecido por dicho segundo elemento, siendo utilizado por lo menos un dato de dicho inventario local de dicho elemento que no tiene capacidad de procesamiento de datos para el establecimiento de dicho inventario parcial o global. El inventario parcial o global comprende, por lo tanto, características de elementos que no tienen capacidad de procesamiento de datos, tales como los interruptores. Los elementos que tienen capacidad de procesamiento de datos son, por ejemplo, los nodos.

40 La invención también tiene por objetivo un programa de ordenador que comprenda instrucciones adecuadas para la implementación de cada una de las etapas del método descrito anteriormente cuando dicho programa se ejecuta en un ordenador. Las ventajas proporcionadas por este programa de ordenador son similares a las mencionadas anteriormente.

Otras ventajas, objetivos y características de la presente invención salen de la siguiente descripción detallada, hecha a título de ejemplo no limitativo, con relación a los dibujos adjuntos en los que:

- La figura 1 ilustra un ejemplo de la topología de un clúster;
- 45 - La figura 2 ilustra un ejemplo de arquitectura de un nodo de clúster;
- La figura 3 ilustra esquemáticamente un ejemplo de la implementación del procedimiento de acuerdo con la invención en un clúster;
- La figura 4 ilustra esquemáticamente determinadas etapas implementadas en un elemento de clúster, por ejemplo, un nodo, para efectuar un inventario local;
- 50 - La figura 5 ilustra esquemáticamente determinadas etapas implementadas en un módulo de inventario de síntesis; y

- La figura 6 ilustra un ejemplo de una secuencia de etapas que permite obtener un inventario global bajo la forma de fichero XML.

5 De manera general, la invención permite efectuar un inventario de los componentes de hardware y/o software de un clúster o parte de un clúster mediante la combinación de una determinación local automática de los componentes de determinados elementos que tienen capacidades de procesamiento de datos (por ejemplo, equipos de tipos predeterminados), o de todos ellos, y una síntesis global de los componentes del conjunto de estos elementos con el fin de automatizar las operaciones de gestión optimizada del clúster.

10 Para estos fines, cada elemento objetivo (que tiene capacidades de procesamiento de datos), por ejemplo, cada equipo de un tipo predeterminado, normalmente cada nodo, de un clúster comprende un módulo de inventario local, mientras que un elemento de gestión, por ejemplo, un nodo de gestión, comprende un módulo de inventario de síntesis. De acuerdo con las necesidades, el módulo de inventario de síntesis puede consultar a los módulos de inventario local y los módulos de inventario local pueden transmitir, de forma automática, informaciones relativas a los componentes del elemento en cuestión, por ejemplo, cuando se identifica un evento específico en este elemento.

15 Los elementos que no disponen de capacidad de procesamiento de datos, por ejemplo, los interruptores, se pueden consultar por los elementos que disponen, por ejemplo, de nodos de administración, los cuales pueden a continuación establecer inventarios locales susceptibles de ser agregados con otros inventarios locales y ser utilizados para actualizar uno o más inventarios parciales y/o globales.

La figura 3 ilustra esquemáticamente un ejemplo de la implementación del método de acuerdo con la invención en un clúster 300.

20 Según se ilustra, el clúster 300 comprende un conjunto de elementos que tienen capacidades de procesamiento de datos, en la presente memoria los nodos 305-1 a 305-5 (generalmente referencias 305) y el nodo de gestión 310, conectados entre sí a través de la red de comunicaciones 315 (que comprende en la presente memoria enlaces e interruptores, no representados).

25 Cada nodo 305 comprende un módulo de inventario de forma genérica referido 320. Por lo tanto, por ejemplo, el nodo 305-1 comprende el módulo de inventario local 320-1 y el nodo 305-2 comprende el módulo de inventario local 320-2. El nodo de gestión 310 comprende un módulo de inventario de síntesis 325.

30 Un inventario local se realiza mediante un módulo de inventario local con la ayuda de comandos estándar, por ejemplo, comandos de Linux. Estos comandos se reagrupan en secuencias, por ejemplo, una secuencia de comandos escrita de acuerdo con el lenguaje PERL. El objetivo de estos comandos es obtener informaciones relativas a los componentes de hardware y software del elemento en cuestión.

35 En el anexo (código 1) se presenta un ejemplo de un extracto de una secuencia de este tipo. Los comandos que se dan en la presente memoria tienen por objetivo la obtención de datos relacionados con un conector Ethernet de un nodo. El comando *ethtool* permite obtener estos datos que son procesados a continuación para extraer el nombre y la referencia del microcódigo (*firmware*) así como el tipo y la versión del componente, la versión de su microcódigo y una descripción del bus. Se utilizan comandos equivalentes para obtener informaciones relativas a los otros componentes (microprocesadores, módulos de memoria, discos duros, etc.).

