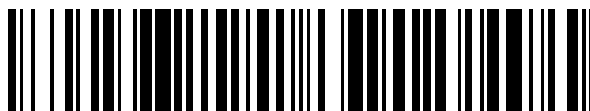


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 729 631**

51 Int. Cl.:

H04L 29/08 (2006.01)

H04L 12/28 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **09.12.2015 PCT/FR2015/053398**

87 Fecha y número de publicación internacional: **07.07.2016 WO16108001**

96 Fecha de presentación y número de la solicitud europea: **09.12.2015 E 15823632 (3)**

97 Fecha y número de publicación de la concesión europea: **20.02.2019 EP 3241308**

54 Título: **Caja de interconexión de equipos de usuario**

30 Prioridad:

31.12.2014 FR 1463500

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

05.11.2019

73 Titular/es:

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

72 Inventor/es:

**JENNEQUIN, MALO y
PELLETIER, BENOIT**

74 Agente/Representante:

ISERN JARA, Jorge

ES 2 729 631 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

Caja de interconexión de equipos de usuario

5 La invención se refiere a un método de procesamiento de datos relacionados con equipos de usuario, en un sistema de vendedores múltiples y servicios multiservicios, que permite la gestión de datos de un conjunto de objetos o equipos conectados a una red.

10 Con la aparición del Internet de las cosas, cada vez se ofrecen más servicios a particulares y profesionales, tales servicios relacionados con hogares u oficinas, comprendiendo los objetos conectados. A modo de ejemplo, en un contexto de domótica, es posible a través de equipos que comprenden sensores y accionadores, desplegados en una casa y conectado a una red, proponer servicios que aseguran:

- 15 - comodidad doméstica, por ejemplo, permitiendo
 - o automatizar y pilotar equipos a partir de una estación fija, o a distancia, por ejemplo, a través de un control remoto, un ordenador portátil, un teléfono;
 - o automatizar acciones gracias a la geolocalización, por ejemplo, abrir un portal, encender la calefacción, gestionar la iluminación, activar una alarma;
- 20 - ahorro de energía, permitiendo
 - o evitar el desperdicio eliminando gastos innecesarios;
 - o asegurar un nivel óptimo de comodidad;
 - 25 o controlar el consumo de electricidad optimizando, por ejemplo, la calefacción, la iluminación o la producción de agua caliente;
- una protección del hogar mediante sistemas que permiten
 - 30 o supervisar a las personas frágiles, ancianos o discapacitados;
 - o supervisar a los niños con el fin de evitar cualquier riesgo de accidente;
 - o detectar intrusiones, caídas, fugas de agua, y desencadenar una alarma si es necesario.

35 El estado de la técnica CN102946425 divulga un marco de integración de servicios de plataforma de comunicación uniforme basado en OSGi (Open Service Gateway Initiative). El marco de integración de servicio de la plataforma de comunicación uniforme comprende tres capas, a saber, una capa de servicio de terminal de cliente, una capa de interacción interna y un contenedor de aplicación, en donde la capa de servicio del terminal de cliente sirve como entrada de acceso de un terminal de cliente, proporciona una interfaz de complemento y se utiliza para realizar la operación de interacción de terminal de cliente y un servicio de integración de terceros en una capa de acceso; la capa de interacción interna crea un canal de mensajes para la capa de terminal del cliente y el contenedor de aplicación y se utiliza para simplificar la gestión de plataforma y equilibrar la carga de plataforma; y el contenedor de aplicación proporciona una interfaz uniforme para el servicio de integración de terceros, transmite un mensaje de gestión interno e interactúa con un sistema de servicio de integración de terceros.

45 Sin embargo, los sistemas existentes que proponen estos servicios presentan diversos inconvenientes o limitaciones:

- los actores del Internet de las cosas son numerosos y variados, a modo de ejemplo: fabricantes, editores, distribuidores, operadores de telecomunicaciones, empresas de energía, aseguradoras. Cada uno de estos actores generalmente construye una oferta basada en un número de equipos, de sensores y/o de accionadores seleccionados de un (otro) proveedor determinado. Por lo tanto, esto requiere un trabajo colaborativo estrecho entre el actor en cuestión y los equipos de investigación y desarrollo del proveedor determinado, en vista de proponer una solución de integración de productos. Ciertos actores del Internet de las cosas proponen, en cuanto a ellos, una amplia variedad de equipos conectados, sin dejar de ser su propio desarrollador de servicios de usuario. Aunque se facilita la colaboración con los equipos de investigación y desarrollo, estos actores siguen sin poder integrar equipos de distintos proveedores en sus plataformas, quienes siguen siendo propietarios. Más generalmente, el conjunto de las soluciones integradas en una misma plataforma son actualmente soluciones que integran a priori equipos de vendedores determinados. Las posibilidades de integraciones a posteriori de equipos, sensores, accionadores, quedan en cuanto a ellas muy limitadas. Estas posibilidades dependen además de las características de las soluciones de los proveedores llevadas a evolucionar, por ejemplo, dependientes de los protocolos propietarios empleados y de sus evoluciones. En consecuencia, estas soluciones carecen de adaptabilidad;
- los equipos, sensores o accionadores conectados a sistemas que proponen servicios, generalmente suelen ser propietarios, es decir, se refieren a un mismo vendedor/proveedor. Una caja de comunicación, comúnmente designado bajo la apelación "box", asegura entonces la interconexión de los diferentes equipos, sensores o accionadores con plataformas de servicios o redes externas. De este modo, para diferentes servicios

proporcionados por diferentes vendedores, el usuario a menudo está obligado a utilizar una caja de comunicación por vendedor, lo que implica problemas ergonómicos en su hogar;

- los equipos propietarios conectado a una misma caja de comunicación son heterogéneos y comúnmente se basan en diferentes protocolos de comunicación no estandarizados. En consecuencia, en el marco de un contexto de vendedores múltiples, si un equipo cambia, su protocolo de comunicación también cambia con frecuencia. Esto implica una dependencia de la caja con respecto a los equipos, teniendo la caja entonces que soportar el nuevo protocolo del equipo, es decir, garantizar los accesos e intercambios con este nuevo equipo;
- se pueden traer varios vendedores, con sus acuerdos, para proponer distintos servicios en una misma caja. Los servicios de vendedores múltiples se ejecutan por lo tanto en paralelo en la misma caja. Esto no implica, sin embargo, compartir el conjunto de los datos o recursos asociados con estos servicios, los vendedores, por ejemplo, que no desean compartir datos confidenciales. Por otro lado, la ejecución de un servicio debe poder desarrollarse independientemente de otros servicios ejecutados en paralelo, por ejemplo, independientemente de los recursos de memoria utilizados por otros servicios de usuario. De este modo, el mal funcionamiento de un servicio no debe afectar los recursos de los servicios vecinos. Por lo tanto, existen necesidades de servicios de "partición" ejecutados en paralelo en la misma caja, particularmente para servicios de usuario procedentes de diferentes vendedores;
- proveedores a cargo de desarrollar servicios de usuario pueden querer acceder a un conjunto de datos confiables con vistas, por ejemplo, a concebir servicios domiciliarios sin ser necesariamente un proveedor de equipos. En un contexto de vendedores múltiples y servicios multiservicios, en particular, de desplegar servicios en una plataforma, garantizando este último tanto el acceso a datos confiables procedentes de equipos de vendedores distintos, y asegurar la comunicación, así como el pilotaje de estos equipos. Generalmente, los proveedores de equipos proponen equipos con posibilidades de pilotarlos a distancia, acceder a sus datos, colocar escenarios, y soportar protocolos (por ejemplo, KNX, Zigbee, Z-Wave). Sin embargo, los servicios propuestos por estos proveedores se limitan al único equipo, o concierne a un solo proveedor/vendedor. Por lo tanto, el proveedor de servicios debe ser un proveedor de equipos y no dispone de acceso a los datos generados por los equipos multivendedores. El proveedor de equipos, por lo tanto, dispone de datos de alcance muy reducido, limitando fuertemente la información útil para la elaboración de un servicio de usuario. En particular, actualmente no existe una plataforma que permita agrupar un conjunto de datos de vendedores múltiples y multiservicios, en vista a garantizar el acceso centralizado a estos datos, un pilotaje de equipos multivendedores y la colocación de escenarios multiservicios (por ejemplo: desencadenar el calentamiento por debajo de una temperatura umbral). La realización de tal plataforma implica problemas técnicos relacionados con la seguridad de los datos, su acceso, así como la gestión de los intercambios entre tal plataforma y los equipos.

La presente invención tiene como objeto responder a los problemas mencionados anteriormente.

Para tal efecto, se propone, según un primer aspecto, una caja de comunicación, conectada a un primer equipo y a un segundo equipo, comunicándose el primer equipo con la caja a través de un primer protocolo, comunicándose el segundo equipo con la caja a través de un segundo protocolo, alojando la caja de comunicación para cada equipo, respectivamente, al menos un primer servicio y un segundo servicio, comprendiendo la caja de comunicación una plataforma de software OSGi, estando esta plataforma configurada para aislar cada servicio de cada equipo en un contenedor estanco e implementar una estructura de acceso unificado que permite el acceso a los equipos.

En esta caja de comunicación, la estructura de acceso unificado comprende una interfaz de programación aplicativa configurada para permitir el acceso del primer servicio al primer equipo y del segundo servicio al segundo equipo, estando el acceso a cada equipo realizado por medio de un adaptador de elemento y de un adaptador protocolario, implementando estos adaptadores la interfaz de programación aplicativa y siendo desplegados en el contenedor estanco de su respectivo servicio.

Ventajosamente, en esta caja de comunicación, la plataforma de software está configurada para realizar además un contenedor compartido, proponiendo este contenedor compartido interfaces Java que constituyen la interfaz de programación aplicativa, adaptadores protocolarios compartidos por diferentes servicios, servicios comunes de usuarios.

Ventajosamente, en esta caja de comunicación, la interfaz de programación aplicativa comprende una interfaz de publicación/suscripción configurada para permitir

- la recepción asíncrona de eventos o de datos procedentes de un equipo;
- la publicación asíncrona de datos o de eventos hacia un equipo.

Ventajosamente, en esta caja de comunicación, la interfaz de programación aplicativa comprende una interfaz de petición configurada para permitir según un modo síncrono, la recuperación, la lectura y la escritura de datos procedentes o con destino a un equipo.

Ventajosamente, en esta caja de comunicación, la estructura de acceso unificado está configurada para simular un modo de acceso asíncrono Push para un equipo, cuando dicho equipo funciona según un modo síncrono Pull, siendo

el modo de acceso asíncrono simulado por la estructura de acceso unificado recuperando de manera periódica un último dato leído a través de la interfaz de petición y devolviendo este valor hacia la interfaz de publicación/suscripción.

5 Ventajosamente, en esta caja de comunicación, la estructura de acceso unificado está configurada para simular un modo de acceso síncrono Pull para un equipo, cuando dicho equipo funciona según un modo síncrono Push, siendo el modo de acceso síncrono simulado mediante la estructura de acceso unificado transmitiendo cada último dato insertado en la interfaz de publicación/suscripción a la interfaz de petición.

10 Se propone, según un segundo aspecto, un producto de programa de ordenador implementado en un soporte de memoria, susceptible de implementarse en el seno de una unidad de procesamiento informático y que comprende instrucciones para la implementación de la caja de comunicación resumida anteriormente.

15 Otros objetos y ventajas de la invención se harán evidentes a la luz de la descripción de los modos de realización, realizada a continuación con referencia a los dibujos adjuntos en donde:

- la figura 1 es un diagrama que representa un sistema de gestión de datos de vendedores múltiples y multiservicios según un modo de realización;
- la figura 2 es un esquema relacional de modelado de datos según un modo de realización;
- 20 - la figura 3 es un diagrama de clase de un nodo del esquema relacional de modelado de datos según un modo de realización;
- la figura 4 es un diagrama de clase de otro nodo del esquema relacional de modelado de datos según un modo de realización;
- la figura 5 es un diagrama topológico que describe un nodo del esquema relacional de modelado de datos según un modo de realización;
- 25 - la figura 6 es un diagrama topológico que describe un nodo del esquema relacional de modelado de datos según un modo de realización;
- la figura 7 es un subconjunto topológico que describe un nodo del esquema relacional de modelado de datos según un modo de realización;
- la figura 8 es un diagrama que ilustra los elementos relacionados con un servicio según un modo de realización;
- 30 - la figura 9 es un diagrama que ilustra la relación entre diferentes servicios según un modo de realización;
- la figura 10 es una figura que ilustra un gestor de acciones según un modo de realización;
- la figura 11 ilustra un flujo de datos relacionado con un consumidor de acción según un modo de realización;
- la figura 12 ilustra el establecimiento de un túnel, para un mecanismo de comunicación entre una caja de comunicación y un proxy según un modo de realización;
- 35 - la figura 13 ilustra el manejo a distancia de la caja de comunicación para el mecanismo de comunicación con un proxy según un modo de realización;
- la figura 14 ilustra la transmisión de una solicitud de acción hacia un accionador asociado con un equipo para el mecanismo de comunicación con un proxy según un modo de realización;
- la figura 15 ilustra el rastreo de una medición de un sensor asociado con un equipo para el mecanismo de comunicación con un proxy según un modo de realización;
- 40 - la figura 16 ilustra el establecimiento de una conexión entre una plataforma central y una caja de comunicación para un mecanismo de comunicación que implementa el protocolo UPnP según un modo de realización;
- la figura 17 ilustra el manejo a distancia de la caja de comunicación para el mecanismo de comunicación que implementa el protocolo UPnP según un modo de realización;
- 45 - la figura 18 ilustra la transmisión de una solicitud de acción hacia un accionador asociado con un equipo para el mecanismo de comunicación que implementa el protocolo UPnP según un modo de realización;
- la figura 19 ilustra el rastreo de una medición de un sensor asociado con un equipo para el mecanismo de comunicación que implementa el protocolo UPnP según un modo de realización;
- la figura 20 ilustra el establecimiento de una conexión entre una plataforma central y una caja de comunicación para un mecanismo de comunicación que implementa el protocolo XMPP según un modo de realización;
- 50 - la figura 21 ilustra el manejo a distancia de la caja de comunicación para el mecanismo de comunicación que implementa el protocolo XMPP según un modo de realización;
- la figura 22 ilustra la transmisión de una solicitud de acción hacia un accionador asociado con un equipo para el mecanismo de comunicación que implementa el protocolo XMPP según un modo de realización;
- 55 - la figura 23 ilustra el rastreo de una medición de un sensor asociado con un equipo para el mecanismo de comunicación que implementa el protocolo XMPP según un modo de realización;
- la figura 24 ilustra el establecimiento de una conexión entre una plataforma central y una caja de comunicación para un mecanismo de comunicación que implementa el protocolo STUN según un modo de realización;
- la figura 25 ilustra el manejo a distancia de la caja de comunicación para el mecanismo de comunicación que implementa el protocolo STUN según un modo de realización;
- 60 - la figura 26 ilustra la transmisión de una solicitud de acción hacia un accionador asociado con un equipo para el mecanismo de comunicación que implementa el protocolo STUN según un modo de realización;
- la figura 27 ilustra el funcionamiento de una plataforma de software según un modo de realización;
- la figura 28 ilustra un ejemplo de distribución de los elementos de la plataforma de software según un modo de realización;
- 65 - la figura 29 ilustra la realización de la estructura de acceso unificado según un modo de realización;

- la figura 30 ilustra un funcionamiento de la estructura de acceso unificado con un equipo que funciona según un modo síncrono según un modo de realización;
- la figura 31 ilustra un funcionamiento de la estructura de acceso unificado con un equipo que funciona según un modo asíncrono según un modo de realización.

5 La figura 1 es un diagrama que representa un sistema 100 de gestión de datos de vendedores múltiples y multiservicios según un modo de realización. El sistema 100 de gestión de datos está interconectado con las siguientes entidades:

- 10 - uno o varios equipos 1 dispuestos localmente en un lugar o un entorno físico predefinido, por ejemplo, en un entorno interior tal como un hogar o una oficina. Los equipos 1 son proporcionados (flecha 3) por los proveedores 2 de equipos, es decir, fabricantes tecnológicos. Ventajosamente, los equipos 1 comprenden sensores y permiten rastrear a una plataforma 4 central de datos, a través de una caja 5 de comunicación de tipo "box", datos cuantificables y/o de cambio de estados asociados con las mediciones o informes de estado de dichos sensores (por ejemplo: medición de la temperatura, posición de un conmutador asociado a un sistema de iluminación). Para hacer esto, los equipos 1 están conectados a la caja 5 de comunicación por medio de una red cableada o inalámbrica. La caja 5 de comunicación intercambia, en cuanto a ella, de manera local o remota, datos con la plataforma 4 central de datos, por medio de otra red, por ejemplo, Internet. Además, los equipos 1 permiten efectuar acciones físicas (por ejemplo: encender la calefacción) a través de uno o varios accionadores. Típicamente, un accionador efectúa una acción después de la recepción de una notificación de acción comunicada desde la caja 5 de comunicación. Los datos intercambiados entre cada equipo 1 y la caja 5 de comunicación, por ejemplo, la retroalimentación de datos medidas por los sensores o las notificaciones de acción con destino a un accionador de un equipo 1, se comunican según uno o varios protocolos 6 simbolizados por una doble flecha. A modo de ejemplo, un protocolo 6 asociado con un equipo puede ser un protocolo KNX, Zigbee, Wifi o Z-wave;
- 15 - proveedores 7 de servicios de usuario, típicamente actores del mundo de Internet, elaborando y proporcionando al usuario servicios de usuario dirigidos a un campo particular, por ejemplo, servicios relacionados con la seguridad, con la comodidad, o con el bienestar. Ventajosamente, estos proveedores 7 de servicios de usuario disponen de lógica empresarial completa y realizan servicios de usuario basándose en datos y servicios básicos, proporcionados previamente a la plataforma 4 central de datos. Ventajosamente, estos proveedores 7 de servicios de usuario interactúan (flecha 8) con la plataforma 4 central de datos según un modo orientado a la arquitectura de servicio, comúnmente conocido como SOA (acrónimo inglés de "*Service Oriented Architecture*");
- 20 - proveedores 9 de datos del mundo de internet, proporcionando, por ejemplo, datos meteorológicos, actualidades o, incluso, procedentes de redes sociales, que permiten rastrear datos cuantificables (por ejemplo: mediciones meteorológicas) o datos de cambio de estado. Ventajosamente, estos proveedores 9 de datos, interactúan (flecha 10) también con la plataforma 4 central de datos según un modo orientado a la arquitectura de servicio SOA;
- 25 - proveedores 11 de servicios básicos, actores del mundo de Internet, proporcionando servicios básicos a la plataforma 4 central de datos (por ejemplo: implementación de un servicio de notificación), siendo estos servicios utilizados por los proveedores 7 de servicios de usuario, como servicios "básicos" que permiten realizar los servicios de usuario. Ventajosamente, estos proveedores 11 de servicios básicos, interactúan (flecha 12) también con la plataforma 4 central de datos según un modo orientado a la arquitectura de servicio SOA.

40 Por otro lado, cualquier proveedor puede posiblemente desempeñar una pluralidad de papeles. Por ejemplo, un proveedor 9 de datos también puede ser un proveedor 2 de equipo y un proveedor 7 de servicios de usuario.

45 Ventajosamente, el sistema 100 de gestión de datos tiene en cuenta varias dimensiones:

- una dimensión multiservicio, es decir, integra y propone una pluralidad de tipos de servicios de usuario, por ejemplo, domótica y e-salud;
- una dimensión de vendedores múltiples, los equipos 1 soportados por el sistema 100 pueden provenir de diferentes fabricantes, y comprenden características específicas de cada uno de estos fabricantes;
- 50 - una dimensión multinivel, los datos y los servicios básicos pueden proporcionarse:
 - o al nivel de los equipos 1 por al menos una caja 5 de comunicación;
 - o al nivel del sistema 100 de gestión de datos. Por ejemplo, el sistema 100 realiza un análisis de los datos proporcionados por los equipos 1 o los proveedores de datos 9, siendo estos datos posiblemente abstractos previamente para preservar su confidencialidad. Los resultados del análisis luego permiten que el sistema 100 haga disponibles las correlaciones, tendencias, predicciones sobre estos datos;
 - o al nivel de los proveedores 7 de servicios de usuario, por la provisión a través del sistema 100 de gestión de datos (posiblemente abstractos previamente), de tendencias/correlaciones relacionadas con estos datos, y servicios básicos para realizar los servicios de usuario.

60 Para hacer esto, el sistema 100 de gestión de datos comprende las siguientes entidades, cuyo funcionamiento se detallará más posteriormente:

- 65 - la caja 5 de comunicación de tipo "box" pudiendo, por ejemplo, ser desplegada en el hogar del usuario. Esta caja permite interconectar uno o varios equipos 1 a través de un enlace cableado o inalámbrico según un protocolo

predeterminado en función del constructor (por ejemplo: Wifi) e intercambiar datos con la plataforma 4 central de datos. Ventajosamente, la caja 5 de comunicación permite

- 5
 - la gestión de equipos 1 de vendedores múltiples, es decir, soporta y gestiona una pluralidad de protocolos, así como las identificaciones y aislamientos de los datos/recursos de cada uno de los servicios ejecutados en paralelo en esta caja;
 - garantizar una comunicación segura y confiable con la plataforma 4 de datos central;
 - proporcionar a la plataforma 4 central de datos de los datos explotables de vendedores múltiples procedentes de los sensores de los equipos 1;
- 10
 - recibir notificaciones procedentes de la plataforma 4 central de datos, refiriéndose, a título de ejemplo a un manejo a distancia para la administración o el envío de acción(es) para ser ejecutadas por uno o varios equipos 1 a los que está interconectado;
- 15
 - la plataforma 4 central de datos concentrando y federando todos los datos que provienen de las cajas 5 de comunicación desplegados en diversos lugares (por ejemplo: hogares, oficinas), los datos procedentes de los proveedores de datos, así como los servicios básicos necesarios para la elaboración de nuevos servicios de usuario. Ventajosamente, la plataforma 4 central de datos está configurada para:
 - 20
 - asegurar un almacenamiento centralizado y organizado de los datos, los datos se pueden organizar, por ejemplo, en función
 - 25
 - de su estructura: datos brutos procedentes de los sensores, consolidados, agregados y/o abstractos. A modo de ejemplo, por razones de confidencialidad, los datos brutos enviados desde un sensor pueden ser formateados/estructurados por la plataforma 4 central de datos según un formato de pivote, que permite abstraer estos datos, tal como un formato de tipo XML;
 - de su origen: datos procedentes de los sensores, enviados por los equipos 1 o que provienen de un proveedor 9 de datos externo;
 - de su derecho de acceso y de su identificación: un proveedor 7 de servicios de usuario puede tener acceso limitado a cierto tipo de datos;
 - 30
 - asegurar la segregación de los datos y de los servicios básicos. Ventajosamente, esto permite, en un entorno de vendedores múltiples, hacer que un conjunto de proveedores 7 de servicios de usuario se beneficien de la naturaleza rica y múltiple de los datos, y esto de manera controlada por
 - 35
 - una identificación y una autenticación de los consumidores de estos datos, aquí, los proveedores 7 de servicios de usuario;
 - una restricción de la accesibilidad a los datos/servicios, de acuerdo con los derechos de los consumidores de estos datos;
 - un seguimiento del uso de los consumos de datos/servicios para cada uno de estos consumidores;
 - 40
 - asegurar la confidencialidad de los datos, por ejemplo, una anonimización de los datos a través de un método de abstracción (por ejemplo: abstracción de datos según un mismo formato de pivote);
 - federar un conjunto de servicios básicos provistos por los proveedores 11 de servicios básicos, para disponer de un ecosistema global, coherente y controlado;
 - 45
 - proporcionar los servicios básicos a los proveedores 7 de servicios de usuario para la realización de dichos servicios, a modo de ejemplo, unos servicios de tipo flujo de trabajo (comúnmente designado con el anglicismo "*Workflow*"), o unos servicios de procesamiento de eventos complejos CEP (acrónimo inglés de "*Complex Event Processing*");
 - garantizar una comunicación segura y confiable con las diferentes cajas 5 de comunicación;
 - 50
 - proporcionar datos utilizables a un módulo 13 analítico de megadatos llamado "*Big Data Analytics*";
 - 55
 - un módulo 13 analítico de megadatos (llamado "*Big Data Analytics*") configurado para analizar los datos de la plataforma 4 central de datos con la que está interconectado. El módulo 13 analítico de megadatos permite, en particular, establecer, después poner a disposición de los proveedores 7 de servicios de usuario, correlaciones, tendencias, y/o predicciones de los datos registrados en la plataforma 4 central de datos, generando informes de análisis (servicios de "*reporting*").

Ventajosamente, los datos procedentes de la plataforma 4 central de datos son identificados, luego clasificados en función de a una topología de datos preestablecida. A modo de ejemplo, las figuras 2 a 7 ilustran un modo de realización de topología de datos preestablecida, según un lenguaje de modelado unificado, llamado UML, acrónimo inglés de "*Unified Modeling Language*", esta topología se detalla posteriormente.

Según, diversos modos de realizaciones, una topología de datos se utiliza por la plataforma 4 central o el módulo 13 analítico de megadatos ("*Big Data Analytics*") para clasificar los datos registrados en la plataforma 4 central de datos. De este modo, en un modo de realización, una topología de datos que describe un modelo de clasificación de datos se graba previamente en la plataforma 4 central de datos, y se usa por este último para clasificar los datos. A modo

de ejemplo, la plataforma 4 central de datos está configurada para identificar la procedencia de un dato, por ejemplo, una medición procedente de un sensor mediante la lectura de sonido en cabeza, luego, en función su procedencia, lo clasifica según la topología grabada previamente. En otro modo de realización, una topología preestablecida está grabada previamente en el módulo 13 analítico de megadatos. Este último se configura entonces para ordenar/estructurar los datos de la plataforma 4 central de datos, en una base de datos de esta plataforma (por ejemplo: una base NoSQL). En otro modo de realización, las funciones de la plataforma 4 central de datos y del módulo 13 analítico de megadatos se realizan mediante el mismo y único módulo (no representado).

Ventajosamente, los datos registrados/manipulados en la plataforma 4 central de datos se presentan en forma de objetos empresariales, es decir, de estructuras de datos relativos, a modo de ejemplo, a equipos 1, objetos, ubicaciones o entidades que interactúan e intercambian datos con el sistema 100 de gestión de datos. En un modo de realización, en un contexto de domótica, el sistema 100 de gestión de datos comprende los objetos empresariales siguientes:

- "*Hogar*": cada hogar está asociado con un identificador único en la plataforma, identificado por un tipo (por ejemplo: particular, empresa) y asociado a sus propias características (cajas, equipos, sensores, accionadores);
- "*Box*": este objeto designa cualquier caja 5 de comunicación que encapsula servicios/funcionalidades y que soporta diferentes protocolos de comunicación. En cada hogar, al menos una caja 5 de comunicación se asocia, es decir, un objeto "*Box*", comprendiendo este objeto un identificador único;
- "*Equipo*", cada equipo 1 está instalado en un hogar, está conectado únicamente a una sola caja 5 de comunicación, y comprende un identificador único;
- "*Sensor*", un sensor permite medir una magnitud física o identificar un cambio y está asociado con una sola pieza del equipo 1;
- "*Accionador*", un accionador permite desencadenar una acción después de un evento y está asociado con un solo equipo 1.

Por otro lado, con el fin de clasificar y discriminar los datos registrados en la plataforma 4 central de datos, como se expuso anteriormente, se implementa una topología de datos preestablecida.

