

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 980 392**

51 Int. Cl.:

G05B 19/418 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **14.10.2020** **E 20201857 (8)**

97 Fecha y número de publicación de la concesión europea: **13.03.2024** **EP 3985465**

54 Título: **Operación mixta de unidades de producción en plantas híbridas**

45 Fecha de publicación y mención en BOPI de la
traducción de la patente:
01.10.2024

73 Titular/es:

ABB SCHWEIZ AG (100.0%)
Bruggerstrasse 66
5400 Baden, CH

72 Inventor/es:

GRUENER, STEN;
HOERNICKE, MARIO;
STARK, KATHARINA;
BRAUN, ROLAND;
VACH, MICHAEL;
SCHOCH, NICOLAI y
DIX, MARCEL

74 Agente/Representante:

LEHMANN NOVO, María Isabel

ES 2 980 392 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

Operación mixta de unidades de producción en plantas híbridas

5 CAMPO DE LA INVENCION

La invención se refiere a la integración de módulos en una planta modular tal como, por ejemplo, una planta modular híbrida.

10 ANTECEDENTES

En las industrias de procesos, el paquete de tipo modular (MTP) es una norma establecida para describir la funcionalidad del módulo y para regir las comunicaciones entre el módulo y el sistema de orquestación. En la fabricación discreta, el lenguaje de máquina de embalaje (PackML) es una norma comúnmente usada para monitorizar y orquestar módulos y líneas de producción. Dentro de las instalaciones híbridas que combinan elementos tanto de proceso como de fabricación discreta, por ejemplo, en la producción de alimentos y bebidas o de celdas de batería, pueden usarse ambas normas (PackML, MTP) para describir y operar unidades/máquinas. Tales normas tienen diferentes formas de integración y operación.

El documento EP 1 889 127 A1 divulga un modelo de datos jerárquico diseñado para representar la disposición de dispositivos y/o procesos dentro de una planta. Un proceso continuo puede asociarse con un proceso discreto dentro de la misma área de la planta.

El documento EP 1 903 411 A1 describe un servidor proxy para la integración de datos de automatización industrial a través de múltiples redes.

SUMARIO

Existe la necesidad de facilitar la integración de módulos en una planta modular híbrida que comprende partes de la fabricación discreta y la industria de procesos. Esta necesidad se satisface con la materia objeto de las reivindicaciones independientes. Se exponen características opcionales en las reivindicaciones dependientes.

La presente divulgación proporciona un enfoque genérico para la interconexión entre varias normas de módulos. La presente divulgación permite una interfaz transparente entre la capa de orquestación (por ejemplo, la capa de orquestación de proceso o POL de MTP) y los módulos descritos por diversas (incompatibles) normas, facilitando por tanto el acceso a la automatización híbrida y permitiendo a los usuarios mezclar módulos en plantas totalmente nuevas, así como para conectar módulos existentes a la POL. En otras palabras, la presente divulgación permite la operación mixta de plantas modulares híbridas (es decir, plantas que implican módulos descritos en diferentes normas) desde una capa de orquestación genérica. Ejemplos de esas normas son MTP y PackML. Se entenderá, sin embargo, que la presente divulgación no está limitada a esas normas. Se propone un mapeo automatizado o semiautomático de las descripciones de conceptos de un módulo, así como una traducción en tiempo de ejecución integrada en la capa de orquestación para permitir la interoperabilidad entre normas. Dado que ambas normas se basan en un protocolo de comunicación común, como UA de OPC, como red troncal de comunicación e intercambio de información, la presente divulgación permite habilitar interfaces PackML dentro de POL a un coste relativamente bajo.

Serán evidentes aspectos, ejemplos y características de la invención y se aclararán con referencia a las realizaciones descritas a continuación en el presente documento.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

A continuación, se dará una descripción detallada, únicamente a modo de ejemplo, con referencia a los dibujos adjuntos, en los que:

La figura 1 ilustra la estructura de una planta modular de acuerdo con la norma de MTP;

La figura 2 ilustra un modelo físico de una planta modular, así como la gestión de recetas de acuerdo con PackML;

La figura 3 ilustra la interconexión de unidades para formar una línea de producción en PackML;

La figura 4 ilustra un método para integrar máquinas de PackML individuales en la capa de orquestación de procesos de MTP;

La figura 5 ilustra las etapas de método que se pueden realizar para importar múltiples máquinas PackML interconectadas a la capa de orquestación de procesos de MTP;

Las figuras 6 y 7 ilustran un método para integrar módulos de MTP en PackML representando los módulos MTP a través de interfaces PackML; y

La figura 8 ilustra un dispositivo informático que se puede usar de acuerdo con los sistemas y métodos divulgados en el presente documento.

DESCRIPCIÓN DETALLADA

La figura 1 ilustra la estructura de una planta modular 100 de acuerdo con la norma de MTP. En el contexto de la automatización de procesos, la planta modular 100 comprende diferentes módulos 102 (un módulo en MTP también se conoce como Conjunto de Equipos de Proceso - PEA), que encapsulan componentes 104 tales como dispositivos y aparatos que también consisten en controladores e instrumentos de procesos. Cada módulo 102 tiene un controlador integrado 106 que se encarga de automatizar las funciones proporcionadas por el módulo 102, así como del manejo de fallos y otras tareas de automatización. Cada módulo 102 cumple al menos una función de proceso, que se encapsula y se ofrece como servicio. Todos los módulos 102 tienen conexiones de proceso con otros módulos 102 y una conexión de información con la Capa de Orquestación de Procesos (POL) 108. La comunicación entre PEA 102 y POL 108 se realiza usando el protocolo de comunicación UA de OPC (Arquitectura Unificada de OPC, un protocolo de comunicación de máquina a máquina para la automatización industrial regido por la Fundación OPC). Un elemento clave dentro del área de plantas de producción modular es la normalización de las interfaces del sistema de automatización entre el sistema de automatización de módulos 110 y la POL 108. Esto ha sido normalizado por el Paquete de Tipo de Módulo (MTP), un archivo de descripción o definición de módulo que permite la integración de los módulos 102 en la POL 108. Un archivo de MTP típicamente describe una interfaz hombre-máquina (HMI) de un módulo 102, estructurado como un diagrama de tuberías e instrumentación (P&ID) 102, describe los servicios del módulo y proporciona una lista de equipos integrados. El MTP es una descripción independiente del proveedor, que define las interfaces a un módulo 102 en formato AutomationML (AML). Por lo tanto, se puede usar para integrar módulos 102 en la POL 108, independientemente del proveedor y tipo de controladores 106 usados para el módulo 102.

La figura 2 ilustra un modelo físico de una planta modular, así como la gestión de recetas de acuerdo con PackML. El Lenguaje de Máquina de Embalaje (PackML) es una norma de automatización industrial para el control de máquinas de embalaje desarrollado por la Organización para la Automatización y el Control de Máquinas (OMAC) en colaboración con la Sociedad Internacional de Automatización (ISA). PackML sugiere una arquitectura de programa normalizada y metodologías de programación con el objetivo de proporcionar una apariencia y un manejo consistentes de las máquinas de embalaje para operadores y técnicos. Los tres elementos principales de la norma son un modelo de máquina de estados, etiquetas (etiquetas de embalaje (PackTags)) para una terminología coherente y la arquitectura de programa modular Make2Pack. El modelo de máquina de estados puede adoptarse en casi cualquier máquina industrial, ya que los estados y transiciones no son exclusivos de las máquinas de embalaje. El código de máquina modularizado que usa PackML coincide con los 3 niveles inferiores del modelo físico ISA 88 / IEC 61512: la unidad 202, el módulo de equipo (EM) 204 y el módulo de control (CM) 206, como se muestra en la figura 2. Una unidad de PackML 202 también puede denominarse máquina. La forma recomendada para la comunicación entre módulos y de línea a módulo es usar UA de OPC. La receta es el conjunto necesario de información que define de forma única los requisitos de producción de un producto específico. La relación entre recetas y equipos se muestra en la figura 2. En particular, la receta de control describe los parámetros que necesitan establecerse en la unidad 202 para la producción de un producto específico. Un ejemplo podría ser una unidad paletizadora 202 que recibe los tres parámetros relacionados con el proceso para una orden de producción: patrón de embalaje, número de capas e intercalado entre capas.