40 Estos comandos pueden consistir, en particular, en consultas de los componentes objetivo o en la obtención, normalmente la lectura, de datos de configuración del elemento en cuestión o datos de los registros de eventos de los sistemas, es decir, datos de ejecución tales como los que se presentan en el apéndice (registro de eventos *syslog*). Este ejemplo muestra, en particular, el número de memorias y microprocesadores disponibles, así como la información de acuerdo con la cual se han transmitido correctamente estos números.

45 Los componentes de hardware de un nodo son, por ejemplo, las unidades centrales de procesamiento o microprocesadores 204, los componentes de acceso aleatorio 206, las interfaces de comunicación 208 y los medios de almacenamiento interno 212 mostrados en la figura 2. Los componentes de software son normalmente módulos del núcleo del sistema operativo del nodo, así como módulos relacionados con este núcleo.

A título ilustrativo, las características obtenidas son las siguientes:

- componentes de hardware (procesadores, memorias, etc.):

- nombre, tipo, vendedor, modelo, ubicación, número de serie, referencia, descripción y microcódigo; y,
- cantidad de cada componente;

50 - componentes de software (incluido el sistema operativo):

- nombre, tipo, versión, fecha de instalación.

En la presente memoria se realiza un inventario local tras la detección de un evento determinado. Un evento de este tipo puede estar relacionado, en particular, con un cambio de estado del elemento en cuestión (es decir, el elemento en el que se efectúa el inventario local), por ejemplo, la detección de un fallo, o con un comando. Este último se puede generar de forma local, por ejemplo, un comando de reinicio (normalmente llamado *reboot* en la terminología anglosajona) o recibido de un dispositivo remoto, por ejemplo, un comando de ejecución del módulo de inventario local recibido del módulo de inventario de síntesis del nodo de gestión, según se ilustra en la figura 3 mediante las flechas de línea de puntos.

Cuando se ha establecido un inventario local, se transmite al módulo de inventario de síntesis del nodo de gestión, según se ilustra en la figura 3 mediante las flechas unidireccionales en línea continua, para formar un inventario global, si es necesario. El inventario global consiste, por ejemplo, en la concatenación de inventarios locales (que pueden comprender, en particular, el inventario local del nodo de gestión). El inventario global se almacena normalmente en una base de datos accesible por el nodo de gestión. También se puede memorizar localmente por este último.

Los nodos 305 y 310 pueden además incluir un módulo de inventario local que permite establecer inventarios locales de elementos que no tienen capacidad de procesamiento de datos (no mostrados en la figura 3).

De acuerdo con una forma de realización particular, los inventarios locales, así como el inventario global se intercambian y/o almacenan en forma de ficheros en formato XML (acrónimo de *Extensible Markup Language* en terminología anglosajona).

A título ilustrativo, en el anexo se presenta un extracto de un fichero XML (Extracto de un fichero de inventario en formato XML). Aunque el ejemplo dado aquí se refiere a varios componentes de diferentes tipos, está simplificado, pudiendo un elemento en particular comprender varias decenas o centenas de equipos de los mismos tipos o de diferentes tipos.

De acuerdo con este ejemplo, el inventario local suministra las características del elemento en cuestión, en la presente memoria un equipo, por ejemplo su nombre (*nombre_1*), su tipo (Nodo), su vendedor (*vendedor_1*), su modelo (*modelo_1*), su dirección IP (acrónimo de *Internet Protocol* en terminología anglosajona) (12.18.64.254), su función (monitorización o *mngt*), su estado (*up*) así como su posición geográfica (referencia de armario de datos y posición en este último). También suministra características sobre los componentes de software instalados en el elemento en cuestión, principalmente su nombre (por ejemplo, *software_1*), su tipo (por ejemplo, *instalación*), su versión (por ejemplo, *1.12*) y su fecha de instalación (por ejemplo, *2011-06-07 08:04:45*). El inventario local suministra también características de los componentes de hardware, por ejemplo sistemas, microprocesadores, memorias de acceso aleatorio y discos duros, del elemento en cuestión, tales como su nombre (por ejemplo, *bullx*), su tipo (por ejemplo, *sistema*), su proveedor (por ejemplo, *Bull SAS*), su modelo (por ejemplo, *modelo_1*) y, si es necesario, su número de serie (por ejemplo, *CR-1763578DF485*), una referencia (o *part number* en terminología anglosajona), una descripción e informaciones relativas al microcódigo, tales como un nombre (por ejemplo, *BIOS*), una versión (por ejemplo, *1.0b*, *Revisión:8.16(14/08/2009)*) y una descripción (por ejemplo, *proveedor de American Megatrends Inc.*).

El inventario global (o uno o más inventarios parciales) se puede utilizar a continuación para automatizar las operaciones de gestión optimizada del clúster, estando la optimización vinculada al inventario global y/o a uno o más inventarios parciales.

Por lo tanto, por ejemplo, conociendo las características de los nodos de un clúster, en términos de disponibilidad y rendimiento, es posible mejorar la distribución de carga en función de los procesos a ejecutar. Para estos fines, se puede utilizar un algoritmo estándar para asignar los recursos identificados.