Un ejemplo de topología se representa en la figura 2, en forma de una arborescencia compuesta de nodos y paradas, de acuerdo con el estándar UML. Esta arborescencia comprende, aquí, un nodo raíz "*Raíz*", refiriéndose los otros nodos a las entidades de la plataforma. Más particularmente, cada nodo de este modelo está asociado con un diagrama de clase, que se describirá posteriormente. Ventajosamente, la relación entre cada entidad está representada por una parada (enlace) y una cardinalidad. Las cardinalidades de la figura 2 se proponen aquí a modo de ejemplo puramente ilustrativo. Por ejemplo, se puede ver en esta figura que la cardinalidad de la parada que conecta el nodo "*Raíz*" al nodo "*Entidad*" es aquí de "1.*", lo que significa que el nodo "*Raíz*" consta de una o varias instancias de clase descritas en la clase "*Entidad*". Las flechas de esta figura, ilustran las dependencias entre los diferentes nodos, siendo los nodos de este modelo son los siguientes:

- nodo "*Entidad*": este nodo designa un lugar, una persona, un animal;
- nodo "*Datos*": este nodo representa los datos de la plataforma 4 central de datos. Las fuentes de estos datos son los sensores, los datos agregados de un sensor, los datos procedentes de los proveedores 9 de datos (por ejemplo: datos ambientales, datos de redes sociales, o más generalmente todos los datos de Internet);
- nodo "*Infraestructura*": este nodo describe los diferentes objetos relacionados con el entorno doméstico, tales como objetos "*Box*", "*Equipo*", "*Sensor*", "*Accionador*" mencionados anteriormente. Se distingue, por otro lado, dos tipos de infraestructuras: gestionadas o no gestionadas. Se designa, aquí, por infraestructura gestionada, cualquier objeto asociado con una caja 5 de comunicación, y más generalmente cualquier objeto conectado, pudiendo intercambiar datos con la plataforma 4 central de datos, y pudiendo ser supervisado por el sistema 100 de gestión de datos. De este modo, una infraestructura gestionada está inventariada en el sistema 100 de gestión de datos y es configurable bajo la responsabilidad de este sistema.
- Por oposición, se entiende por infraestructura no gestionada, cualquier objeto o equipo 1 que no puede ser supervisado por el sistema 100 de gestión de datos. Sin embargo, tal infraestructura sigue siendo inventariada, es decir, conocido por el sistema 100 de gestión de datos;
- nodo "*Actor*": tal nodo designa a los actores de la plataforma. Un actor puede ser un usuario de la plataforma, un proveedor 7 de servicios de usuario, un proveedor 2 de equipos o incluso un consumidor de servicios. Ventajosamente, el papel de cada actor permite definir las autorizaciones de acceso a los datos y servicios;
- nodo "*Proveedor de tecnología de la información*", comúnmente conocido por el anglicismo "*Proveedor de IT*": este nodo describe el conjunto de los proveedores de la plataforma 4 central de datos: proveedores 7 de servicios de usuario y/o proveedores 11 de servicios básicos;
- nodo "*Cuenta de cliente*": este nodo describe la cuenta de un cliente que se ha suscrito a un conjunto de servicios ofrecidos por la plataforma 4 de datos central, o cualquier usuario que disponga de los equipos 1 que soportan los servicios propuestos por la plataforma 4 central de datos.

La figura 3 describe después el diagrama de clase asociado con el nodo "*Datos*" descrito anteriormente, es decir, las clases que heredan el nodo "*Datos*" que permiten clasificar los diferentes tipos de datos. Como se puede constatar en este esquema, un dato:

- puede constar de una unidad de medición, por ejemplo, referirse a una temperatura o una presión medida por un sensor. Además, una magnitud física puede estar asociada con una pluralidad de unidades de mediciones. A modo de ejemplo, para una medición de temperatura se pueden asociar unidades Celsius y Fahrenheit;
- 5 - un dato no puede constar de una medición, por ejemplo, cuando se trata de un estado (por ejemplo: estado abierto o cerrado de una puerta);
- un dato puede ser del tipo de medio, por ejemplo, ser texto, una imagen o vídeo.

Los datos pueden constar, además, de una pluralidad de fuentes

- 10 - "*Datos sensores*": estos datos son procedentes de sensores, y son, por ejemplo, datos brutos (no procesados) devueltos por los sensores, o un conjunto de datos agregados según una estructura predeterminada (por ejemplo: estructurados según un formato de pivote), relacionándose este conjunto con las mediciones de al menos un sensor;
- 15 - "*Datos ambientales*": estos son datos que no son proporcionados por el sistema 100. Estos datos se relacionan con un conjunto de datos agregados de una zona geográfica predeterminada (por ejemplo: región, ciudad) y pueden ser ejemplos de temperaturas, presiones o tasas de partículas;
- "*Datos de Internet*", llamado "*datos de la web*": estos datos pueden ser texto, imagen, vídeo o cualquier otro soporte multimedia propuesto por un proveedor 9 de datos.

La figura 4 ilustra un diagrama de clases que describe las instancias del nodo "*Entidad*". Con referencia a esta figura, en este modelo de topología, un dato que se refiere a la clase "*Entidad*" puede clasificarse en una de las instancias ("subclases") siguientes:

- 25 - clase "*Ser vivo*", comprendiendo como posibles instancias
 - o una clase "*Persona*": como ejemplo para los servicios de e-Salud, una persona puede estar equipada con sensores para medir su tensión, su temperatura y disponer de un dispositivo conectado (por ejemplo: mostrar) para la transmisión de dichas mediciones asociadas a los sensores;
 - 30 o una clase "*Animal*": un animal puede estar equipado, por ejemplo, con un collar o un chip electrónico que permita localizarlo;
- clase "*Lugar*": cualquier espacio equipado con sensores, por ejemplo, una empresa o un hogar;
- clase "*Objeto*": generalmente, cualquier objeto dispuesto en un espacio físico (por ejemplo: oficina, casa) utilizando uno o varios sensores para medir una magnitud física o un estado. A modo de ejemplo, en un hogar, un objeto puede ser un refrigerador equipado con un sensor de temperatura y de medios de emisión/recepción. Ventajosamente, un lugar puede comprender una pluralidad de objetos.

La figura 5 es un diagrama topológico que describe los diferentes tipos de lugar, así como sus composiciones. El sistema 100 tiene como objetivo gestionar el conjunto de los equipos 1 u objetos conectados desplegados en diferentes lugares, estando dichos equipos 1 u objetos dispuestos en zonas específicas y provistos de sensores. En este ejemplo, un lugar es o una empresa, o bien, un hogar. De este modo, en este ejemplo, unos datos relacionados con la clase "*Lugar*" pueden clasificarse en las instancias "*Hogar*" o "*Empresa*".

La clase "*Hogar*" comprende aquí, por ejemplo, las siguientes clases: "*Cocina*", "*Comedor*", "*Estancia*", "*Habitación*", "*Habitación de invitados*", "*Sótano*", "*Jardín*", "*Garaje*".

La clase "*Empresa*" puede ser implementada por las siguientes instancias: "*Bienvenida*", "*Sala de reunión*", "*Oficina*", "*Oficina de dirección*", "*Espacio abierto*", comúnmente denominado con el anglicismo "*Espacio abierto*", "*Local técnico*", "*Archivo*", "*Sanitario*".

Se entiende que el modelado de estos lugares es posteriormente extensible en función de las necesidades.

Ventajosamente, las características de la clase "*Sensor*" introducidas en la figura 2, permiten proporcionar datos a los proveedores 11 de servicios básicos para el desarrollo de servicios confiables. Comúnmente, un sensor puede ser descrito por las siguientes características:

- su alcance de medición: pudiendo los valores extremos ser medidos por el sensor;
- su resolución: la variación de magnitud medible más pequeña por el sensor;
- 60 - su sensibilidad: la variación de la señal de salida con respecto a la variación de la señal de entrada;
- su precisión: capacidad del sensor para dar una medición cercana al valor verdadero;
- su rapidez: tiempo de reacción del sensor;
- su protocolo: protocolo soportado, interfaz de comunicación utilizada.

La figura 6 ilustra un diagrama topológico que permite clasificar datos que se refieren a la clase "*Sensor*". En esta figura, la clase "*Sensor*" comprende, por ejemplo:

- una clase de "*Sensor de magnitud física*": este tipo de sensor permite mediciones de una magnitud física como un caudal de agua, de gas, un consumo de electricidad, un valor de temperatura, de presión, o incluso una tasa de humedad;
- una clase "*Sensor de estado*": este tipo de sensor permite identificar el estado de un objeto, por ejemplo, el estado de una puerta (cerrada o abierta). Se pueden citar como ejemplos, un sensor de presencia, o un sensor de imagen.

El nodo "Actor" de la figura 2 comprende, por ejemplo, el nodo "*Papel*", pudiendo un actor del sistema 100 acumular diferentes papeles.

La figura 7 ilustra el subconjunto topológico relacionado con esta última entidad. Ventajosamente, la clase "*Papel*" comprende las siguientes instancias:

- clase "*Proveedor de datos*": un proveedor 9 de datos proporciona datos que serán utilizados por los servicios proporcionados por la plataforma 4 de datos central;
- clase "*Proveedor de equipos*": un proveedor 2 de equipos proporciona objetos de hardware para el sistema 100, tales como sensores, equipos 1, accionadores;
- clase "*Proveedor de servicios*": un proveedor 7 de servicios de usuario proporciona y expone los servicios de usuario en la plataforma 4 central de datos;
- clase "*Consumidor de servicios*": este ejemplo designa a los actores que explotan los servicios expuestos por la plataforma 4 central de datos. A modo de ejemplo, un constructor de equipos electrodomésticos puede aprovechar los datos de la plataforma 4 central de datos para adaptar su oferta. Por otro lado, un proveedor 7 de servicios de usuario o un proveedor 2 de equipos puede utilizar los servicios expuestos por la plataforma 4 central de datos. En este caso, el actor es tanto un proveedor de servicios/de equipos como un consumidor de servicios;
- clase "*Usuario*": esta instancia designa una persona, situada, por ejemplo, en un hogar o una empresa, que dispone al menos una caja 5 de comunicación de tipo "*Box*".

La figura 8, es un diagrama topológico que ilustra, según el estándar UML, el modelado de un servicio. En esta figura, el nodo que simboliza la clase "*Servicio*" está conectado por paradas a las siguientes instancias: "*Proveedor*", "*Consumidor*", "*Interfaz*", "*Tipo de servicio*". Ventajosamente, un servicio del sistema 100 de gestión de datos consta de las siguientes características:

- se expone (parada "proporcionar por") por un proveedor (proveedor 7 de servicios de usuario y/o proveedores 11 de servicios básicos) en la plataforma 4 central de datos;
- se consume (parada "ofrecer por") por actores (por ejemplo: clientes, proveedores) del sistema 100 de gestión de datos;
- se asocia con una interfaz predeterminada. En la plataforma 4 central de datos, en la mayoría de los casos, el servicio dispone (parada "disponer") de una interfaz de red;
- se emplea en la plataforma 4 central de datos, es decir, en la "*Nube*", o desplegado localmente en la caja 5 de comunicación. El lugar de despliegue condiciona el tipo de servicio (parada "tener").

Ventajosamente, el sistema 100 de gestión de datos permite proporcionar servicios de usuario y gestionar estos servicios desde su concepción técnica hasta su despliegue. Para hacer esto, la plataforma 4 central de datos propone servicios básicos y servicios elaborados, estando estos últimos realizados a través de un número determinado de servicios básicos. La figura 9 ilustra la relación topológica existente entre estos diferentes servicios, se distingue:

- los servicios de tipo petición: estos servicios se basan en los datos. Ventajosamente, consultando la base de datos que constituye la plataforma 4 central de datos a través de peticiones, es posible proporcionar datos a uno o varios actores del sistema 100. A modo de ejemplo, los datos pueden ser datos brutos relacionados con los sensores o datos agregados;
- los servicios de accionamiento: estos son servicios que utilizan los accionadores de los equipos 1 con el fin de realizar una acción;
- los servicios de notificación: estos servicios efectúan un conjunto de operaciones determinadas, apoyándose en los datos almacenados en la plataforma 4 central de datos. A modo de ejemplo, estos servicios permiten realizar un diagnóstico, o recomendar un escenario para un equipo 1, con el fin de optimizar el consumo de energías no renovables tal como la electricidad, el gas, y el agua;
- los servicios elaborados: se trata de servicios que se realizan según una secuencia predeterminada de flujo de trabajo, comúnmente designados con el anglicismo "*workflow*". Estos servicios permiten realizar un conjunto de operaciones haciendo un uso comúnmente de los servicios básicos.

Ventajosamente, el conjunto de los nodos descritos anteriormente, así como sus subconjuntos topológicos, constituyen un modelo semántico de datos. Basándose en este modelo semántico, el módulo 13 analítico de megadatos llamado "*Big Data Analytics*" (o la plataforma 4 central de datos), está configurado para aplicar un procesamiento en los datos registrados en la plataforma 4 central. A modo de ejemplo, el módulo 13 analítico de megadatos clasifica, segmenta, agrega, abstrae y/o formatea cualquier dato de la plataforma 4 central. Por ejemplo, suponiendo que cada dato comprende un identificador característico de su origen, por ejemplo, un identificador relacionado con la dirección de

una red de Internet o una caja 5 de comunicación, el módulo 13 analítico de megadatos clasifica, en función de estos identificadores, los datos de usuarios brutos o agregados procedentes de los sensores, o los datos de Internet proporcionados por los proveedores 9 de datos. Por otro lado, el módulo 13 analítico de megadatos procede opcionalmente sobre estos datos en un paso de abstracción/anonimización, permitiendo preservar la confidencialidad, después, los hace accesibles a los proveedores 7 de servicios de usuario.

Ventajosamente, el módulo 13 analítico de megadatos proporciona servicios para diferentes fases de uso de los datos almacenados en la plataforma 4 de datos central. Se distinguen las siguientes fases:

- 10 - fase de vigilia y tendencias: esta fase permite identificar tendencias mediante el análisis de los datos procedentes de Internet y de las redes sociales. Para esta fase, el módulo 13 analítico de megadatos pone a disposición las herramientas y los métodos de análisis de datos (por ejemplo: Pig/Hive) con destino a los proveedores 7 de servicios de usuario. Ventajosamente, estas herramientas permiten a los proveedores 7 de servicios de usuario identificar temas relacionados con los datos registrados en la plataforma 4 de datos central, y elaborar indicadores estadísticos en relación con estos temas. A modo de ejemplo, el módulo 13 analítico de megadatos conduce un análisis sobre datos procedentes de redes sociales o peticiones de motores de búsqueda, siendo estos datos proporcionados por los proveedores 9 de datos a la plataforma 4 de datos central. El resultado del análisis realizado por el módulo 13 analítico de megadatos se devuelve entonces al proveedor 7 de servicios de usuario en forma de palabras clave, permitiéndole identificar un tema pertinente de actualidad con el fin de realizar servicios de usuario;
- 15 - fase de elaboración de los servicios: esta fase permite a los proveedores 7 de servicios de usuario elaborar servicios. Para hacer esto, la plataforma 4 central de datos proporciona datos confiables (posiblemente abstractos), y el módulo 13 analítico de megadatos propone herramientas que permiten el análisis de estos datos. Según diversos modos de realizaciones, y el módulo 13 analítico de megadatos propone herramientas estadísticas basadas en
 - 20 ○ métodos de correlación de datos, por ejemplo, correlaciones de
 - 25 ■ Pearson para variables cuantitativas continuas;
 - 30 ■ Spearman para datos ordinales teniendo en cuenta sus rangos;
 - 35 ■ Kendall para datos ordinales teniendo en cuenta sus rangos;
 - métodos de análisis de datos, tales como
 - 35 ■ métodos de análisis de componentes principales (ACP) para estudiar y visualizar correlaciones entre varias variables;
 - 38 ■ métodos de análisis de correspondencia múltiple (ACM) para analizar el enlace entre variables cualitativas;
 - 40 ○ métodos de agrupación de datos comúnmente denominados con el anglicismo "*clustering*".

Ventajosamente, los datos confiables que se analizarán/correlacionarán mediante el módulo 13 analítico de megadatos se pondrán a disposición de los diferentes proveedores 7 de servicios de usuario a través de un servicio de informes (del tipo "*reporting*") y se relacionan, a modo de ejemplo, a los datos de los sensores, datos ambientales, o datos de usuarios;

- 45 - fase de despliegue de un servicio: esta fase permite a diferentes proveedores 7 de servicios de usuario desplegar sus servicios en el sistema 100 de gestión de datos. Para esta fase, el módulo 13 analítico de megadatos pone a disposición de los proveedores 7 de servicios de usuario herramientas de gestión del ciclo de vida del servicio y herramientas de control adecuada para garantizar los requisitos previos para el despliegue de un servicio en el sistema 100 gestión de datos;
- 50 - fase de recomendación de los servicios: esta fase implementa algoritmos de recomendación para recomendar servicios a los actores del sistema 100 de gestión de datos. Se recomienda, por ejemplo, durante esta fase, los servicios básicos a los proveedores 7 de servicios de usuario, servicios de usuario y/o escenarios de uso de servicios de usuario a particulares. Ventajosamente, los servicios de recomendación proporcionados por el sistema 100 de gestión de datos se basan en métodos de filtrado colaborativo. Se realiza un filtrado colaborativo, a modo de ejemplo, por el módulo 13 analítico de megadatos mediante la aplicación de un método capaz de comparar los usuarios entre sí (por ejemplo: tipo de servicio usado, tipo de datos consumidos, comportamiento del usuario), o elementos previamente anotados (por ejemplo: servicios de usuario previamente anotados por sus clientes). En un modo de realización, el sistema 100 de gestión de datos propone durante esta fase un servicio de autoajuste de equipos 1. Para hacer esto, el ajuste de un equipo 1 se memoriza durante un período predefinido en la plataforma 4 de datos central, el ajuste se aplica a continuación a uno o varios otros equipos 1 del mismo tipo. En un modo de realización, un servicio llamado "del mejor escenario" propuesto por el sistema 100 de gestión de datos, con el fin de recomendar o reproducir el mejor escenario para un servicio de usuario, tal como el ajuste un aparato o la gestión de un recurso (por ejemplo: agua, gas, electricidad). Ventajosamente, la recomendación del mejor escenario, se basa en el resultado de un análisis conducido por el módulo 13 analítico de megadatos, siendo este análisis realizado sobre la información enviada por los equipos 1 de una categoría de usuarios similares. Para

hacer esto, se utilizan algoritmos de agrupación ("*clustering*") de tipo "*k-means*" o "*canopy*". A modo de ejemplo, la configuración del servicio de mejor escenario se realiza a través de los siguientes pasos:

- 5 o selección del equipo 1 o del recurso;
- o segmentación de los usuarios;
- o aplicación de un método de tipo "*canopy*" para calcular un conjunto de clústeres de usuarios;
- o aplicación de un método de tipo "*k-means*", con el fin de identificar el mejor escenario relativo a cada clúster;
- fase de puntuación de los servicios ("*scoring*"): esta fase, permite asignar una puntuación e identificar servicios no pertinentes. El sistema 100 de gestión de datos, por ejemplo, propone a los clientes calificar los servicios de usuario que usan en una tienda en línea, o los proveedores 7 de servicios de usuario de anotar los servicios básicos puestos a su disposición para la realización de los servicios de usuario. Ventajosamente, tal fase permite anticipar la obsolescencia de los servicios. Para ello la plataforma utiliza algoritmos de aprendizaje, basados en métodos estadísticos, por ejemplo, métodos de regresión logística o métodos de árbol. Ventajosamente, para anticipar la obsolescencia de los servicios, el módulo 13 analítico de megadatos está configurado para asociar con cada servicio un umbral, y calcular una puntuación relativa a dicho servicio. Si la puntuación del servicio es inferior al umbral, entonces el servicio se identificó como obsoleto por el módulo 13 analítico de megadatos, los proveedores 7 de servicios de usuario se ven entonces notificados.

20 Uno de los desafíos de la plataforma 4 central de datos se refiere al acceso (con segregación) a los datos y servicios de usuario. La plataforma 4 central de datos desempeña un papel de pivote en la gestión y la transmisión de datos con las cajas 5 de comunicación, el módulo 13 analítico de megadatos ("*Big Data Analytics*") y los diferentes actores/consumidores de datos, tales como los proveedores 7 de servicios de usuario. De este modo, en un modo de realización, se asocia a la plataforma 4 central de datos con un equipo que implementa el protocolo AAA (acrónimo inglés de "*Authentication, Authorization, Accounting*") que realiza funciones de autenticación, de autorización, y la trazabilidad de los datos y servicios expuestos por la plataforma 4 central de datos. Ventajosamente, esto permite garantizar que un consumidor de datos no acceda a los datos o servicios a los que no tiene derecho o que son susceptibles de derogar las reglas de privacidad. A modo de ejemplo, después de la autenticación, un consumidor está autorizado para llamar a un servicio de la red (comúnmente denominado con el anglicismo "service web") del tipo REST (acrónimo de "*Representational State Transfer*") o SOAP (acrónimo de "*Simple Object Access Protocol*"), o llamar incluso a un servicio un cierto número de veces por unidad de tiempo (por ejemplo: mil veces al mes). La llamada de servicios REST o SOAP permite, en particular, la lectura de los datos que emanan de los equipos 1 o del módulo 13 de megadatos, y permite el envío de acciones hacia los equipos 1. Además, la plataforma 4 central de datos expone una o varias interfaces de programación aplicativas llamadas "*API*" (acrónimo inglés de "*Application Programming Interface*"), y opcionalmente una interfaz gráfica de usuario llamada "*GUI*" (acrónimo inglés de "*Graphical User Interface*"), permitiendo el suministro de datos de los consumidores de datos (clientes, proveedores 7 de servicios de usuario), aplicaciones consumidoras de datos, y permitiendo gestionar las autorizaciones de acceso a los datos, por ejemplo, en función de los papeles y umbrales configurados previamente.

40 Ventajosamente, la exposición de una o varias interfaces de programación aplicativa API por la plataforma 4 central de datos está dedicada a los proveedores y permite:

- a los proveedores 11 de servicios básicos gestionar un conjunto de servicios ("*bundles*") creados por los diferentes proveedores y desplegados en la plataforma 4 central de datos (por ejemplo: validación del despliegue, de las versiones, mutualización, estadísticas);
- a proveedores 7 de servicios de usuario, consumidores de servicios básicos y datos, acceder de forma autenticada, controlada y contabilizada, a través de una capa frontal (por ejemplo: sitio de Internet, aplicaciones móviles), a la información producida por el módulo 13 analítico de megadatos ("*Big Data Analytics*"), después de que este último módulo se haya procesado (por ejemplo: analizado, correlacionado, agrupado, formateado, abstraído) los datos de usuarios enviados por las diferentes cajas 5 de comunicación.

Igualmente, se pone a disposición en cada caja 5 de comunicación un conjunto de interfaces de programación aplicativa API dedicadas esta vez a los usuarios finales. Ventajosamente, la exposición de una o varias interfaces de programación aplicativa API dedicada a los usuarios finales, permite a través de una capa frontal (por ejemplo: sitio de Internet, aplicaciones móviles) proponer

- a los usuarios finales de equipos 1, la compra de servicios de usuario en línea, por ejemplo, a través de una tienda en línea propuesta por una aplicación;
- el acceso tanto a la información global (por ejemplo: datos estadísticos) sobre la flota de cajas 5 de comunicación, como a la información individual por servicio comprado (acceso autenticado, controlado y contabilizado);
- a la parametrización de cada servicio de usuario (configuración personalizada);
- el acceso a los servicios de puntuación y recomendación para cada servicio comprado.

Ventajosamente, la integración de los datos en la plataforma 4 central de datos es vista por las diferentes entidades como una capa intermedia, asegurando el tránsito de flujos de información entre:

- las cajas 5 de comunicaciones, es decir, la información procedente de los sensores de cada equipo 1;
- una base semántica, tal como el modelo semántico de datos anteriormente descrito;
- el conjunto de los servicios accesibles al consumidor final que constituye un portal de servicios;
- las interfaces de programación aplicativa API que constituyen un portal de gestión de servicios. Ventajosamente, tal portal facilita las interacciones automatizadas con los equipos 1 o los sistemas de información de los proveedores.

En particular, en un contexto de vendedores múltiples, los diferentes dispositivos (equipos 1, sensores, accionadores) conectados a una caja 5 de comunicación pueden provenir de diferentes vendedores, que no tienen conocimiento previo entre sí, siendo estos dispositivos soportados tanto por la caja 5 de comunicación como por la plataforma 4 central de datos. La plataforma 4 central de datos luego trivializa los datos procedentes de dichos dispositivos, exponiéndolos como servicios (por ejemplo: de acción, de alerta, de datos), permitiendo de este modo el desarrollo de servicios relacionados con múltiples dispositivos y/o múltiples vendedores, siendo estos servicios implementados posteriormente al nivel de la plataforma 4 central de datos o de las diferentes cajas 5 de comunicación.

Por otro lado, la integración de los datos en la plataforma 4 central de datos permite posteriormente realizar una lista de acciones, comunicada por la plataforma 4 central de datos a al menos una caja 5 de comunicación, estando dicha caja conectada a un número determinado de equipos 1. Ventajosamente, estas acciones se determinan en función del modelo semántico, y posiblemente estén relacionadas:

- a datos de métricas de equipos 1 reenviados por una caja 5 de comunicación;
- un evento desencadenante de acción preconfigurado, a modo de ejemplo, una acción configurada para desencadenarse en una fecha o una hora predefinidas;
- notificaciones insertadas por las interfaces de programación aplicativa API expuestas por la plataforma 4 central de datos, insertadas, por ejemplo, a través de un portal de Internet o una aplicación móvil, relacionándose con una solicitud de acción de usuario o proveedor.

De este modo, durante la recepción un evento tal como una notificación o una métrica, la plataforma 4 central de datos a través de una capa de integración de datos:

- registra en un historial el evento recibido si se trata de un evento procedente de una caja 5 de comunicación. El evento puede contener opcionalmente parámetros asociados con un equipo 1 determinado;
- con ayuda del módulo 13 analítico de megadatos clasifica el evento con la ayuda de una base de datos semántica, y establece una correlación entre este evento y un servicio asociado;
- si el servicio existe, envía datos asociados con el evento hacia un mecanismo de ejecución (por ejemplo: asociado al equipo 1), estando dicho mecanismo configurado para generar una acción asociada con el servicio, así como con el evento;
- inserción a través de un mecanismo del tipo "*push*" la acción generada hacia la caja 5 de comunicación correspondiente.

Por otro lado, otro desafío de la plataforma 4 central de datos se refiere a su interfaz con las cajas 5 de comunicación. La interfaz de la plataforma 4 central de datos con las cajas 5 de comunicación debe gestionar los flujos de datos:

- en sentido descendente: de la plataforma 4 central de datos hacia una caja 5 de comunicación, por ejemplo, durante el envío de una orden hacia un equipo 1 de tipo de accionador;
- en sentido ascendente: de una caja 5 de comunicación hacia la plataforma 4 central de datos, por ejemplo, durante el rastreo de una medición procedente de un sensor de un equipo 1.

Comúnmente, una caja 5 de comunicación está ubicada en una red local "interna" (por ejemplo: LAN) y se conecta a una red remota "externa" (por ejemplo: Internet) por medio de un dispositivo de acceso integrado, comúnmente conocido con el anglicismo IAD ("*Integrated Access Device*"). El dispositivo de acceso integrado IAD es proporcionado por un proveedor de acceso de red a Internet y permite intercambiar flujos de datos de diferentes naturalezas a través de una única conexión. De este modo, cada caja 5 de comunicación descrita anteriormente se conecta a la plataforma 4 central de datos detrás de un dispositivo de acceso integrado IAD. Con el fin de asegurar la comunicación entre cada caja 5 de comunicación y la plataforma 4 central de datos, el dispositivo de acceso integrado IAD establece una conexión con la plataforma 4 central de datos. Para establecer esta conexión, el dispositivo de acceso integrado IAD y la plataforma 4 central de datos disponen de direcciones públicas, mientras que la caja 5 de comunicación, ubicada detrás del dispositivo de acceso integrado IAD, dispone de una dirección privada, no direccionable desde la plataforma 4 central de datos. Ventajosamente, tal arquitectura permite que cualquier caja 5 de comunicación ubicada detrás del dispositivo de acceso integrado IAD, puede ser la iniciativa de una conexión de datos (flujo ascendente) hacia cualquier plataforma de Internet en el sentido amplio, mientras se mantiene protegido de amenazas externas. De este modo, en sentido ascendente, cualquier caja 5 de comunicación llega por medio del dispositivo de acceso integrado IAD para llegar a la plataforma 4 central de datos, por ejemplo, durante el rastreo de mediciones o de eventos reenviados por sensores. Sin embargo, en sentido descendente, por ejemplo, para la gestión de acciones a enviar a los equipos 1, la plataforma 4 central de datos de la plataforma no dispone de medios que permitan alcanzar la caja 5 de comunicación, debido a su direccionamiento privado detrás del dispositivo de acceso integrado IAD. Por otro lado, la caja 5 de

comunicación puede no estar disponible momentáneamente desde el punto de vista del dispositivo de acceso integrado IAD, por ejemplo, en caso de desconexión temporal o de una desconexión eléctrica simple. Comúnmente, cualquier dispositivo de acceso integrado IAD propone una función de reglas de traducción de dirección de red, comúnmente designadas con el anglicismo NAT ("*Network Address Translation*"), permitiendo hacer coincidir una dirección pública/un puerto de salida del dispositivo de acceso integrado IAD con una dirección privada/un puerto de entrada relacionado con una caja 5 de comunicación. Sin embargo, tal función no se realiza de forma predeterminada y requiere para su activación la configuración de reglas.

Además, el desarrollo de mecanismos que permitan el manejo a distancia (por ejemplo: para administración) y el envío de acción a los equipos 1, conectados con una caja 5 de comunicación, desde la plataforma 4 central de datos no se implementa en la arquitectura actual.