Dentro de MTP, se requiere que la POL 108 coordine los módulos 102 usados en la planta modular 100, por ejemplo, ejecutando una receta de proceso. En PackML, la operación se basa principalmente en la comunicación entre unidades. Por lo tanto, PackML no requiere una POL, sino que puede implementarse simplemente conectando las unidades 202 en la línea de producción 210, como se muestra, por ejemplo, en la figura 3. Conciliar estos dos conceptos opuestos de operación en plantas híbridas ha resultado ser un desafío para la industria.

En el presente documento se describen métodos para integrar módulos en una planta modular híbrida (que comprende una parte de fabricación discreta y una parte de fabricación continua) mediante la implementación de un mapeo de PackML a MTP, por ejemplo, proporcionando el mapeo de máquinas de paquete de PackML a embalajes de MTP para ingeniería y envoltura en tiempo de ejecución para la orquestación en tiempo de ejecución, o implementando un mapeo de MTP a PackML, por ejemplo, creando una envoltura compatible con MTP para unidades de PackML. Esta envoltura permite la ingeniería, monitorización y orquestación transparentes de los módulos de PackML 202 y las líneas de producción basadas en PackML 210 en la POL de MTP 108, o la monitorización y control de los módulos de MTP 102 a través de interfaces de PackML, respectivamente. Las secuencias de control o recetas que se ejecutan en la parte continua se pueden configurar para ejecutar servicios de la parte discreta y/o las secuencias de control o recetas que se ejecutan en la parte discreta se pueden configurar para ejecutar servicios de la parte continua.

De acuerdo con un primer ejemplo, se implementa un mapeo de PackML a MTP para integrar la parte discreta de la planta modular híbrida (que comprende aquellas unidades 202 que están interconectadas usando PackML) en la POL

108 de la parte continua de la planta modular (que comprende los módulos de MTP 102). Por lo tanto, la siguiente parte de la descripción se centra en un enfoque de PackML a MTP para envolver PackML para su uso en POL basada en MTP 108. Las variantes del primer ejemplo o mapeo de PackML a MTP incluyen mapear unidades de PackML únicas 202 una a una como módulos de MTP 102 individuales en la POL de MTP 108, y mapear múltiples unidades de PackML 202 combinadas en una línea de producción 210 como un único módulo de MTP 102 en la POL de MTP 108. Por supuesto, dentro de una única planta híbrida, algunos módulos de PackML se pueden integrar individualmente mientras que otros se integran como una línea de producción. Las máquinas de PackML 202 se importan en tiempo de ejecución o en tiempo de diseño/ingeniería. Cuando se importan las máquinas en tiempo de ejecución, las máquinas están presentes y se pueden explorar a través de una interfaz de UA de OPC. La importación de las máquinas en tiempo de diseño/ingeniería depende de la disponibilidad de la máquina: se usa cualquier modelo de información de UA de OPC disponible de una máquina de PackML 202, por ejemplo, a través de un archivo de establecimiento de nodo (NodeSet) de UA de OPC; como alternativa, se supone un conjunto normalizado de estados de máquina, modos operativos y etiquetas de embalaje, donde modificaciones adicionales, tales como etiquetas o modos operativos, están disponibles como exportación, por ejemplo, a través de un archivo de hoja de cálculo tal como Excel.

Como se mencionó anteriormente, en una variante del primer ejemplo, las unidades de PackML 202 se integran una a una en la POL de MTP 108. En un ejemplo, las secuencias de control o recetas que se ejecutan en la parte continua no se pueden aplicar a la parte discreta. La integración de las unidades de PackML 202 implica envolver/mapear la descripción de una única máquina de PackML 202 en una abstracción compatible con MTP que incluye, por ejemplo, uno o más de las siguientes etapas: monitorizar información de la máquina de PackML 202 tal como el estado actual, las transiciones disponibles e información de error/alarma a través de interfaces compatibles con MTP; realizar un cambio de modo de operación y estado de la máquina de PackML 202 usando la POL de MTP 108 y la abstracción de estado del módulo de MTP 102; acceder a información no funcional (incluida en etiquetas de embalaje de PackML) tal como la velocidad de la máquina o parámetros de la máquina personalizados a través de constructos de ensamblaje de datos (DataAssembly) o parámetro de servicio (ServiceParameter) de PackML, permitiendo su utilización en transiciones de recetas de POL; resumir múltiples acciones de la máquina de PackML 202 como un servicio o procedimiento de MTP especializado (por ejemplo, establecer el modo de la máquina, escribir un conjunto de parámetros personalizados de la máquina, modificar la velocidad de la máquina).

El contenido de un módulo PackML se puede resumir de la siguiente manera: modelo de estado (normalizado, pero también extensible mediante subestados personalizados); lista de modos de la máquina (tres modos normalizados, por ejemplo, manual, automático, con modos personalizados adicionales); etiquetas de embalaje de estado (Status PackTags) de solo lectura que describen el estado de la máquina, por ejemplo, velocidad de la máquina, que incluyen algunos parámetros normalizados y personalizados que son específicos de la máquina); etiquetas de embalaje de comando (Command PackTags) de solo escritura que permiten que se inflencie el estado y el modo de la máquina, por ejemplo, velocidad de la máquina, que incluyen algunos parámetros normalizados y personalizados que son específicos de la máquina); etiquetas de embalaje de administración (Admin PackTags) de solo lectura que contienen algunos datos de OEE normalizados (por ejemplo, recuentos totales y de productos defectuosos) y algunos datos de advertencia específicos de la máquina.

La figura 4 ilustra las etapas de método que se pueden realizar para importar una máquina PackML individual 202 a la POL de MTP 108. El método incluye las etapas de construir 401 un archivo de MTP 400; importar y parametrizar 402 el archivo 400 en la POL 108; y traducción en tiempo de ejecución 403 entre comunicaciones de PackML y MTP.

En la etapa 401, se construye el archivo de MTP 400. Esto puede implicar la construcción automatizada o semiautomatizada del archivo de MTP 400 basándose en la interfaz de PackML (UA de OPC) en el tiempo de la ingeniería. La construcción de archivos se puede llevar a cabo usando una herramienta de cara al usuario que puede explorar servidores de PackML en línea o cargar una representación fuera de línea de PackML (por ejemplo, un archivo de establecimiento de nodo o una hoja de cálculo) e importarlos a la POL de MTP 108, ya sea directamente o usando un archivo de MTP intermedio. Por tanto, el archivo de MTP 400 puede construirse escaneando la representación de PackML y generando una representación compatible con MTP en la que el archivo o módulo de MTP contiene uno o más de (i) modos o procedimientos de la unidad de acuerdo con modos de la unidad integrados dentro de la máquina de PackML; (ii) variables de la sección de etiquetas de embalaje contenidas dentro del modelo de información de PackML. Además, se puede definir un marcador especializado del MTP, permitiendo que la POL 108 realice traducciones en tiempo de ejecución en la etapa 403, como se describe además más adelante. El mapeo de la interfaz de PackML existente se puede realizar basándose en una o más de una interfaz de una unidad existente 202 (como se muestra en la figura 4), y la información fuera de línea, por ejemplo, una exportación de establecimiento de nodo del servidor de UA de OPC u hojas de Excel de las etiquetas de embalaje contenidas. El mapeo comprende uno o más de un mapeo de servicio, un mapeo de ensamblaje de datos y un mapeo de HMI.

Para el mapeo de servicio, la entidad principal de PackML que se puede mapear al servicio de MTP es el modo de operación de la unidad. Más allá de los servicios convencionales como la operación automática o manual, un módulo de PackML puede contener un número arbitrario de modos de operación personalizados. El mapeo automático puede ocurrir basándose en estos servicios en caso de que no se hayan de mapear parámetros personalizados. En los modos de operación personalizados, se puede obtener asistencia de usuario para marcar qué etiquetas de embalaje de PackML se usan en qué modos de operación. A continuación, esas etiquetas de embalaje pueden incluirse en la

definición de servicio de MTP de modo que puedan establecerse en una llamada. Hay al menos dos estrategias de mapeo diferentes previstas en la presente divulgación: una primera estrategia es implementar un mapeo de uno a uno de los modos de operación de PackML a los servicios de MTP, permitiendo un control granular preciso sobre la unidad; una segunda estrategia es implementar un mapeo complejo de múltiples modos de módulo en un servicio de MTP, por ejemplo, cambiando de un modo automático a un modo manual con una reducción intermedia de la velocidad de la máquina.