Todavía a título ilustrativo, el inventario permite comprobar el correcto desarrollo de las actualizaciones de los componentes de software, a escala de clúster, de un equipo o más, en general, de un elemento que tenga capacidades de tratamiento de datos. Del mismo modo, el inventario permite verificar el material que debe estar presente en un equipo en relación con el material previsto contractualmente. Unas verificaciones de este tipo se pueden efectuar comparando las características de los componentes del clúster, tal y como se definen en el inventario global y/o en uno o más inventarios locales con datos predeterminados, por ejemplo, datos de configuración del clúster.

La figura 4 ilustra esquemáticamente determinadas etapas implementadas en un elemento de un clúster, por ejemplo, un nodo, para efectuar un inventario local.

Según se ilustra, una primera etapa (etapa 400) tiene por objetivo la detección de un evento en particular, por ejemplo, un evento que pertenece a una lista de eventos predeterminados. Un evento de este tipo se puede relacionar en particular con el elemento para el que se va a establecer el inventario o con un evento externo, tal como una solicitud de establecimiento de un inventario local recibida de un nodo de gestión.

En las etapas siguientes (etapas 405 y 410) se identifican los componentes de hardware y/o software. Según se indicó anteriormente, estos componentes se pueden identificar a partir de consultas o lecturas que pueden tomar la forma de una secuencia de comandos, en particular de una secuencia de comandos descrita en el lenguaje PERL. Las características de estos componentes, por ejemplo, las referencias de versión y de cantidades, se obtienen ventajosamente de forma similar y se asocian con los componentes identificados.

Los componentes identificados y las características asociadas obtenidas se utilizan a continuación para formar un inventario local, preferiblemente de acuerdo con un formato predeterminado, según se describió anteriormente (etapa 415), que se transmite a un módulo de inventario de síntesis que se puede implementar en particular en un nodo de gestión (etapa 420), que se agrega con otros inventarios locales para formar un inventario parcial y/o global.

- 5 Normalmente, la implementación del algoritmo descrito en referencia a la figura 4 se divide en dos partes. Una primera parte, referenciada 425, que comprende la etapa 400, está intrínsecamente ligada al sistema operativo implementado en el elemento en cuestión (o en conexión con este último) con el fin de detectar un evento en particular y, en respuesta a esta detección, iniciar la ejecución de un módulo de construcción y transmisión de un inventario local. La segunda parte, referenciada 430, que corresponde al módulo de construcción y transmisión de un inventario local, comprende en la presente memoria las etapas 405 a 420. Según se indicó anteriormente, esta segunda parte utiliza, por ejemplo, el lenguaje PERL.

La figura 5 ilustra esquemáticamente determinadas etapas implementadas en un módulo de inventario de síntesis. Según se describió anteriormente, un módulo de inventario de síntesis suele pertenecer a un nodo de gestión.

- 15 Una primera etapa (etapa 500), opcional (según se representa mediante la utilización de líneas de puntos), tiene por objetivo el envío de una solicitud de inventario. De acuerdo con las necesidades, una solicitud de este tipo se dirige a un elemento en particular, conforme a un modo de comunicación llamado *unicast*, a un conjunto de elementos, por ejemplo, una isla, conforme a un modo de comunicación llamado *multicast* o a todos los elementos conforme a un modo de comunicación llamado *broadcast*. La solicitud puede tomar en particular la forma de un mensaje, utilizando su transmisión un protocolo de comunicación estándar

- 20 En una etapa posterior (etapa 505), se recibe un inventario local de un elemento, por ejemplo, de un nodo, pudiendo este último utilizar un algoritmo tal como el descrito en referencia a la figura 4.

- 25 El inventario recibido se utiliza a continuación para actualizar (etapa 510) uno o más inventarios parciales, por ejemplo, un inventario vinculado a una isla y un inventario vinculado a un conjunto de islas y/o un inventario global. Una etapa de este tipo consiste, por ejemplo, en modificar, en uno o más inventarios parciales y/o globales, los datos correspondientes a las informaciones recibidas (inventario local).

En la presente memoria se observa que, si el o los inventarios parciales y/o global no existen, se crea o se crean.

Según se ilustra mediante la flecha de línea de puntos, las etapas 505 y 510 se repiten de forma ventajosa para todos los inventarios locales recibidos.

- 30 El o los inventarios parciales y/o global se puede o se pueden utilizar para automatizar operaciones de gestión optimizada del clúster, por ejemplo, optimizando la distribución de carga (etapa 515-1), verificando la conformidad del clúster (etapa 515-2) y/o verificando una actualización de uno o varios elementos del clúster (etapa 515-n). Los algoritmos de optimización de la distribución de la carga, de verificación de la conformidad del clúster y de verificación de la actualización de uno o varios elementos del clúster son en la presente memoria algoritmos estándar.