Según diversos modos de realizaciones, el manejo a distancia y el envío de acción a uno o varios equipos 1, conectados con una caja 5 de comunicación, desde la plataforma 4 central de datos, se realizan mediante la implementación de cuatro mecanismos cuyo funcionamiento general se describe brevemente aquí:

- mecanismo de comunicación de tipo "túnel mediante proxy". Comúnmente, para este mecanismo, una caja 5 de comunicación de tipo "Box" establece una conexión a través de un protocolo de control de transmisiones, posteriormente designado con el apelativo TCP, acrónimo inglés de "*Transmission Control Protocol*". Esta conexión se establece a través de la creación de un túnel desde la caja 5 de comunicación hacia un servidor proxy asociado con una plataforma remota. De este modo, en los modos de realización desarrollados posteriormente, se establece un túnel entre una caja 5 de comunicación y un proxy dispuesto en la plataforma 4 central de datos. El túnel se mantiene entonces abierto por la caja 5 de comunicación, por ejemplo, mediante el envío de paquetes ficticios ("*dummy packets*"), o reabierta en caso de desconexión. La creación de este túnel se efectúa en la dirección ascendente, es decir, de la caja 5 de comunicación hacia la plataforma 4 central de datos. El establecimiento del túnel permite, por lo siguiente, la circulación de información ascendente o descendente entre la caja 5 de comunicación y la plataforma 4 central de datos. Ventajosamente, gracias al proxy, la plataforma 4 central de datos ve entonces la caja 5 de comunicación, como si el dispositivo de acceso integrado IAD faltara e implementara un mecanismo de intercambio de datos bidireccional. No obstante, tal mecanismo sigue consumiendo muchos recursos en los servidores proxy al mantener conexiones TCP abiertas con las diferentes cajas 5 de comunicación;
- mecanismo de comunicación basado en el protocolo de control del "dispositivo de pasarela de Internet", posteriormente designado con el apelativo protocolo IGD, acrónimo inglés de "*Internet Gateway Device Protocol*". El protocolo IGD se describe en el estándar UPnP (acrónimo inglés de "*Universal Plug and Play*"). Comúnmente las "box" de los operadores, es decir, los dispositivos de acceso integrado IAD en este documento, proponen funciones de tipo rúter, "cortafuegos" ("*firewall*" en inglés) y "UPnP". Ciertas aplicaciones de redes, tales aplicaciones de tipo de pares P2P, a veces proponen al instalar en una máquina informática una opción de configuración automática mediante el uso de un mecanismo de tipo "UPnP". El principio de este mecanismo consiste, a través de un controlador UPnP presente en el dispositivo de acceso integrado IAD, en configurar a través del protocolo IGD, una función de "traducción" de dirección de red "NAT" ("*Network Address Translation*" en inglés) del cortafuegos. Esta configuración permite, en particular, establecer una correspondencia ("*mapping*" en inglés) entre los puertos públicos/las direcciones públicas de la IAD, y puertos privados/direcciones privadas de objetos conectados detrás del cortafuegos del IAD, y esto de una manera transparente para el usuario. Por ejemplo, el protocolo IGD establecido a través de la función NAT, una correspondencia entre una dirección privada y un puerto privado hacia una caja 5 de comunicación desplegada en una red local, y una dirección pública y un puerto público hacia la plataforma 4 central de datos desplegada en Internet. Ventajosamente, el uso de este mecanismo, permite entonces que la plataforma 4 central de datos vea la caja 5 de comunicación, como si el dispositivo de acceso integrado IAD faltara y propone un mecanismo de intercambio de datos bidireccional. Sin embargo, este mecanismo presenta riesgos del tipo de seguridad: la reconfiguración del cortafuegos está potencialmente abierta a cualquier software de terceros conectado a la red local del IAD, y una vez que el "*mapping*" está en su lugar, la caja 5 de comunicación está potencialmente expuesta a riesgos de ataques procedentes de Internet. Por otro lado, tal mecanismo requiere el soporte de la UPnP por parte de la IAD y que esta función se active en la IAD;
- mecanismo de comunicación que usa el protocolo de "extensible de presencia y de mensajería", designado posteriormente con la denominación protocolo XMPP, acrónimo inglés de "*Extensible Messaging and Presence Protocol*". Este protocolo de mensajería se basa en protocolos TCP, XML y reemplaza el protocolo Jabber, que es un protocolo de mensajería instantánea. El protocolo XMPP se describe en la memoria descriptiva TR-069. En particular, el anexo K.2 de la versión 1.4 de esta memoria descriptiva (noviembre de 2013), describe un mecanismo que utiliza el protocolo XMPP para procesar a través de la función NAT, el problema de una petición de conexión "*Connection Request*" con destino a un "Equipo en los locales del suscriptor", comúnmente denominado con la denominación CPE, acrónimo inglés de "*Customer Premises Equipment*". Un CPE es, a modo de ejemplo, la caja 5 de comunicación desplegada en una red local. Con fines de comprensión, este anexo se resume aquí, pudiendo los detalles adicionales encontrarse en este:

- o un servidor de configuración automática ACS (acrónimo inglés de "*Auto-Configuration Server*") establece una conexión con un servidor XMPP. El servidor de configuración automática de ACS y el servidor XMPP son, a modo de ejemplo, instalados en una red de Internet;

- o dicho servidor de autoconfiguración ACS activa el uso del protocolo XMPP al nivel de un CPE mediante la configuración de un objeto XMPP "*XMPP Connection object*", opcionalmente, proporcionar un conjunto de identificadores Jabber autorizados;
- o dicho CPE establece una conexión XMPP (por la vía ascendente) con el servidor XMPP;
- o cuando el servidor de configuración automática de ACS intenta comunicarse con el CPE, envía un mensaje de "*XMPP Connection Request*" al servidor XMPP. Este mensaje es una "estrofa" de XMPP, designada comúnmente con la denominación "*XMPP IQ Stanza*", y comprende una petición de conexión "*Connection Request*" que indica un identificador para originar un identificador Jabber autorizado, refiriéndose al servidor de configuración automática de ACS, e indicando para destino un identificador relativo al CPE.
- o el servidor XMPP transmite envía el mensaje "*XMPP IQ Stanza*" al CPE apropiado;
- o el CPE envía un de vuelta un mensaje de "*Inform Request*" con destino al servidor de configuración automática ACS.

En los modos de realización descritos a continuación, el servidor de configuración automática ACS se asocia con la plataforma 4 central de datos. Ventajosamente, esto permite realizar un mecanismo de tipo "activación", que permite disminuir el consumo de recursos en comparación con los mecanismos anteriores: cualquier CPE, por ejemplo, cada caja 5 de comunicación, se le previene ("activación") de que debe ponerse en contacto con la plataforma 4 central de datos mediante un flujo ascendente con el fin de recuperar el flujo de datos en sentido descendente. Ventajosamente, tal mecanismo permite una mejora de la seguridad de la caja 5 de comunicación; mecanismo de comunicación basado en el protocolo "cruzada simple UDP a través de los NAT", designado posteriormente con la denominación protocolo "*STUN*", acrónimo inglés de "*Simple Traversal of UDP through NATs*". El protocolo STUN permite una aplicación de un CPE (por ejemplo: la caja 5 de comunicación) conectado a un IAD detrás de un cortafuegos, para descubrir la posible presencia de una función (es decir, una pasarela) NAT de la IAD, y obtener la correspondencia ("*mapping*") de la aplicación con la dirección pública y el puerto UDP (anglicismo de "*User Datagram Protocol*") del dispositivo de acceso integrado IAD atribuido por la pasarela NAT. El uso de este protocolo requiere la asistencia de un tercero, es decir, un servidor STUN desplegado en una red pública tal como Internet. El protocolo STUN se describe en la memoria descriptiva TR-111 ("Applying TR-069 to Remote Management of Home Networking Devices", diciembre de 2005). En particular, la parte de "2.2 Procedures" de esta memoria descriptiva describe el procedimiento del protocolo STUN para que un CPE pueda recibir una petición de conexión UDP desde un servidor de configuración automática ACS remoto. El servidor de configuración automática ACS y el servidor STUN se despliegan en el lado de la dirección pública de la función NAT del dispositivo de acceso integrado IAD. Con fines de comprensión, la parte 2 de la memoria descriptiva TR-111 se resume aquí, pudiendo los detalles adicionales encontrarse en este:

- o el servidor de configuración automática ACS permite el uso del protocolo STUN para el CPE (si esta configuración no está activada de forma predeterminada) y designa un servidor STUN para comunicarse con el CPE;
- o el CPE luego utiliza el protocolo STUN, para descubrir si se sitúa detrás de una pasarela NAT con una dirección privada asignada;
- o en caso afirmativo, el CPE utiliza el procedimiento definido por el estándar STUN para descubrir la expiración de su enlace de datos detrás de la pasarela NAT ("*binding timeout*");
- o con el fin de efectuar el paso de "*mapping*", el CPE envía periódicamente peticiones de enlaces STUN con destino al servidor STUN, llamadas "*STUN Binding Requests*". Esto permite mantener abierta el enlace del CPE a través de la pasarela NAT, permitiendo este enlace al CPE escuchar posibles peticiones de conexión UDP ("*UDP Connection Requests*");
- o cuando el CPE determina la dirección IP pública y el puerto público utilizado para el enlace de la pasarela NAT (utilizados para escuchar mensajes "*UDP Connection Requests*"), el CPE transfiere la información de "*mapping*" determinadas al servidor de configuración automática de ACS, por ejemplo, mediante el envío de un mensaje de "*STUN Binding Request*";
- o el servidor de configuración automática de ACS establece entonces una conexión UDP con el CPE, mediante el envío de un mensaje de petición de conexión UDP "*UDP Connection Request*" hacia el puerto y la dirección pública de la pasarela NAT, determinados por el CPE;

En los modos de realización descritos a continuación, el servidor de configuración automática ACS y el servidor STUN se asocia con la plataforma 4 central de datos. Ventajosamente, esto permite realizar un mecanismo de tipo "activación", que permite disminuir el consumo de recursos en comparación con los mecanismos anteriores: cualquier CPE, por ejemplo, cada caja 5 de comunicación, se le previene ("activación") de que debe ponerse en contacto con la plataforma 4 central de datos mediante un flujo ascendente con el fin de recuperar el flujo de datos en sentido descendente. Ventajosamente, tal mecanismo es bastante ineficiente en recursos en la plataforma 4 central de datos. Sin embargo, el envío periódico de peticiones "STUN Binding request" que deja abierta la pasarela NAT, expone potencialmente el CPE a ataques externos.

El tablero siguiente, identifica las ventajas e inconvenientes de cada uno de los mecanismos introducidos anteriormente:

| | Túnel | UPnP | XMPP | STUN |
|-----------|-------|------|------|------|
| Seguridad | Alta | Baja | Alta | Baja |

ES 2 729 631 T3

| | | | | |
|-----------------|----------|----------|-------|----------|
| Implementación | Complejo | Complejo | Fácil | Complejo |
| Consumo | Alta | Baja | Medio | Baja |
| Dependencia IAD | Baja | Medio | Baja | Medio |

Actualmente, los mecanismos introducidos anteriormente no permiten el manejo a distancia (por ejemplo: para administración) y el envío de acción a los equipos 1, conectados con una caja 5 de comunicación, desde la plataforma 4 central de datos.

5 De este modo, después de la implementación, cada uno de los mecanismos introducidos anteriormente será adecuado para proponer el manejo a distancia, así como el envío de acción a equipos 1 conectados con una caja 5 de comunicación.

10 Según diversos modos de realizaciones, durante el despliegue del sistema 100 de gestión de datos, particularmente, cajas 5 de comunicación, se selecciona uno o varios de estos cuatro mecanismos en función de las restricciones técnicas y operacionales. Ventajosamente, se selecciona un único mecanismo para cada caja 5 de comunicación en función, a modo de ejemplo, del nivel de seguridad del mecanismo, su complejidad de implementación, de su consumo de recursos, y/o de su nivel de dependencia en el dispositivo de acceso integrado IAD. Esta elección puede, a modo de ejemplo, apoyarse en el tablero expuesta anteriormente.

15 Según diversos modos de realizaciones, durante la implementación de estos mecanismos, luego se despliegan servidores de configuración automática ACS, permitiendo gestionar las cajas 5 de comunicación y la inicialización de los diferentes mecanismos.

20 Con referencia a la figura 10, se realiza, por otra parte, una capa de abstracción a través de un gestor 14 de acciones, integrado al nivel de la plataforma 4 central de datos, ofreciendo una interfaz unificada, es decir, independientemente del tipo de mecanismo implementado y, por lo tanto, del protocolo empleado por este mecanismo. La capa de abstracción, a modo de ejemplo, se realiza con ayuda de un método que permite formatear los datos recibidos/registrados por esta capa según un formato de pivote.

25 Ventajosamente, el gestor 14 está configurado para gestionar (por ejemplo: recibir, transmitir, poner en espera) las acciones con destino a los diferentes equipos 1. En la figura 10, cada caja 5 de comunicación utiliza un solo mecanismo de tipo "túnel mediante proxy", "UPnP", "XMPP", o "STUN", seleccionado durante su despliegue y de acuerdo con las restricciones operacionales. Además, para cada caja 5 de comunicación, el mecanismo seleccionado se memoriza en la plataforma 4 central de datos, a través de una base 15 de datos asociada al gestor 14 de acciones.

30 Ventajosamente, durante la recepción de una petición 16 de acción (por ejemplo: una acción insertada por un consumidor de acción) con destino a un equipo 1 conectado a una caja 5 de comunicación, el gestor 14 de acción está configurado para:

- almacenar la acción recibida en la base 15 de datos, permitiendo así
 - o una puesta en espera temporal, el tiempo en que la caja 5 de comunicación recibe la petición 16 de acción. La puesta en espera es particularmente ventajosa cuando la caja 5 de comunicación no está disponible temporalmente, por ejemplo, durante una pérdida de conexión con los datos de la plataforma 4 central;
 - o para procesar la petición 16 de acción, por ejemplo: identificar el equipo objetivo (identificador, dirección), identificar la caja 5 de comunicación a la que está conectado, identificar el mecanismo de comunicación a emplear con dicha caja 5 de comunicación para transmitir la petición 16 de acción. Estas diferentes identificaciones, se realizan, por ejemplo, comparando el identificador del equipo con un conjunto de información registrada previamente en la base 15 de datos;
- desencadenar la acción en el equipo 1 transmitiendo la petición 16 de acción a su caja 5 de comunicación, a través de un mecanismo de comunicación adecuado, identificado en la base 15 de datos.

35 Por otro lado, como se expuso anteriormente, los mecanismos basados en los protocolos XMPP y STUN son de tipo "activación": la caja 5 de comunicación, se le previene de que debe ponerse en contacto con la plataforma 4 central de datos mediante un flujo ascendente con el fin de recuperar el flujo de datos en sentido descendente. Para soportar este mecanismo, entonces se realizan dos componentes:

- un agente de activación ("*Wakeup Agent*") realizado en la caja 5 de comunicación al colocar una capa intermedia, comúnmente designada con el anglicismo capa "*middleware*". Ventajosamente, la realización de tal capa permite que cualquier aplicación de la caja 5 de comunicación se suscriba a un servicio de activación, permitiéndole ser activado por una aplicación central;
- un servidor de activación ("*Wakeup Server*") a través de la realización de una capa de intermedia en la plataforma 4 central de datos, permitiendo a cualquier servicio "consumidor" de acción (integrado o no en la plataforma 4 central de datos) activar una aplicación de la caja 5 de comunicación.

La figura 11, ilustra el flujo funcional de datos relacionados con un consumidor 17 de acción, tal como un proveedor 7 de servicios de usuario:

- 5 - el consumidor 17 de acción inserta (flujo 18) una petición 16 de acción hacia el gestor 14 de acciones de la plataforma 4 central de datos. Esta petición 16 de acción consta de información sobre el tipo de acción que se efectuará y el equipo 1 objetivo por esta acción (por ejemplo: identificador del equipo, descripción de la acción). Esta petición 16 de acción puede opcionalmente ir acompañada de una fecha de caducidad y de una dirección, por ejemplo, una dirección de tipo "localizador uniforme de recurso" llamada URL, para notificar la recepción y/o el procesamiento de la petición 16 de acción por la caja 5 de comunicación;
- 10 - el gestor 14 de acciones almacena (flujo 19) en la cola en la base 15 de datos esta solicitud y reconoce (flujo 20) la petición 16 de acción emitida para notificar al consumidor 17 de la acción que se ha tenido bien en cuenta;
- 15 - a partir de la identificación del equipo 1 objetivo, el gestor 14 de acciones identifica la caja 5 de comunicación y su mecanismo de acceso correspondiente, luego transmite la petición 16 de acción a la caja 5 de comunicación. La transmisión de la petición 16 de acción es en función del tipo de mecanismo de comunicación, pudiendo dos situaciones surgir
 - o si el mecanismo de comunicación es de tipo bidireccional (de tipo "túnel mediante proxy" o UPnP):
 - 20 ■ el gestor 14 de acciones se pone en contacto con la caja 5 de comunicación a través de la dirección pública proporcionada por el túnel, o por el dispositivo de acceso integrado IAD en el caso de un mecanismo UPnP;
 - 25 ■ si, según el caso, el túnel se abre o la configuración UPnP se efectúa, el gestor 14 de acciones envía (flujo 21) una petición a la caja 5 de comunicación para proporcionar la petición 16 de acción. De lo contrario, el gestor 14 de acciones espera el restablecimiento del túnel o la reconfiguración UPnP para enviar entonces la petición 16 de acción, posiblemente gestionando las fechas de caducidad de la petición 16 de acción;
 - o si el mecanismo es de tipo "activación" (STUN o XMPP)
 - 30 ■ el gestor 14 de acciones entra en contacto con el servidor de activación ("*Wakeup Server*", capa intermedia) de la plataforma 4 central de datos;
 - 35 ■ este servidor de activación luego intenta activar la caja 5 de comunicación mediante un protocolo de bajo nivel (en STUN o XMPP, según sea el caso). Ventajosamente, el contenido del mensaje de activación con destino a la caja 5 de comunicación es limitado, y se puede restringir al tipo de activación, por ejemplo: una activación por acción en el equipo 1 o una activación para un manejo a distancia del equipo 1. Además, este mensaje está idealmente relacionado con un protocolo en modo no conectado, por ejemplo, el "protocolo de datagrama de usuario" llamado UDP. El servidor de activación también gestiona las funcionalidades de retransmisión, ya que la caja 5 de comunicación no es necesariamente accesible en el momento de la eliminación de la acción;
 - 40 ■ una vez la caja 5 de comunicación es accesible, el mensaje de activación llega al agente de activación ("*Wakeup Agent*") de este último, quién está a cargo de analizar el tipo de activación y de iniciar la conexión hacia la plataforma 4 central de datos y hacia la aplicación correcta para recuperar las acciones en espera;
 - 45 ■ la plataforma 4 central de datos a través de su gestor 14 de acciones proporciona (flujo 21) luego las peticiones de acciones en espera (incluyendo dicha petición 16 de acción) a la caja 5 de comunicación, posiblemente gestionando la fecha de caducidad de cada una de las peticiones;
- o la caja 5 de comunicación luego se ejecuta o transmite a los equipos 1 a los que está conectada, la petición 16 de acción y devuelve (flujo 22) un informe al gestor 14 de acciones;
- 50 o el gestor 14 de acciones notifica (flujo 23) entonces al consumidor 17 de acciones, la ejecución correcta de la acción asociada con su solicitud inicial (si se especifica en la solicitud inicial).

Ventajosamente, el flujo funcional descrito anteriormente está unificado, es decir, consta de los mismos pasos, y cualquiera que sea el mecanismo considerado. El funcionamiento de dichos mecanismos implementados se describe ahora para los casos de las figuras siguientes: manejo a distancia de un equipo 1, envío de acción a un equipo 1, rastreo de medición del sensor de un equipo 1 con destino a la plataforma 4 central de datos.

La figura 12 ilustra los principales flujos de un mecanismo de comunicación, para implementar un túnel entre la caja 5 de comunicación y un servidor 24 proxy. En esta figura, una caja 5 de comunicación está interconectada con un dispositivo 25 de acceso integrado IAD, a través de un primer puerto de comunicación, la caja 5 de comunicación siendo únicamente direccionable mediante una dirección privada situada detrás del dispositivo 25 de acceso integrado IAD. El dispositivo 25 de acceso integrado IAD dispone, por otro lado, de un segundo puerto de comunicación, que le permite ser direccionable desde una red 26 externa, por ejemplo, una red de internet, a través de una dirección pública. Una plataforma 4 central de datos comprende un gestor 14 de acciones asociado a una base 15 de datos, así como un servidor 27 de configuración automática ACS tal como se describe en la memoria descriptiva TR-069. Por otro lado, por razones de extensibilidad, el servidor 24 proxy se despliega en una plataforma 28 mandatária, adecuada para intercambiar datos con la plataforma 4 central de datos. Ventajosamente, estando cada caja 5 de comunicación

conectada permanentemente con un servidor 24 proxy, tal arquitectura permite que un mismo servidor 24 de proxy pueda gestionar una pluralidad de cajas 5 de comunicaciones en una plataforma 28 mandataria específica. Cada plataforma 28 mandataria, utiliza en cuanto a ella un número determinado de conexiones TCP con la plataforma 4 central de datos, siendo este número independiente del número de cajas 5 de comunicación a las que está conectado.

5 Para hacer esto, es posible utilizar técnicas de multiplexación. Ventajosamente, tal arquitectura permite que los servidores 24 de proxys se vean desde la plataforma 4 central de datos, como instancias virtuales de cajas 5 de comunicación, pero desplegados en una red pública (por ejemplo: Internet). Tal configuración es particularmente ventajosa, ya que permite superar los problemas relacionados con el direccionamiento de las cajas 5 de comunicación en una red privada. Por otro lado, en función del número de cajas 5 de comunicación, es posible, si es necesario,

10 agregar plataformas 28 mandatarias adicionales, que participan así en la extensibilidad de esta arquitectura. El servidor 27 de configuración automática ACS dispuesto en la plataforma 4 central de datos, permite informar a cualquier caja 5 de comunicación, durante la inicialización, del servidor 24 de proxy al que está conectado. Ahora se describe el establecimiento del túnel entre la caja 5 de comunicación y el servidor 24 de proxy. El establecimiento de este túnel ocurre en cada inicialización de la caja 5 de comunicación y comprende los siguientes pasos:

- durante la inicialización, la caja 5 de comunicación se declara a la plataforma 4 central de datos a través del servidor 27 de configuración automática ACS según el protocolo de gestión TR-069;
- la caja 5 de comunicación entonces se tiene en cuenta por la plataforma 4 central de datos, y esta elige un servidor 24 de proxy para establecer un túnel con la caja 5 de comunicación, y la socia a la caja 5 de comunicación. A modo de ejemplo, el servidor 24 de proxy es elegido por la plataforma 4 central de datos, en función de una proximidad geográfica y de una disponibilidad;
- esta asociación se transmite al servidor 27 de la configuración automática ACS;
- luego a través de un mensaje "SetParameterValues", cuya estructura se define en el protocolo TR-069, el servidor 27 de configuración automática ACS activa en la caja 5 de comunicación, el uso del túnel proporcionando (flujo 29 de la figura) la dirección del servidor 24 de proxy;
- la caja 5 de comunicación establece (flujo 30 de la figura) entonces una conexión TCP con el servidor 24 proxy, y envía un mensaje (no estándar) de enlace "BIND" para indicarle su identificación. Ventajosamente, el mensaje "BIND" comprende el identificador de la caja 5 de comunicación, así como los parámetros de la conexión TCP (por ejemplo: dirección y puerto remoto, enchufe). La recepción de este mensaje por parte del servidor 24 de proxy, por lo tanto, permitirá a este último gestionar la instancia virtual de la caja 5 de comunicación, conservando para esta instancia el identificador de la caja 5 de comunicación, así como los parámetros de la conexión TCP;
- esta conexión TCP se mantiene abierta permanentemente por la caja 5 de comunicación.

La figura 13, luego ilustra para la arquitectura anterior, los principales flujos de datos que permiten el manejo a distancia de la caja 5 de comunicación y/o de los equipos 1:

- un actor 31 externo, tal como un proveedor 7 de servicios de usuario, envía (flujo 32) una petición de manejo, para una caja 5 de comunicación o un equipo 1, al servidor 27 de configuración automática ACS;
- el servidor 27 de configuración automática ACS envía (flujo 33) un mensaje de petición de conexión "Connection Request", de acuerdo con la memoria descriptiva TR-069, al servidor 24 de proxy, especificando en este mensaje un identificador relativo al destinatario final del manejo, por ejemplo, relacionado con una caja 5 de comunicación (o un equipo 1) objetivo;
- gracias al identificador de la caja 5 de comunicación, el servidor 24 proxy selecciona entonces la instancia virtual de la caja 5 de comunicación, y, por lo tanto, el correspondiente túnel 34;
- el servidor proxy 24 luego transmite a través del correspondiente túnel 34 la petición de conexión "Connection Request" a la caja 5 de comunicación (flujo 35);
- la caja 5 de comunicación luego procesa la petición y emite (flujo 36) un mensaje de "Inform request" de acuerdo con la memoria descriptiva TR-069, con destino al servidor 27 de configuración automática ACS, para notificarle el procesamiento de la petición;
- el manejo de la caja 5 de comunicación (o del equipo 1), luego sigue el protocolo TR-069, como cualquier tipo de topología de red con un CPE (aquí la caja 5 de comunicación).

La figura 14, ilustra para la misma arquitectura, los flujos de datos principales que permiten transmitir una petición 16 de acción hacia un accionador asociado con un equipo 1:

- un actor 37 externo tal como un consumidor de acción envía (flujo 38) una petición 16 de acción hacia el gestor 14 de acciones de la plataforma 4 central de datos;
- el gestor 14 de acciones almacena entonces dicha petición 16 de acción en la base 15 de datos, luego envía la petición 16 de acción al servidor 24 proxy. Para hacer esto, el gestor 14 de acciones abre una conexión TCP y está configurado para enviar (flujo 39) dos peticiones al servidor 24 de proxy:

- o una primera petición, que es una petición (no estándar) de conexión "CONNECT", que comprende la identificación de la caja 5 de comunicación objetivo. Ventajosamente, el mensaje "CONNECT" permite especificar al servidor 24 de proxy que las siguientes peticiones, a saber, solicitudes de acción, deben dirigirse a la caja 5 de comunicación;

o una segunda petición correspondiente a la petición 16 de acción, según una sintaxis idéntica al caso de una caja 5 de comunicación directamente visible desde Internet, por ejemplo, una sintaxis de http(s);

- 5 - ventajosamente, gracias al mensaje de petición de conexión "CONNECT" (primera petición), el servidor 24 de proxy selecciona la instancia virtual de la caja 5 de comunicación objetivo y el correspondiente túnel 34. Ventajosamente, esto realiza una asociación temporal en el seno del servidor 24 de proxy, entre la conexión del gestor 14 de acciones a este servidor y el túnel 34 de la caja 5 de comunicación;
- 10 - el servidor 24 de proxy luego reconoce al gestor 14 de acciones la buena recepción del mensaje de petición de conexión "CONNECT". Mientras el gestor 14 de acciones no reciba este acuse de recibo, por ejemplo, cuando el túnel 34 no está establecido, el gestor 14 de acciones está configurado para reemitir periódicamente este mensaje o esperar al restablecimiento del túnel 34;
- 15 - cuando el acuse de recibo del mensaje "CONNECT" se recibe por el gestor 14 de acción, este último luego transmite la segunda petición, es decir, la solicitud 16 de acción al servidor 24 de proxy;
- el servidor 24 de proxy envía (flujo 40) luego a través del túnel 34 la petición 16 de acción tal como se recibe a la caja 5 de comunicación objetivo;
- la caja 5 de comunicación ejecuta entonces la acción solicitada o, de lo contrario, la transmite al equipo 1 correspondiente para su ejecución, luego devuelve (flujo 41) después un informe al servidor 24 de proxy a través del túnel 34;
- 20 - el servidor 24 de proxy transmite (flujo 42) entonces el informe tal como se recibe al gestor 14 de acciones;
- si se solicita, el gestor 14 de acciones luego notifica (flujo 43) al actor 37 externo de la ejecución exitosa de su solicitud de acción inicial por parte de la caja 5 de comunicación o el equipo 1 en cuestión.

La figura 15, todavía se ilustra para el mismo mecanismo, los principales flujos de datos que permiten rastrear una medición de un sensor asociado con un equipo 1. En esta figura, la plataforma 4 central de datos comprende, además, un gestor 44 de mediciones asociado con una base 441 de datos, que permiten gestionar (almacenar y/o poner a disposición) los datos asociados con los sensores de los diferentes equipos 1. El proceso de rastreo de medición hacia la plataforma 4 central de datos desde el sensor de un equipo 1 conectado a una caja 5 de comunicación es el siguiente:

- 30 - la caja 5 de comunicación recupera la medición procedente de dicho sensor y la transmite al equipo 1;
- la caja 5 de comunicación envía (flujo 45) entonces al servidor 24 de proxy, por el túnel 34 previamente establecido, la medición mediante un mensaje que indica que el objetivo es el gestor 44 de mediciones, por ejemplo, utilizando un mensaje del tipo "GET http(s)";
- 35 - el servidor 24 de proxy inicia entonces una conexión TCP hacia el gestor 44 de mediciones y transmite (flujo 46) la medición.

