

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 843 680**

51 Int. Cl.:

H04L 29/08 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **14.08.2014 PCT/GB2014/052499**

87 Fecha y número de publicación internacional: **19.02.2015 WO15022540**

96 Fecha de presentación y número de la solicitud europea: **14.08.2014 E 14761392 (1)**

97 Fecha y número de publicación de la concesión europea: **28.10.2020 EP 3033867**

54 Título: **Procedimiento, sistema y programa informático para la transmisión de datos**

30 Prioridad:

16.08.2013 GB 201314732

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

20.07.2021

73 Titular/es:

**SPARKLE CS LTD (100.0%)
2nd Floor Hygeia House, 66 College Road
Harrow, GB**

72 Inventor/es:

**BRIGHTON, MARK y
FERRER, JUDD**

74 Agente/Representante:

ISERN JARA, Jorge

ES 2 843 680 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

Procedimiento, sistema y programa informático para la transmisión de datos

5 Campo de la invención

La presente invención pertenece al campo de la transmisión de datos. Particularmente, pero no de forma exclusiva, la presente invención se refiere a la inyección de datos en un canal de comunicaciones de un dispositivo de hardware para permitir la transmisión de datos a, por ejemplo, un sistema informático heredado.

10 Antecedentes

A menudo es conveniente agregar funcionalidad a las arquitecturas de tecnología implementadas. Sin embargo, la introducción de nuevas funciones a menudo requiere una modificación sustancial de la tecnología ya implementada. La modificación es generalmente inconveniente ya que puede introducir efectos inesperados dentro de la tecnología interoperativa y aumentar las posibilidades de que ocurran errores dentro de la tecnología compleja.

Un campo que encuentra un desafío para mejorar la funcionalidad es el sector minorista, particularmente en lo que respecta a sus terminales de punto de venta, que tienden a ser instalados y operados por proveedores pertenecientes a monopolios de tecnología.

El comercio minorista es, a la inversa, un campo que obtendría enormes ventajas de los nuevos procedimientos. Por ejemplo, los cupones en nuevas formas han atraído el interés y la participación de los consumidores. A los minoristas les gustaría implementar procedimientos nuevos e inteligentes para administrar los cupones en sus tiendas.

En consecuencia, existe el deseo de que se añadan funciones adicionales que implementen nuevos procedimientos a los sistemas informáticos heredados, en particular en el comercio minorista.

Con los cupones, hay varias mejoras que mejorarían de manera útil la funcionalidad de los sistemas minoristas existentes. Por ejemplo, puede ser conveniente que un minorista aplique fácilmente cupones, como cupones digitales, que no pueden escanearse.

Para implementar estas mejoras, se requiere un procedimiento y un sistema de transmisión de datos mejorados que sea fácil de implementar en entornos heredados.

Es un objeto de la presente invención proporcionar un sistema de transmisión de datos que supere las desventajas de la técnica anterior, o al menos proporcione una alternativa útil.

El documento de patente 2011214131 de Estados Unidos divulga un sistema y un procedimiento para habilitar las funciones o servicios adicionales de un dispositivo mediante el uso de una puerta de enlace/proxy transparente.

Sumario de la invención

45 La invención se describe en las reivindicaciones independientes. Las realizaciones adicionales se describen en las reivindicaciones dependientes.

Breve descripción de los dibujos

50 Las realizaciones de la invención se describirán ahora, a manera de ejemplo solamente, con referencia a los dibujos acompañantes, en los cuales:

Figura 1: muestra un sistema de transmisión de datos de acuerdo con una realización de la invención;

55 Figura 2: muestra otro sistema de transmisión de datos de acuerdo con una realización de la invención;

Figura 3: muestra una arquitectura de comunicaciones de acuerdo con una realización de la invención;

60 Figura 4: muestra otra arquitectura de comunicaciones de acuerdo con una realización de la invención;

Figura 5: muestra un diagrama de flujo que ilustra un procedimiento de acuerdo con una realización de la invención.

65 Figura 6: muestra un diagrama de bloques que ilustra un formato de mensaje de señales usadas dentro de una realización de la invención;

Figura 7: muestra un diagrama de bloques que ilustra un módulo de inyección de software dentro de un sistema operativo de acuerdo con una realización de la invención;

5 Figura 8: muestra un diagrama de secuencia que ilustra el flujo de mensajes dentro de un procedimiento de acuerdo con una realización de la invención; y

Figura 9: muestra un sistema de procesamiento de datos de acuerdo con una realización de la invención;

Descripción detallada de las realizaciones preferentes

10 La presente invención proporciona un procedimiento de transmisión de datos.

15 El inventor ha descubierto que los datos pueden transmitirse a un sistema informático heredado, como un sistema de terminal POS, inyectando datos en un canal de comunicaciones con un dispositivo de hardware y el sistema informático.