Para el mapeo del ensamblaje de datos, una posibilidad es mapear etiquetas de embalaje de estado o de administración a la estructura de ensamblaje de datos dentro del módulo de MTP, por ejemplo, para permitir que la POL 108 lea y procese parámetros de la unidad de PackML 202, por ejemplo, velocidad de la máquina actual. El ingeniero puede implementar este mapeo manualmente, por ejemplo, para etiquetas de embalaje no convencionales que se incluyen en el módulo, o automáticamente mapeando todas las etiquetas de embalaje no convencionales a parámetros de servicio. El sistema de ingeniería puede registrar todos los mapeos manuales, de modo que el mapeo para módulos similares o recurrentes se pueda implementar automáticamente.

En la etapa 402, se realiza la importación y parametrización del archivo de MTP 400 en la POL de MTP 108. Los parámetros relevantes incluyen, por ejemplo, la dirección IP de la instancia MTP y parámetros para servicios PackML envueltos, por ejemplo, velocidad de la máquina actual o modo de operación de la unidad. La importación del envoltorio de MTP 400 generado a la POL 108 se puede realizar sin mayores ajustes. Como se mencionó anteriormente, se puede incluir un marcador en el MTP de envoltorio 400 de modo que la POL 108 pueda reaccionar y realizar la traducción en tiempo de ejecución en la etapa 403. Este marcador se puede configurar usando la estructura de datos normalizada de MTP, por ejemplo, mediante un esquema de nomenclatura de tipo de módulo especializado o un parámetro de MTP propietario. Tales marcadores pueden comprender uno o más prefijos o sufijos del elemento de UA de OPC mapeado a MTP, un tipo adicional no de MTP mencionado en un archivo AutomationML y/o un comentario en el nivel de XML.

En la etapa 403, se realiza la traducción en tiempo de ejecución entre las comunicaciones de PackML y MTP. Después de cargar el envoltorio-MTP 400 en la POL 108, se realizarán llamadas a la interfaz de PackML de la unidad 202 para activar servicios y monitorizar parámetros. Por lo tanto, la POL 108 puede complementarse con una funcionalidad de traducción en tiempo de ejecución (no mostrada en la figura 4) para traducir entre interfaces de comunicación de MTP y PackML, por ejemplo, traducir valores que codifican el estado de la unidad 202, para superar las diferencias entre la representación de información en tiempo de ejecución de MTP y PackML. La funcionalidad de traducción puede implementarse dentro de la POL 108 o mediante un componente de puerta de enlace adicional que esté ubicado fuera de la POL 108. Basado en el marcador en el archivo de envoltorio-MTP 400, la POL 108 o el componente de pasarela mapea información dentro de la unidad de PackML 202 a una representación de MTP compatible. Como alternativa, se pueden utilizar atributos opcionales del sistema de control distribuido (por ejemplo, el sistema ABB 800xA) en tiempo de ejecución. Este mapeo en tiempo de ejecución incluye, pero sin limitación: información de estado, por ejemplo, estado actual y estados de transición disponibles; representación de estado de operación de módulo y capacidad de cambiar entre estos modos de manera similar al estado de módulo; y remapeo de la información necesaria para usar mapeos de ensamblaje de datos. El mapeo de información de estado se puede realizar no únicamente para los estados de PackML que están presentes en el modelo de estado de MTP, sino también para los estados requeridos por MTP, pero ausentes en la definición de módulo de MTP, ya que PackML puede omitir tales estados. Lo mismo se aplica a estados adicionales que la unidad PackML 202 puede contener pero que no están presentes en el conjunto de estados de MTP fijo). La lectura del estado actual se puede realizar de manera diferente en un nivel de comunicación de POL a módulo. Por ejemplo, el modelo de UA de OPC de PackML define el cambio entre estados por medio de llamadas de método de UA de OPC, mientras que MTP usa escrituras variables.

Para el mapeo de HMI, dado que PackML no proporciona información especializada sobre su estructura física, por ejemplo, interfaces de producto o estructura interna necesarias o proporcionadas, esta porción de información del MTP puede reemplazarse por un marcador de posición genérico o un mapeo específico de unidad especializada, por ejemplo, para máquinas complejas que tienen múltiples entradas y salidas de productos. También se puede realizar un aprovisionamiento automático o semiautomático, por ejemplo, generación de placas frontales de HMI de parámetros basadas en etiquetas de embalaje existentes de PackML.

Como se mencionó anteriormente, en la segunda variante del primer ejemplo, la parte discreta está integrada como un único módulo de MTP 102 dentro de la parte continua de la planta modular híbrida. Cada máquina de PackML 202 proporciona un único servicio, estando la línea de producción 210 representada por un único módulo de MTP 102 que combina todos los servicios de las máquinas de PackML 202 de la línea 210, por ejemplo, llenar el producto en múltiples tipos de embalaje. La línea de producción completa 210 está controlada por la POL 108 que opera en la parte continua. Además de esto, la ingeniería puede integrarse tanto para la parte discreta como para la continua. Esto significa que, la línea de producción podría definirse como un módulo de acuerdo con la norma correspondiente (por ejemplo, VDI 2658, Ingeniería de automatización de sistemas modulares en la industria de procesos - Concepto general e interfaces) y cada máquina podría ser un servicio, como se define por la misma norma. El MTP resultante podría usarse en ingeniería y es posible que solo se necesite un envoltorio 500 para el tiempo de ejecución (principalmente convertir los números enteros usados en PackML a bits usados en MTP). Las soluciones descritas

para la conversión de MTP a PackML también se pueden aplicar al revés, es decir, para representar módulos de MTP mediante interfaces de PackML, como se describe más adelante.

La figura 5 ilustra las etapas de método que se pueden realizar para importar múltiples máquinas de PackML interconectadas 202 a la POL de MTP 108. El método se realiza para representar la línea de producción 210 que comprende múltiples módulos de PackML 202 como un único módulo de MTP 102 desde el punto de vista de la POL de MTP 108. Como con la figura 4, el método de la figura 5 incluye las etapas de construir 501 un archivo de MTP 500; importar y parametrizar 502 el archivo 500 en la POL 108; y traducción en tiempo de ejecución 503 entre comunicaciones de PackML y MTP.

El procedimiento de mapeo para construir el archivo de MTP 500 en la etapa 501 se puede realizar de manera similar al mapeo de un único módulo como se ha descrito anteriormente en relación con la etapa 401, excepto que se usa adicionalmente un mapeo de línea basado en la topología de línea 510 (que comprende, por ejemplo, el orden físico de unidades de PackML 202). Usando la información de topología 510, la POL 108 puede determinar un estado general de la línea 210, por ejemplo, la condición de toda la línea 210 puede cambiarse en el caso de un cambio de condición en alguna unidad de PackML 202 no redundante. En el momento de la ingeniería, la lógica se puede definir y envolver en un servicio de MTP, tal como iniciar o detener toda la línea 210 y cambiar el procesamiento producido dentro de la línea 210. En términos generales, la descripción de envoltura/mapeo de múltiples máquinas de PackML 202 combinadas en la línea de producción 210 en la abstracción compatible con MTP 500 que representa la línea de producción 210 como un módulo de MTP 102 puede comprender: monitorizar la línea de producción 210 usando su información de topología 510 y definir estados de operación de la línea de producción 210 como estados del módulo de MTP, por ejemplo, un defecto en un módulo de la línea provocaría que cambiara el estado de toda la línea. Por ejemplo, el método puede comprender reducir de manera síncrona la velocidad de todas las máquinas de la línea o cambiar el modo de producción de la línea, por ejemplo, cambiando el formato o distribución de los embalajes usados.

La etapa 502 se puede realizar como se ha descrito anteriormente en relación con la etapa 402.

La traducción en tiempo de ejecución 503 se vuelve más compleja que en la etapa 403, ya que se pueden ordenar múltiples llamadas a interfaces de PackML y se puede evaluar cualquier lógica de servicio personalizada definida en la etapa 501 por la POL 108, por ejemplo, ejecutando algunas rutinas precompiladas.

Un método alternativo de mapeo de la línea de producción implica una orquestación de múltiples módulos de MTP 102 que se convierten desde PackML usando las técnicas descritas anteriormente. Para algunos casos de uso, una representación de caja negra de MTP podría ser un enfoque más adecuado, por ejemplo, para reducir la complejidad dentro de la POL 108. Además, se pueden definir servicios basados en líneas complejas personalizados durante el procedimiento de mapeo/construcción que podrían usar una sintaxis diferente a la de la POL 108.

De acuerdo con un segundo ejemplo, se toma un enfoque de PackML en primer lugar implementando un mapeo de MTP a PackML para representar los módulos de MTP 102 y para permitir el control del módulo de MTP 102 mediante una interfaz de PackML.