La figura 16 ilustra los principales flujos implementados para el establecimiento de una conexión entre la plataforma 4 central de datos, y una caja 5 de comunicación, para un mecanismo de comunicación que implementa el protocolo UPnP. Al igual que anteriormente:

- 40 - la una caja 5 de comunicación está interconectada con un dispositivo 25 de acceso integrado IAD, a través de un primer puerto de comunicación, siendo la caja 5 de comunicación direccionable únicamente a través de una dirección privada;
- 45 - el dispositivo 25 de acceso integrado IAD dispone de un segundo puerto de comunicación, que le permite ser direccionable desde una red 26 externa, por ejemplo, una red de internet, a través de una dirección pública;
- la plataforma 4 central de datos comprende un gestor 14 de acciones asociado a una base 15 de datos (no representada en esta figura), así como un servidor 27 de configuración automática ACS tal como se describe en la memoria descriptiva TR-069.

El siguiente proceso se ejecuta en cada inicialización de la caja 5 de comunicación:

- durante la inicialización, la caja 5 de comunicación se declara a la plataforma 4 central de datos por el servidor 27 de configuración automática ACS a través del protocolo de gestión TR-069;
- 55 - la caja 5 de comunicación entonces se tiene en cuenta por la plataforma 4 central de datos, y ésta activa en la caja 5 de comunicación el uso del protocolo UPnP;
- la caja 5 de comunicación luego envía las peticiones UPnP hacia el dispositivo 25 de acceso integrado IAD, para abrir un puerto de comunicación TCP público, permitiendo así la activación (flujo 47) de la función de traducción de dirección de red NAT;
- 60 - la caja 5 de comunicación transmite (flujo 48) entonces al servidor 27 de configuración automática ACS, a través del protocolo TR-069, el puerto TCP público abierto en el dispositivo 25 de acceso integrado IAD correspondiente a la traducción de la dirección NAT hacia la caja 5 de comunicación.

La figura 17, luego ilustra para el mecanismo anterior, los principales flujos de datos, que permiten el manejo a distancia de la caja 5 de comunicación y/o de los equipos 1:

65

- en primer lugar, un actor 31 externo tal como un proveedor 7 de servicios de usuario envía (flujo 49) una petición de manejo al servidor 27 de configuración automática ACS;
- el servidor 27 de configuración automática ACS envía (flujo 50) un mensaje "*Connection Request*", de acuerdo con la memoria descriptiva TR-069, hacia la caja 5 de comunicación. Para hacer esto, envía el mensaje "*Connection Request*" a la dirección pública del dispositivo 25 de acceso integrado IAD, en el puerto TCP público abierto durante el establecimiento UPnP. El dispositivo 25 de acceso integrado IAD luego transmite el mensaje a la caja 5 de comunicación aplicando la función de traducción de dirección de red NAT;
- la caja 5 de comunicación luego procesa la orden y emite (flujo 51) un mensaje de "*Inform request*", de acuerdo con el estándar TR-069, con destino al servidor 27 de configuración automática ACS para notificarle el procesamiento de la petición;
- el manejo de la caja 5 de comunicación (o del equipo 1), seguido entonces por el protocolo TR-069, como cualquier tipo de topología de red con un CPE (aquí la caja 5 de comunicación).

La figura 18, ilustra para el mismo mecanismo, los flujos de datos principales que permiten transmitir una solicitud de acción hacia un accionador asociado con un equipo 1:

- un actor 37 externo tal como un consumidor de acción pública (flujo 52) una solicitud de acción hacia el gestor 14 de acciones de la plataforma 4 central de datos;
- el gestor 14 de acciones almacena entonces dicha solicitud en la base 15 de datos, luego envía (flujo 53) la solicitud a la caja 5 de comunicación. Para hacer esto, envía la solicitud a la dirección pública del dispositivo 25 de acceso integrado IAD en el puerto TCP público abierto durante el establecimiento de la UPnP. El dispositivo 25 de acceso integrado IAD transmite entonces, mediante el uso de la función de traducción de dirección de red NAT, la solicitud a la caja 5 de comunicación;
- la caja 5 de comunicación ejecuta entonces la acción solicitada o, de lo contrario, la transmite al equipo 1 correspondiente para su ejecución, después, devuelve (flujo 54) después un informe al gestor 14 de acciones. Mientras el gestor 14 de acciones no reciba este informe, por ejemplo, cuando la función NAT UPnP no está establecida, el gestor 14 de acción está configurado para reemitir periódicamente este mensaje o esperar al restablecimiento de la función de traducción de dirección de red NAT UPnP;
- si se solicita, el gestor 14 de acciones luego notifica (flujo 55) al actor 37 externo de la ejecución exitosa de su solicitud de acción inicial por parte de la caja 5 de comunicación o el equipo 1 en cuestión.

La figura 19, todavía se ilustra para el mismo mecanismo, los principales flujos de datos que permiten rastrear una medición de un sensor asociado con un equipo 1, hacia el gestor 44 de mediciones de la plataforma 4 central de datos:

- la caja 5 de comunicación recupera la medición procedente de dicho sensor y la transmite al equipo 1;
- la caja 5 de comunicación inicia (flujo 56) luego una conexión, por ejemplo, de tipo "https(s)" hacia el gestor 44 de mediciones de la plataforma 4 central de datos;
- la caja 5 de comunicación transmite finalmente la medición mediante un mensaje que indica que el objetivo es el gestor 44 de mediciones, por ejemplo, utilizando un mensaje del tipo "*GET http(s)*".

La figura 20 ilustra después los principales flujos implementados para el establecimiento de una conexión entre la plataforma 4 central de datos, y una caja 5 de comunicación, para un mecanismo de comunicación que implementa el protocolo XMPP. Al igual que anteriormente:

- la una caja 5 de comunicación está interconectada con un dispositivo 25 de acceso integrado IAD, a través de un primer puerto de comunicación, siendo la caja 5 de comunicación direccionable únicamente a través de una dirección privada;
- el dispositivo 25 de acceso integrado IAD dispone de un segundo puerto de comunicación, que le permite ser direccionable desde una red 26 externa, por ejemplo, una red de internet, a través de una dirección pública;
- la plataforma 4 central de datos comprende un gestor 14 de acciones asociado a una base 15 de datos, así como un servidor 27 de configuración automática ACS tal como se describe en la memoria descriptiva TR-069.

Por otro lado, por razones de extensibilidad, un servidor 57 de proxy http y un servidor 58 de proxy XMPP, realizados de acuerdo con el estándar TR-069 se despliegan en una plataforma 28 mandataria, la plataforma 28 mandataria estando adaptada para intercambiar datos con la plataforma 4 central de datos. Ventajosamente, estando cada caja 5 de comunicación conectada permanentemente con un servidor 58 de proxy XMPP. Ventajosamente, la realización de tal arquitectura permite que un mismo servidor 58 de proxy XMPP pueda gestionar una pluralidad de cajas 5 de comunicaciones en una plataforma 28 mandataria específica. Como se expone posteriormente, los servidores 57 de proxy http en cuanto a ellos, se utilizan durante el rastreo de mediciones o recuperación de acciones a través de peticiones http. Cada plataforma 28 mandataria utiliza, en cuanto a ella, un número determinado de conexiones con la plataforma 4 central de datos, siendo este número de conexiones independiente del número de cajas 5 de comunicación a las que está conectada. Para hacer esto, es posible utilizar técnicas de multiplexación. Por otro lado, en función del número de cajas 5 de comunicación, es posible, si es necesario, agregar plataformas 28 mandatarias adicionales, así como servidores 58 de proxy XMPP y servidores 57 de proxy http, que participan así en la extensibilidad de esta arquitectura. El servidor 27 de configuración automática ACS dispuesto en la plataforma 4 central de datos, luego permite informar a cualquier caja 5 de comunicación durante su inicialización del servidor 58 proxy

XMPP al que está conectado. Igualmente, durante la configuración de la caja 5 de comunicación, durante su arranque, el servidor 27 de configuración automática ACS se encarga de informar a la caja 5 de comunicación de su servidor 57 de proxy http mandatario.

5 Ahora se describe el establecimiento de una conexión XMPP entre la caja 5 de comunicación y el servidor 58 de proxy XMPP. El establecimiento de esta conexión ocurre en cada inicialización de la caja 5 de comunicación y comprende los siguientes pasos:

- 10 - durante la inicialización, la caja 5 de comunicación se declara a la plataforma 4 central de datos por el servidor 27 de configuración automática ACS a través del protocolo de gestión TR-069;
- la caja 5 de comunicación entonces se tiene en cuenta por la plataforma 4 central de datos, y ésta elige un servidor 58 de proxy XMPP mandatario, por ejemplo, en función de una proximidad geográfica y de una disponibilidad de servidor 58 de proxy XMPP, y la socia a la caja 5 de comunicación;
- 15 - esta asociación se transmite entonces al servidor 27 de la configuración automática ACS;
- el servidor 27 de configuración automática ACS, establece (flujo 59) entonces una conexión con el servidor 58 de proxy XMPP, que luego se mantendrá abierta por el servidor 27 de configuración automática ACS;
- 20 - después a través del mensaje "*SetParameterValues*" del protocolo TR-069, el servidor 27 de configuración automática ACS activa en la caja 5 de comunicación, el uso del protocolo XMPP proporcionándole (flujo 60) la dirección del servidor, así como los identificadores Jabber autorizados. Ventajosamente, los identificadores Jabber autorizados se declaran de manera idéntica en el servidor 58 de proxy XMPP y en el servidor 27 de configuración automática ACS, y permiten a cualquier cliente XMPP, por ejemplo, a la caja 5 de comunicación, identificarse durante los intercambios XMPP;
- la caja 5 de comunicación establece (flujo 61 de la figura) entonces una conexión XMPP con el servidor 58 de proxy XMPP, esta conexión se mantendrá abierta permanentemente por la caja 5 de comunicación.

25 La figura 21, ilustra para la arquitectura y el mecanismo anterior, los principales flujos de datos que permiten el manejo a distancia de la caja 5 de comunicación y/o de los equipos 1:

- 30 - en primer lugar, un actor 31 externo tal como un proveedor 7 de servicios de usuario envía (flujo 62) una petición de manejo al servidor 27 de configuración automática ACS;
- el servidor 27 de configuración automática ACS envía (flujo 63) entonces, de acuerdo con el anexo K.2 de la memoria descriptiva TR-069, un mensaje de petición de conexión "*Connection Request*" en XMPP ("*estrofa*" XMPP de tipo "*XMPP IQ Stanza*") al servidor 58 de proxy XMPP. El servidor 27 de configuración automática ACS especifica en este mensaje, el destinatario final del manejo, por ejemplo, una caja 5 de comunicación objetivo, así como la fuente de este mensaje identificada por un identificador Jabber autorizado. La estructura de tal mensaje se detalla, en particular, en el anexo K.2.3.1 de la memoria descriptiva TR-069;
- 35 - el servidor 58 de proxy XMPP luego transmite (flujo 64) el mensaje a la caja 5 de comunicación a través del protocolo XMPP;
- la caja 5 de comunicación recibe entonces el mensaje, y genera un mensaje de respuesta XMPP ("*estrofa*" XMPP de tipo "*XMPP IQ Stanza*") de tipo "*resultado*" si el mensaje anterior se tiene en cuenta correctamente, o de tipo "*error*" de lo contrario. Las estructuras de estos mensajes de respuesta, se detalla respectivamente en el anexo K.2.3.2 y K.2.3.3 de la memoria descriptiva TR-069;
- 40 - la caja 5 de comunicación luego procesa la petición y emite (flujo 65) un mensaje de "*Inform request*" de acuerdo con la memoria descriptiva TR-069, con destino al servidor 27 de configuración automática ACS, para notificarle el procesamiento de la petición;
- 45 - el manejo de la caja 5 de comunicación (o del equipo 1), seguido entonces por el protocolo TR-069, como cualquier tipo de topología de red con un CPE (aquí la caja 5 de comunicación).

50 La figura 22, ilustra para la misma arquitectura y el mismo protocolo, los flujos de datos principales que permiten transmitir una petición 16 de acción hacia un accionador asociado con un equipo 1:

- un actor 37 externo tal como un consumidor de acción envía (flujo 66) una petición 16 de acción hacia el gestor 14 de acciones de la plataforma 4 central de datos;
- el gestor 14 de acciones almacena entonces dicha petición 16 de acción en la base 15 de datos, luego envía la petición 16 de acción a través del mensaje "*Action Request*" en XMPP (mensaje de tipo "*XMPP IQ Stanza*") al servidor 58 proxy XMPP. Ventajosamente, el mensaje "*Action Request*" comprende como origen un identificador Jabber autorizado y como destinatario de la caja 5 de comunicación. Según diversos modos de realizaciones, no estando el mensaje "*Action Request*" descrito en la memoria descriptiva TR-069, se puede inspirar en la estructura del mensaje "*Connection Request*" descrito en el anexo K.2.3.1 de esta memoria descriptiva, para implementar este mensaje. A continuación, se proporciona un ejemplo de realización del mensaje "*Action Request*". En este ejemplo, se especifica en particular una solicitud de acción 16 (campos "*actionRequest*") tipo "*obtener*", un identificador Jabber que se refiere al emisor de la solicitud de acción 16 (campos "*de=...*"), aquí, el gestor de acciones 14, un identificador Jabber para un destinatario (campos "*a=...*") de la solicitud de acción 16, aquí la caja de comunicación 5, un identificador de mensaje (campos "*identificación*") y una información de inicio de sesión, como un nombre de usuario y contraseña (campos "*nombre de usuario*", "*contraseña*"), esta información constituye un mensaje del tipo "*XMPP IQ Stanza*" (campos "*iq*");

```
iq from = "[PF-identity]" to = "L@D/R" id = "cr001" type = "get">
<actionRequest xmlns = "urn:bull-org:cwmp:xmppActReq-1-0">
<username>username</username>
<password>password</password>
</actionRequest>
</iq>
```

- el servidor 58 de proxy XMPP, recibe el mensaje "Action Request", después actúa entonces como un servidor de activación ("Wakeup server"), transmitiendo (flujo 68) el mensaje a la caja 5 de comunicación a través del protocolo XMPP. Por otro lado, el servidor 58 de proxy XMPP es capaz de almacenar dicho mensaje si la caja 5 de comunicación está desconectada, después la reemite durante la reconexión de la caja 5 de comunicación;
- la caja 5 de comunicación recibe entonces el mensaje, y genera un mensaje de respuesta XMPP (estrofa XMPP "XMPP IQ Stanza") de tipo "resultado" si el mensaje anterior se tiene en cuenta correctamente, o de tipo "error" de lo contrario. Las estructuras de estos mensajes de respuesta, se detalla respectivamente en el anexo K.2.3.2 y K.2.3.3 de la memoria descriptiva TR-069;
- la caja 5 de comunicación luego inicia una conexión http(s) con el fin de recuperar las solicitudes de acciones en espera. Esta conexión se establece (flujo 69) con el servidor 57 de proxy http de la plataforma 28 mandataria, que a su vez interrogará (flujo 70) al gestor 14 de acciones para recuperar las solicitudes de acciones, por ejemplo, a través de un mensaje de tipo "GET http". Ventajosamente, el uso del servidor 57 de proxy http en la plataforma 28 mandataria, permite entonces optimizar los recursos de la plataforma 4 central de datos, evitando múltiples conexiones http iniciadas directamente desde las diferentes cajas 5 de comunicaciones;
- el gestor 14 de acción proporciona entonces, en respuesta, las solicitudes de acción a la caja 5 de comunicación, a través de la conexión establecida con el servidor 58 de proxy XMPP. El gestor 14 de acción gestiona, por otro lado, la expiración de las solicitudes de acciones según dos modos:
 - o según un modo asíncrono si la caja 5 de comunicación no está disponible, a través del mensaje XMPP de activación transmitido por el servidor 57 de proxy http;
 - o según un modo síncrono, por una transmisión "en tiempo real" de las acciones a la caja 5 de comunicación si está disponible (sujeto a la validez temporal de la acción, función por ejemplo de un umbral temporal configurado previamente);
- la caja 5 de comunicación ejecuta entonces la acción solicitada o, de lo contrario, la transmite al equipo 1 correspondiente para su ejecución, después ven seguida vuelve, por medio del servidor 57 de proxy http, un informe al gestor 14 de acción;
- si se solicita, el gestor 14 de acciones luego notifica (flujo 71) al actor 37 externo de la ejecución exitosa de su solicitud de acción inicial por parte de la caja 5 de comunicación o el equipo 1 en cuestión.

La figura 23, ilustra también para la misma arquitectura, los principales flujos de datos que permiten rastrear una medición de un sensor asociado con un equipo 1. En esta figura, la plataforma 4 central de datos comprende, un gestor 44 de mediciones asociado con una base 441 de datos, que permiten gestionar (almacenar y/o poner a disposición) los datos asociados con los sensores de los diferentes equipos 1. El proceso de rastreo de medición hacia la plataforma 4 central de datos desde el sensor de un equipo 1 conectado a una caja 5 de comunicación es el siguiente:

- la caja 5 de comunicación recupera la medición procedente de dicho sensor y la transmite al equipo 1;
- la caja 5 de comunicación inicia (flujo 72) una conexión http(s) hacia el servidor 57 de proxy http de la plataforma 28 mandataria;
- el servidor 57 de proxy http inicia entonces una conexión de tipo https(s) hacia el gestor 44 de mediciones de la plataforma 4 central de datos;
- la caja 5 de comunicación envía (flujo 73) entonces al servidor 57 de proxys http, a través de la conexión http(s) establecida, la medición mediante un mensaje que indica que el objetivo es el gestor 44 de mediciones, por ejemplo, utilizando un mensaje del tipo "GET http(s)".

La figura 24 ilustra los principales flujos implementados para el establecimiento de un túnel, para un mecanismo de comunicación entre la caja 5 de comunicación y la plataforma 4 central de datos, para un mecanismo de comunicación que implementa el protocolo STUN. Al igual que anteriormente:

- la una caja 5 de comunicación está interconectada con un dispositivo 25 de acceso integrado IAD, a través de un primer puerto de comunicación, siendo la caja 5 de comunicación direccionable únicamente a través de una dirección privada. Esta dirección privada es, a título de ejemplo, direccionable a través de una pasarela/función de traducción de dirección de red NAT constitutiva del dispositivo 25 de acceso integrado IAD;
- el dispositivo 25 de acceso integrado IAD dispone de un segundo puerto de comunicación, que le permite ser direccionable desde una red 26 externa, por ejemplo, una red de internet, a través de una dirección pública;
- la plataforma 4 central de datos comprende un gestor 14 de acciones asociado a una base 15 de datos (no representada en esta figura), así como un servidor 27 de configuración automática ACS tal como se describe en la memoria descriptiva TR-069.

Por otro lado, la plataforma 4 central de datos comprende un servidor 74 STUN, de acuerdo con la memoria descriptiva TR-111. Como se explica a continuación, tal servidor permitirá establecer un "túnel" de activación entre los datos de la plataforma 4 central de datos y la caja 5 de comunicación.

5 El establecimiento de la conexión entre la caja 5 de comunicación y la plataforma 4 central de datos se describe ahora. El establecimiento de esta conexión ocurre en cada inicialización de la caja 5 de comunicación y comprende los siguientes pasos:

- 10 - durante la inicialización, la caja 5 de comunicación se declara a la plataforma 4 central de datos por el servidor 27 de configuración automática ACS a través del protocolo de gestión TR-069;
- la caja 5 de comunicación entonces se tiene en cuenta por la plataforma 4 central de datos, y ésta activa en la caja 5 de comunicación el uso del protocolo STUN;
- 15 - la caja 5 de comunicación emite (flujo 75) entonces peticiones de enlaces STUN "*STUN Binding Requests*" con destino al servidor 74 STUN de la plataforma 4 central de datos, de acuerdo con la parte 2.2 de la memoria descriptiva TR-111. Las respuestas del servidor 74 STUN a estas peticiones, permiten entonces que la caja 5 de comunicación descubra la dirección pública del dispositivo 25 de acceso integrado IAD. La caja 5 de comunicación transmite (flujo 76) entonces la dirección pública identificada del dispositivo 25 de acceso integrado IAD, al servidor de configuración automática ACS 27 a través del protocolo de gestión TR-069. Ventajosamente, la identificación de la dirección pública del dispositivo 25 de acceso integrado IAD, permite a la caja 5 de comunicación descubrir la presencia de una pasarela/función NAT en el dispositivo 25 de acceso integrado IAD, y establecer a través de esta pasarela/función una conexión UDP con el servidor 74 STUN;
- 20 - con el fin de mantener esta conexión abierta, la caja 5 de comunicación, envía (flujo 77) entonces a continuación periódicamente un mensaje de petición de enlace STUN de tipo "*Binding Request*". Ventajosamente, este período se configura de manera que el dispositivo 25 de acceso integrado IAD no desactive la función NAT después de un período de inactividad, y al mismo tiempo el tiempo suficiente con el fin de evitar la saturación en recursos del servidor 74 STUN. De este modo, en un modo de realización, la caja 5 de comunicación efectúa un mecanismo de aprendizaje, emitiendo inicialmente mensajes de petición de conexión de "*Binding Request*" con un período bajo, luego aumentando progresivamente el período de emisión de estos mensajes. Se determina de este modo un período límite durante el cual la caja ya no recibe una respuesta de conexión de "*Binding Response*" por parte del servidor 74 STUN. A partir de este estado, el período identificado como utilizable por la caja 5 de comunicación es el período anterior a este período límite.

35 La figura 25, ilustra para el mecanismo anterior, los principales flujos de datos que permiten el manejo a distancia de la caja 5 de comunicación y/o de los equipos 1:

- en primer lugar, un actor 31 externo tal como un proveedor 7 de servicios de usuario envía (flujo 78) una petición de manejo al servidor 27 de configuración automática ACS;
- 40 - el servidor 27 de configuración automática ACS entra en contacto (flujo 79) con el servidor 74 STUN, de acuerdo con la memoria descriptiva TR- 111, y le solicita enviar un mensaje de petición de conexión "*Connection Request*" según el protocolo UDP, con destino a la caja 5 de comunicación. La realización de tal mensaje se describe en la parte 2.2.2.3 de esta memoria descriptiva. El servidor 74 STUN es entonces el único elemento autorizado por el dispositivo 25 de acceso integrado IAD para utilizar la función de traducción de dirección de red NAT hacia la caja 5 de comunicación. En efecto, tras el establecimiento de la conexión con el servidor 74 STUN, el dispositivo 25 de acceso integrado IAD conoce solo la dirección relativa al servidor 74 STUN, así como el puerto UDP utilizado con este servidor;
- 45 - el servidor 74 STUN envía (flujo 80) luego la caja 5 de comunicación un mensaje UDP de petición de conexión "*Connection Request*", hacia el puerto UDP público del dispositivo 25 de acceso integrado IAD (utilizado por la función/pasarela de traducción de dirección de red NAT para establecer la conexión con la caja 5 de comunicación). El dispositivo 25 de acceso integrado IAD luego transmite este mensaje a la caja 5 de comunicación a través la función/pasarela de traducción de dirección de red NAT;
- 50 - la caja 5 de comunicación luego procesa la orden y emite un mensaje de "*Inform request*", de acuerdo con el estándar TR-069, con destino al servidor 27 de configuración automática ACS (flujo 81) para notificarle el procesamiento de la petición;
- 55 - el manejo de la caja 5 de comunicación (o del equipo 1), seguido entonces por el protocolo TR-069, como cualquier tipo de topología de red con un CPE (aquí la caja 5 de comunicación).

La figura 26, se ilustra para el mismo mecanismo, los flujos de datos principales que permiten transmitir una petición 16 de acción hacia un accionador asociado con un equipo 1:

- 60 - un actor 37 externo tal como un consumidor de acción envía (flujo 82) una petición 16 de acción hacia el gestor 14 de acciones de la plataforma 4 central de datos;
- el gestor 14 de acciones almacena entonces dicha petición 16 de acción en la base 15 de datos, a continuación, entra en contacto entonces con el servidor 74 STUN (flujo 83), de acuerdo con la memoria descriptiva TR-111, y le solicita enviar un mensaje "*Action Request*" en UDP, con destino a la caja 5 de comunicación. Según diversos modos de realizaciones, no estando el mensaje "*Action Request*" descrito en la memoria descriptiva TR-111, se puede inspirar en la estructura del mensaje "*Connection Request*" descrito en la parte 2.2.2.3 de esta memoria descriptiva, para
- 65

implementar este mensaje. A continuación, se proporciona un ejemplo de realización del mensaje "Action Request". Esto especifica una ruta no vacía asociada con el identificador uniforme de recurso llamado "URI" de este mensaje. Para construir este mensaje, se agrega la ruta resaltada "ActionRequest", entre un campo de tipo "hostname", aquí "10.1.1.1:8080", y los argumentos de este mensaje especificados después del "?":

```
http://10.1.1.1:8080/ActionRequest?ts=1120673700&id=1234&un=CPE57689&cn=XTG
RWIPC6D3IPXS3&sig=3545F7B5820D76A3DF45A3A509DA8D8C38F13512
```

- El servidor 74 STUN es entonces el único elemento autorizado por el dispositivo 25 de acceso integrado IAD para utilizar la función NAT hacia la caja 5 de comunicación. En efecto, tras el establecimiento de la conexión con el servidor 74 STUN, el dispositivo 25 de acceso integrado IAD conoce solo la dirección relativa al servidor 74 STUN, así como el puerto UDP utilizado con este servidor;

- el servidor 74 STUN envía (flujo 84) luego la caja 5 de comunicación un mensaje UDP de petición de acción "Action Request", hacia el puerto UDP público del dispositivo 25 de acceso integrado IAD (utilizado por la función/pasarela de traducción de dirección de red NAT para establecer la conexión con la caja 5 de comunicación). El dispositivo 25 de acceso integrado IAD luego transmite este mensaje a la caja 5 de comunicación aplicando la función de traducción de dirección de red NAT;

- la caja 5 de comunicación recibe el mensaje y emite un mensaje de confirmación (respuesta de acción) "Action Response", de acuerdo con el estándar TR-111, con destino al servidor 74 STUN. Ventajosamente, el mensaje "Action Response", es un mensaje UDP realizado de manera similar al mensaje "Action Request", a través del uso de una ruta no vacía. A continuación, se proporciona un ejemplo de realización de este mensaje:

```
http://10.1.1.1:8080/ActionRequest?ts=1120673700&id=1234&un=CPE57689&cn=XTG
RWIPC6D3IPXS3&sig=3545F7B5820D76A3DF45A3A509DA8D8C38F13512
```

- Mientras el servidor 74 STUN no reciba esta respuesta de acuse de recibo "Action Response", por ejemplo, cuando la función NAT UDP no está establecida, el servidor 74 STUN está configurado para reemitir periódicamente el mensaje UDP "Action Request" o esperar al restablecimiento de la función de traducción de dirección de red NAT;

- la caja 5 cuadro de comunicación luego inicia (por vía ascendente) una conexión http(s) hacia el gestor 14 de acciones (flujo 85) con el fin de recuperar las solicitudes de acciones en espera;

- el gestor 14 de acción proporciona entonces, en respuesta, las solicitudes de acción a la caja 5 de comunicación, a la vez que se gestiona la expiración de las solicitudes de acciones según dos modos:

- o según un modo asíncrono si la caja 5 de comunicación no está disponible, a través de un mensaje UDP de activación;
- o según un modo síncrono, por una transmisión "en tiempo real" de las acciones a la caja 5 de comunicación si está disponible (sujeto a la validez temporal de la acción, función por ejemplo de umbral temporal configurado previamente);

- la caja 5 de comunicación ejecuta entonces la acción solicitada o, de lo contrario, la transmite al equipo 1 correspondiente para su ejecución, después devuelve en seguida un informe al gestor 14 de acción;

- si se solicita, el gestor 14 de acciones luego notifica (flujo 86) al actor 37 externo de la ejecución exitosa de su petición 16 de acción inicial por parte de la caja 5 de comunicación o el equipo 1 en cuestión.

Por otro lado, para esta arquitectura, los principales flujos de datos que permiten rastrear una medición de un sensor asociado con un equipo 1, hacia el gestor 44 de mediciones de la plataforma 4 central de datos, son los mismos que los de la figura 19, que describe estos flujos para un mecanismo que implementa la UPnP:

- la caja 5 de comunicación recupera la medición procedente de dicho sensor y la transmite el equipo 1;
- la caja 5 de comunicación inicia (flujo 56) luego una conexión, por ejemplo, de tipo "https(s)" hacia el gestor 44 de mediciones de la plataforma 4 central de datos;
- la caja 5 de comunicación transmite finalmente la medición mediante un mensaje que indica que el objetivo es el gestor 44 de mediciones, por ejemplo, utilizando un mensaje del tipo "GET http(s)".

Los desafíos técnicos y los modos de realización que se refieren a la caja 5 de comunicación se describen ahora. Como se expuso anteriormente, una caja 5 de comunicación (de tipo de "box") se despliega en un entorno o lugar físico determinado, comprendiendo uno o varios equipos 1, proponiendo cada uno de estos equipos 1 servicios de usuario. Ventajosamente, para cada caja 5 de comunicación sirve:

- de pasarela protocolaria entre las redes (cableadas o inalámbricas) de los equipos 1 locales, sensores, accionadores y la plataforma 4 central de datos, este último accesible a través de una red 26 externa desde un dispositivo 25 de acceso integrado IAD;
- tras la ejecución de servicios, cuando es preferente ejecutarlos localmente en lugar de al nivel de la plataforma 4 central de datos, por ejemplo, por razones de autonomía de los equipos 1.