20 Por ejemplo, un terminal POS puede configurarse para recibir información de código de barras desde un dispositivo de entrada de hardware de escáner de código de barras al sistema informático del terminal POS. Los datos pueden falsificarse para que parezcan que provienen de un lector de código de barras y se inyectan en el canal de comunicaciones del software de aplicación del terminal POS.

En la Figura 1, se muestra un sistema de transmisión de datos 100 de acuerdo con una realización de la invención.

25 Se muestra un sistema informático 101. El sistema informático 101 se configura para ejecutar una aplicación de software. El sistema informático 101 puede ser un terminal de punto de venta (POS) configurado para ejecutar una aplicación de software de punto de venta electrónico (EPOS).

El sistema informático 101 incluye un procesador y una memoria.

30 Se muestra un aparato de inyección 102 que implementa un módulo de inyección. El aparato de inyección 102 puede conectarse al sistema informático 101, por ejemplo, mediante un puerto en el sistema informático 101. La conexión permite que las señales se transmitan y reciban entre el aparato de inyección 102 y el sistema informático 101.

35 El aparato de inyección 102 incluye un procesador y una memoria.

Se muestra un dispositivo periférico 103. El dispositivo periférico 103 puede ser un dispositivo de hardware de entrada, como un escáner QR, un lector NFC, o una cámara.

40 En una realización alternativa, el sistema de transmisión de datos 100 no requiere el dispositivo periférico 103.

El dispositivo periférico 103 puede conectarse al aparato de inyección 102 de manera que las señales puedan transmitirse y recibirse entre el aparato de inyección 102 y el dispositivo periférico 103.

45 Se muestra un dispositivo periférico virtual 104. El dispositivo periférico virtual 104 puede ser un dispositivo de hardware de entrada, como un escáner de código de barras o cualquier otro tipo de dispositivo de entrada. El dispositivo periférico virtual 104 es virtual en el sentido de que no existe, pero su existencia está suplantada a la aplicación en el sistema informático 101 por el aparato de inyección 102.

50 El aparato de inyección 102 puede configurarse para crear un canal de comunicaciones que parece provenir del dispositivo periférico virtual 104 y la aplicación que se ejecuta en el sistema informático 101. El aparato 102 de inyección puede configurarse además para inyectar señales en el canal de comunicaciones entre el dispositivo periférico virtual 104 y la aplicación en el sistema informático 101.

55 El aparato de inyección 102 puede configurarse para procesar señales recibidas desde el dispositivo periférico real 103 y generar señales en respuesta a la inyección en el canal de comunicaciones.

En la Figura 2, se muestra un sistema de transmisión de datos 200 de acuerdo con otra realización de la invención.

60 Se muestra un sistema informático 201. El sistema informático 201 se configura para ejecutar una aplicación de software 201a. El sistema informático 201 puede ser un terminal de punto de venta (POS) configurado para ejecutar una aplicación de software de punto de venta electrónico (EPOS).

El sistema informático 201 incluye un procesador y una memoria.

65

Se muestra un dispositivo periférico 202. El dispositivo periférico 202 puede ser un dispositivo de hardware de entrada, como un escáner QR, un lector NFC, o una cámara.

En una realización alternativa, el sistema de transmisión de datos 200 no requiere el dispositivo periférico 202.

5 El dispositivo periférico 202 está conectado al sistema informático 201 de manera que las señales pueden transmitirse y recibirse entre el sistema informático 201 y el dispositivo periférico 202.

10 Se muestra un dispositivo periférico virtual 203. El dispositivo periférico virtual 203 puede ser un dispositivo de hardware de entrada, como un escáner de código de barras o cualquier otro tipo de dispositivo de entrada. El dispositivo periférico virtual 203 es virtual en el sentido de que no existe, pero su existencia está suplantada a la aplicación 201a en el sistema informático 201 mediante un módulo de inyección 204.

15 El sistema informático 201 puede configurarse además para ejecutar el módulo de inyección 204.

20 El módulo de inyección 204 puede configurarse, a su vez, para crear un canal de comunicaciones que parece provenir del dispositivo periférico virtual 203 y la aplicación 201a que se ejecuta en el sistema informático 201. El módulo de inyección 204 puede configurarse además para inyectar señales en el canal de comunicaciones entre el dispositivo periférico virtual 203 y la aplicación 201a en el sistema informático 201.

25 El módulo de inyección 204 puede configurarse para procesar señales recibidas desde el dispositivo periférico real 202 y generar señales en respuesta a la inyección en el canal de comunicaciones.

En la Figura 3, se muestra una implementación ilustrativa de un sistema de transmisión de datos 300 dentro de una arquitectura de red de acuerdo con una realización de la invención.