Las figuras 6 y 7 ilustran el procedimiento con más detalle para un módulo de MTP 102 y el archivo de MTP 600 correspondiente.

En la etapa 601, se importa y traduce el archivo de MTP 600. El archivo de MTP 600 y la información de instancia de MTP (por ejemplo, punto final de UA de OPC) pueden importarse en tiempo de ingeniería o tiempo de ejecución, por ejemplo, reutilizando información de POL existente. En más detalle, en la etapa 601, se parametriza un componente proxy de tiempo de ejecución 608 usando el archivo de MTP importado 600 y una dirección de punto final del módulo de MTP 102, y se construye una representación compatible con PackML 202 del módulo 102 usando un algoritmo de mapeo.

El algoritmo de mapeo es similar al descrito anteriormente en relación con PackML a MTP de una manera contraria, e incluye una o más de las siguientes etapas: se realiza un mapeo de modelo de estado (el modelo de estado de MTP no es extensible, por lo que es posible casi un mapeo directo); (ii) los servicios de MTP se mapean a los modos de operación del proxy de PackML; (iii) los ensamblajes de datos dentro de MTP se mapean a etiquetas de embalaje de la unidad PackML de acuerdo con un mapeo predefinido o proporcionado por el usuario.

Dicho de otra manera, envolver/mapear la descripción de un único módulo de MTP 102 en una interfaz de PackML 610 para una única máquina de PackML 202 puede incluir una o más de las siguientes etapas: monitorizar información de la máquina como el estado actual del módulo de MTP 102, transiciones disponibles e información de error/alarma a través de interfaces compatibles con PackML (estado, etiquetas de embalaje de modo); cambiar el modo de operación y llamar a los servicios del módulo de MTP 102 a través de interfaces de PackML; acceder a información funcional y no funcional adicional de MTP (por ejemplo, parámetros de módulo y ensamblajes de datos (DataAssemblys)) a través de etiquetas de embalaje de PackML; generación automatizada o semiautomatizada de documentación de etiqueta de embalaje basándose en la estructura de MTP existente/usada; y definir la lógica entre las partes de MTP y PackML de la planta modular híbrida representada en la POL 108. La definición de la lógica puede

incluir: cambiar el comportamiento de la parte de planta de MTP basándose en las condiciones de equipo de PackML, por ejemplo, regular la producción aguas arriba, por ejemplo, un proceso de elaboración de cerveza en caso de un mal funcionamiento aguas abajo, por ejemplo, el proceso de llenado de botellas, o, por ejemplo, aumentar la velocidad de llenado de la botella basándose en algunas condiciones del proceso de elaboración de cerveza; y recopilar datos basándose en condiciones, eventos de ambas partes de la planta, por ejemplo, para el análisis de causa raíz.

Dado que MTP puede contener múltiples servicios, puede requerirse un mapeo de varias interfaces de PackML lógicas 610, como se indica en la figura 7, que muestra múltiples representaciones de los servicios de MTP como unidades de PackML respectivas 202 ("Servicios de unidad 1-n"), incluyendo cada una Interfaz de PackML 610.

En la etapa 602, se realiza la traducción en tiempo de ejecución. Más detalladamente, el componente de proxy de tiempo de ejecución 608 inicia un servidor de UA de OPC compatible con PackML, y las solicitudes a la interfaz de PackML 610 del proxy de tiempo de ejecución 608 se traducen a continuación en llamadas a la interfaz de MTP 612 del módulo de MTP 102. Por ejemplo, las llamadas de método de UA de OPC del modelo de PackML se convierten en lecturas/escrituras variables de la interfaz de MTP.

Por lo tanto, para integrar la parte continua en la parte discreta, la etapa de construir una o más interfaces 610 que representen cada módulo de parte continua 102 como una o más unidades de parte discreta 202 respectivas puede comprender además importar y traducir (601) un archivo 600 que representa dicho módulo de parte continua 102, construir (601) una representación de parte discreta 202 de dicho módulo de parte continua 102, estando asociada la representación de parte discreta con dicha interfaz 610, y realizar (602) traducciones en tiempo de ejecución de comunicaciones entre dicha interfaz 610 y una interfaz 612 del módulo de parte continua 102.

La presente divulgación es aplicable a aplicaciones de elaboración de cerveza, farmacéuticas, de producción de celdas y de alimentos y bebidas donde se usan módulos de proceso y no de proceso (discretos).

Haciendo referencia ahora a la figura 8, se ilustra una ilustración de alto nivel de un dispositivo informático 800 ilustrativo que se puede usar de acuerdo con los sistemas y metodologías divulgados en el presente documento. El dispositivo informático 800 incluye al menos un procesador 802 que ejecuta instrucciones que están almacenadas en una memoria 804. Las instrucciones pueden ser, por ejemplo, instrucciones para implementar la funcionalidad descrita como llevada a cabo por uno o más componentes analizados anteriormente o instrucciones para implementar uno o más de los métodos descritos anteriormente. El procesador 802 puede acceder a la memoria 804 por medio de un bus de sistema 806. Además de almacenar instrucciones ejecutables, la memoria 804 también puede almacenar entradas conversacionales, puntuaciones asignadas a las entradas conversacionales, etc.

El dispositivo informático 800 incluye adicionalmente un almacén de datos 808 al que puede acceder el procesador 802 a través del bus de sistema 806. El almacén de datos 808 puede incluir instrucciones ejecutables, datos de registro, etc. El dispositivo informático 800 también incluye una interfaz de entrada 810 que permite que los dispositivos externos se comuniquen con el dispositivo informático 800. Por ejemplo, la interfaz de entrada 810 puede usarse para recibir instrucciones de un dispositivo informático externo, de un usuario, etc. El dispositivo informático 800 también incluye una interfaz de salida 812 que interconecta el dispositivo informático 800 con uno o más dispositivos externos. Por ejemplo, el dispositivo informático 800 puede mostrar texto, imágenes, etc., a través de la interfaz de salida 812.

Se contempla que los dispositivos externos que se comunican con el dispositivo informático 800 a través de la interfaz de entrada 810 y la interfaz de salida 812 puedan incluirse en un entorno que proporcione sustancialmente cualquier tipo de interfaz de usuario con la que un usuario pueda interactuar. Ejemplos de tipos de interfaz de usuario incluyen interfaces gráficas de usuario, interfaces de usuario naturales y así sucesivamente. Por ejemplo, una interfaz gráfica de usuario puede aceptar entradas de un usuario que emplea dispositivo o dispositivos de entrada tales como un teclado, ratón, control remoto o similares y proporcionar salida en un dispositivo de salida tal como una pantalla. Además, una interfaz de usuario natural puede permitir que un usuario interactúe con el dispositivo informático 800 de una manera libre de restricciones impuestas por dispositivos de entrada tales como teclados, ratones, controles remotos y similares. En su lugar, una interfaz de usuario natural puede depender del reconocimiento de voz, el reconocimiento táctil y de lápiz óptico, el reconocimiento de gestos tanto en la pantalla como adyacentes a la pantalla, los gestos aéreos, el seguimiento de la cabeza y los ojos, la voz y el habla, la visión, el tacto, los gestos, la inteligencia artificial y así sucesivamente.

Además, aunque se ilustra como un sistema único, debe entenderse que el dispositivo informático 800 puede ser un sistema distribuido. Por lo tanto, por ejemplo, varios dispositivos pueden estar en comunicación a través de una conexión de red y pueden realizar colectivamente tareas descritas como realizadas por el dispositivo informático 800.