La caja 5 de comunicación se comparte, por otro lado, entre varios actores, por ejemplo, vendedores de equipos 1 o proveedores 7 de servicios de usuario. En consecuencia, la caja 5 de comunicación aloja componentes de software comunes y específicos (en particular, los servicios de usuario) para dichos actores, permitiendo gestionar la comunicación con los diferentes equipos 1 o efectuar un procesamiento de datos local sobre los datos procedentes de estos equipos 1. Se habla comúnmente, en este contexto, de arquitectura de "tenencia múltiple". La caja 5 de comunicación implementa de este modo una plataforma intermedia de servicios. En esta plataforma, las funciones propias de cada actor, por ejemplo, el protocolo de comunicación empleado con un equipo 1 o el procesamiento específico de datos asociados con un equipo 1, son desplegados por los actores en forma de componentes de software orientados a servicios. Un fallo o mal uso de un componente de software asociado con un actor, no debe afectar ni degradar los comportamientos de los componentes de software relativos a otros actores alojados en una misma plataforma. A modo de ejemplo, un desbordamiento de memoria resultante de una anomalía de codificación de un servicio no debe propagarse en la plataforma, a riesgo de provocar la interrupción ("*crash*") de otros servicios. Además, en ausencia de estándar, los equipos 1 conectados a la caja 5 de comunicación son heterogéneos y están basados en diferentes protocolos de comunicación. De este modo, comúnmente un componente de software que se comunica inicialmente con los equipos 1 según un protocolo, tendrá que evolucionar cuando el equipo 1 cambie ya que el protocolo asociado también está evolucionando. Además, el cambio de un equipo 1 también puede generar una modificación del método de acceso de la caja 5 de comunicación a los datos del equipo 1. A modo de ejemplo, la caja 5 de comunicación accede inicialmente a los datos de un equipo 1 a través de un método de tipo "*pull*": el equipo 1 expone una interfaz de programación aplicativa API y la caja 5 de comunicación toma la iniciativa de recuperar las mediciones de un sensor asociado con este equipo 1. Un cambio de este equipo 1 conduce entonces al uso de un método de acceso de tipo "*push*": los datos se insertan por iniciativa del equipo 1 hacia la caja 5 de comunicación. En consecuencia, el componente de software que permite comunicarse con este equipo también debe evolucionar.

De este modo, para superar sus inconvenientes se realiza en la caja 5 de comunicación, una plataforma de software de servicios basados en el lenguaje Java y de acuerdo con el estándar OSGi (acrónimo inglés de "*Open Services Gateway initiative*"), tenencia múltiple e integrando una interfaz de programación aplicativa de API unificada de acceso a los equipos 1 o servicios conectados. Ventajosamente, tal plataforma de software permite asegurar:

- una fiabilidad y un fuerte aislamiento entre los componentes de software de diferentes usuarios;
- una independencia de los servicios aplicativos con respecto a los protocolos de comunicación con los equipos 1 o los servicios de datos;
- una independencia de los servicios aplicativos con respecto a los métodos de acceso (por ejemplo: "*pull*", "*push*") a los equipos 1 o a los servicios de datos.

Tal plataforma de software se realiza mediante la agregación:

- de una capa de acceso a los equipos 1 conectados, cuya realización se describe posteriormente;
- de una plataforma java de tenencia múltiple que asegura el aislamiento entre los servicios alojados por la plataforma. Se puede, para este tipo de plataforma, apoyarse en plataformas existentes de la técnica anterior. Como ejemplo, una plataforma Java/OSGi basada en un modelo Haas (acrónimo inglés de "*Hardware as a Service*") se realiza a través de componentes de aplicativos dispuestos en un contenedor compartido llamado "*Kernel*" y en contenedores separados y estancos llamados "*Features*". Ventajosamente, tal plataforma se realiza de modo que se pueda acceder al espacio de memoria asignado a cada contenedor, solo si este último configurado para autorizar el acceso. El acceso de un usuario autorizado a un contenedor, por ejemplo, durante la invocación de un método, a continuación, se efectúa a través de una interfaz aplicativa de programación API, que limita el acceso del usuario a las necesidades básicas. Por lo tanto, es imposible utilizar técnicas de desbordamiento de pila para evitar el control y acceder ilícitamente a la zona de memoria del contenedor desde el exterior. Ventajosamente, la memoria del contenedor no puede ser accedida o corrompida desde el exterior del contenedor, el contenedor compartido "*Kernel*" y los contenedores estancos "*Features*" ejecutándose en el seno de una máquina virtual Java JVM, y ésta solo autorizando en su seno la ejecución de contenedores aislados o comunicándose solo a través de métodos explícitamente expuestos y controlados. El código de programación de componentes del "*Kernel*" se considera confiable entonces (sin constar de anomalías), mientras que los servicios de usuario se agrupan en "*Features*" distintas específicas para cada proveedor. Ventajosamente, tal estructura garantiza entonces un fuerte aislamiento entre los servicios de usuario, advirtiendo que cualquier fallo de un servicio tenga un impacto en los otros, asegura una gestión de los recursos de hardware (por ejemplo: ocupación de memoria, recursos de computación) por "*Features*" y limita la propagación de errores de cualquier componente de software.

La figura 27 ilustra el concepto de funcionamiento de una plataforma de software con las características descritas anteriormente. La plataforma de software 87 es una plataforma compartida entre varios usuarios, dos en este ejemplo, y aloja todos los servicios de los usuarios, cada uno relacionado con un dispositivo. En este ejemplo, un primer servicio 88 se refiere a un primer equipo 89, y un segundo 90 servicio se refiere a un segundo equipo 91. El primer equipo 89 y el segundo equipo 91 están conectados externamente a la plataforma 87 de software de la caja 5 de comunicación, y utilizan para sus intercambios respectivamente un primer y un segundo protocolo. Por otro lado, en otro modo de realización, cualquier servicio remoto/alejado, accesible a través de una interfaz de programación aplicativa API, puede sustituir a un equipo 89, 91. Se entiende, aquí, por servicio remoto/alejado cualquier servicio que permite generalizar

una fuente de datos tal como un equipo 89,91 o cualquier servicio desplegado en el software de tipo "Nube" de tipo "software como un servicio" llamado SaaS, acrónimo inglés de "*Software as a Service*". Por ejemplo, el elemento 91 puede ser un servicio remoto en lugar del segundo equipo, utilizando este servicio el segundo protocolo. Ventajosamente, los componentes específicos de software, que comprenden los servicios 88, 90, específicos para cada usuario están aislados respectivamente en contenedores 92, 93 estancos, comúnmente designados con el apelativo "*sandbox*" o "*Feature*". Una interfaz 94 de programación aplicativa de API unificada permite el acceso a los equipos 89, 91 o a servicios remotos desde los servicios 88, 90 aplicativos, o más generalmente a cualquier elemento de software o de hardware exterior, llamado "*eThing*", conectado a la plataforma 87 de software de la caja 5 de comunicación. De este modo, para asegurar tanto el soporte de elementos (por ejemplo: equipos 89, 91 o servicios remotos) gestionados por un usuario, y el soporte de los protocolos de comunicación asociados a estos elementos, se despliega entre la interfaz 94 de programación aplicativa API unificada y cualquier elemento "*eThing*", un adaptador protocolario relativo al protocolo del elemento, así como un adaptador de elemento, dicho adaptador "*eThing*", presentándose estos adaptadores en forma de componentes de software, tales como servicios OSGi.

De este modo, en el ejemplo ilustrado, un primer y un segundo adaptador 95, 96 protocolario está asociado respectivamente con el primer y el segundo equipo 89,91, estando cada uno de estos adaptadores respectivamente interconectado con un primer y un segundo adaptador 97,98 "*eThing*", implementando estos adaptadores la interfaz 94 de programación aplicativa API unificada. Ventajosamente, los adaptadores específicos para el mismo equipo están aislados en el mismo contenedor estanco ("*Feature*"). Por ejemplo, los adaptadores 95, 96 protocolarios y los adaptadores 97, 98 de elemento "*eThing*" están desplegados respectivamente en los contenedores 92, 93. Sin embargo, un mismo adaptador de protocolo puede ser compartido entre varios usuarios. Ventajosamente, la interfaz 94 de programación aplicativa API unificada, los adaptadores 95, 96 protocolarios y los adaptadores 97, 98 de los elementos "*eThing*" constituyen una estructura 99 ("*framework*") de acceso unificado, es decir, una capa de acceso a los equipos 1, 89, 91 conectados.

La figura 28 ilustra un ejemplo de distribución de los elementos de la plataforma 87 de software anteriormente descrita, combinada con un enfoque Java/OSGi basado en un modelo Haas (acrónimo inglés de "*Hardware as a Service*") realizado por componentes de aplicativos dispuestos en un contenedor 100 compartido llamado "*Kernel*", y en contenedores 101, 102 separados y estancos llamados "*Features*". En este ejemplo:

- el contenedor 100 compartido "*Kernel*", considerado confiable, embarca
 - o interfaces (103) Java que constituyen la interfaz (94) de programación aplicativa API unificada, detalladas posteriormente, y clases de datos correspondientes a los objetos intercambiados al nivel de la interfaz 94 de programación aplicativa, por ejemplo, clases de datos estructurados (por ejemplo: binarios, booleanos) retroalimentados por los equipos 1, 89, 91;
 - o adaptadores 104 protocolarios compartidos por todos los usuarios, es decir, compartidos por diferentes servicios, por ejemplo, adaptadores Zigbee, o puertos en serie;
 - o servicios 105 comunes de usuarios, por ejemplo, un motor de reglas, y un servicio de publicación de datos hacia una interfaz de tipo REST (acrónimo de "*Representational State Transfer*");
- cada contenedor 101, 102 separado "*Features*" (componentes de software), específica para un usuario, embarca
 - o servicios 106, 107 específicos para un usuario, por ejemplo, servicios de procesamiento previo en los datos de un equipo 1, 89, 91;
 - o adaptadores 108, 109 protocolarios específicos para su equipo;
 - o adaptadores 110, 111 de elementos "*eThing*" que implementan la interfaz 94 de programación aplicativa API unificada.

La figura 29 ilustra la realización de la estructura 99 ("*framework*") de acceso unificado, que comprende la interfaz 94 de programación aplicativa API unificada. Este último está interconectado con un adaptador de elementos "*eThing*" a través de una interfaz designada a continuación como interfaz 112 de elemento "*eThing*", siendo esta interfaz un servicio OSGi. En este ejemplo, el adaptador de elementos "*eThing*" es el adaptador 97 de elementos "*eThing*" del primer equipo 89 (no representado). El adaptador 97 de elementos "*eThing*" está él mismo interconectado con el adaptador 95 protocolario. La interfaz de programación aplicativa API 94 unificada, el adaptador de elementos "*eThing*" 97 y el adaptador protocolario 95 constituyen, por lo tanto, la estructura 99 de acceso unificado. En esta estructura, cada equipo 89, o más generalmente cualquier elemento conectado "*eThing*", está representado en la estructura 99 de acceso unificado por un servicio OSGi que implementa la interfaz 112 de elemento "*eThing*", este servicio consta de los siguientes atributos: un estado, un identificador único, un nombre, un vendedor, una versión, un número de serie, una descripción.

De este modo, en el siguiente ejemplo, un adaptador 97 de elementos "*eThing*" implementa la interfaz OSGi "*eThing*", está interconectado con un adaptador 95 protocolaria a través de un método "*bindProtocolAdapter()*", estando este método asociado a cualquier adaptador 95 protocolario con cualquier elemento "*eThing*", y comprende para campos: un estado "*Status*", un identificador único "*UID*", un nombre "*NAME*", un vendedor "*VENDOR*", una versión "*VERSION*", un número de serie "*SERIAL_NUMBER*", una descripción "*DESCRIPTION*"

```

package com.bull.everythink.etmf.api.ethings;

import com.bull.everythink.etmf.api.protocoladapters.ProtocolAdapter;

public interface EThing {
    enum Status {
        UNRESOLVED,
        STOPPED,
        STARTED
    }

    String PROPERTY_NAME = "eThing.name";
    String PROPERTY_VENDOR = "eThing.vendor";
    String PROPERTY_VERSION = "eThing.version";
    String PROPERTY_SERIAL_NUMBER = "eThing.serial.number";
    String PROPERTY_DESCRIPTION = "eThing.description";
    String PROPERTY_UID = "eThing.uid";

    Status getStatus();
    void bindProtocolAdapter(ProtocolAdapter protocol Adapter);
    void unbindProtocolAdapter();

    void start();
    void stop();

    String getUID();
}

```

Las funciones del equipo 1, 89, 91 (o del servicio) están representadas por los servicios OSGi que implementan una interfaz 113 abstracta de función de elementos llamada "eThingFunction" y tres interfaces básicas:

- 5
- una interfaz 114 de alarma llamada "Alarma", que permite generar las alarmas procedentes del equipo, estas alarmas están descritas por una clase 115 llamada "AlarmData";
 - una interfaz 116 de control "Control", que permite controlar el estado de un equipo y/o actuar por encima, y comprendiendo dos funciones
 - 10
 - o una función 117 de control binario llamada "BinaryControl", en el caso de que el estado asociado con el sensor de un equipo sea de tipo binario. El valor de este estado está representado entonces por una clase llamada "BinaryData";
 - 15
 - o una función 118 de control multinivel llamada "MultiLevelControl", en el caso de que el estado del sensor asociado con un equipo comprenda varios estados (multivalor). El valor de este estado es representado por una clase llamada "MultiLevelData";
 - una interfaz 119 de sensor llamada "Sensor", que permite recopilar, periódicamente o no, un valor procedente de un sensor de un equipo 1, 89, 91, según dos funciones:
 - 20
 - o una función 120 de sensor binario llamada "BinarySensor", si el sensor asociado es de tipo binario. El valor es representado por la clase llamada "BinaryData";
 - 25
 - o una función 121 de sensor multinivel llamada "MultiLevelSensor", si el caso del sensor asociado es multivalor. El valor es representado entonces por la clase llamada "MultiLevelData".

Ejemplos de realización de la interfaz 113 abstracta de la función de elementos "eThingFunction", de la interfaz básica "Alarm" y de la clase "AlarmData" se dan a continuación:

```

public interface EThingFunction<D extends Data> {

    String PROPERTY_NAME = "eThing.function.name";
    String PROPERTY_DESCRIPTION = "eThing.function.description";
    String TOPIC_DATA_PUBLISHED = "com/bull/everthink/etmf/ethings/functions/data/PUBLISHED";
    Map<String, ?> getProperties();
    Class<D> supportedDataQ;

}

```

```

public interface Alarm<A extends AlarmData> extends EThingFunction<A> {

}

public class AlarmData implements Data {

    private long timestamp;

    private Map<String, String> metadata = new HashMap<String, String>();

    private int type;

    public AlarmData(int type, Map<String, String> metadata) {
        this(type, new Date().getTime(), metadata);
    }

    public AlarmData(int type, long timestamp, Map<String, String> metadata) {
        this.timestamp = timestamp;
        this.type = type;
        if (metadata != null) {
            this.metadata.putAll(metadata);
        }
    }

    public int getType() {
        return type;
    }

    public Map<String, String> getMetadata() {
        return metadata;
    }

    public long getTimestamp() {
        return timestamp;
    }

}

```

Ventajosamente, un nuevo equipo 1, 89, 91 o el servicio agregado es compatible con el sistema agregando el adaptador 97 de elementos "eThing", a través de una clase "eThingAdapter" implementando las interfaces descritas anteriormente.

5 De este modo, para la comunicación con el primer equipo 89, el adaptador 97 de elementos "eThing" está interconectado con el adaptador 95 protocolario, este último expone un servicio con una interfaz OSGi adaptada al protocolo de comunicación específico de este equipo 89. Ventajosamente, las interfaces específicas adaptadas a los protocolos de comunicación, constan de dos interfaces básicas llamadas "CommProtocolAdapter" y "ProtocolAdapterFactory".

10 Cada servicio en la interfaz "ProtocolAdapter" expone al menos las siguientes dos propiedades: tipo, vendedor. Cada tipo de adaptador 95 protocolario define entonces una interfaz OSGi que amplía la interfaz "CommProtocolAdapter". Extendiendo una interfaz (compuesta por métodos, parámetros y valores de retorno asociados), se entiende, aquí, retomar una interfaz existente a un nivel de abstracción más genérico, y para agregar nuevos métodos que le son propios. Además, se asocia a cada adaptador protocolario una fábrica de adaptador protocolario, llamada "Protocol Adapter Factory" (PAF), que permiten crear un adaptador protocolario con las propiedades apropiadas. Ventajosamente, cada fábrica de adaptador protocolario PAF define una interfaz que extiende la interfaz "ProtocolAdapterFactory". El adaptador protocolario creado por la fábrica de adaptador protocolario PAF, luego se asocia con un adaptador de elemento "eThing". El adaptador de elemento "eThing" podrá después

notificar opcionalmente a la fábrica de adaptador protocolario PAF que ya no usa el adaptador protocolario, permitiendo así liberar posibles recursos.

5 Un ejemplo de realización de adaptador 95 protocolario, extendiendo la clase OSGi "*ProtocolAdapter*", presentando para propiedades un tipo "*TIPO*" y un vendedor "*VENDOR*" se dan a continuación:

```
package com.bull.everythink.etmf.api.protocoladapters;
public interface ProtocolAdapter {
    String PROPERTY_TYPE = "protocol.adapter.type";
    String PROPERTY_VENDOR = "protocol.adapter.vendor";
}
```

10 Para este ejemplo, cada adaptador protocolario está asociado a una fábrica de adaptador protocolario PAF, extendiendo la interfaz "*ProtocolAdapterFactory*" y realizada de la siguiente manera:

```
package com.bull.everythink.etmf.api.protocoladapters;

import java.util.Dictionary;

public interface ProtocolAdapterFactory {
    ProtocolAdapter createProtocolAdapter(Dictionary<?, ?> properties) throws
    ProtocolAdapterException;

    void releaseProtocolAdapter(Protocol Adapter protocol Adapter) throws
    ProtocolAdapterException;
}
```

Por otro lado, la estructura 99 ("*framework*") de acceso unificado, más precisamente la interfaz 94 de programación aplicativa, comprende las dos interfaces programáticas siguientes:

- 15 - una interfaz 122 de publicación/suscripción llamada "*publish/subscribe*", que permite, según un modo eventual
- o de datos procedentes de un equipo 1, 89, 91 o de un servicio, después de un paso de abono/suscripción del usuario, hacia la estructura 99 ("*framework*") de acceso unificado (por ejemplo: con destino a un servicio 88, 90 específico);
 - 20 o la publicación asíncrona desde la estructura 99 ("*framework*") de acceso unificado (por ejemplo: a partir de un servicio 88, 90 específico), de datos o de eventos hacia un equipo 1, 89, 91 o servicio relacionado con uno o varios usuarios abonados;
- 25 - una interfaz 123 de petición llamada "*solicitud*" que permite, según un modo síncrono, la recuperación, la lectura/escritura de datos procedentes de/con destino a un equipo 1, 89, 91 o servicio, hacia/desde la estructura 99 ("*framework*") de acceso unificado (por ejemplo: con destino a/desde un servicio 88, 90 específico).

30 Ventajosamente, la interfaz 123 de petición llamada "*request*" permite enviar peticiones sobre los equipos 1, 89, 91 (servicios "*eThing*") datos de lectura (mensaje de tipo "*GET*") o de escritura (mensaje de tipo "*SET*" en un atributo o invocación de una operación). En un modo de realización para crear peticiones hacia un elemento (equipo 1, 89, 91 o servicio), se utiliza un método estático, que comprende como argumentos el identificador "uid" del elemento "*eThing*", así como el nombre de la función de elemento "*eThingFunction*".

35 Luego se usa un método para:

- leer los datos en un equipo (orden "*getData*"):

```
<D extends Data> D getData(Class<D> rawData Type)
```

- 40 - escribir datos hacia un equipo (orden "*sendData*"):

```
<D extends Data> Request sendData(D data)
```

La interfaz 122 de publicación/suscripción llamada "*publish/subscribe*", permite, en cuanto a ella:

- 45 - producir datos en esta interfaz rellenando un identificador de equipo "*eThing*":

```
<D extends Data> Publisher publishData(D data, String eThingName, String eThingVersion, String eThingUid,
String eThingFunctionName)
```

Ventajosamente, esto permite reinyectar datos desde un servicio aplicativo o de un nivel superior al de la interfaz 94 de programación aplicativo API unificada en esta interfaz, por ejemplo, desde un motor de procesamiento de eventos complejos CEP, acrónimo inglés de "*Complex Event Processing*". Tal motor CEP puede ser tanto un consumidor como un productor de datos en la interfaz 94 de programación aplicativo API. En este ejemplo, la producción de datos resulta de la creación de un nuevo evento por el motor CEP (por ejemplo: correlación de otros eventos), siendo estos datos luego consumidos por las aplicaciones a través de la interfaz 122 de publicación/suscripción "*publish/subscribe*";

- suscribirse a los datos producidos en los elementos "*eThings*", definiendo un filtro de eventos llamado "*EventFilter*", que permiten registrar un "oyente" de evento en el servicio, comúnmente designado por el anglicismo "*listener*". Cuando se publiquen los datos correspondientes al filtro, se usa una función de rellamada, comúnmente designada con el anglicismo "*callback*". Ventajosamente, la función de rellamada "*callback*" permite implementar una interfaz 124 de evento relacionada con la función de elemento "*eThing*" llamada interfaz 124 "*EThingFunctionEvent*" que tiene un solo método:

```
void handleEThingFunctionEvent(EThingFunctionEvent<D> event)
```

Un ejemplo de uso de "*listener*", que permite una aplicación consumidora de datos de un equipo 1, 89, 91 registrar una "*callback*" (o invocar un método cuando sus datos estarán disponibles) se proporciona a continuación: se da a continuación:

```
EventFilter eventFilter = EventFilter. eventFilter(context);
MyHandler<Data> hdl = new MyHandler(...);
eventFilter.registerEThingFunctionEventHandler(hdl, Data.class, getName(), getVersion(), getUid(), null);
```

Ventajosamente, el conjunto de servicios aplicativos de usuarios se basa en la interfaz 94 de programación aplicativo API unificada, permitiendo una abstracción de los equipos 1, 89, 91 y de protocolos de comunicación subyacentes. De este modo, la sustitución del equipo 1, 89, 91, por un nuevo equipo basados en un protocolo de comunicación diferente, pero proponiendo los mismos servicios, es decir, las mismas funciones, entonces no tiene ningún impacto en el código de servicio de usuario asociado con la interfaz 94 de programación aplicativo API unificada. Además, la interfaz 94 de programación aplicativo API unificada ofrece un medio para abstraer el modo de acceso a los equipos 1, 89, 91 independientemente de si es síncrono (modo de extracción de datos "*pull*") para la lectura o escritura de datos, o independientemente de si es asíncrono o está vinculado a un evento (modo de datos insertados "*push*") para la lectura de datos de equipos 1, 89, 91, a través de las interfaces 122, 123 de publicación/suscripción "*publish/subscribe*" y de petición "*request*". Ventajosamente, cualquier modo no implementado de forma nativa por un equipo 1, 89, 91 entonces se simula por la estructura 99 ("*framework*") de acceso unificado. La figura 30, ilustra a modo de ejemplo, un equipo 1 que funciona, según un modo síncrono (modo de extracción "*pull*"), interconectado con la estructura 99 ("*framework*") de acceso unificado que comprende la interfaz 94 de programación aplicativo API unificada, estando esta estructura interconectada con un primer servicio de usuario 125 basado en modo síncrono, y un segundo servicio de usuario 126 basado en un modo asíncrono. Por lo tanto, el primer servicio 125 de usuario usa, por lo tanto, la interfaz 123 de petición "*request*", mientras que el segundo servicio 126 de usuario utiliza la interfaz 122 de publicación/suscripción "*publish/subscribe*" de la interfaz de programación aplicativo API unificada 94 de la estructura 99 ("*framework*") de acceso unificado. En este ejemplo, los modos de peticiones que emanan del primer servicio 125 de usuario están representados por la flecha 127, mientras que los modos de publicación/suscripción "*publish/subscribe*" están representados respectivamente por las flechas 128, 129.

El modo síncrono "*request*" en escritura y lectura, se implementa por una simple delegación, es decir, por una simple llamada de método, en la interfaz nativa del equipo 1: luego se usa los mensajes de tipo "*SET*" en la escritura propagada (flecha 130) en el equipo 1 y "*GET*" en la lectura que devuelve el último valor/datos leído en el equipo 1.

Como el modo eventual es asíncrono, este último modo entonces se simula por la estructura 99 ("*framework*") de acceso unificado recuperando periódicamente (flecha 131), a través de la interfaz 123 de petición "*request*", los datos del equipo y memorizando del último dato leído. Este último dato se envía entonces hacia la interfaz 122 de publicación/suscripción "*publish/subscribe*".

Ventajosamente, el período de recopilación es configurable a través de la interfaz 113 abstracta de la función de elementos "*eThingFunction*". La figura 31, ilustra, un equipo 1 que funciona, según un modo eventual (modo "*push*"), interconectado con la estructura 99 ("*framework*") de acceso unificado que comprende la interfaz 94 de programación aplicativo API unificada, estando esta estructura interconectada con un primer servicio 125 de usuario basado en modo síncrono, y un segundo servicio 126 de usuario basado en un modo asíncrono. Por lo tanto, el primer servicio de usuario 125 usa, por lo tanto, la interfaz 123 de petición "*request*", mientras que el segundo servicio 126 de usuario utiliza la interfaz 122 de publicación/suscripción "*publish/subscribe*" de la interfaz 94 de programación aplicativo API unificada de la estructura 99 ("*framework*") de acceso unificado. En este ejemplo, los modos de peticiones que emanan

del primer servicio 125 de usuario están representados por la flecha 127, mientras que los modos de publicación/suscripción "*publish/subscribe*" están representados respectivamente por las flechas 128, 129. A diferencia del ejemplo anterior, el equipo 1 inserta sus datos hacia la interfaz 122 publicación/suscripción "*publicar/suscribir*". De este modo, la flecha 128 se refiere a la propagación de datos que el equipo 1 inserta (flecha 132) en cada nuevo evento. Como el equipo 1 está esta vez basado en modo asíncrono, el acceso unificado de la estructura 99 ("*framework*") simula esta vez el modo síncrono memorizando el último dato insertado por el equipo 1 a través de la interfaz 122 de publicación/suscripción "*publish/subscribe*". Este dato entonces se envía hacia la interfaz 123 de petición "*request*". A continuación, como anteriormente, el modo síncrono "*request*" en escritura y lectura, se implementa por una simple delegación en la interfaz nativa del equipo: luego se usa los mensajes de tipo "*SET*" en la escritura propagada (flecha 130) en el equipo 1 y "*GET*" en la lectura que devuelve el último valor leído en el equipo 1.

REIVINDICACIONES

- 5 1. Caja (5) de comunicación, conectada a un primer equipo (89) y a un segundo equipo (91), comunicándose el primer equipo (89) con la caja (5) a través de un primer protocolo, comunicándose el segundo equipo (91) con la caja (5) a través de un segundo protocolo, alojando la caja (5) de comunicación para cada equipo (89,91) respectivamente al menos un primer servicio (88) y un segundo servicio (90), comprendiendo la caja (5) de comunicación una plataforma (87) de software OSGi, comprendiendo esta plataforma (87) para
- 10 - cada servicio (88,90) de cada equipo (89,91) un contenedor (92, 93) estanco de tipo caja de arena, para garantizar un aislamiento entre los servicios (88,90);
- 15 - una estructura (99) de acceso unificado que permite el acceso a los equipos (89,91), realizada por una interfaz (94) de programación aplicativa configurada para permitir el acceso del primer servicio (88) al primer equipo (89) y del segundo servicio (90) al segundo equipo (91), realizándose el acceso a cada equipo (89, 91) por medio de un adaptador (97,98) de elemento y de un adaptador (95,96) protocolario, implementando estos adaptadores la interfaz (94) de programación aplicativa y siendo desplegados en el contenedor (92, 93) estanco de su servicio (88,90) respectivo.
- 20 2. Caja (5) de comunicación según la reivindicación 1, en donde la plataforma (87) de software está configurada para realizar además un contenedor (100) compartido, proponiendo este contenedor (100) compartido interfaces (103) Java que constituyen la interfaz (94) de programación aplicativa, adaptadores (104) protocolarios compartidos por diferentes servicios, servicios (105) comunes de usuarios.
- 25 3. Caja de comunicación (5) según la reivindicación 1 o 2, en donde la interfaz (94) de programación aplicativa comprende una interfaz (122) de publicación/suscripción configurada para permitir
- la recepción asíncrona de eventos o de datos procedentes de un equipo (89,91);
- la publicación asíncrona de datos o de eventos hacia un equipo (89,91).
- 30 4. Caja de comunicación (5) según una cualquiera de las reivindicaciones 1 a 3, en donde la interfaz (94) de programación aplicativa comprende una interfaz (123) de petición configurada para permitir según un modo síncrono, la recuperación, la lectura y la escritura de datos procedentes o con destino a un equipo (89,91).
- 35 5. Caja de comunicación (5) según la reivindicación 4, en donde la estructura (99) de acceso unificado está configurada para simular un modo de acceso asíncrono Push para un equipo (89,91), cuando dicho equipo (89,91) funciona según un modo síncrono Pull, siendo el modo de acceso asíncrono simulado por la estructura (99) de acceso unificado recuperando de manera periódica un último dato leído a través de la interfaz (123) de petición y devolviendo este valor hacia la interfaz (122) de publicación/suscripción.
- 40 6. Caja de comunicación (5) según la reivindicación 4 o 5, en donde la estructura (99) de acceso unificado está configurada para simular un modo de acceso síncrono Pull para un equipo (89, 91), cuando dicho equipo (89,91) funciona según un modo síncrono Push, siendo el modo de acceso síncrono simulado mediante la estructura (99) de acceso unificado transmitiendo cada último dato insertado en la interfaz (122) de publicación/suscripción a la interfaz (123) de petición.
- 45 7. Producto de programa de ordenador implementado en un soporte de memoria, susceptible de implementarse en el seno de una unidad de procesamiento informático y que comprende instrucciones para la implementación de la plataforma de software de la caja (5) de comunicación según una de las reivindicaciones 1 a 6.