Se muestra un sistema de transmisión de datos 300 como se describió en relación con las Figuras 1 o 2.

30 El sistema de transmisión de datos 300 puede configurarse además para comunicarse con un servidor 301 a través de una red 302.

En la Figura 4, se muestra una implementación ilustrativa alternativa de un sistema de transmisión de datos dentro de una arquitectura de red de acuerdo con una realización de la invención.

35 Se muestra una pluralidad de sistemas de transmisión de datos 400, 401, y 402 como se describe en relación con las Figuras 1 o 2.

40 Los sistemas de transmisión de datos 400, 401 y 402 pueden conectarse a una red 403. La red 403 puede conectarse a una segunda red 404 a través de una puerta de enlace 405.

Los sistemas de transmisión de datos 400, 401, y 402 pueden configurarse además para comunicarse con un servidor 406 a través de la primera red 403, la puerta de enlace 405, y la segunda red 404.

45 Se apreciará que pueden contemplarse implementaciones alternativas del sistema de transmisión de datos. Por ejemplo, con las arquitecturas de computación en la nube, las funciones pueden distribuirse entre una pluralidad de dispositivos o servidores, que pueden interconectarse mediante diversas configuraciones de red y comunicaciones

En la Figura 5, se muestra un procedimiento 500 de acuerdo con una realización de la invención.

50 En la etapa 501, se elabora un canal de comunicaciones entre un primer periférico y una aplicación que se ejecuta en el sistema informático mediante un módulo de inyección. El canal de comunicaciones puede falsificarse, ya que el primer periférico puede no existir.

55 En la etapa 502, el módulo de inyección puede recibir señales de otro dispositivo periférico. Las señales pueden estar relacionadas con datos capturados por el dispositivo periférico adicional. Por ejemplo, el dispositivo periférico adicional puede capturar y codificar datos QR, información NFC o información fotográfica. El dispositivo periférico adicional puede no ser compatible directamente con el software de aplicación del sistema informático.

60 En una realización como se muestra en la etapa 503, pueden recibir señales desde un servidor externo. Las señales pueden recibirse en respuesta a señales recibidas en la etapa 502 y transmitidas al servidor, o en respuesta a eventos activados en el servidor.

65 En una realización, el módulo de inyección puede, como parte del procedimiento, generar la visualización de un aviso para el operador del sistema informático. El aviso puede visualizarse en una pantalla del sistema informático o el aviso puede visualizarse en una pantalla externa conectada, por ejemplo, al aparato de inyección. La respuesta

del operador al aviso puede recibirse mediante el módulo de inyección y determinar el contenido o tipo de señales a inyectar.

5 En la etapa 504, el módulo de inyección inyecta señales en el canal de comunicaciones entre el primer dispositivo periférico para un sistema informático y una aplicación que se ejecuta en el sistema informático, para que las reciba la aplicación.

Las señales inyectadas pueden ser las recibidas, o pueden crearse nuevas señales en respuesta a esas señales.

10 Por ejemplo, el módulo de inyección puede reprocesar las señales recibidas en un formato de mensaje que el software de aplicación espera, y/o el módulo de inyección puede capturar las señales recibidas y generar nuevas señales en respuesta ocasionalmente y en una secuencia esperada mediante el software de aplicación.

15 Las señales pueden recibirse desde múltiples dispositivos periféricos y procesarse mediante el módulo de inyección para generar una o más señales nuevas para inyectarlas en el canal de comunicaciones falso.

20 En una realización, el módulo de inyección también puede configurarse para interceptar señales de un tercer dispositivo periférico, como un periférico configurado para su uso con el software de aplicación (es decir, un escáner de código de barras para usar con un terminal POS), que forma un canal de comunicaciones con el software de aplicación. El módulo de inyección puede configurarse para procesar esas señales junto con las señales recibidas desde el periférico adicional para generar nuevas señales en el canal de comunicaciones falso.

25 La Figura 6 muestra el formato de señales esperadas por el software de aplicación del dispositivo periférico virtual en una realización de la invención.

Dentro de esa figura:

PID = ID de paquete que identifica el tipo de paquete que se envía;

30 ADDR = Campo de dirección del dispositivo para el que está destinado el paquete;

ENDP = Punto final;

35 CRC = Verificación de Redundancia Cíclica; y

EOP = Fin del paquete.

40 Los paquetes token 600 se transmiten desde el software de aplicación para que los reciba el dispositivo periférico virtual y son interceptados por el módulo de inyección.

En respuesta a la recepción de los paquetes token 600, los paquetes de datos 601 se construyen y envían desde el módulo de inyección al software de aplicación.