Varias funciones descritas en el presente documento se pueden implementar en hardware, software o cualquier combinación de los mismos. Si se implementan en software, las funciones pueden almacenarse o transmitirse como una o más instrucciones o códigos en un medio legible por computadora. Los medios legibles por ordenador incluyen medios de almacenamiento legibles por ordenador. Un medio de almacenamiento legible por computadora puede ser cualquier medio de almacenamiento disponible al que pueda acceder un ordenador. A modo de ejemplo, y sin limitación, tales medios de almacenamiento legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-

ROM u otro almacenamiento en disco óptico, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético, o cualquier otro medio que pueda usarse para llevar o almacenar el código de programa deseado en forma de instrucciones o estructuras de datos y al que se puede acceder mediante un ordenador. Disco físico y disco óptico, como se usa en el presente documento, incluyen disco compacto (CD), disco láser, disco óptico, disco versátil digital (DVD), disquete y disco Blu-ray (BD), donde los discos físicos generalmente reproducen datos magnéticamente y los discos ópticos generalmente reproducen datos ópticamente con láseres. Además, una señal propagada no está incluida dentro del alcance de los medios de almacenamiento legibles por ordenador. Los medios legibles por ordenador también incluyen medios de comunicación, que incluyen cualquier medio que facilite la transferencia de un programa informático de un lugar a otro. Una conexión, por ejemplo, puede ser un medio de comunicación. Por ejemplo, si el software se transmite desde un sitio web, servidor u otra fuente remota mediante un cable coaxial, cable de fibra óptica, par trenzado, línea de abonado digital (DSL) o tecnologías inalámbricas tales como infrarrojos, radio y microondas, a continuación, en la definición de medio de comunicación se incluyen el cable coaxial, el cable de fibra óptica, el par trenzado, DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas. Las combinaciones de lo anterior también deben incluirse dentro del alcance de los medios legibles por ordenador.

Como alternativa, o, además, la funcionalidad descrita en el presente documento se puede realizar, al menos en parte, mediante uno o más componentes lógicos de hardware. Por ejemplo, y sin limitación, los tipos ilustrativos de componentes lógicos de hardware que se pueden usar incluyen matrices de puertas programables en campo (FPGA), circuitos integrados específicos de programa (ASIC), productos convencionales específicos de programa (ASSP), sistemas de sistema en un chip (SOC), dispositivos lógicos programables complejos (CPLD), etc.

Por la presente, el solicitante divulga de forma aislada cada característica individual descrita en el presente documento y cualquier combinación de dos o más de tales características, en la medida en que tales características o combinaciones puedan llevarse a cabo basándose en la presente memoria descriptiva en su conjunto a la luz del conocimiento general común de un experto en la materia, independientemente de si tales características o combinaciones de características resuelven cualquier problema divulgado en el presente documento.

Si bien la invención se ha ilustrado y descrito en detalle en los dibujos y en la descripción anterior, tal ilustración y descripción han de considerarse ilustrativas y no restrictivas. La invención no se limita a las realizaciones divulgadas. A la vista de la descripción y los dibujos anteriores, resultará evidente para un experto en la materia que se pueden realizar diversas modificaciones dentro del alcance de la invención, como se define en las siguientes reivindicaciones.

REIVINDICACIONES

1. Un método para integrar módulos en una planta modular híbrida que comprende una parte de fabricación discreta y una parte de fabricación continua, comprendiendo el método:
5 integrar la parte discreta en la parte continua, que comprende construir (401; 501) al menos un archivo de definición de módulo (400, 500) que mapea una o más unidades de parte discreta (202) de la parte discreta a un módulo de parte continua (102) e importar (402; 502) el archivo de definición de módulo (400, 500) en una capa de orquestación (108) de la parte continua; y
10 ejecutar la planta modular híbrida, que comprende realizar traducciones en tiempo de ejecución de las comunicaciones entre la parte discreta y la parte continua, en donde las partes discreta y continua usan un protocolo de comunicación común.
2. El método de la reivindicación 1, en donde integrar la parte discreta en la parte continua comprende integrar las unidades de parte discreta (202) individualmente en la capa de orquestación (108) como respectivos módulos de parte
15 continua (102).
3. El método de la reivindicación 1 o 2, que comprende además (i) configurar secuencias de control o recetas que se ejecutan en la parte continua para ejecutar servicios de la parte discreta, (ii) configurar secuencias de control o recetas que se ejecutan en la parte discreta para ejecutar servicios de la parte continua, o ambos de (i) y (ii).
20
4. El método de la reivindicación 1, en donde integrar la parte discreta en la parte continua comprende integrar una línea de producción (210) de la parte discreta como un único módulo de parte continua (102) dentro de la parte continua de la planta modular híbrida.
- 25 5. El método de la reivindicación 4, que comprende además usar información de topología (510) perteneciente a la línea de producción (210) de la pieza discreta para definir estados de operación de la línea de producción (210) como estados de módulo de parte continua.
- 30 6. El método de cualquier reivindicación anterior, en donde construir (401; 501) el al menos un archivo de definición de módulo (400, 500) comprende generar un mapeo de servicios que mapea al menos un modo de operación de unidad de dicha unidad de parte discreta (202) a un servicio de módulo de parte continua.
7. El método de la reivindicación 6, que comprende generar el mapeo de servicio como un mapeo uno a uno de dicho modo de operación de unidad único de dicha unidad de parte discreta (202) al servicio de módulo de parte continua.
35
8. El método de la reivindicación 6, que comprende generar el mapeo de servicio como un mapeo complejo de múltiples dichos modos de operación de unidad al servicio de módulo de parte continua.
9. El método de cualquier reivindicación anterior, en donde construir (401; 501) el al menos un archivo de definición
40 de módulo (400; 500) comprende generar un ensamblaje de datos que mapea etiquetas o sellos para parámetros de la parte discreta a etiquetas o sellos equivalentes en la parte continua.
10. El método de cualquier reivindicación anterior, en donde construir (401; 501) el al menos un archivo de definición
45 de módulo (400; 500) comprende generar un mapeo de interfaz hombre-máquina - HMI - que mapea elementos de HMI de dichas unidades de parte discreta (202) a elementos equivalentes de un módulo de parte continua (102) usando un marcador de posición genérico o un mapeo específico de unidad especializada.
11. El método de la reivindicación 1, en donde la realización de las traducciones en tiempo de ejecución se basa en un marcador añadido al archivo de definición de módulo (400, 500).
50
12. Un método para integrar módulos en una planta modular híbrida que comprende una parte de fabricación discreta y una parte de fabricación continua, comprendiendo el método:
integrar la parte continua en la parte discreta, que comprende construir una o más interfaces (610) que representan
55 cada módulo de parte continua (102) como una o más unidades de parte discreta respectivas (202); y
ejecutar la planta modular híbrida, que comprende realizar traducciones en tiempo de ejecución de las comunicaciones entre la parte discreta y la parte continua, en donde las partes discreta y continua usan un protocolo de comunicación común.
13. El método de cualquier reivindicación anterior, en donde la parte de fabricación discreta comprende una parte de
60 PackML, en donde la parte de fabricación continua comprende una parte de paquete de tipo módulo - MTP - y en donde el protocolo de comunicación es una arquitectura unificada de OPC - UA de OPC.
14. Un dispositivo informático (800) que comprende un procesador (802) configurado para realizar el método de
65 cualquiera de las reivindicaciones 1-13.

15. Un medio legible por ordenador (804, 808) que comprende instrucciones que, cuando se ejecutan por un dispositivo informático (800), hacen que el dispositivo informático lleve a cabo el método de cualquiera de las reivindicaciones 1-13.