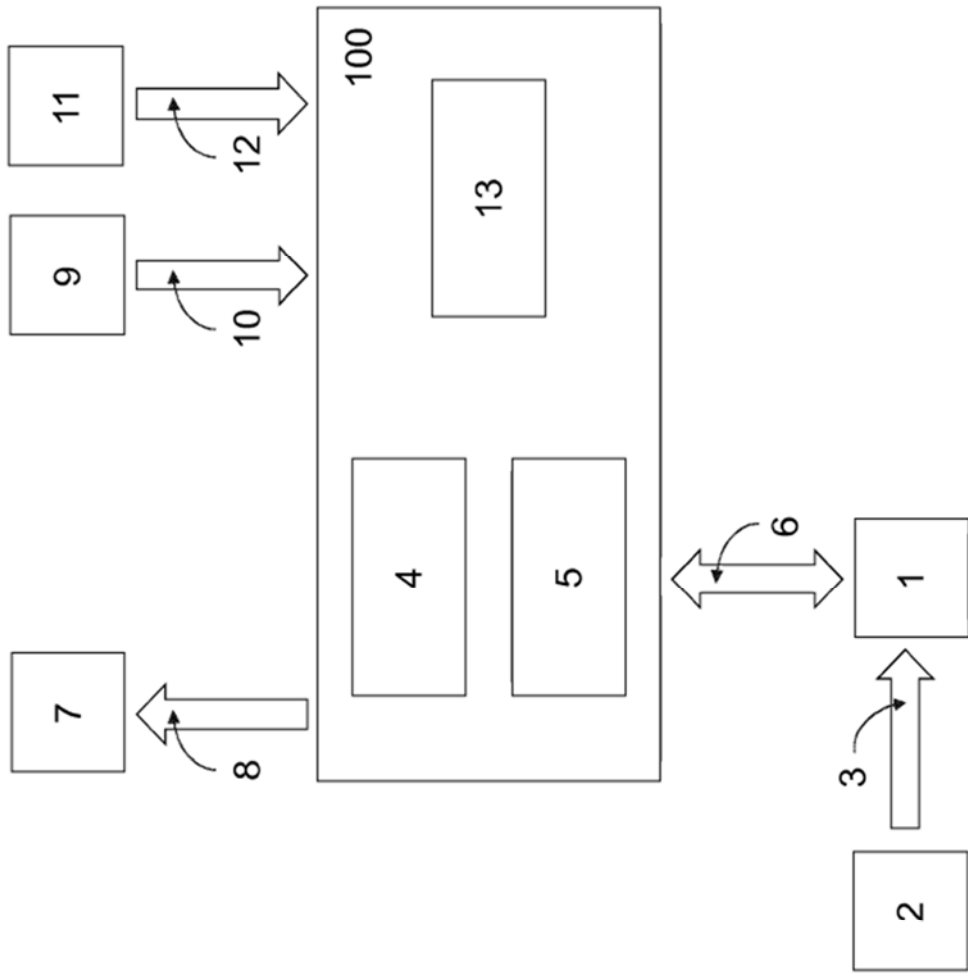


FIG. 1

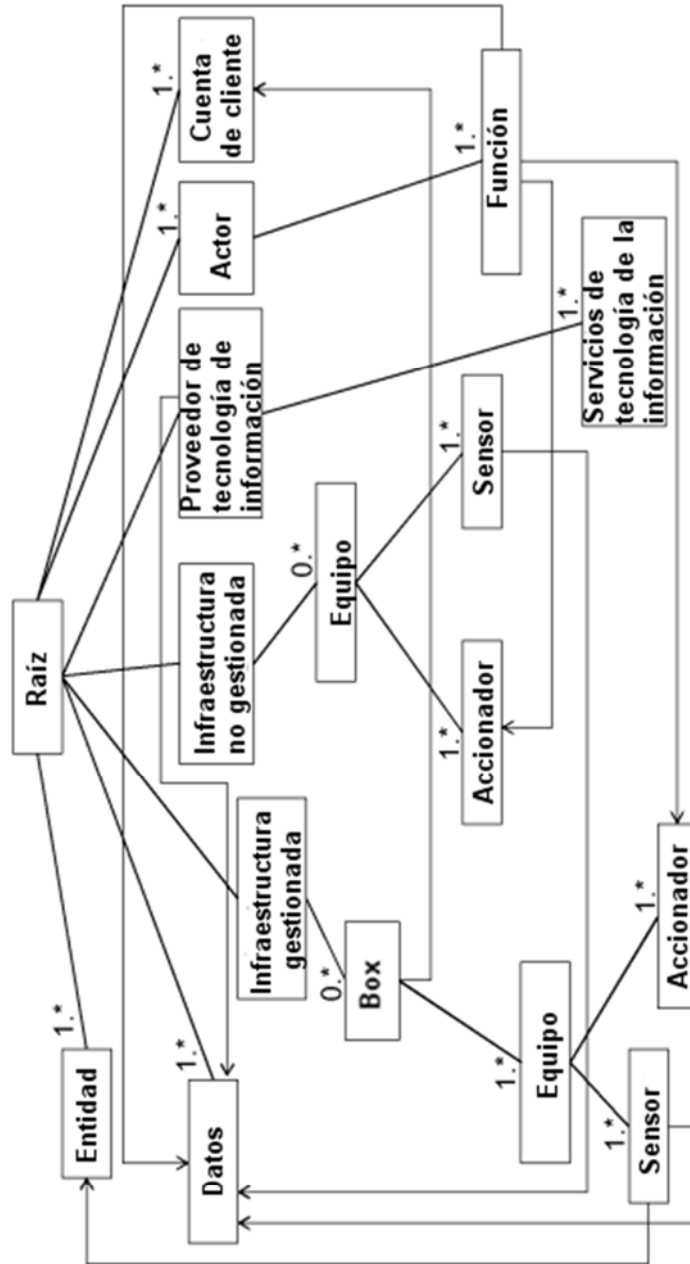


FIG. 2

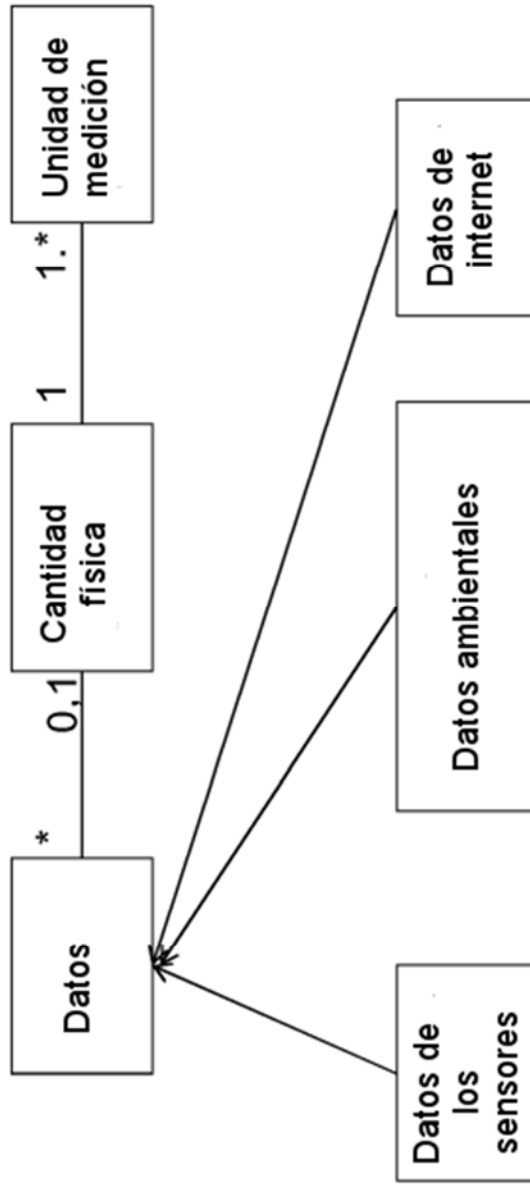


FIG. 3

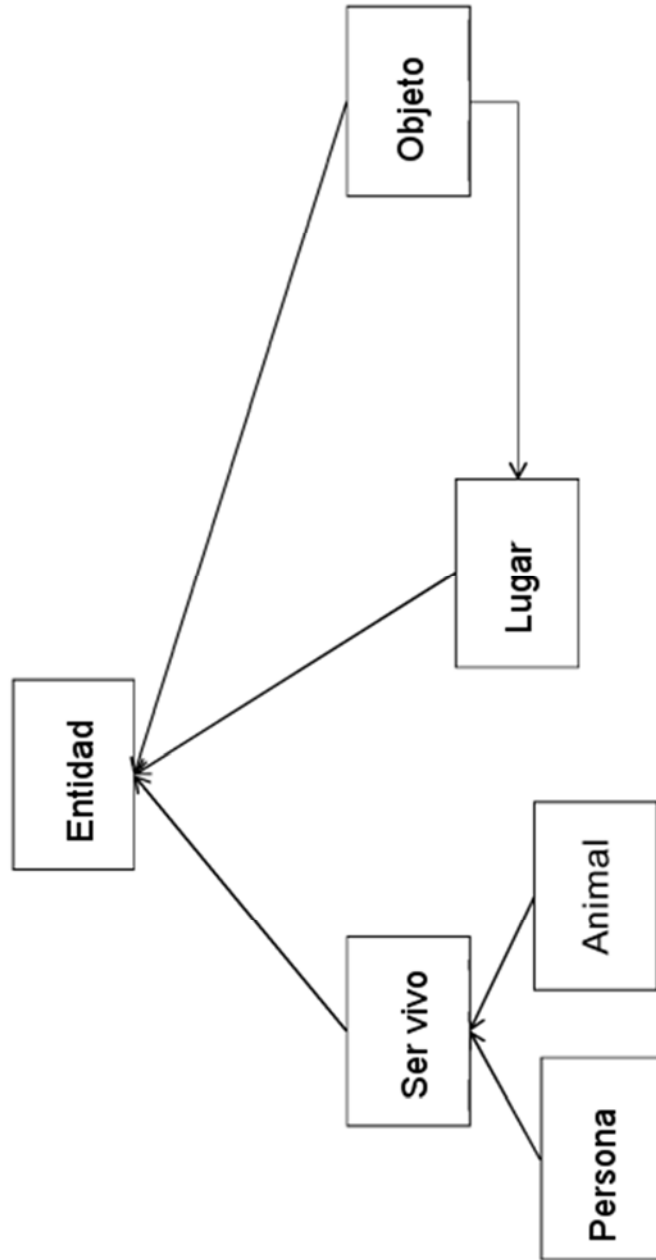


FIG. 4

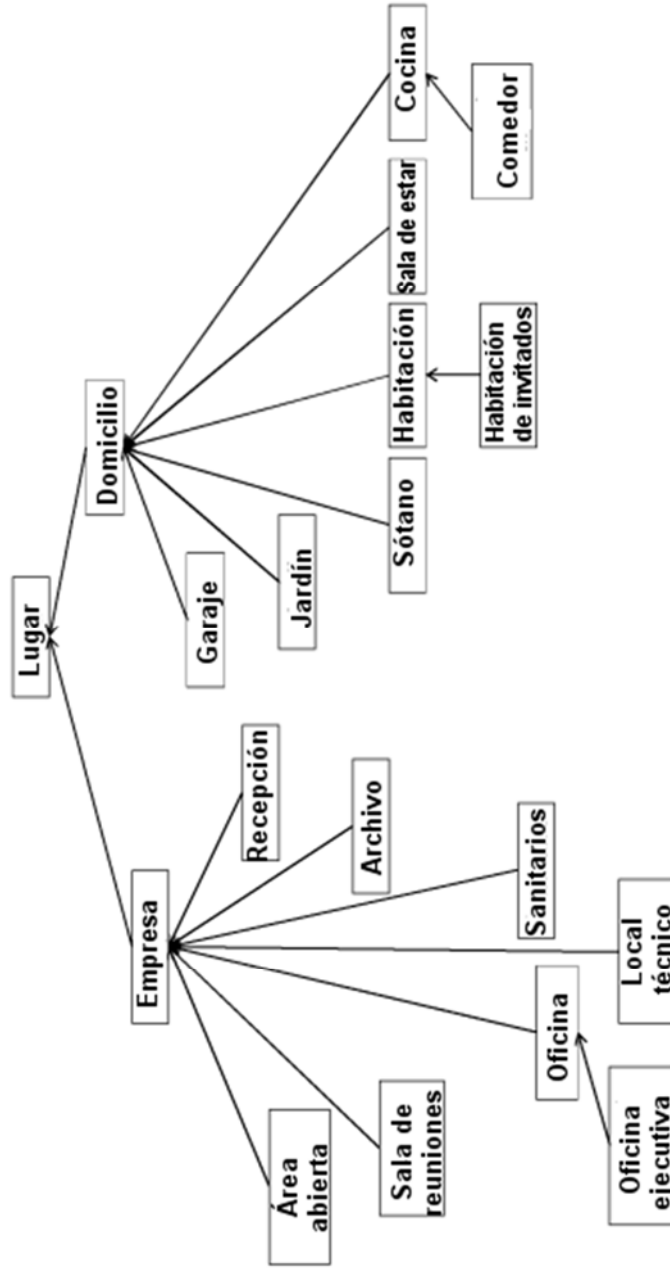


FIG. 5

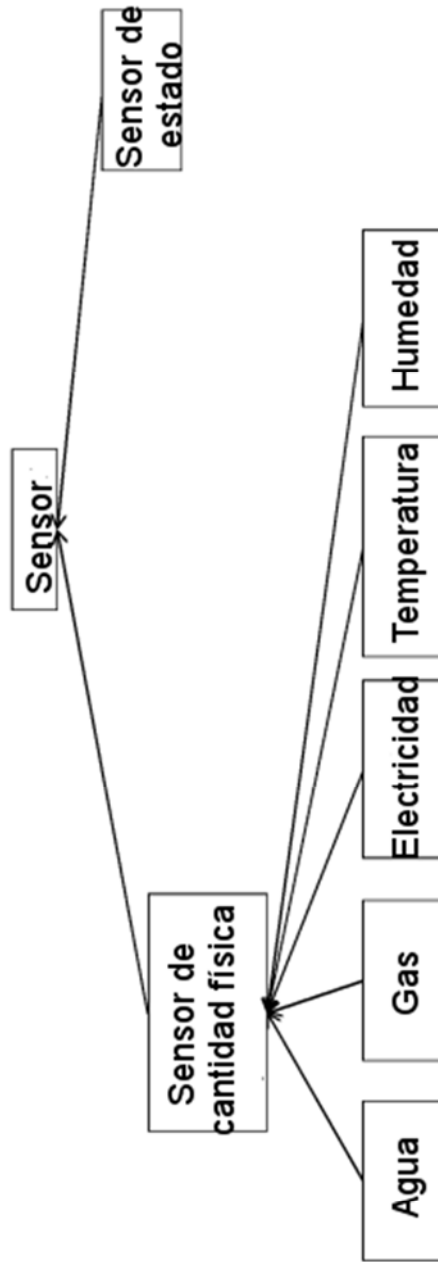


FIG. 6

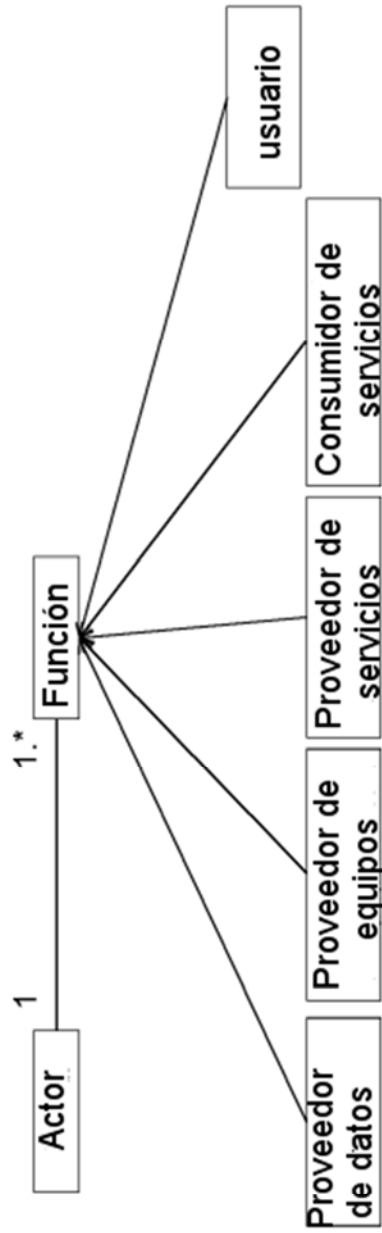


FIG. 7

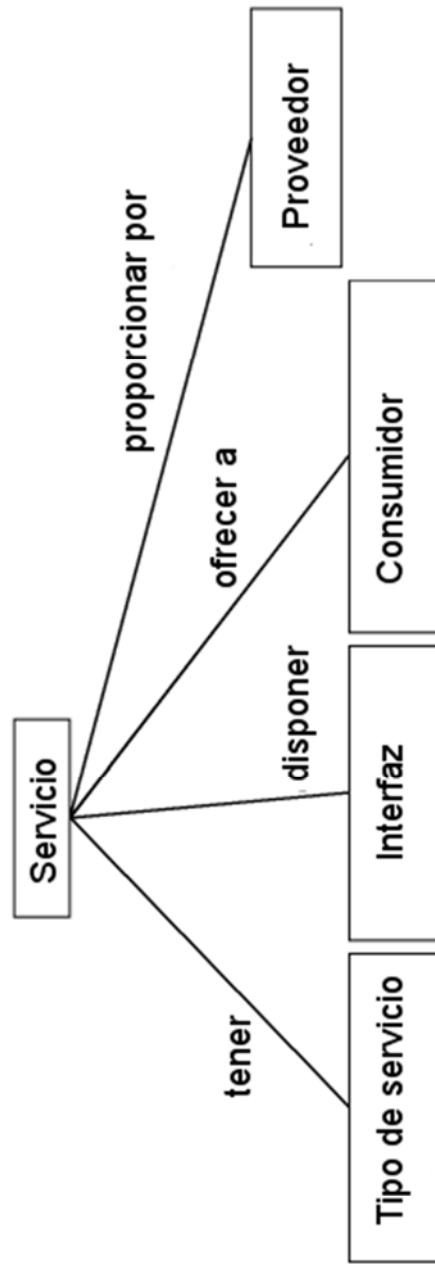


FIG. 8

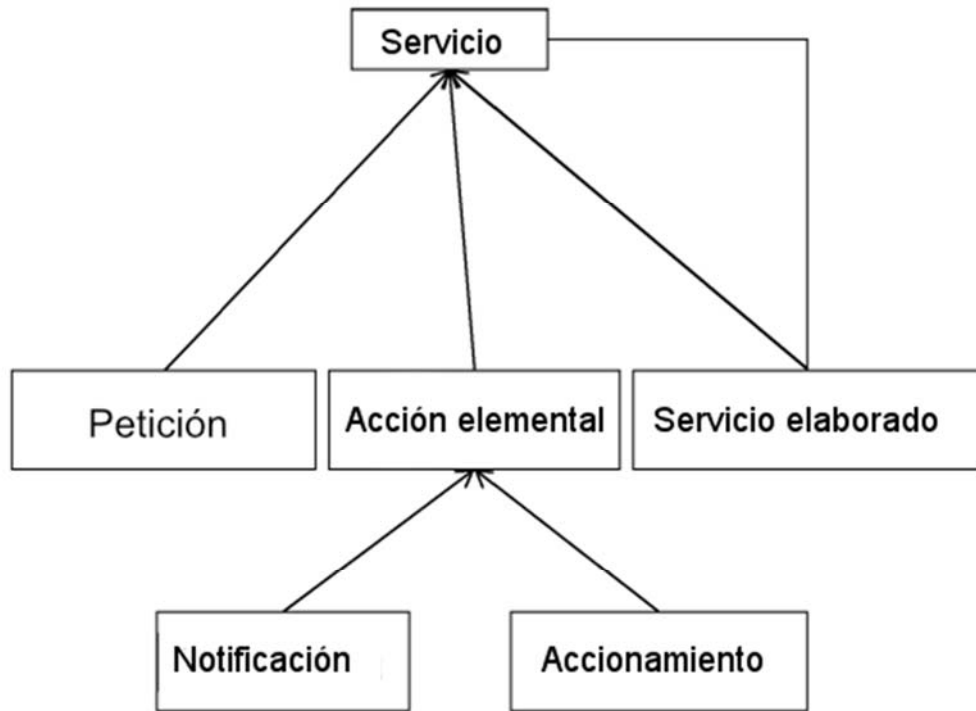


FIG. 9

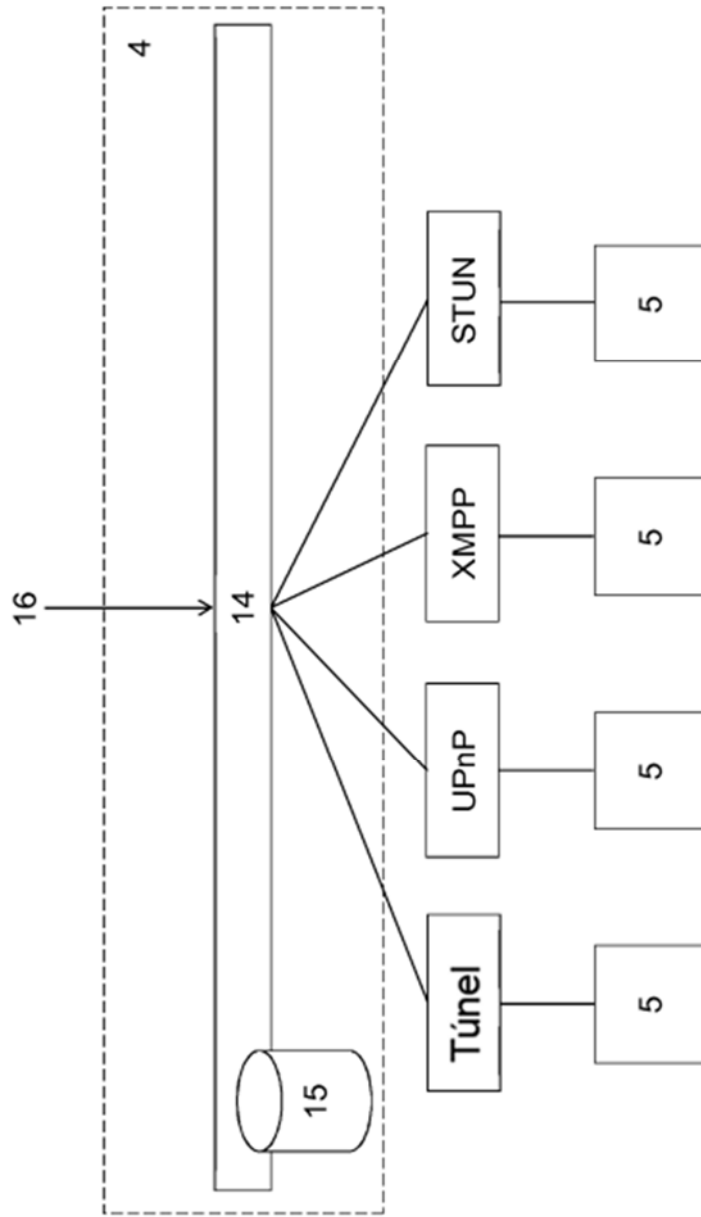


FIG. 10

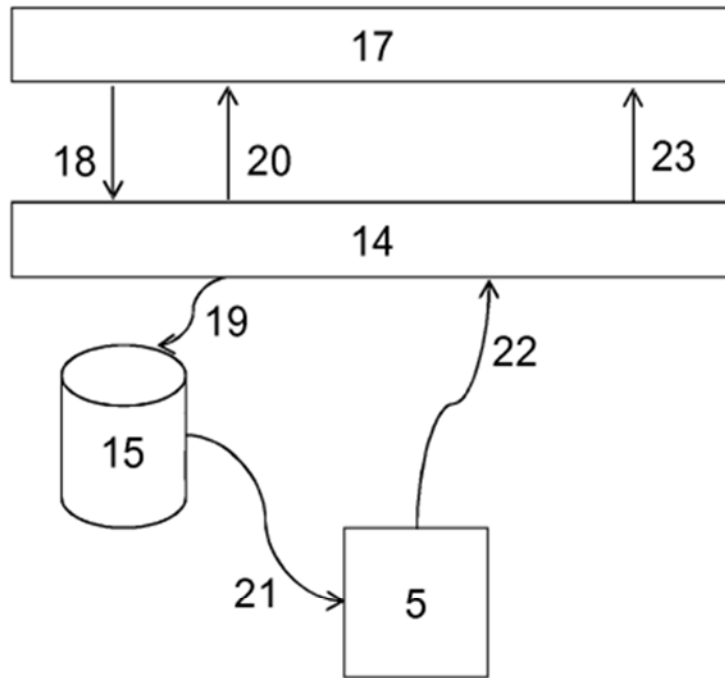