45 En respuesta a la recepción de los paquetes de datos 601, los paquetes de estado 602 se construyen y se envían desde el software de aplicación para que los reciba el dispositivo periférico virtual, pero son interceptados por el módulo de inyección.

50 Puede verse a partir de lo anterior que el módulo de inyección puede configurarse para generar señales formateadas de acuerdo con un formato de mensaje, como el descrito anteriormente. De esta forma, el software de aplicación que recibe los datos interpreta que proviene de un periférico existente y la existencia del módulo de inyección es transparente.

55 Se describirá ahora un módulo de inyección de software de acuerdo con una realización de la invención con referencia a la Figura 7.

El procedimiento se describirá en relación con el uso de un módulo de inyección de software 700 que se ejecuta en el sistema informático y en relación con el uso para mejorar un sistema informático que ejecuta una aplicación de software de punto de venta (POS) 701.

60 El módulo de inyección de software 700 se instala a nivel de sistema operativo en el mismo dispositivo (el sistema informático) que aloja el software controlador 702 para los periféricos 703 esperados por el POS 701.

Para crear e instalar el módulo de inyección de software 700:

65

1. Se crea un objeto de software "wedge" 704 que presenta la misma interfaz para la aplicación POS 701 que la clase de periférico original 705 para un tipo de periférico que es conocido por el POS 701 pero cuya existencia 705a será falsificada al POS 701.

2. El software POS 701 se configura para cargar el objeto wedge 704 en lugar de la clase original 705.

5 3. El módulo de inyección de software 700 a su vez carga un objeto de servicio 706 para un periférico 706a de un tipo no conocido por el POS 701. Un componente del constructor de cestas 707 puede, por ejemplo, cargar el objeto de servicio y gestionar la comunicación hacia/desde el nuevo periférico 706a.

10 La siguiente descripción es para POS para .NET pero, con las modificaciones adecuadas, la invención de sustituir el registro de un objeto por otro funcionaría igualmente bien con otros estándares de controladores periféricos como OPOS y JPOS.

15 Inicialmente se crea un objeto de software 704 que presenta la misma interfaz al POS que la clase de escáner de barras original. En POS para .NET estos objetos se denominan Objetos de Servicio o SOs. El software POS 701 debe configurarse para cargar el SO creado en lugar de la clase original 705. La forma en que esto suceda dependerá del software del cliente 701. Por ejemplo, si el software del cliente 701 tiene un archivo de configuración que contiene el nombre de un SO para cargar entonces esto puede modificarse para cargar el wedge 704 en su lugar. Si tal archivo de configuración no está disponible y el SO se está cargando directamente por nombre, entonces el objeto de reemplazo 704 podría recibir el mismo nombre que el original 705 y copiarse en su lugar en el sistema de archivo.

20 Puede usarse un objeto de navegador de objetos de servicio POS para .NET para recuperar una lista de SOs disponibles que pueden consultarse de acuerdo con el tipo de dispositivo que representan. Una clave de registro contiene valores que representan directorios donde residen los binarios de objetos de servicio. El navegador se usa para encontrar los SO. El directorio del objeto wedge se agrega a esta clave. Existe un mecanismo en POS 701 para .NET para redefinir el dispositivo de hardware con el que está asociado un SO, y esto puede usarse para vincular dinámicamente el SO wedge 704 con un ID de dispositivo de hardware particular en el momento de la instalación. Se escribe un archivo de configuración que asigna el ID de hardware al nombre de un SO en particular, y esto se coloca en una carpeta que se agrega a otra clave de registro usada por el POS para .NET para ubicar dichos archivos de configuración. En el momento de la instanciación del dispositivo POS para .NET cargará cualquier archivo de configuración que encuentre y usará dinámicamente el mapeo contenido para cargar el SO apropiado para el dispositivo con la ID de hardware referenciada que se está abriendo. El ID del hardware que se reemplazará puede extraerse de los metadatos expuestos por el administrador de hardware del dispositivo.

25 30 El SO wedge 704 puede crearse de manera que implemente de forma completa la jerarquía de clases que se está falsificando. Por ejemplo el estándar POS para .NET define una clase para lectores que tiene en su nivel superior una clase llamada ScannerBase. Esto a su vez hereda de muchas otras clases, todas las cuales pueden implementarse para mantener la jerarquía de herencia y la naturaleza polimórfica de la clase.

35 40 Una vez que se completa este procedimiento, cuando el POS 701 llama a un método de interfaz, está llamando al método asociado en el wedge 704 y cuando un evento es activado por nuevos datos, de manera similar se retroalimenta al POS 701 a través del wedge 704. Siempre que el objeto wedge 704 y el constructor de cestas 707 administren los procedimientos y eventos del software POS 701 y reempaquetar cualquier mensaje del nuevo periférico 706a, el software de POS 701 funcionará como si el objeto wedge 704 no existiera e interpretará que interactúa con un periférico conocido 705a.