- 35 Los algoritmos descritos en referencia a las figuras 4 y 5 se implementan en elementos que tienen capacidades de procesamiento de datos, normalmente equipos tales como nodos cuya arquitectura es similar a la descrita en referencia a la figura 2.

La figura 6 ilustra un ejemplo de una secuencia de etapas que permite obtener un inventario global bajo la forma de fichero XML.

- 40 Una primera etapa (etapa 600) en la presente memoria tiene como objetivo la ejecución del comando "*clminventory inventory -name nom_1*" que tiene como objetivo la ejecución de la función "*clminventory*" con el nombre "*nom_1*" como parámetro, que permite la obtención del inventario local del elemento cuyo nombre se le da como parámetro.

- 45 En una etapa posterior (etapa 605), se determina el tipo de elemento cuyo inventario local se debe establecer, es decir, se determina el tipo de elemento que tiene por nombre "*nom_1*". El tipo de elemento se determina a partir de su nombre con la ayuda de datos de configuración del clúster. Si el elemento en cuestión no tiene capacidades de procesamiento de datos, el inventario local se establece mediante otro elemento, de forma convencional (no representada).

Por el contrario, si el elemento en cuestión dispone de capacidades de procesamiento de datos, se le envía una instrucción, por ejemplo, en forma de evento en un mensaje que se le transmite, con el fin de que en respuesta transmita su inventario local (etapa 610).

- 50 El elemento objetivo, es decir, el elemento denominado "*nom_1*", crea su inventario local según se describió anteriormente (etapa 615), utilizando por ejemplo datos de ejecución (*syslog*). El inventario local se crea en la presente memoria bajo la forma de un fichero XML. A continuación, se transmite al elemento que originó la solicitud, es decir, el elemento que inició el inventario remoto.

Después de haber obtenido el fichero en el formato XML que representa el inventario local (etapa 620), el contenido de este fichero se utiliza para actualizar un inventario parcial o global del clúster (etapa 625) también presentado bajo la forma de un fichero XML que a continuación se almacena (etapa 630). La actualización del inventario parcial o global puede consistir en agregar informaciones que falten y/o en modificar informaciones memorizadas previamente.

- 5 Las etapas 600 a 610 y 620 a 630 se implementan en la presente memoria en un elemento 635 desde el cual es posible iniciar de forma automática o manual un inventario de un elemento del clúster al que pertenece. El elemento 635 es normalmente un nodo administrativo. La etapa 615 se implementa en un elemento 640 del clúster que tiene capacidades de procesamiento de datos, por ejemplo, un nodo de cálculo.

- 10 Naturalmente, para satisfacer necesidades específicas, una persona competente en el campo de la invención podrá aplicar modificaciones en la descripción anterior.

ANEXO

```

# kawela firmware
my $kawela = qx(ethtool -i eth0);
5 my $kawelaComponent = Bull::CLM::Inventory::Component->new({'name','Kawela'});
my $kawelaFirmware = Bull::CLM::Inventory::Firmware->new({'name','Firmware Kawela'});
my @k_lines = split( /\n/, $kawela );
foreach my $line (@k_lines) {
10   if( $line =~ /^driver: (.*)/){
       $kawelaComponent->{'type'} = $1;
   }
   elsif( $line =~ /^version: (.*)/){
       $kawelaComponent->{'description'} .= ' version : '.$1;
   }
15   elsif( $line =~ /^firmware-version: (.*)/){
       $kawelaFirmware->{'version'} = $1;
   }
   elsif( $line =~ /^bus-info: (.*)/){
       $kawelaComponent->{'description'} = ' bus-info : '.$1;
20   }
}

```

Código 1

```

25 1308061409 2011 Jun 14 16:23:29 berlin32 local0 info inventory [inventory@clmBull
action="cpu_memory" memory_available="36" cpu_available="8" host="nom_1"
status="success"] Cpu_available and memory_available successfully sent

```

Registro de eventos syslog

```

30 <?xml version="1.0" encoding="UTF-8"?>
<bcm:configuration xmlns:bcm="http://www.xml.bcm.bull/inventory">
  <date>2011-06-14 12:33:06</date>
  <equipment>
    <name>nom_1</name>
    <type>Node</type>
35   <vendor>vendeur_1</vendor>
    <model>modele_1</model>
    <ip>12.18.64.254</ip>
    <function>mnngt</function>
    <status>up</status>
40   <location>
    <rackLabel>M3-0</rackLabel>
    <x>A</x>
    <y>3</y>
    <z>K</z>
45   <slot>1</slot>
  </location>
  <software>
    <name>logiciel_1</name>
    <type>install</type>
50   <version>1.12</version>
    <installDate>2011-06-07 08:04:45</installDate>
  </software>
  <component>