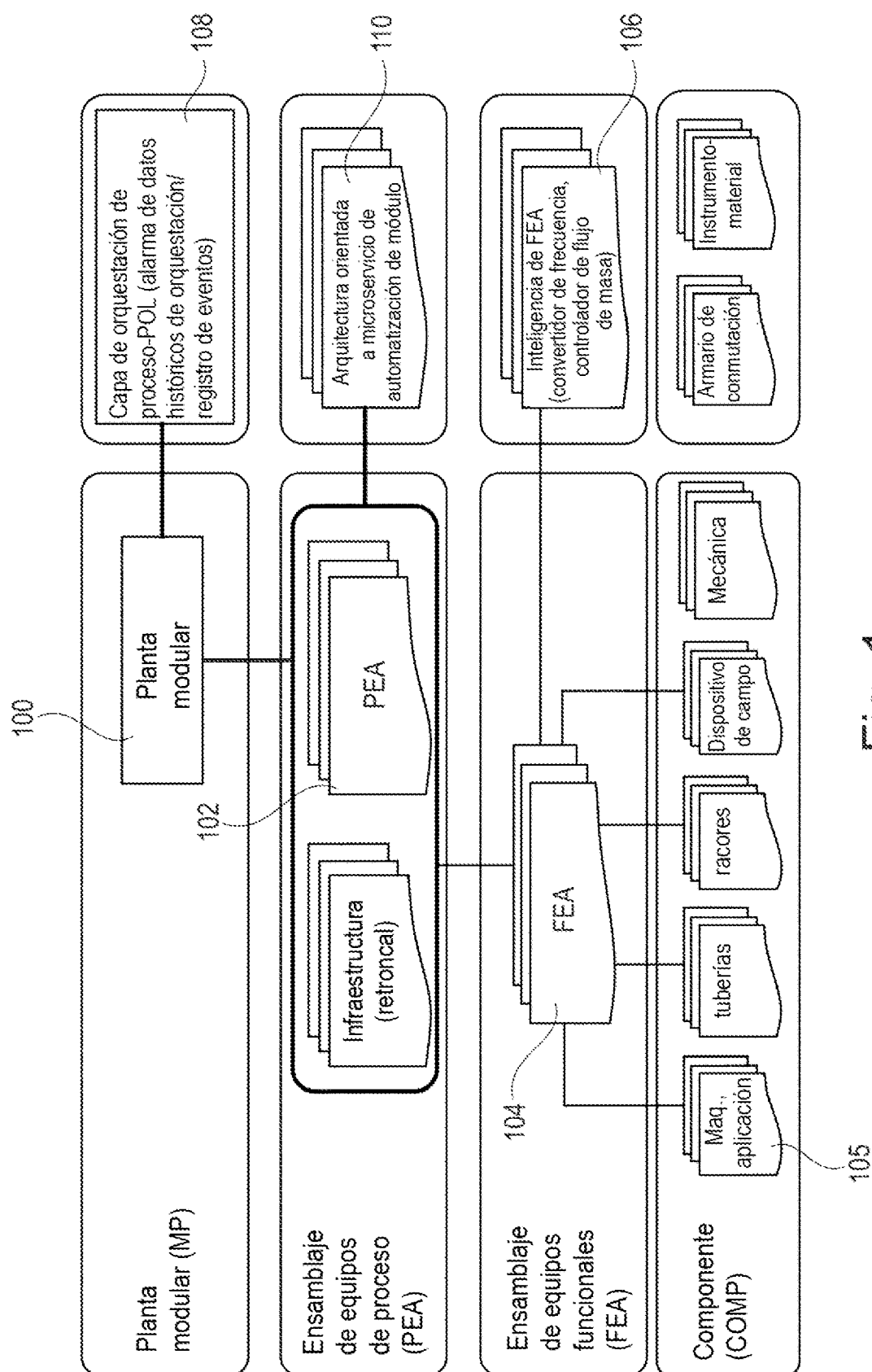


Fig. 1

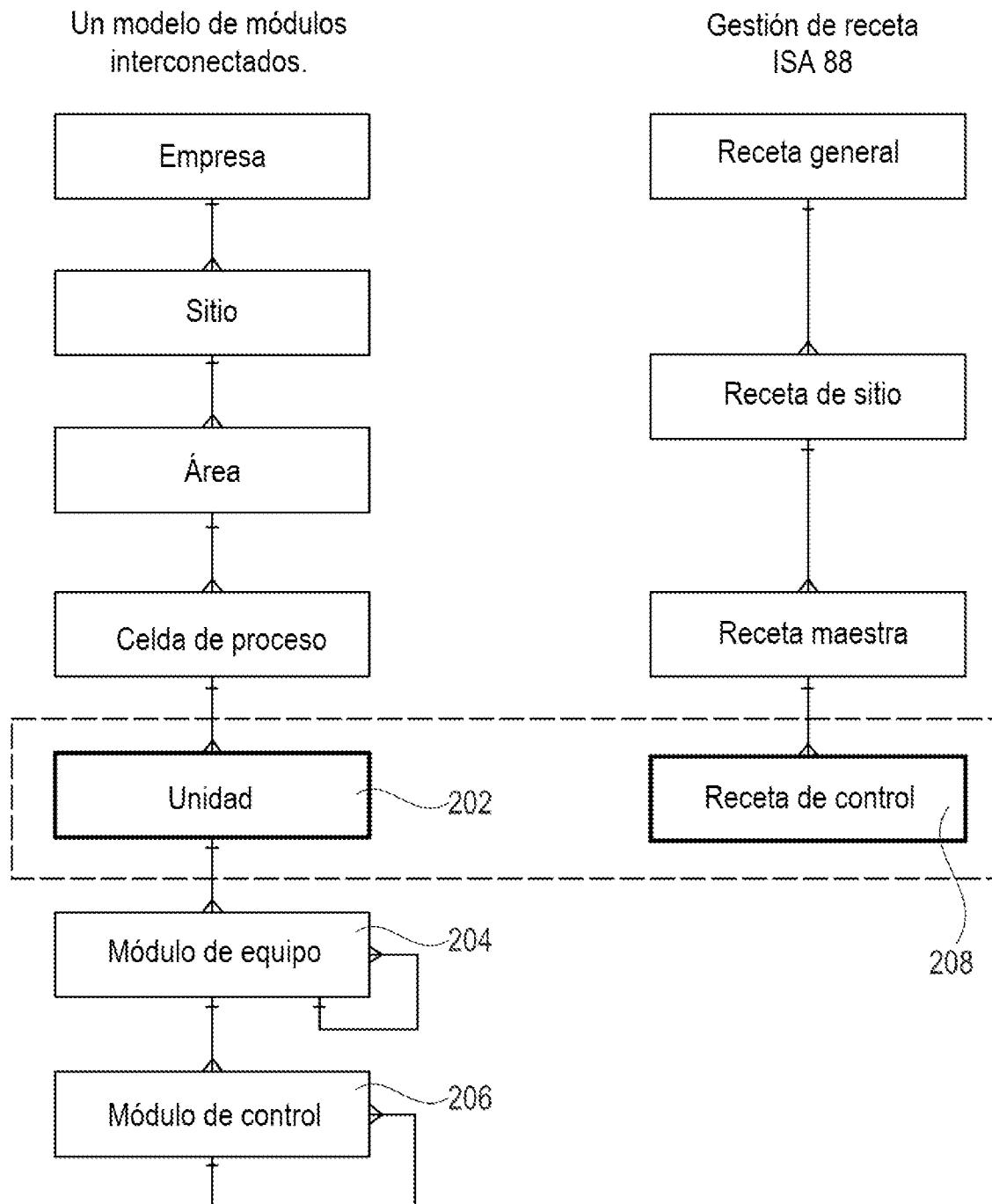


Fig. 2

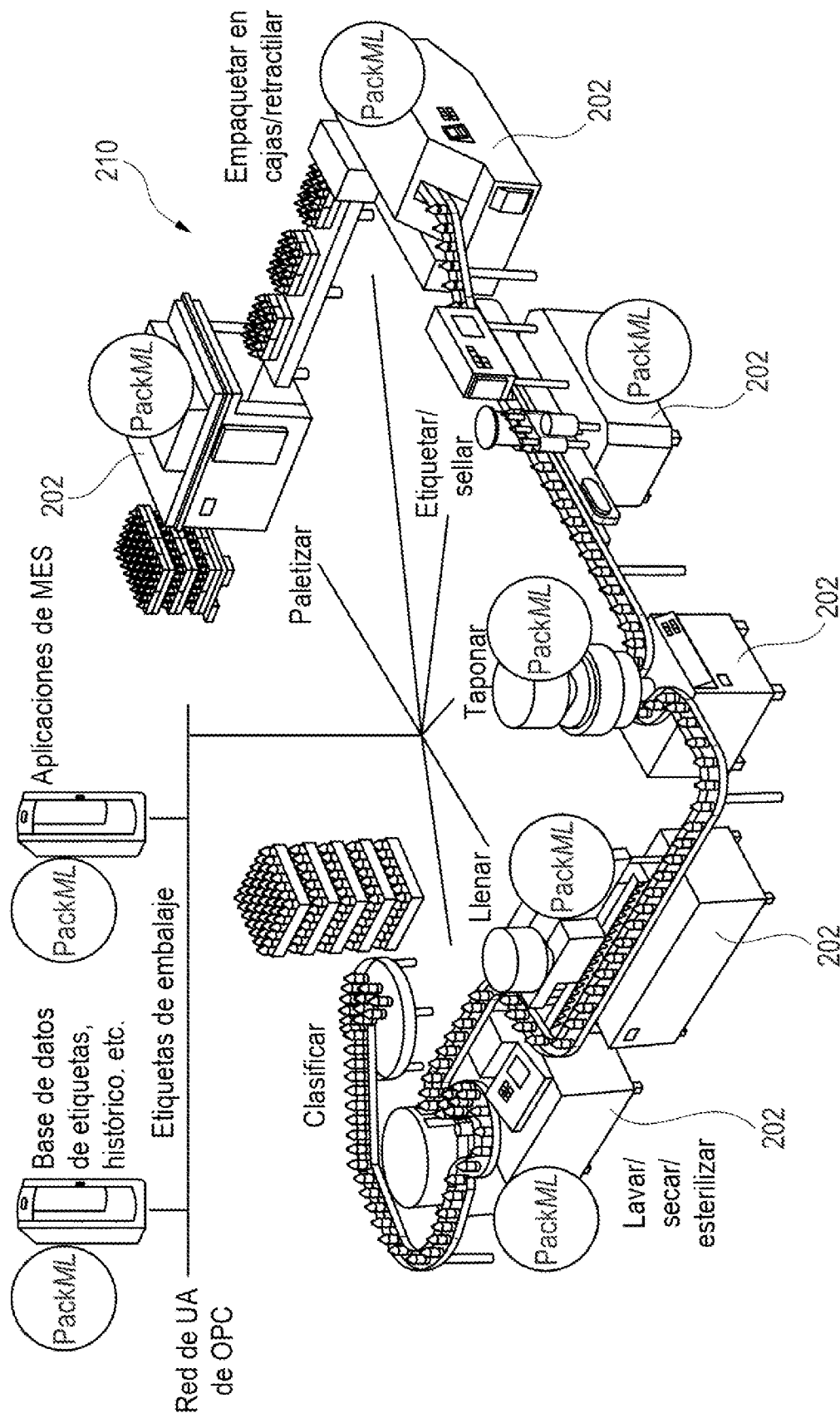