45 50 El wedge 704 y el constructor de cestas 707 pueden incluir lógica de procesamiento adicional para redirigir el procedimiento y los datos del evento a otros módulos de software del 708 al 710 para procesar los datos recibidos y, en base a las instrucciones de esos módulos, reemplazar, descartar o inyectar nuevos datos en el transporte de datos originales hacia/desde el nuevo periférico 706a.

Se describirá ahora un diagrama de secuencia que muestra el flujo de mensajes dentro de un sistema de transmisión de datos de acuerdo con una realización de la invención con referencia a la Figura 8.

55 Un código se escanea, por ejemplo, mediante un escáner QR o RFID y los datos se envían en la etapa 800 al software de aplicación para un terminal de punto de venta electrónico (EPOS). Es posible que el terminal EPOS no conozca el escáner QR o RFID.

60 El código puede presentarse para escanear durante el escaneo de una serie de productos para un cliente. La serie de productos y cualquier cupón asociado, identificadores personales y tokens de pago, se denominarán cesta.

Un módulo de inyección recibe el código y envía el código a un módulo evaluador y constructor de cestas en la etapa 801.

65

El módulo de constructor y evaluador de cestas agrega cualquier producto que identifica dentro del código a la cesta que está rastreando para el cliente en la etapa 802. El módulo de inyección también puede recibir códigos para construir la cesta desde otros periféricos, como lectores de código de barras.

5 El módulo evaluador y constructor de cestas puede entonces realizar una solicitud de búsqueda mediante el uso del código de un servidor remoto en la etapa 803. Por ejemplo, si el código se relaciona con un cupón presentado y no con un producto en particular, es posible que el creador de la cesta y el evaluador necesiten verificar si el cupón es válido.

10 El módulo de constructor y evaluador de cestas puede evaluar la cesta actual para que el cliente determine si los productos dentro de la cesta califican para una acción particular (por ejemplo, un descuento) en la etapa 804.

El módulo de constructor y evaluador de cestas puede solicitar una entrada del usuario/operador de EPOS antes de aplicar una acción en la etapa 805. Por ejemplo, puede haber condiciones locales que el usuario debe verificar (por ejemplo, si el cliente tiene más de 65 años o es mujer, etc.) Estas condiciones locales pueden ser necesarias, por ejemplo, en relación con un cupón presentado o a los descuentos particulares actualmente en vigor. Alternativamente, el aviso puede informar al operador de EPOS que un cupón parece no ser válido y darles la opción de aceptar o rechazar el cupón; o el aviso puede solicitar un código para cupones protegidos por CAN (Número de Autenticación de Cupón) de un solo uso, que el cliente o el operador ingresa (el código puede luego verificar el uso anterior).

20 El módulo de constructor de cestas y evaluador puede solicitar información en diferentes momentos durante el procedimiento.

25 El módulo de constructor y evaluador de cestas puede generar instrucciones, en la etapa 806, para el módulo de inyección en base a un o varios productos de la constitución de la cesta, la validez de cualquier cupón presentado y la verificación de los elementos requeridos.

30 Las instrucciones pueden dar como resultado que el módulo de inyección genere datos en la etapa 807 que instruirían una acción conocida por el EPOS (por ejemplo, aplicar un descuento de una cantidad o porcentaje específico). Los datos pueden enviarse en la etapa 808 al software EPOS, por ejemplo, inyectando los datos en un canal de comunicaciones falso creado por el módulo de inyección entre un periférico virtual y el software EPOS.

35 Aunque la Figura 8 muestra un lector RFID como fuente de un código escaneado, el módulo de inyección puede igualmente interceptar la alimentación de datos de uno o más periféricos adicionales, incluidos los periféricos EPOS estándar (por ejemplo, caja registradora, impresora de recibos), periféricos patentados, y periféricos de productos básicos.

40 Las instrucciones devueltas al módulo de inyección por el constructor de cestas y el módulo evaluador pueden ser instrucciones múltiples. Por lo tanto, un código QR escaneado por un lector óptico patentado podría generar varios códigos de barras inyectados que se envían como si fueran de un lector de código de barras.

El módulo de inyección puede basarse en hardware o software y la mensajería es la misma.

45 Un módulo de inyección implementado dentro de un aparato de inyección de acuerdo con una realización de la invención se describirá ahora con referencia a la Figura 9.

El aparato de inyección puede comprender un procesador y una memoria dentro de un nodo de comunicaciones 900 o dentro de una pantalla interactiva adjunta 901 (una pantalla y una entrada).