```

```

5     <name>bullx</name>
      <type>System</type>
      <vendor>Bull SAS</vendor>
      <model>modele_1</model>
      <serialNumber>CR-1763578DF485</serialNumber>
      <firmware>
10        <name>BIOS</name>
          <version>1.0b, Revision: 8.16 (08/14/2009)</version>
          <description>Vendor from American Megatrends Inc.</description>
      </firmware>
      </component>
      <component>
15        <name>CPU_1</name>
          <type>Processor</type>
          <vendor>Intel</vendor>
          <model>XeonIntel(R) Xeon(R) CPU X5550 @ 2.67GHz</model>
          <serialNumber>13547RE2156</serialNumber>
          <partNumber>1385735465499FER546DA</partNumber>
          <description>Core Count: 4, Core Enabled: 4, Thread Count: 4</description>
20      </component>
      <component>
          <name>P1_DIMM1A#0</name>
          <type>Memory</type>
          <vendor>Qimonda</vendor>
25        <model>DIMM Type: Other</model>
          <location>P1_DIMM1A (BANK0)</location>
          <serialNumber>2665C30329</serialNumber>
          <partNumber>IMSH2G13A1F1C-10F</partNumber>
          <description>Size: 2048 MB</description>
30      </component>
      <component>
          <name>host6/disk#0</name>
          <type>Disk</type>
          <vendor>SEAGATE</vendor>
35        <model>ST314664855SS</model>
          <location>Enclosure 0, Slot 0</location>
          <serialNumber>3LN7BHIV900009922LBM1</serialNumber>
          <description>Size: 140014 MB, Protocol: SAS, Status: OK, </description>
40      </component>
      </equipment>
      <checksumEquipment>0b0eefac72d93530243f4aef134765</checksumEquipment>
</bcm:configuration>

```

Extracto de fichero de inventario en el formato XML

REIVINDICACIONES

- 5 1. Método para identificar de forma dinámica los componentes hardware y/o software (204, 206, 208, 210) de al menos un elemento de un clúster (300) que comprende varios elementos (305, 310) y automatizar las operaciones de gestión optimizada de dicho clúster, perteneciendo dicho al menos un elemento a dichos varios elementos y teniendo cada elemento de dichos varios elementos capacidades de procesamiento de datos, caracterizándose este método por comprender las siguientes etapas,
- en dicho al menos un elemento (305), llamado al menos un primer elemento, detección (425) de un evento y, en respuesta a dicha detección,
- 10 • identificación (405, 410) de al menos un componente de hardware y/o software de dicho al menos un primer elemento,
- establecimiento (415) de al menos un inventario local, comprendiendo dicho al menos un inventario local al menos una característica de al menos un componente de hardware y/o software de dicho al menos un primer elemento; y,
- 15 • transmisión (420) de dicho al menos un inventario local a un elemento de dichos varios elementos de dicho clúster, distinto de dicho al menos un primer elemento, llamado segundo elemento;
- en dicho segundo elemento,
- recepción (505) de dicho al menos inventario;
- establecimiento (510) de un Inventario parcial o global que comprenda al menos dicha al menos una característica de dicho al menos un componente de hardware y/o software de dicho al menos un primer elemento; y,
- 20 • inicio de una operación de gestión de dicho clúster de acuerdo con dicho método de inventario parcial o global; método caracterizado por dicha operación de gestión de dicho clúster que comprende una optimización de la distribución de la carga entre varios componentes de hardware de elementos de dichos varios elementos.
- 25 2. Método de acuerdo con la reivindicación 1 que comprende, además, en dicho segundo elemento, una etapa de envío de una solicitud de inventario, comprendiendo dicha etapa de detección de un evento una detección de una recepción de dicha solicitud,
3. Método de acuerdo con la reivindicación 1 de acuerdo con el cual dicho evento detectado se genera en dicho elemento al menos en un primer elemento.
- 30 4. Método de acuerdo con una cualquiera de las reivindicaciones anteriores de acuerdo con el cual dicho al menos un inventario local comprende un identificador de dicho al menos un componente de hardware y/o software de dicho al menos un primer elemento.
5. Método de acuerdo con una cualquiera de las reivindicaciones anteriores de acuerdo con el cual se repiten dichas etapas de recepción de dicho al menos un inventario y de establecimiento de un inventario parcial o global para todos los elementos de dichos varios elementos de un tipo predeterminado.
- 35 6. Método de acuerdo con una cualquiera de las reivindicaciones anteriores de acuerdo con el cual se establece al menos un Inventario local de un elemento de dicho clúster que no tiene capacidad de procesamiento de datos por dicho segundo elemento, utilizándose al menos un dato de dicho inventario local de dicho elemento que no tiene capacidad de procesamiento de datos para el establecimiento de dicho Inventario parcial o global.
- 40 7. Método de acuerdo con una cualquiera de las reivindicaciones anteriores de acuerdo con el cual dichos elementos son nodos.
8. Programa de ordenador que comprende instrucciones adaptadas para la implementación de cada una de las etapas del método de acuerdo con una cualquiera de las reivindicaciones anteriores cuando dicho programa se ejecuta en un ordenador.