Fig. 3

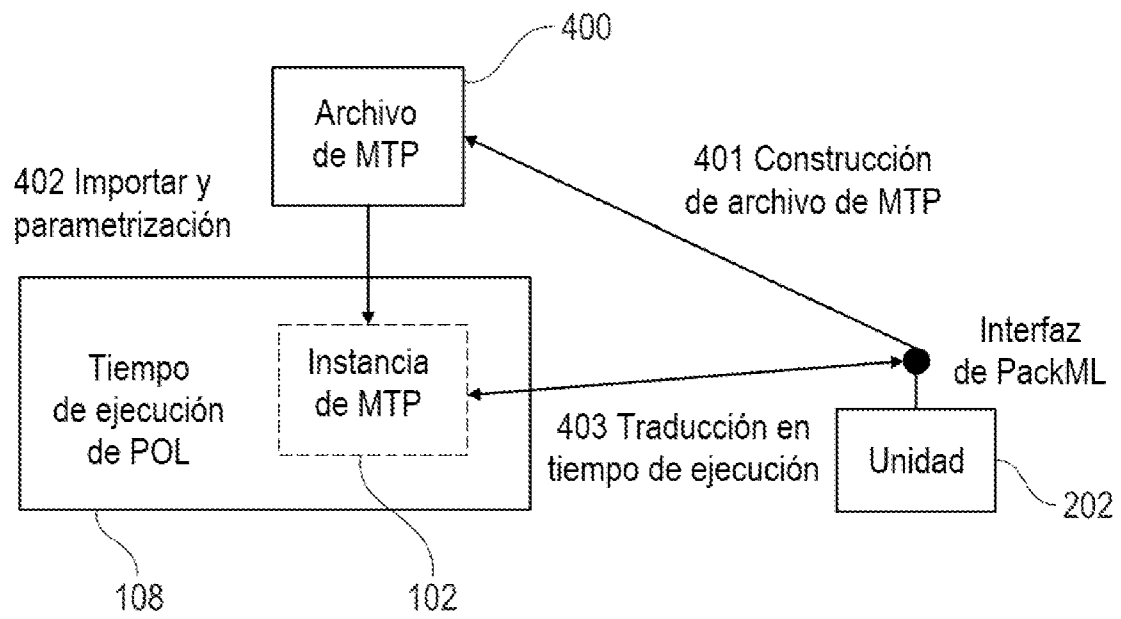


Fig. 4

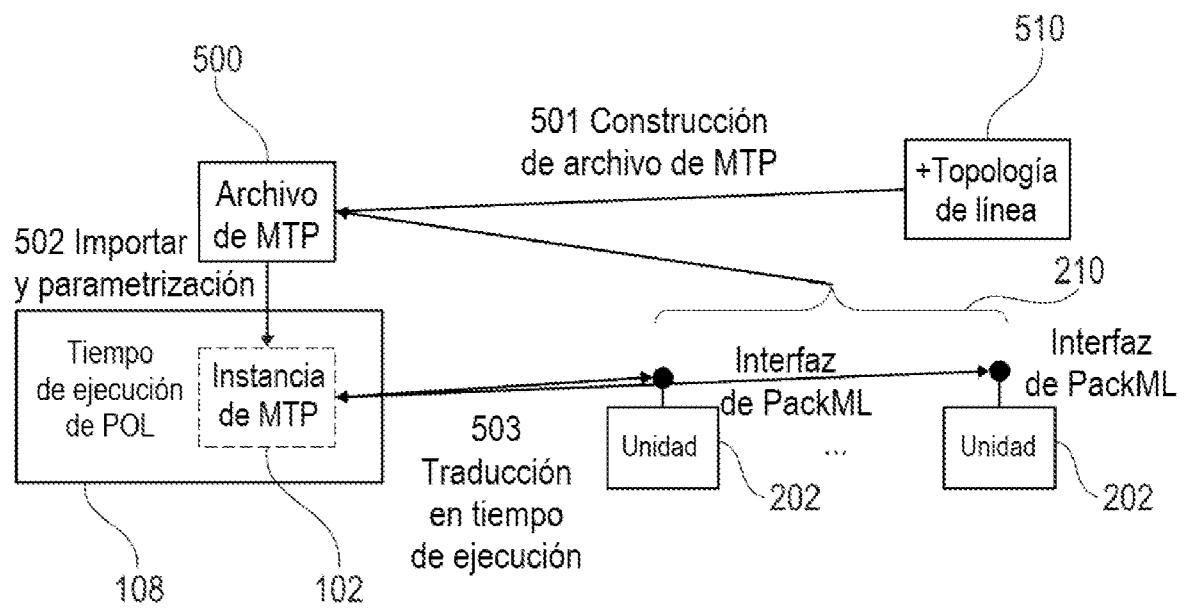


Fig. 5