50 Las comunicaciones enrutadas a través del nodo 900 pueden recibirse mediante el módulo de inyección.

El nodo de comunicaciones 900 puede tener conexiones USB u otras entradas de comunicaciones. Pueden usarse adaptadores para conectar las entradas/salidas que no sean USB a las conexiones USB.

55 Los periféricos de entrada y salida del 902 al 907 pueden conectarse al nodo de comunicaciones 900.

60 El nodo de comunicaciones 900 puede estar conectado a un terminal de punto de venta electrónico (POS) 908 a través de una o más conexiones de comunicación. Por ejemplo, el terminal POS 908 puede utilizar conexiones de entrada/salida estándar de la industria como "entrada de lector", "salida de impresora", y "salida de caja de efectivo".

El nodo de comunicaciones 900 puede incluir un dongle wifi 909 y/o una conexión Ethernet 910 para comunicarse con un servidor externo 911.

65 Cuando el módulo de inyección inyecta un nuevo código de barras, "se apropiará" de un paquete de token enviado desde el POS 908 y devolverá su propio paquete de datos que contiene el código inyectado. Después de recibir el

paquete de estado resultante del POS, y suponiendo que ninguna otra configuración lo impida, se reanudan las comunicaciones normales entre el periférico y el POS.

5 Tenga en cuenta que en una configuración alternativa, el módulo de inyección nunca puede bloquear ninguna comunicación entre un periférico estándar del 905 al 907 y el host 908. En esta configuración, se activará un aviso después de que ya se haya pasado un código de barras de cupón al POS 908 (posiblemente confiando en que el operador del POS elimine manualmente un cupón no válido) y se inyectará un nuevo código (por ejemplo) falsificando un lector de código de barras adicional. Esta configuración puede ser necesaria para una mayor fiabilidad en un entorno minorista.

10 Una ventaja potencial de algunas realizaciones de la presente invención es que los datos pueden transmitirse a una aplicación en un sistema informático donde la aplicación se configura para interoperar con un periférico externo esperado sin requerir que la aplicación sea modificada. En consecuencia, un sistema de transmisión de datos de acuerdo con la invención da como resultado un sistema informático mejorado que puede ejecutar un software heredado sin dejar de ser confiable.

20

25

30

35

40

45

50

55

60

65

REIVINDICACIONES

1. Un procedimiento de transmisión de datos, que incluye:
 - a) Un módulo de inyección de software que construye un canal de comunicaciones entre un dispositivo periférico para un punto de venta electrónico, EPOS, un terminal y una aplicación EPOS que se ejecuta en el terminal; y
 - b) El módulo de inyección de software que inyecta señales en el canal de comunicaciones para que las reciba la aplicación EPOS; en el que el módulo de inyección de software incluye un objeto de software wedge, el software wedge presenta la misma interfaz para la aplicación EPOS como una clase de periférico original para un tipo de periférico que es conocido por la aplicación EPOS, pero cuya existencia será falsificada para la aplicación EPOS, la aplicación EPOS carga el objeto wedge en lugar de la clase original, y el módulo de inyección de software carga un objeto de servicio para un periférico de un tipo desconocido para la aplicación EPOS.
2. Un procedimiento de transmisión de datos como se reivindicó en la reivindicación 1, en el que el dispositivo periférico es un dispositivo periférico virtual y el canal de comunicaciones se construye falsificando la existencia del dispositivo periférico virtual.
3. Un procedimiento de transmisión de datos como se reivindicó en cualquier reivindicación anterior, en el que el dispositivo periférico es un dispositivo de entrada.
4. Un procedimiento de transmisión de datos de acuerdo con cualquier reivindicación anterior, en el que las señales se crean en respuesta a señales precedentes interceptadas desde otro dispositivo periférico a la aplicación.
5. Un procedimiento de transmisión de datos de acuerdo con cualquier reivindicación anterior, en el que las señales se crean, al menos en parte, de acuerdo con un conjunto de instrucciones almacenadas.
6. Un procedimiento de transmisión de datos de acuerdo con cualquier reivindicación anterior, en el que las señales se crean, al menos en parte, con la ayuda de un servidor externo.
7. Un procedimiento de transmisión de datos de acuerdo con cualquier reivindicación anterior, en el que las señales se inyectan en el nivel del sistema operativo.
8. Un procedimiento de transmisión de datos de acuerdo con cualquier reivindicación anterior, en el que las señales son inyectadas mediante un dispositivo de hardware intermediario en el que está conectado el dispositivo periférico y que a su vez está conectado al dispositivo informático.
9. Un procedimiento de transmisión de datos de acuerdo con cualquier reivindicación anterior, en el que el dispositivo periférico es un lector de código de barras.
10. Un sistema de transmisión de datos, que comprende:
 - un procesador configurado para establecer un canal de comunicaciones entre un dispositivo periférico para un punto de venta electrónico, EPOS, un terminal y una aplicación EPOS que se ejecuta en el terminal e inyecta señales en el canal de comunicaciones para que las reciba la aplicación mediante el uso de un módulo de inyección de software; y
 - un módulo de comunicaciones configurado para comunicarse con una aplicación EPOS que se ejecuta en el terminal a través del canal de comunicaciones;
 - en el que el módulo de inyección de software incluye un objeto de software wedge, el software wedge presenta la misma interfaz para la aplicación EPOS como una clase de periférico original para un tipo de periférico que es conocido por la aplicación EPOS, pero cuya existencia será falsificada para la aplicación EPOS, la aplicación EPOS carga el objeto wedge en lugar de la clase original, y el módulo de inyección de software carga un objeto de servicio para un periférico de un tipo desconocido para la aplicación EPOS.
11. Un sistema de transmisión de datos como se reivindicó en la reivindicación 10, que comprende además:
 - el terminal; en el que el terminal comprende el procesador;
 - un aparato interceptor; en el que el aparato interceptor comprende el procesador;
 - una pantalla configurada para mostrar avisos a un usuario del terminal;
 - una entrada configurada para recibir respuestas a las solicitudes del usuario; en el que el procesador se configura además para generar la visualización de avisos y recibir las respuestas, y en el que las señales se generan, al menos en parte, en las respuestas; y
 - uno o más periféricos, en el que el procesador se configura además para recibir datos de uno o más periféricos y en el que las señales se generan, al menos en parte, sobre los datos recibidos de uno o más periféricos.
12. Un sistema de transmisión de datos como se reivindicó en la reivindicación 11, en el que al menos uno de los periféricos transmite datos en un formato desconocido para la aplicación EPOS.
13. Un sistema de transmisión de datos como se reivindicó en una cualquiera de las reivindicaciones de la 10 a la 12, en el que el periférico es un periférico virtual.