FIG. 11

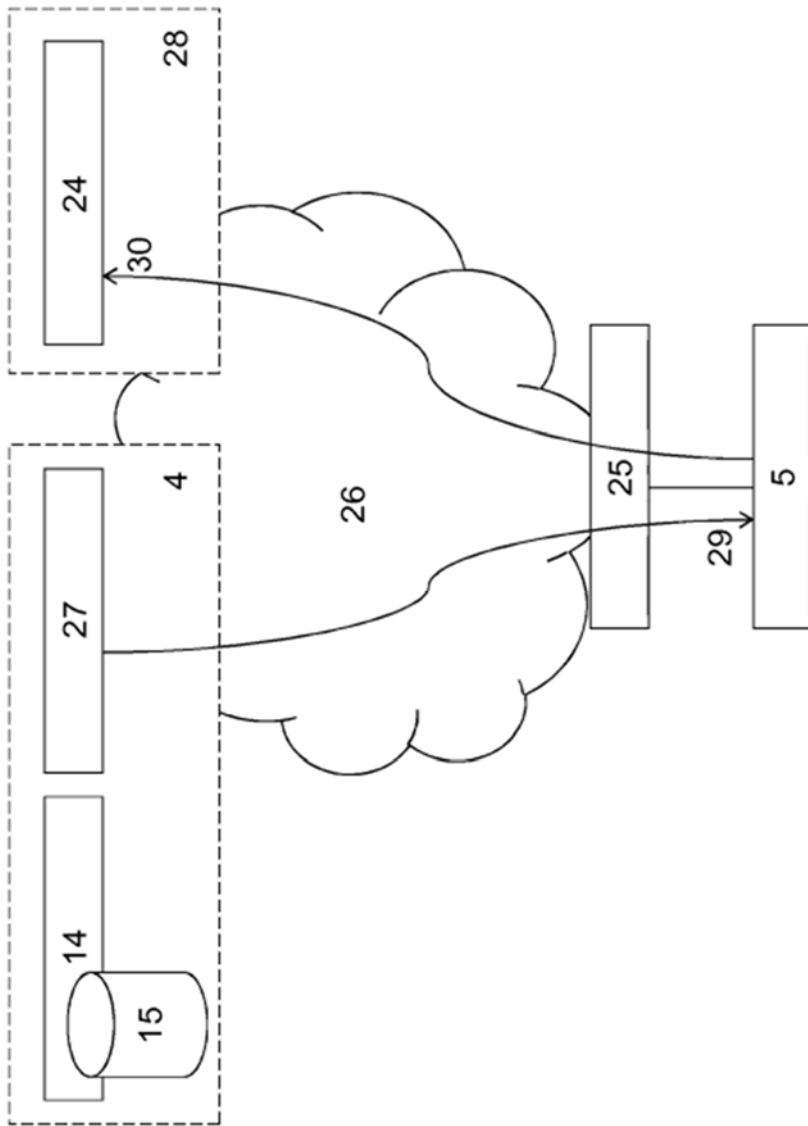


FIG. 12

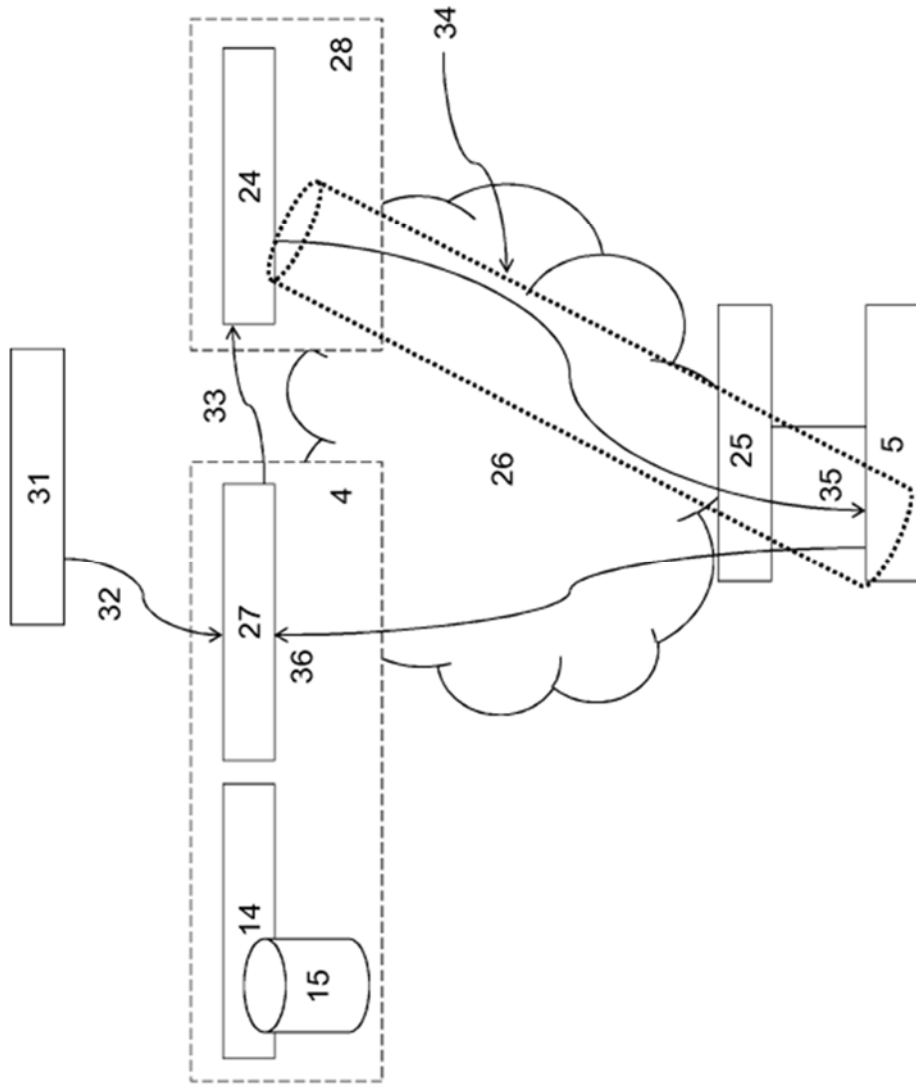


FIG. 13

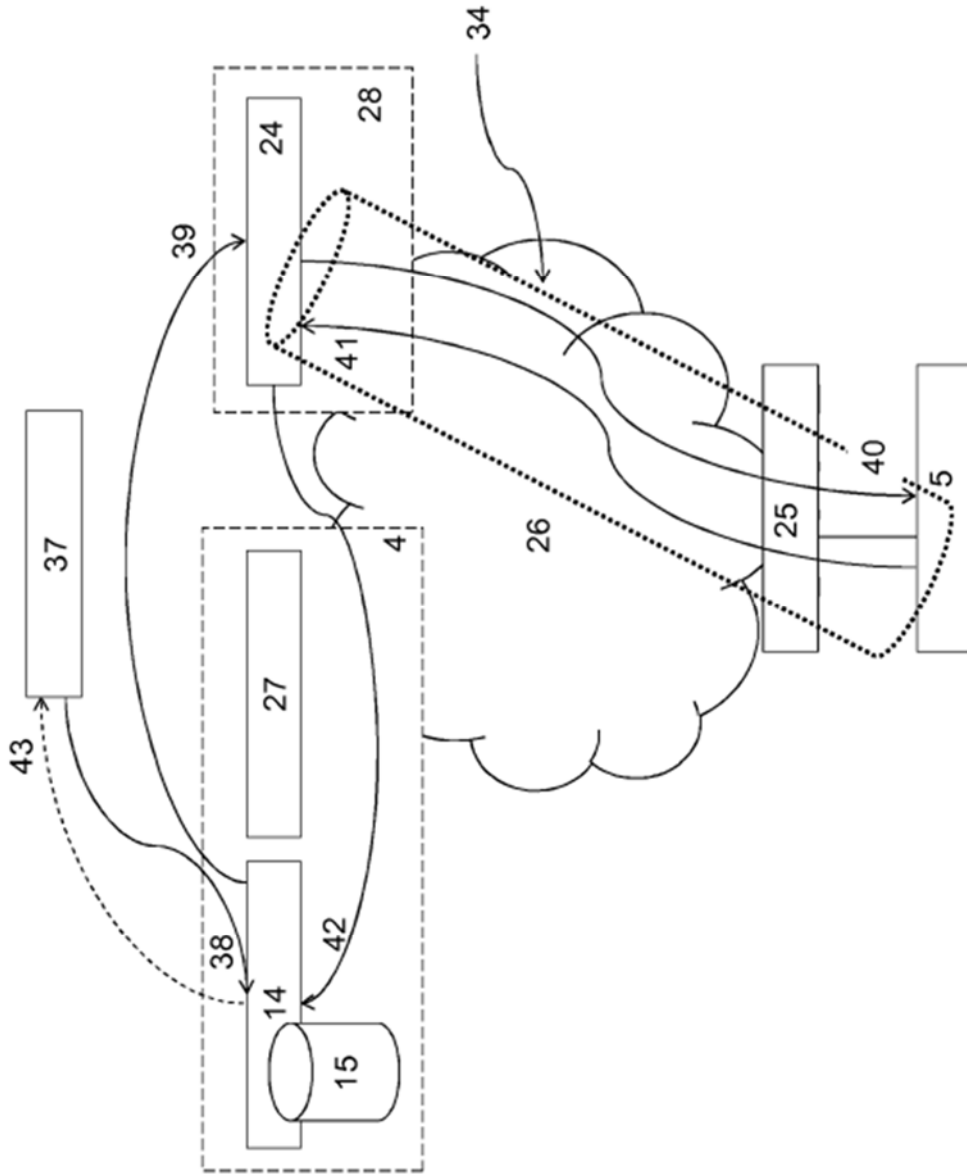


FIG. 14

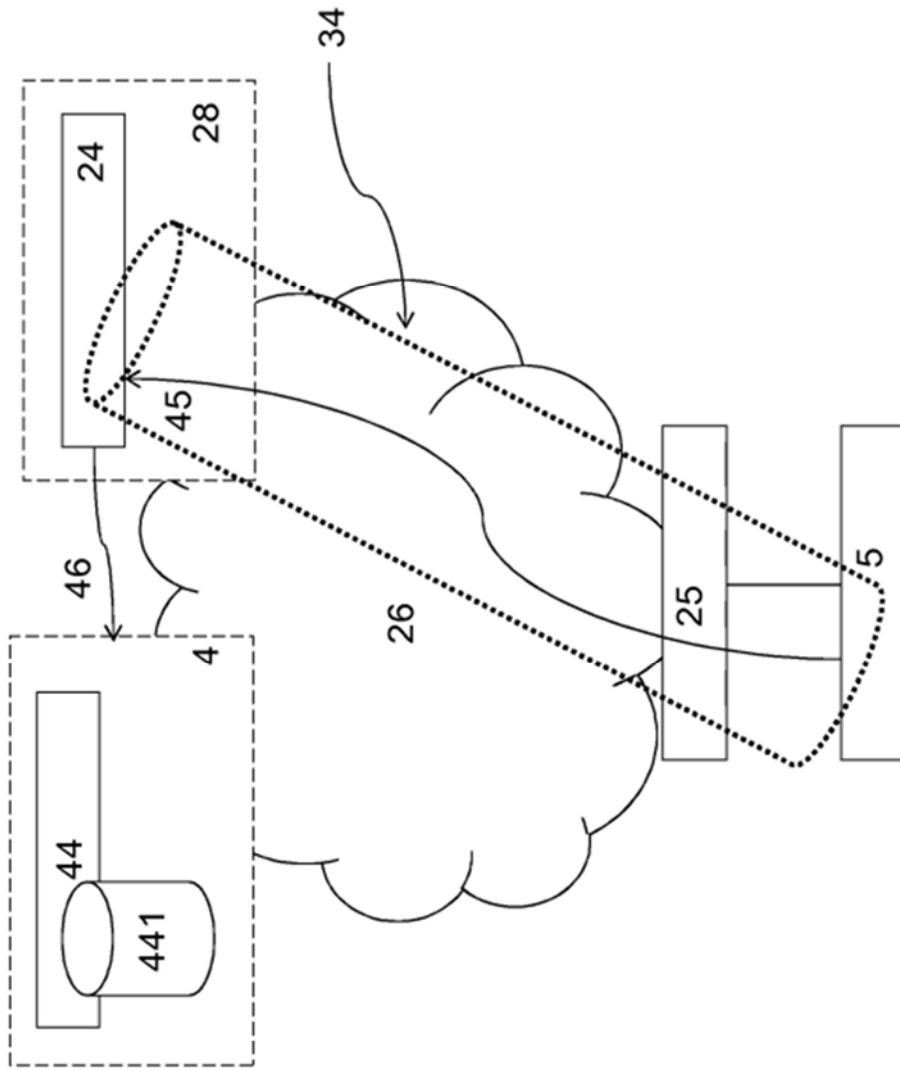


FIG. 15

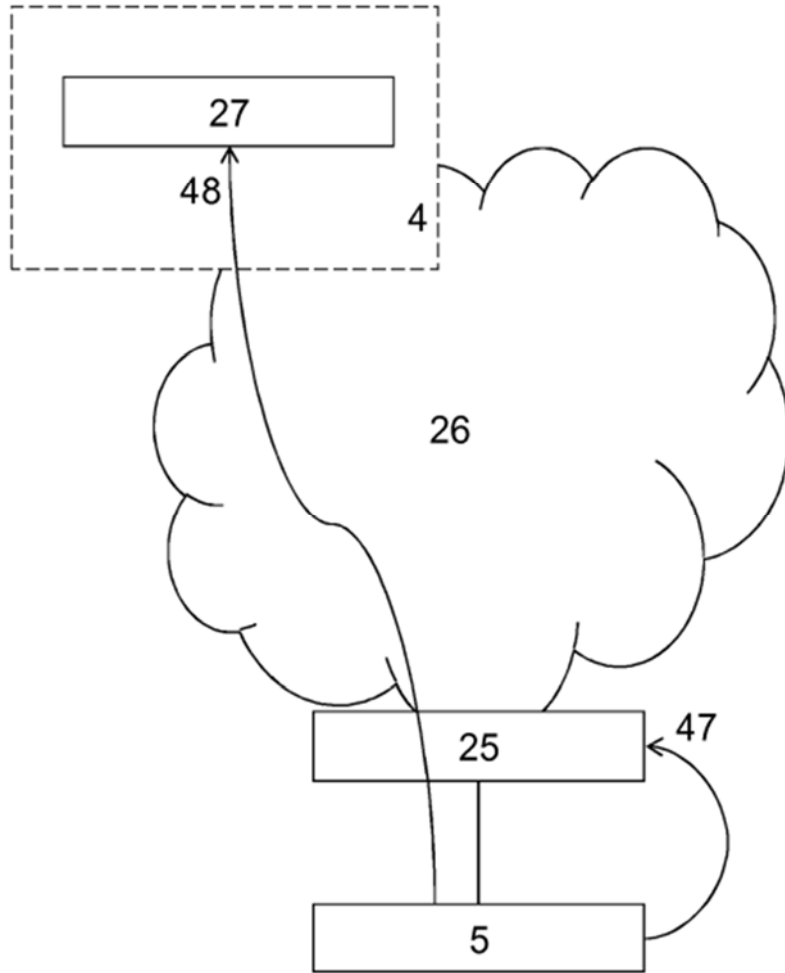


FIG. 16

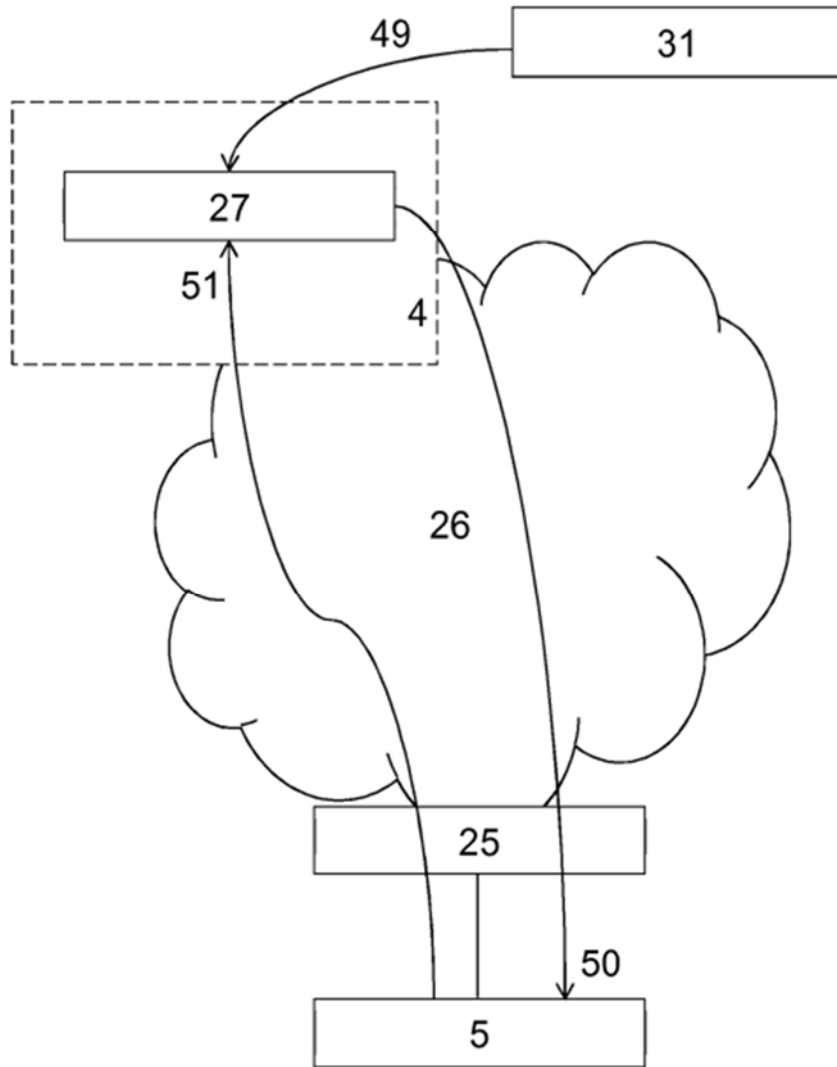


FIG. 17

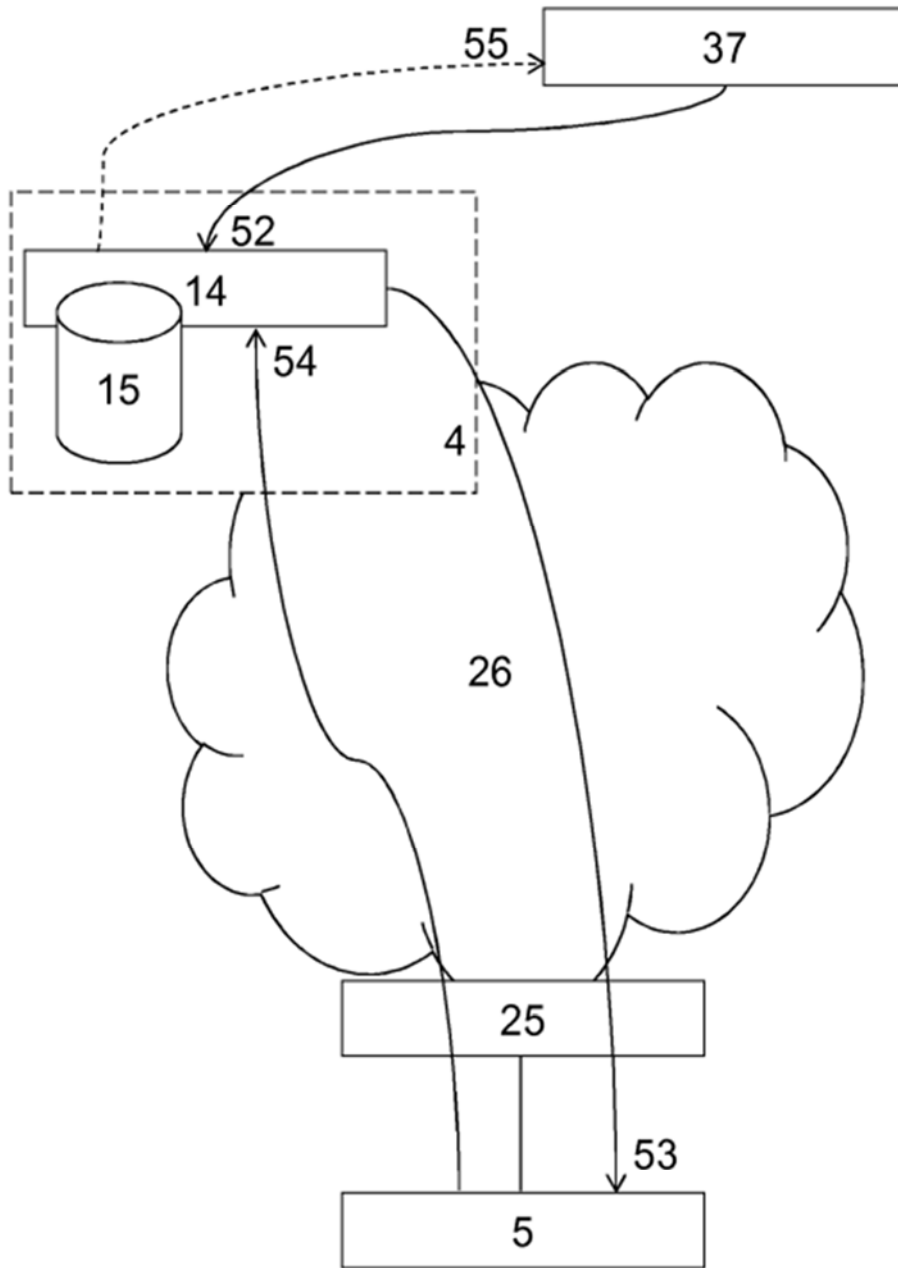


FIG. 18

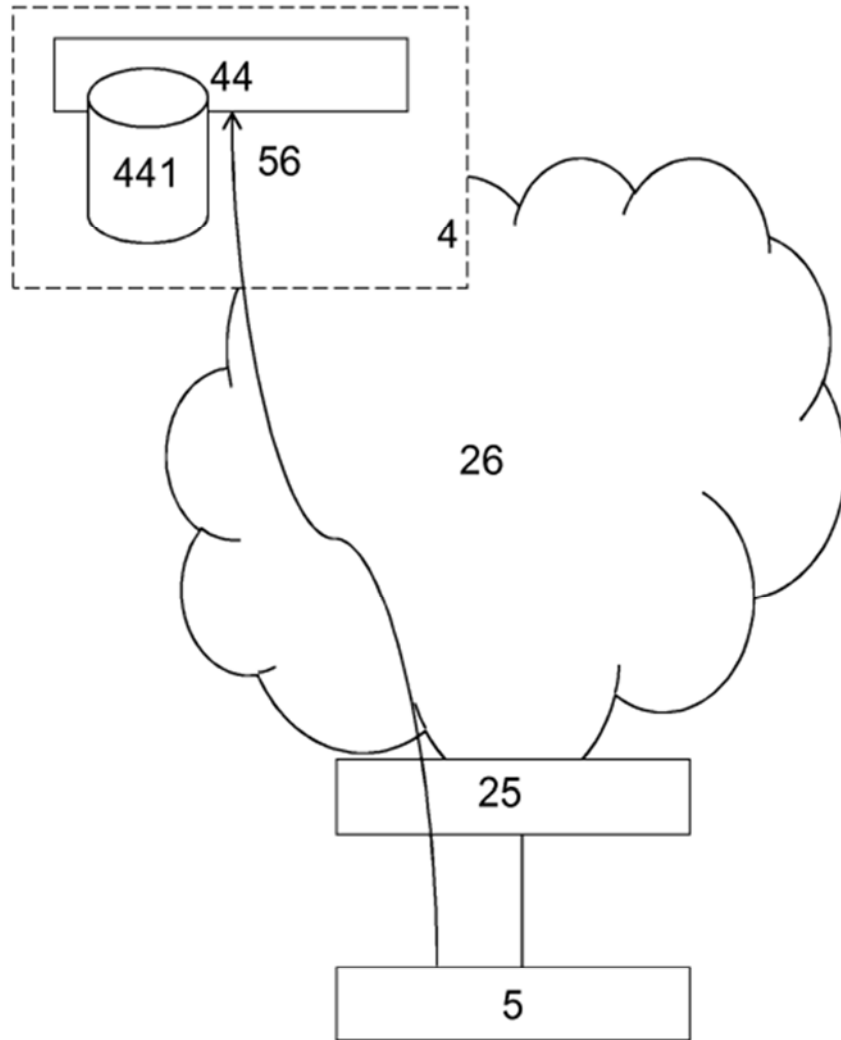


FIG. 19

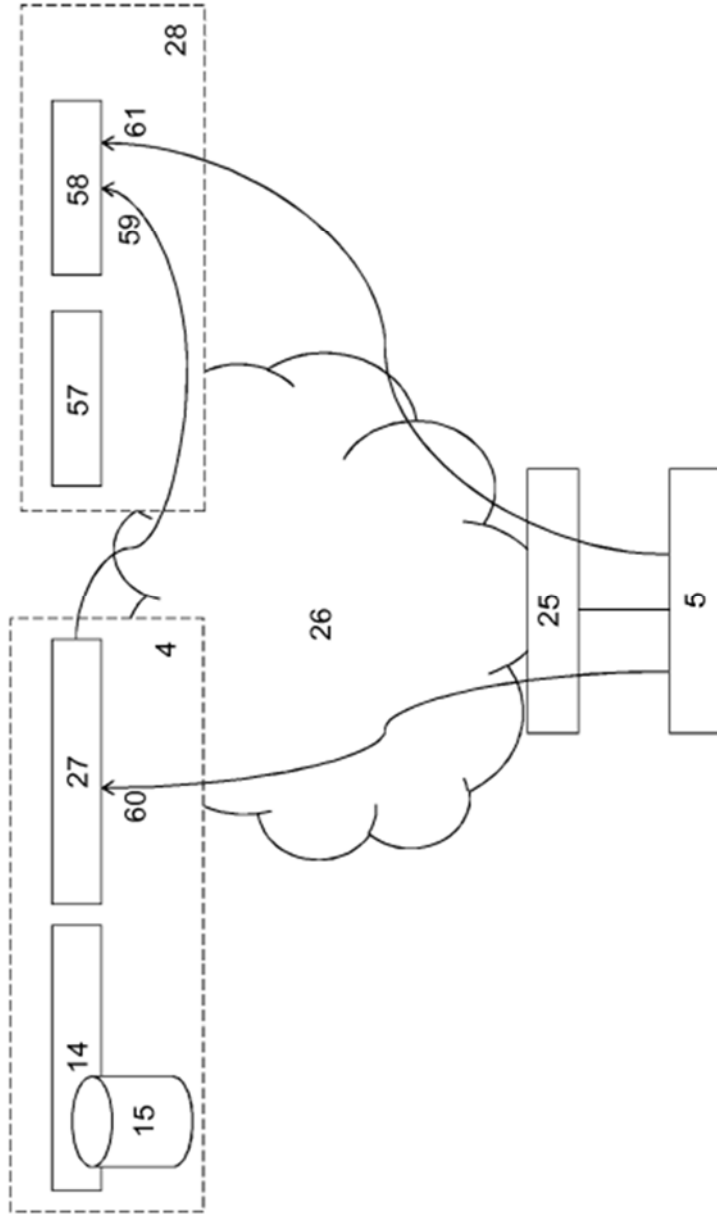


FIG. 20

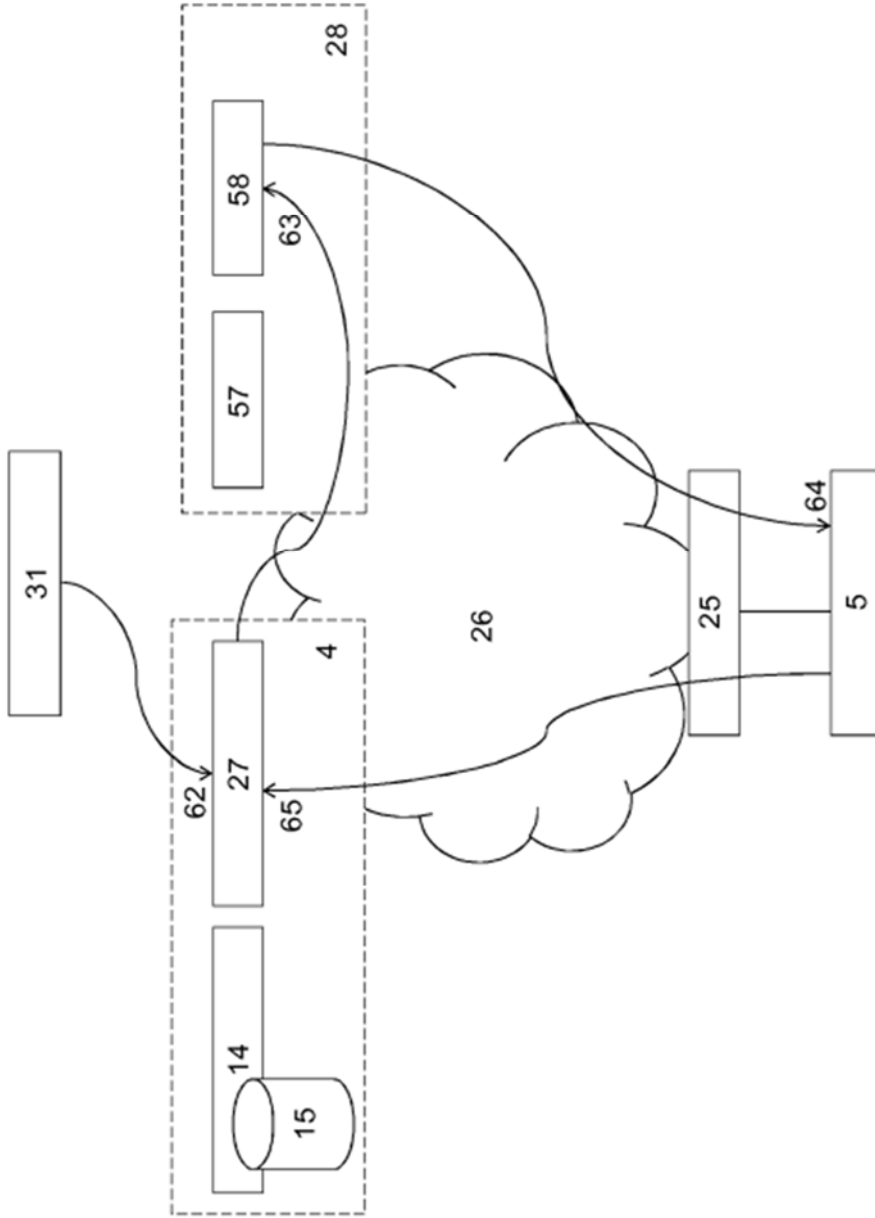