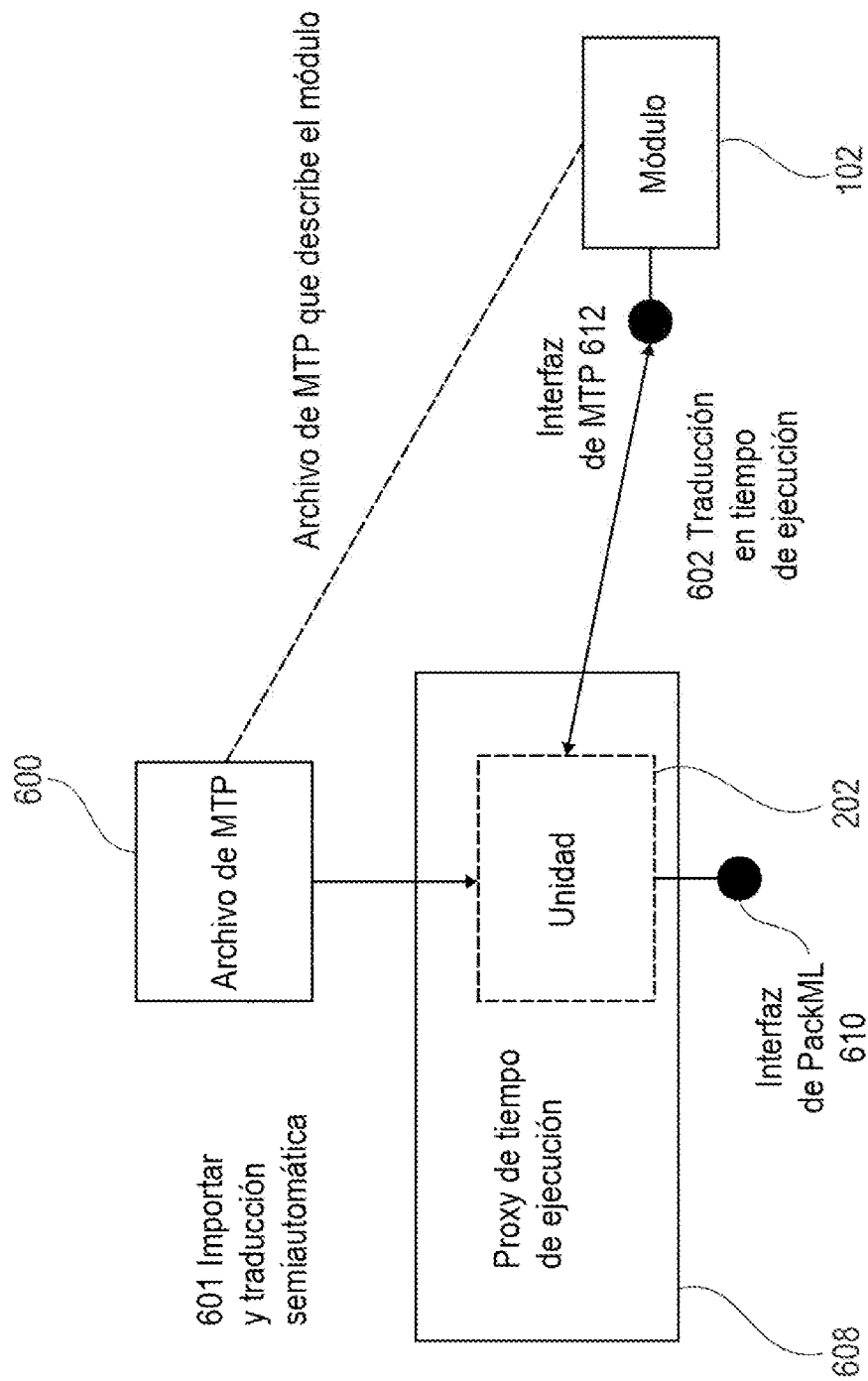


Fig. 6

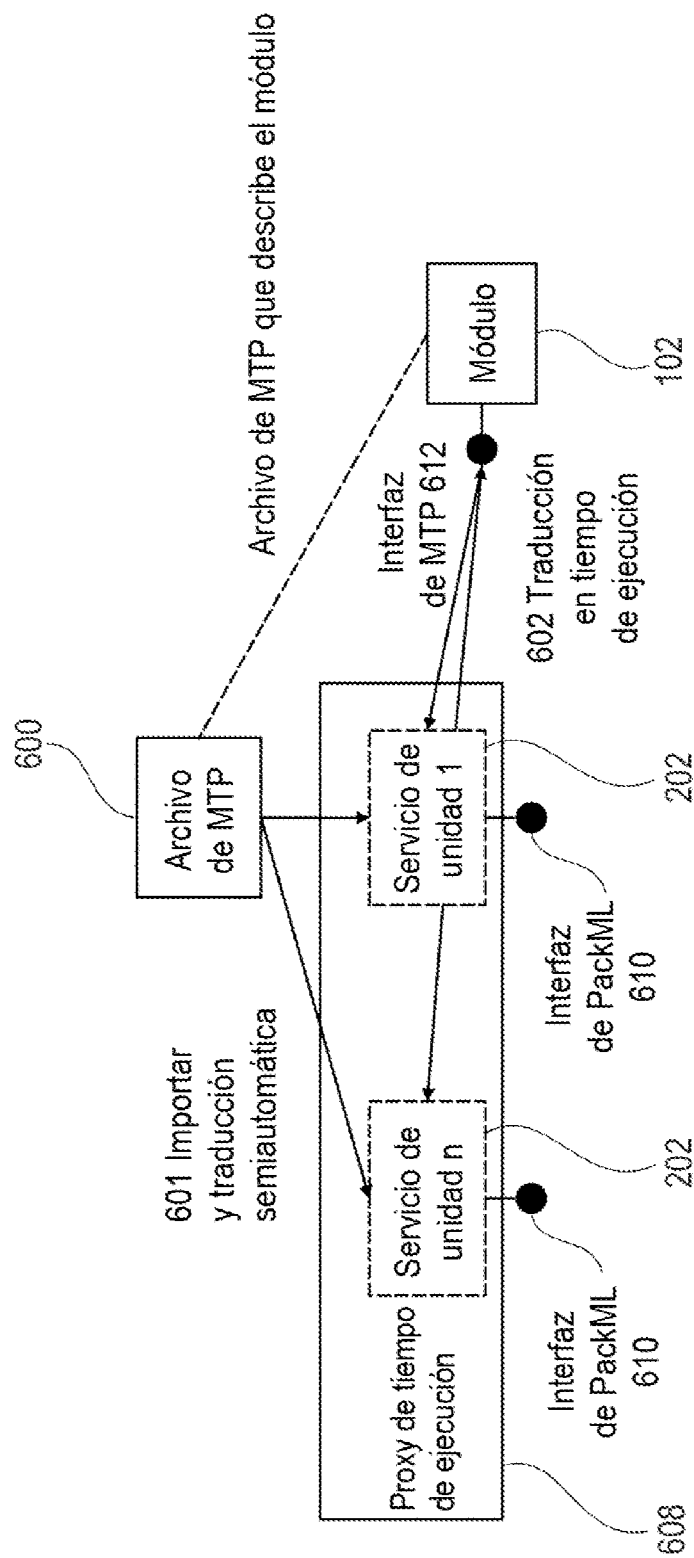


Fig. 7

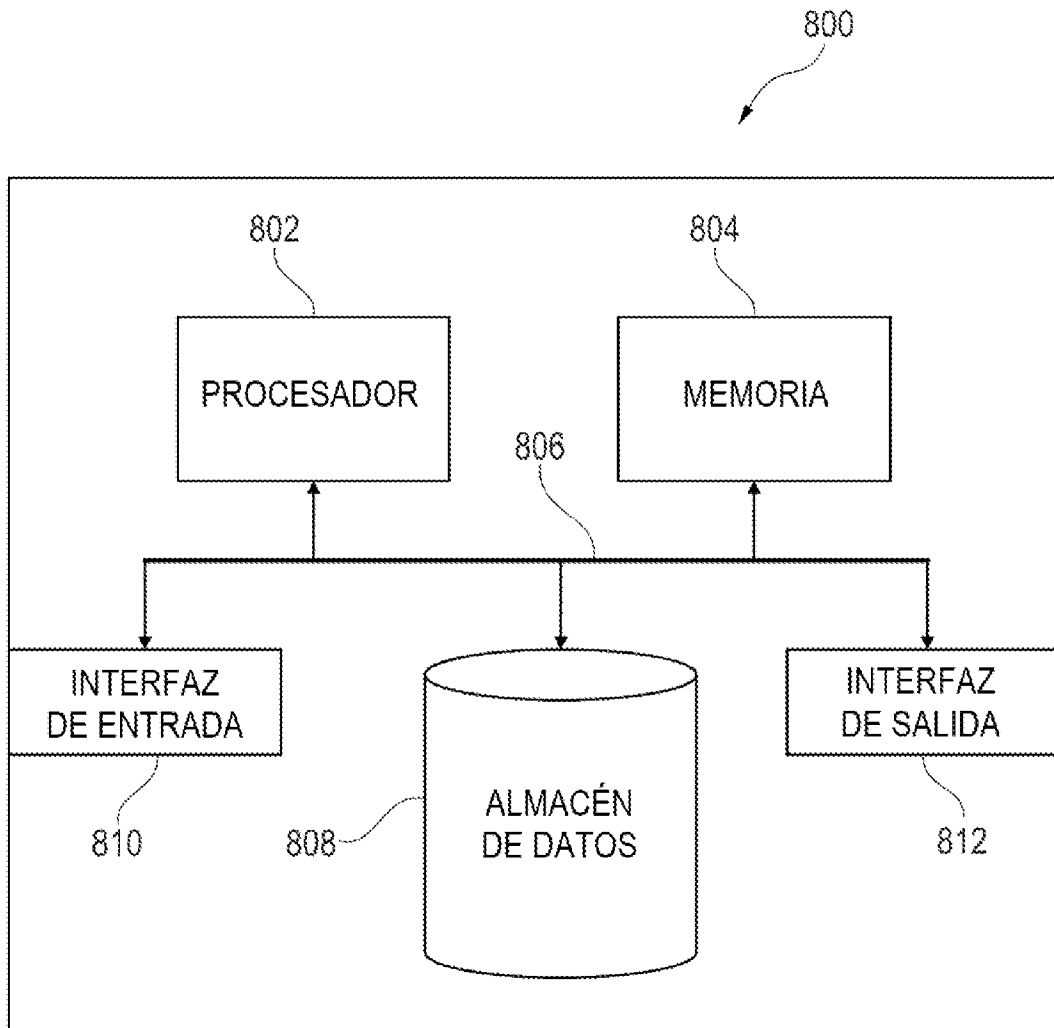


Fig. 8