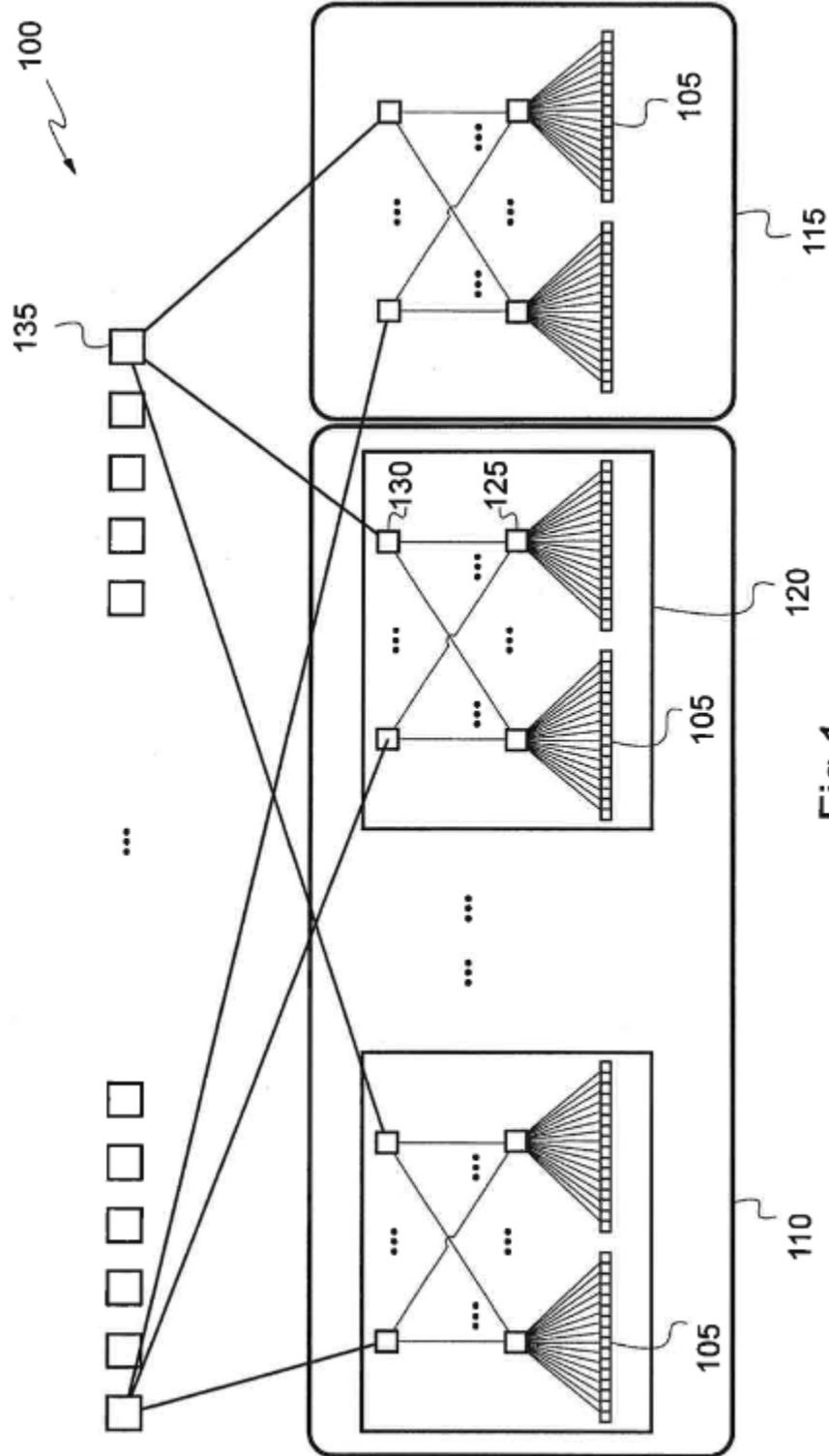
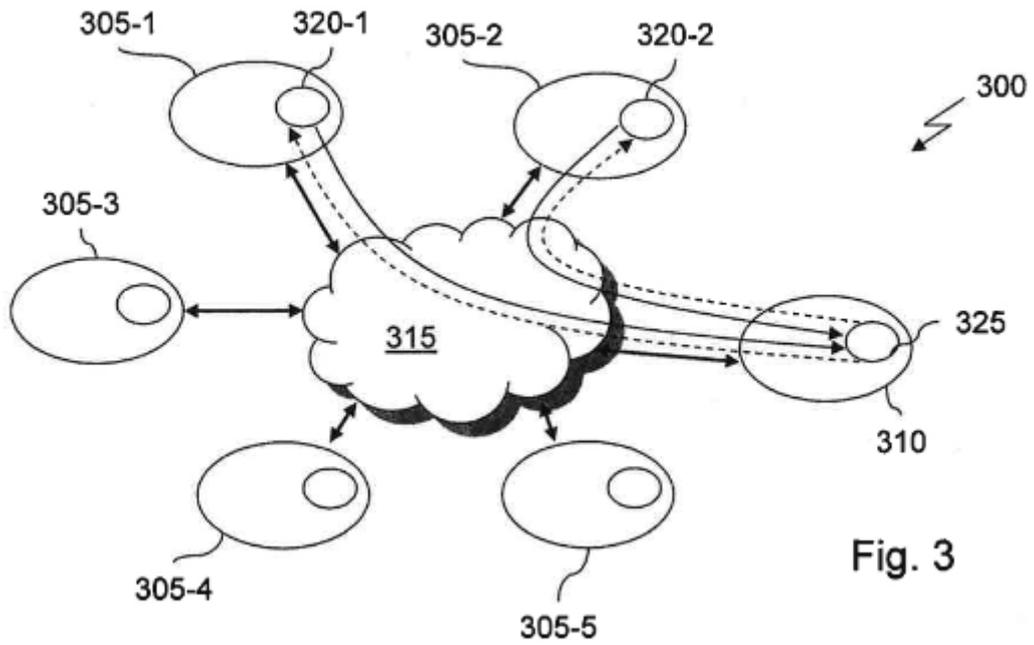
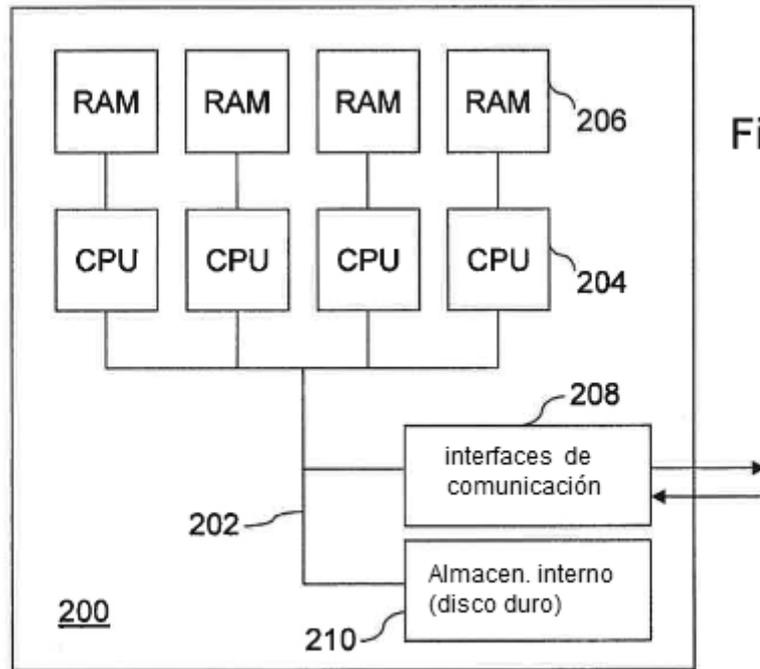