FIG. 21

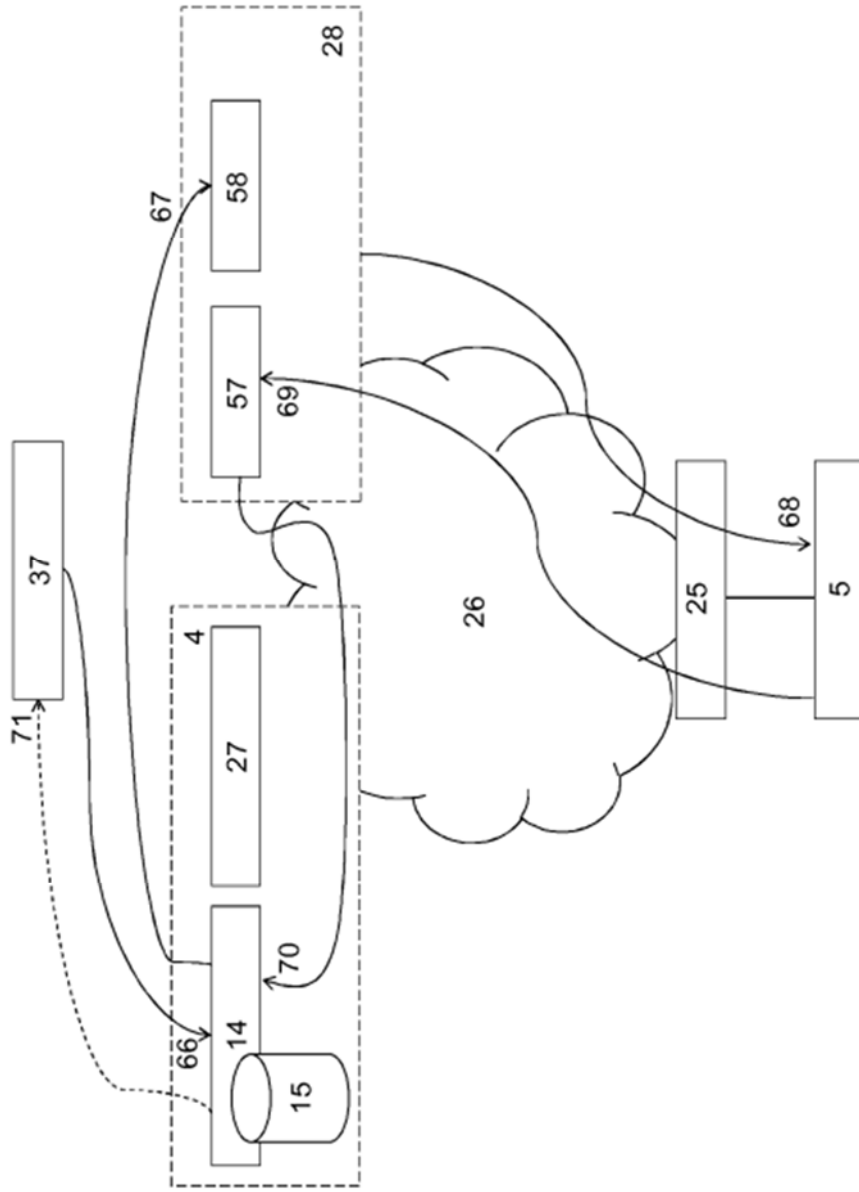


FIG. 22

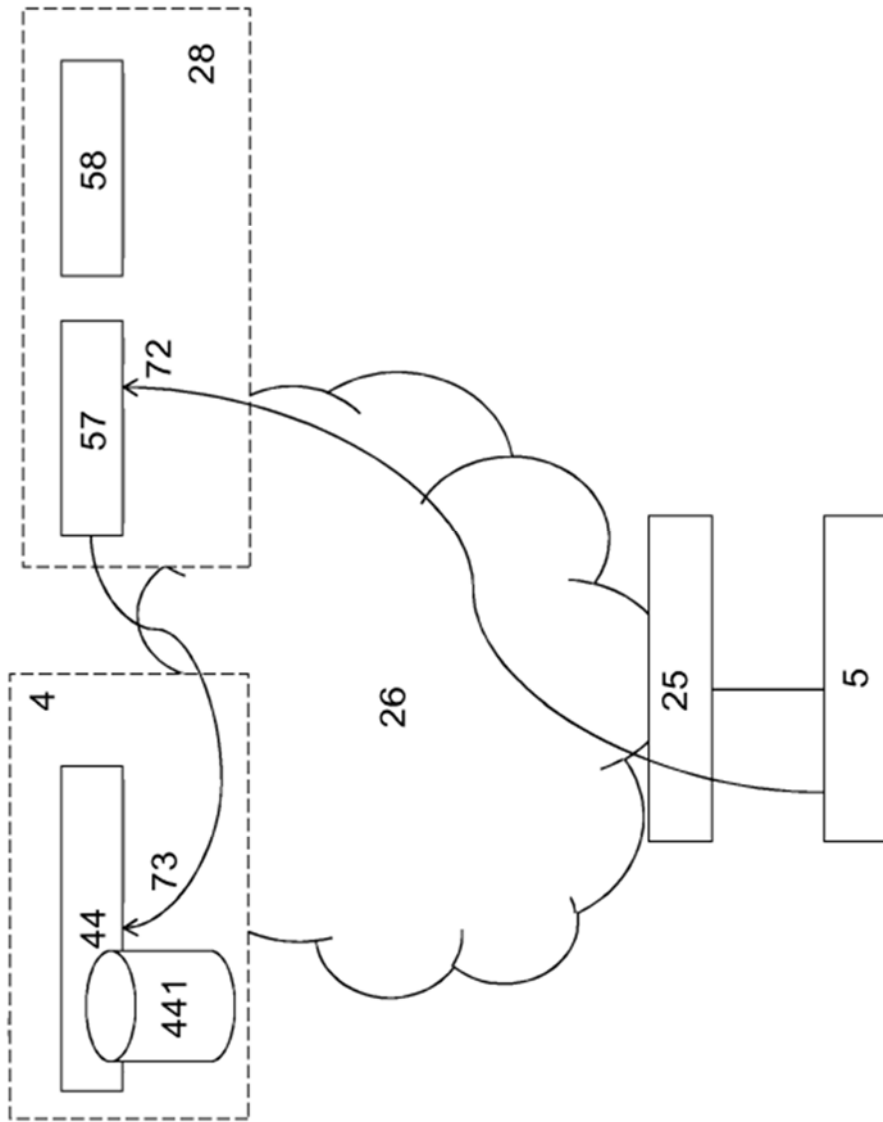


FIG. 23

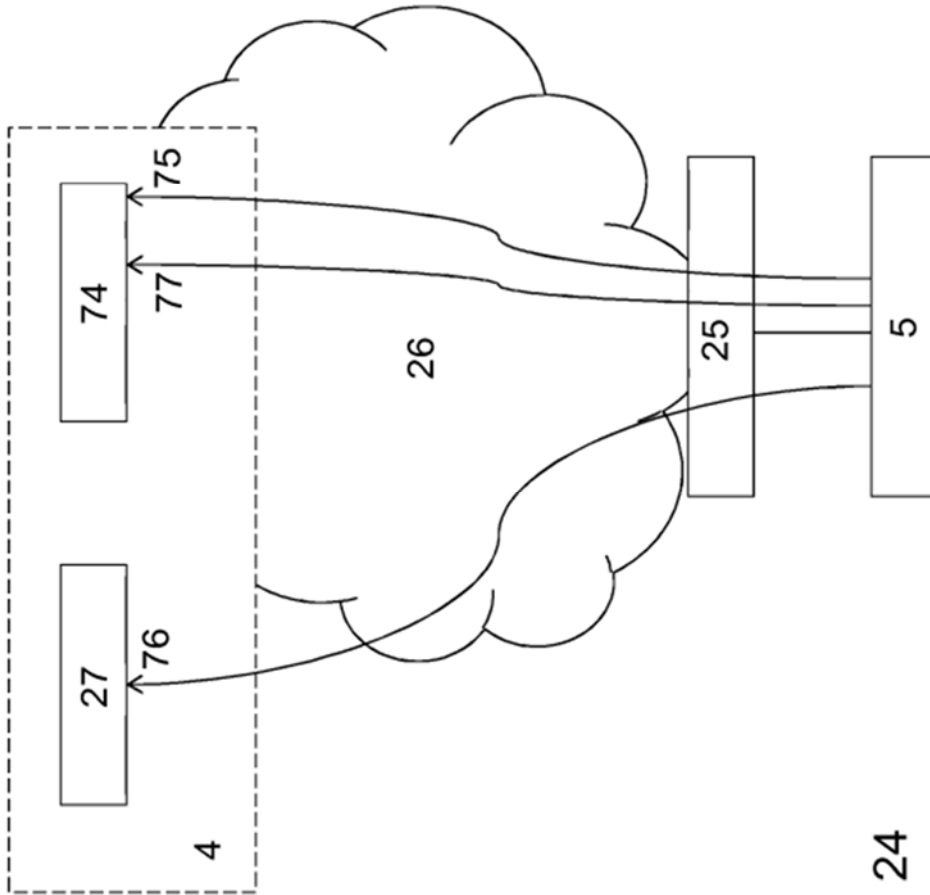


FIG. 24

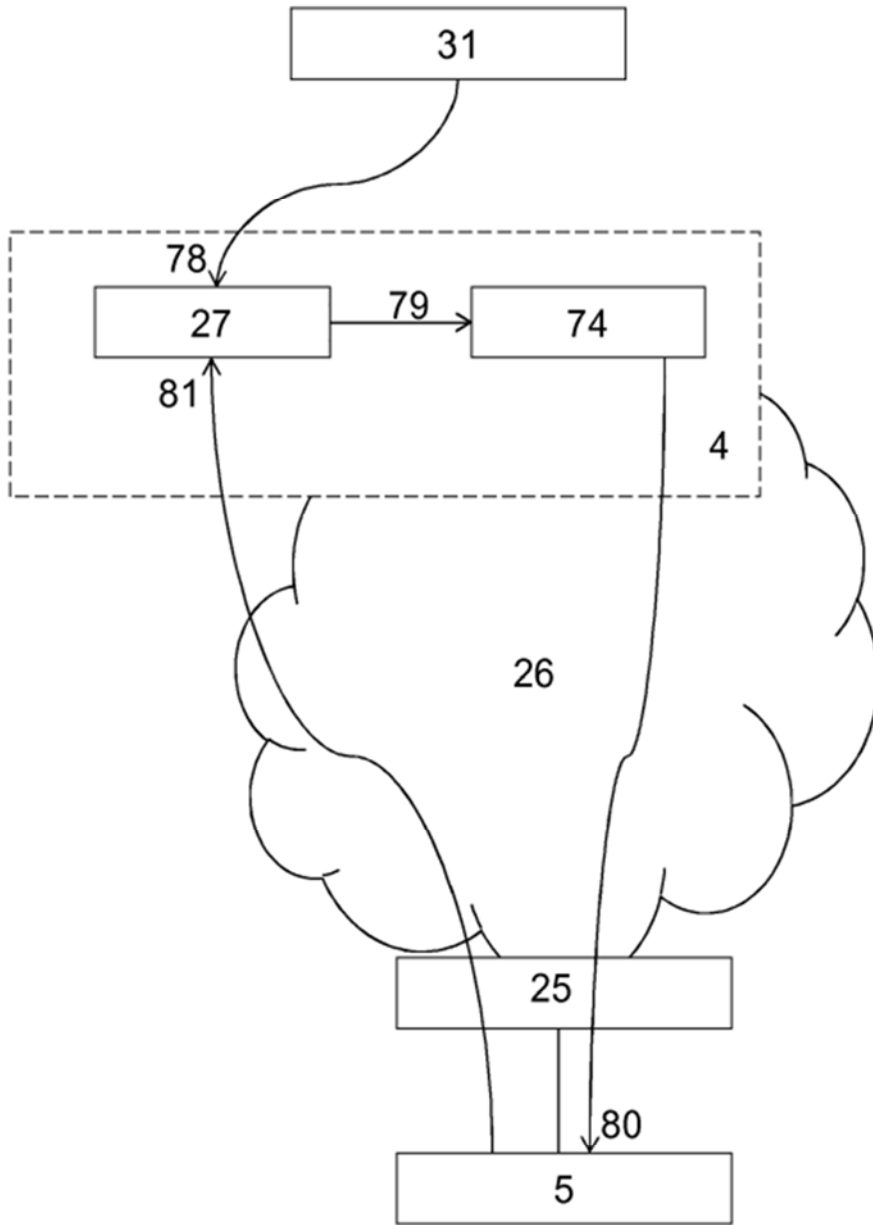


FIG. 25

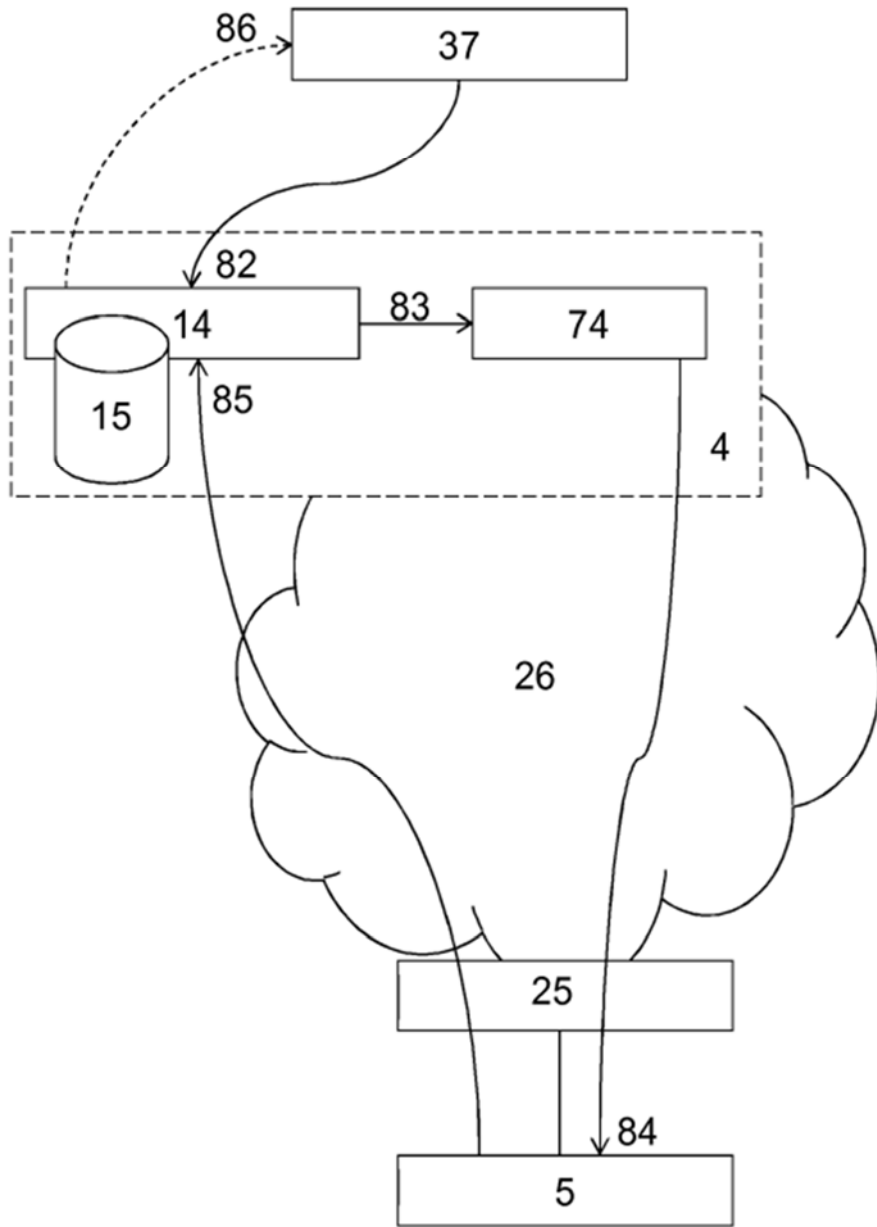


FIG. 26

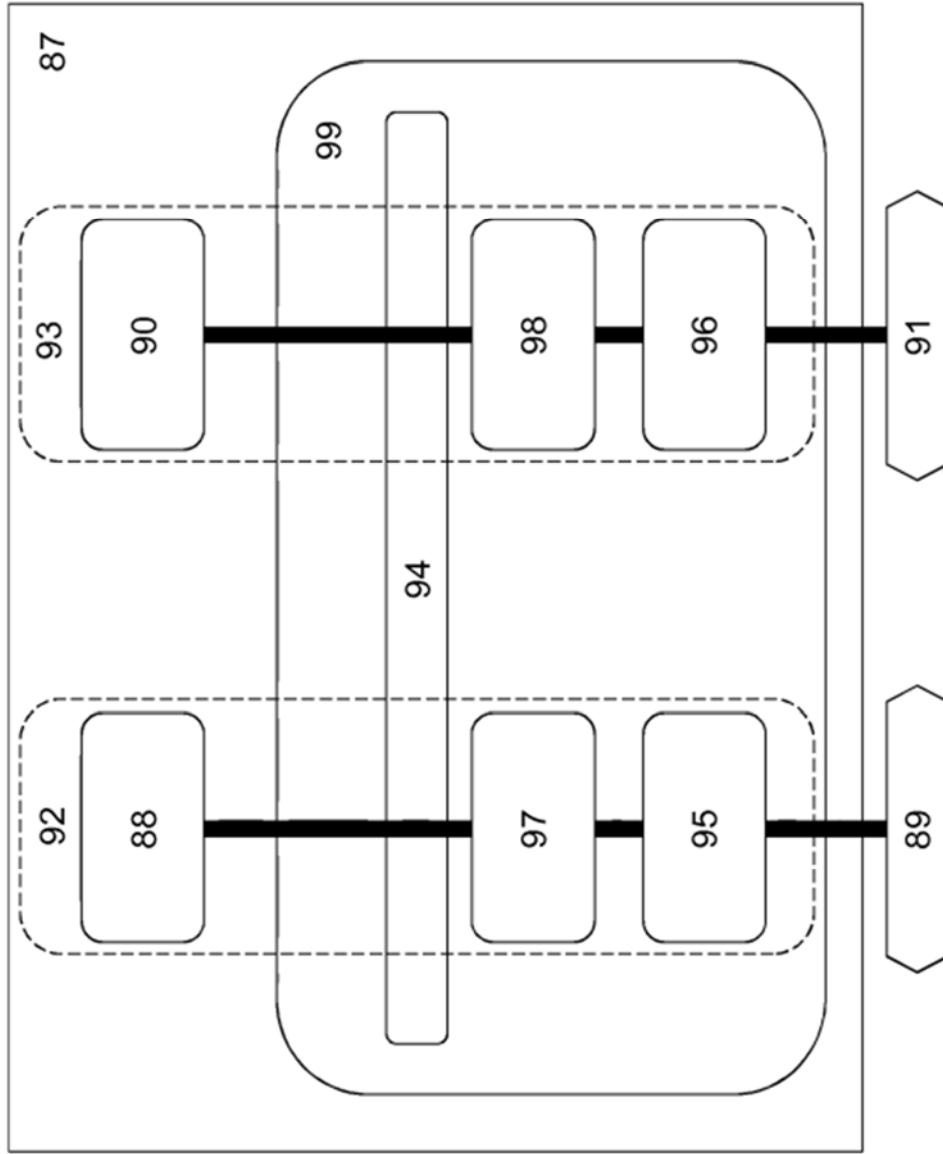


FIG. 27

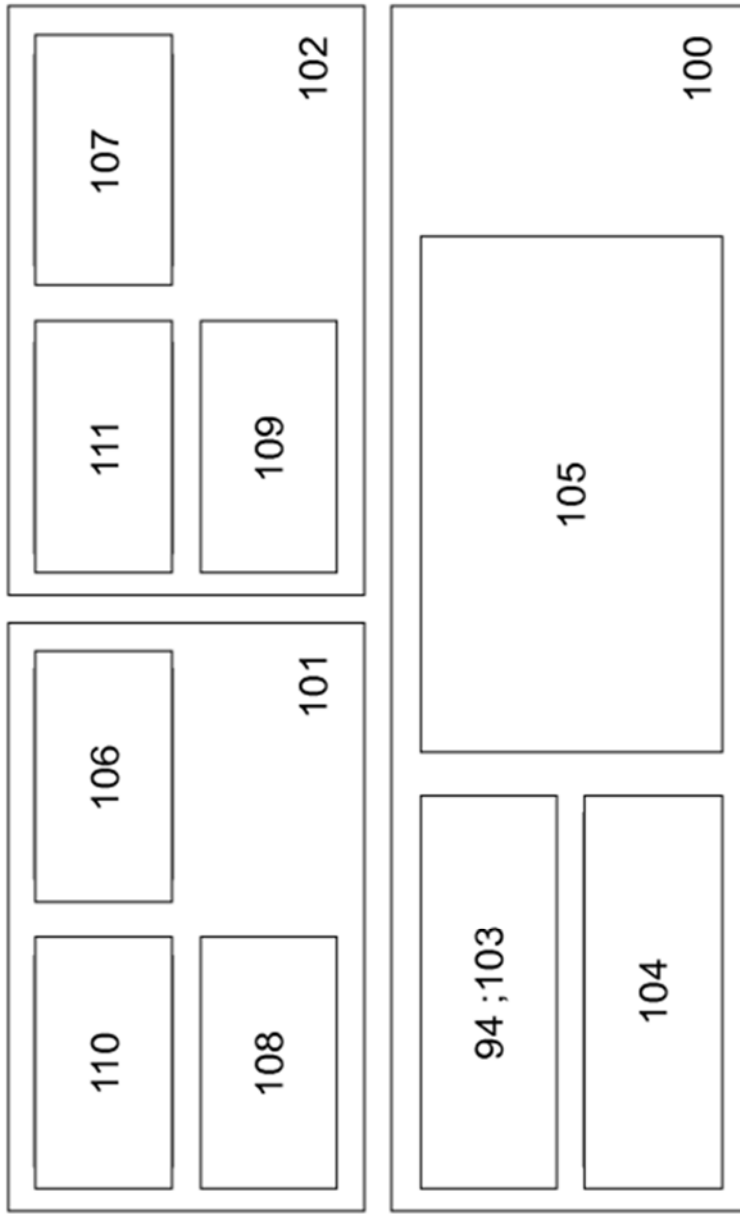


FIG. 28

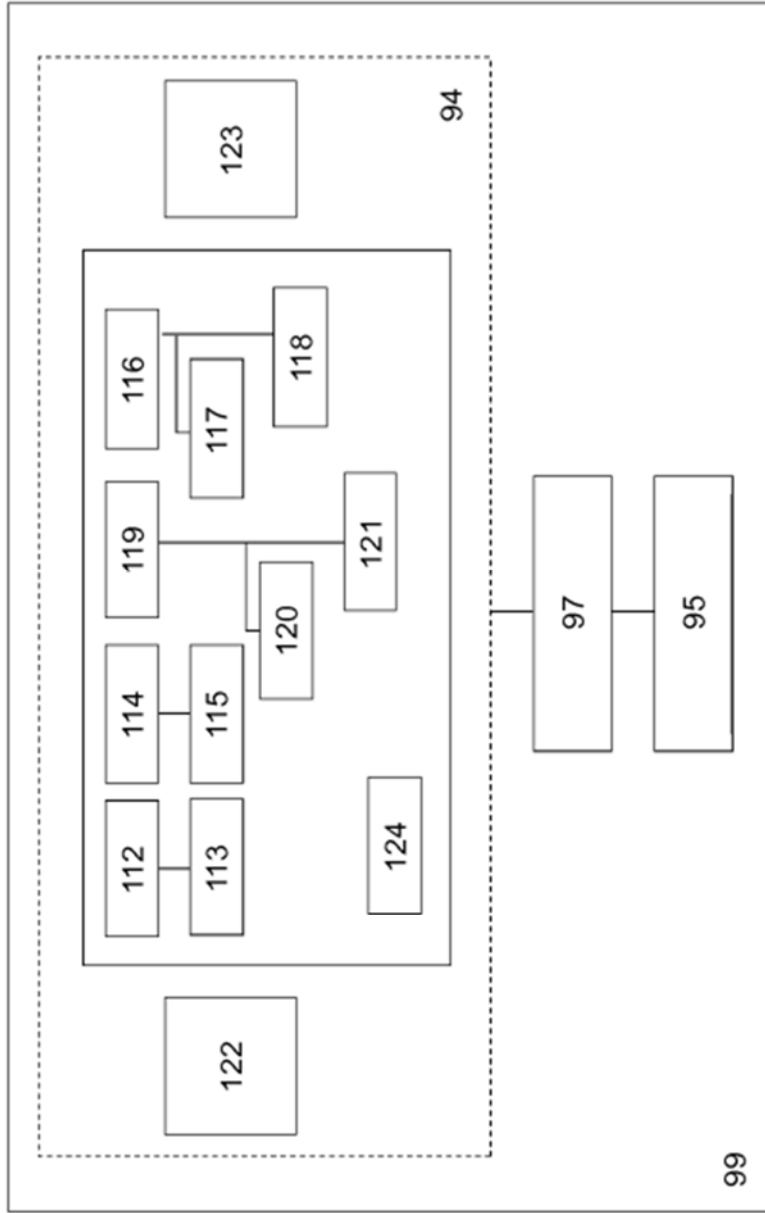


FIG. 29

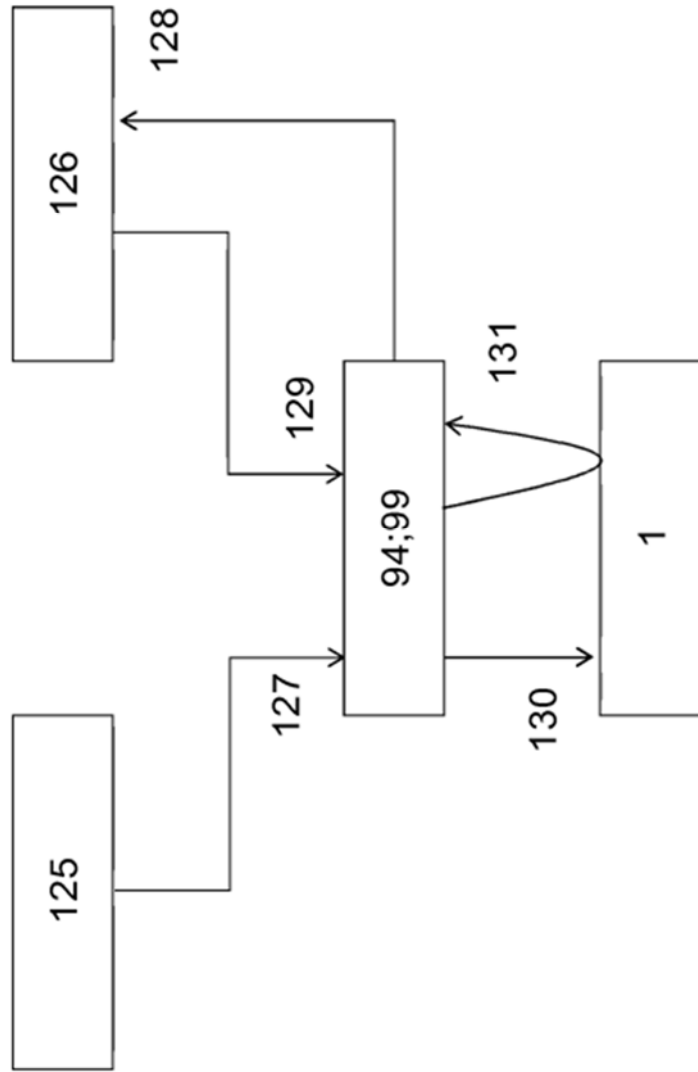


FIG. 30

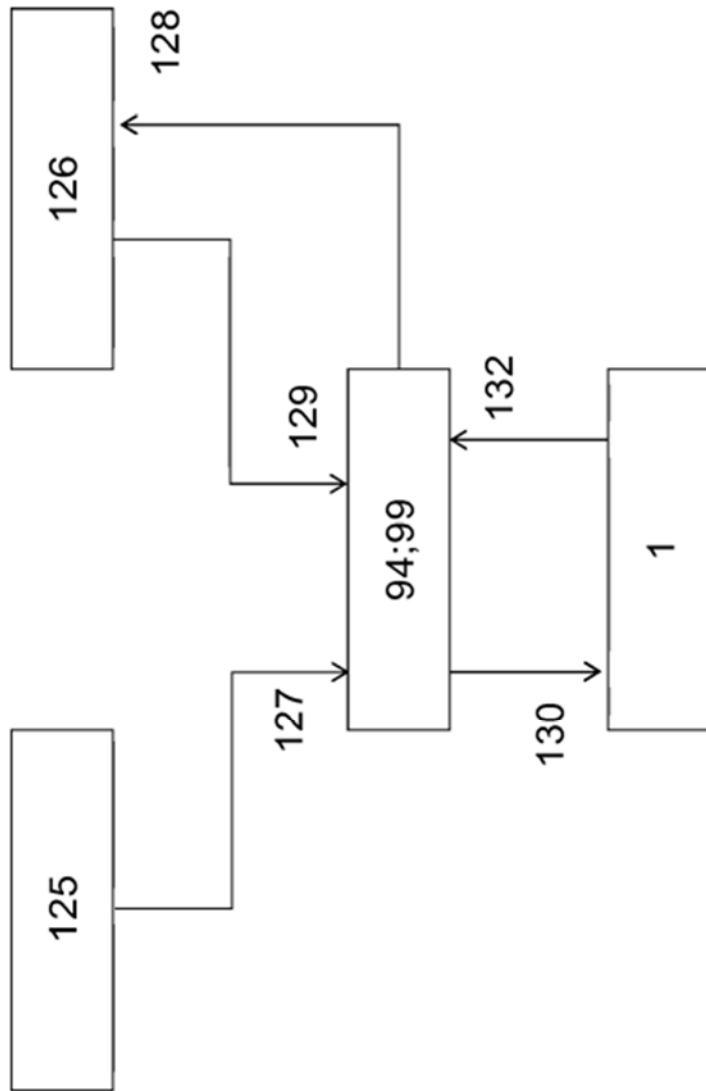


FIG. 31