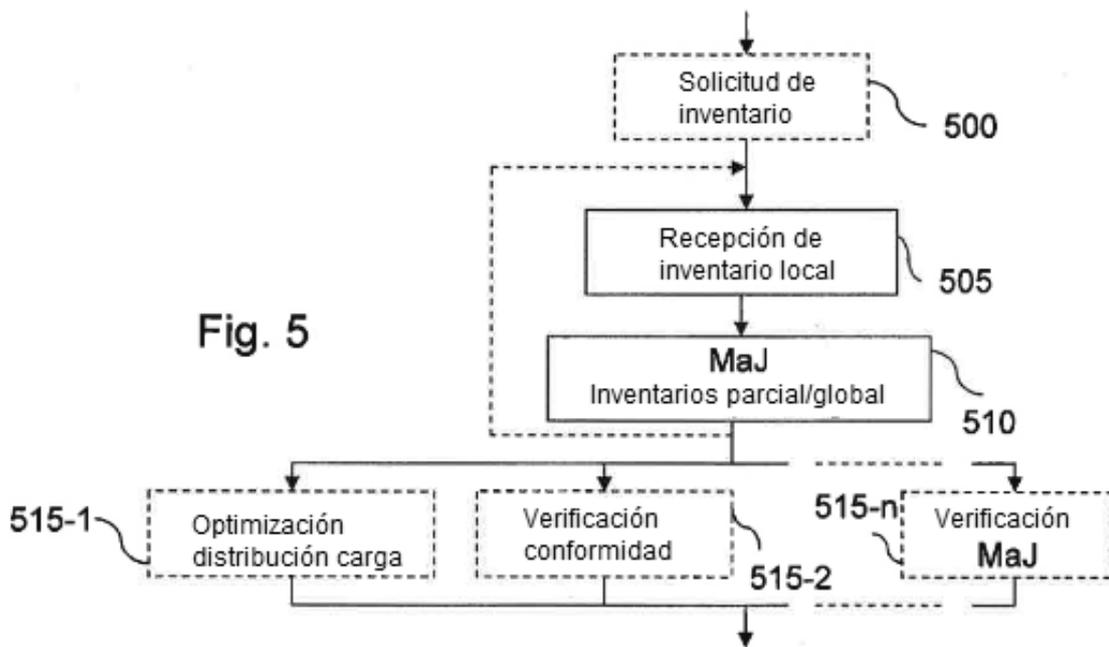
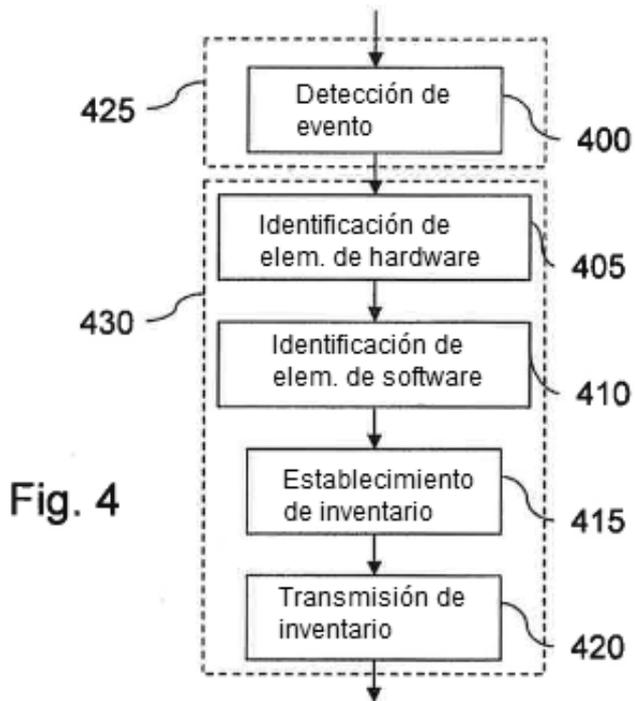


Fig.1





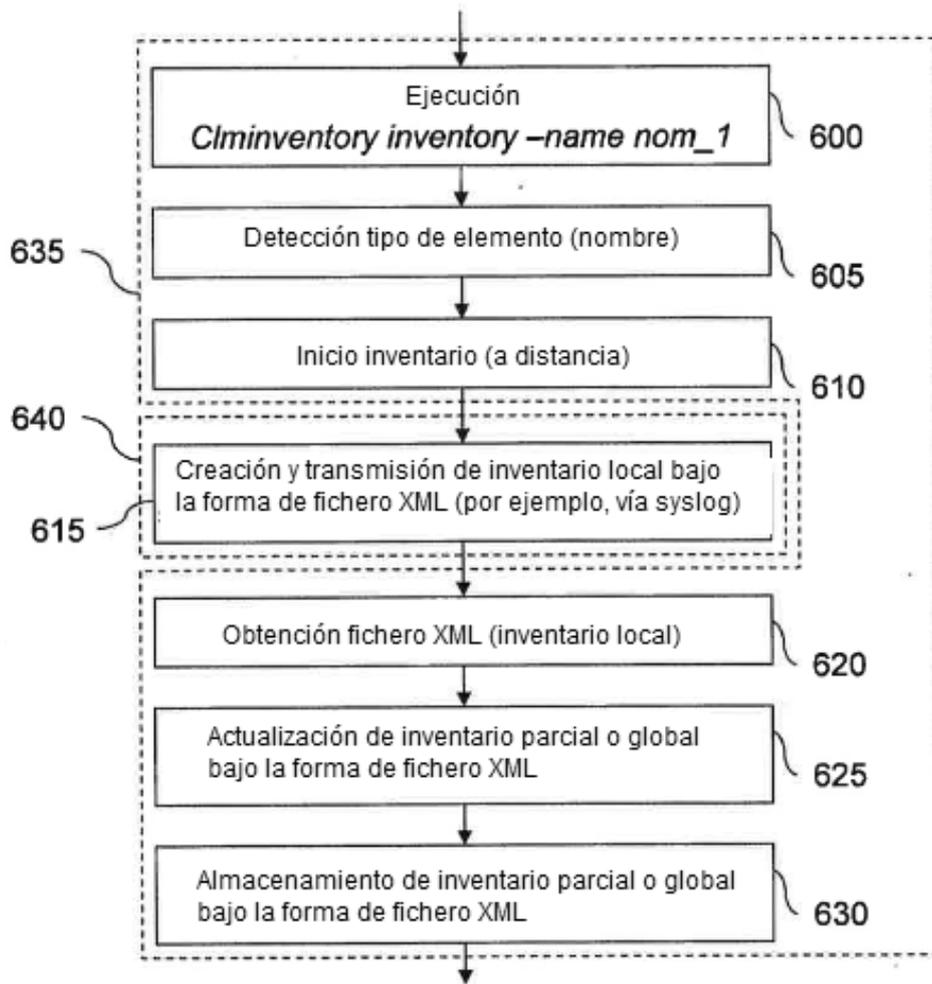


Fig. 6