14. Un programa informático configurado para realizar el procedimiento de una cualquiera de las reivindicaciones de la 1 a la 9.

5

10

15

20

25

30

35

40

45

50

55

60

65

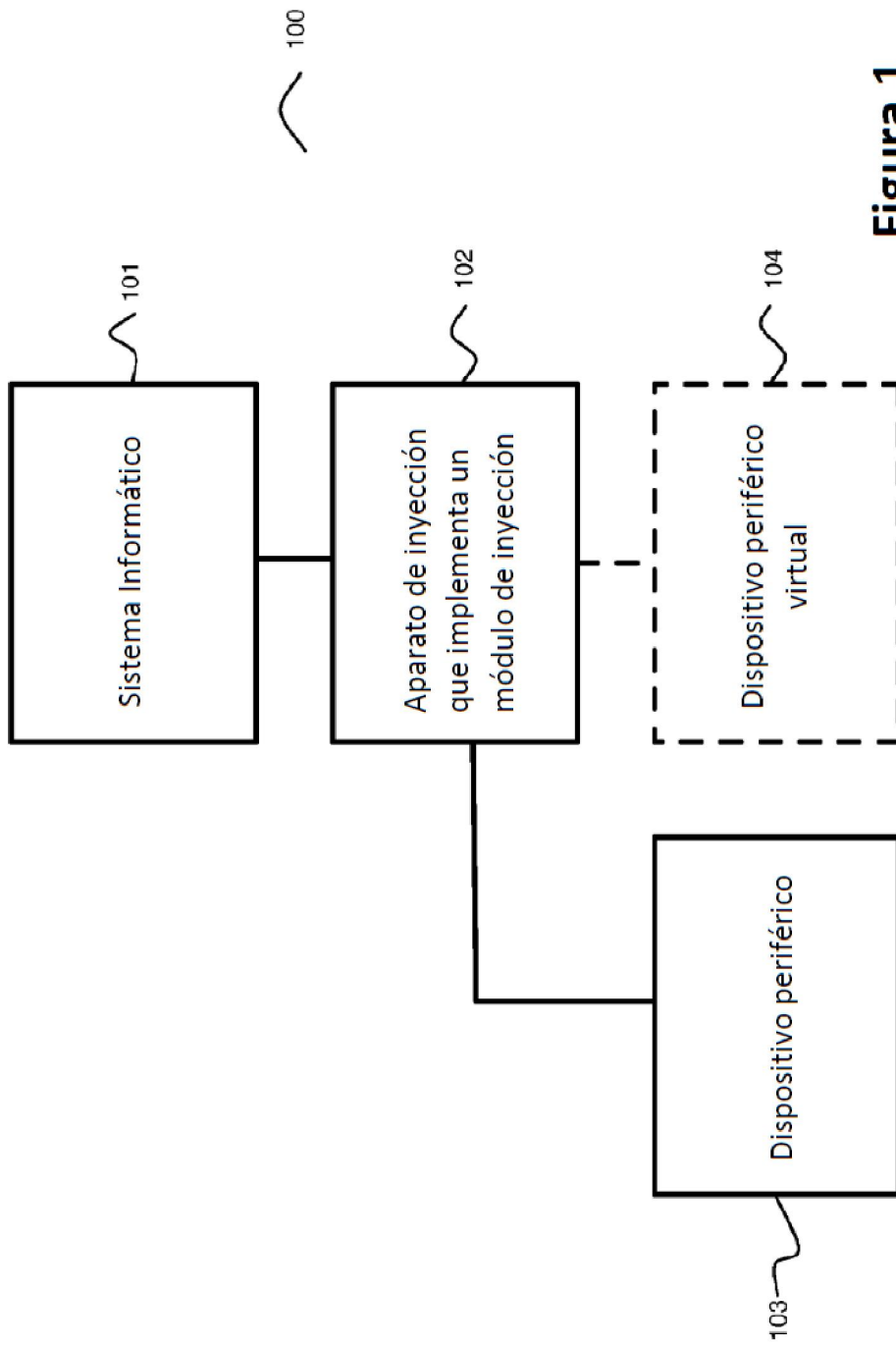


Figura 1

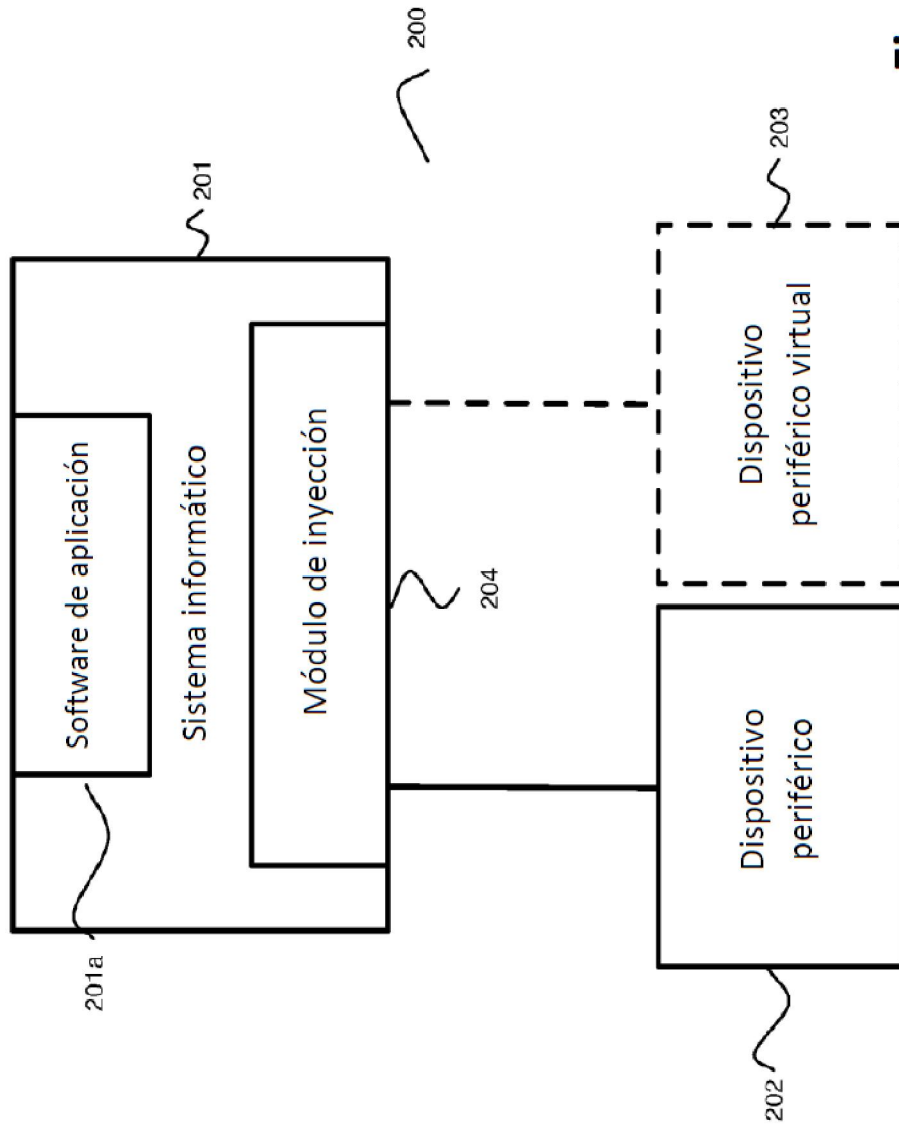


Figura 2

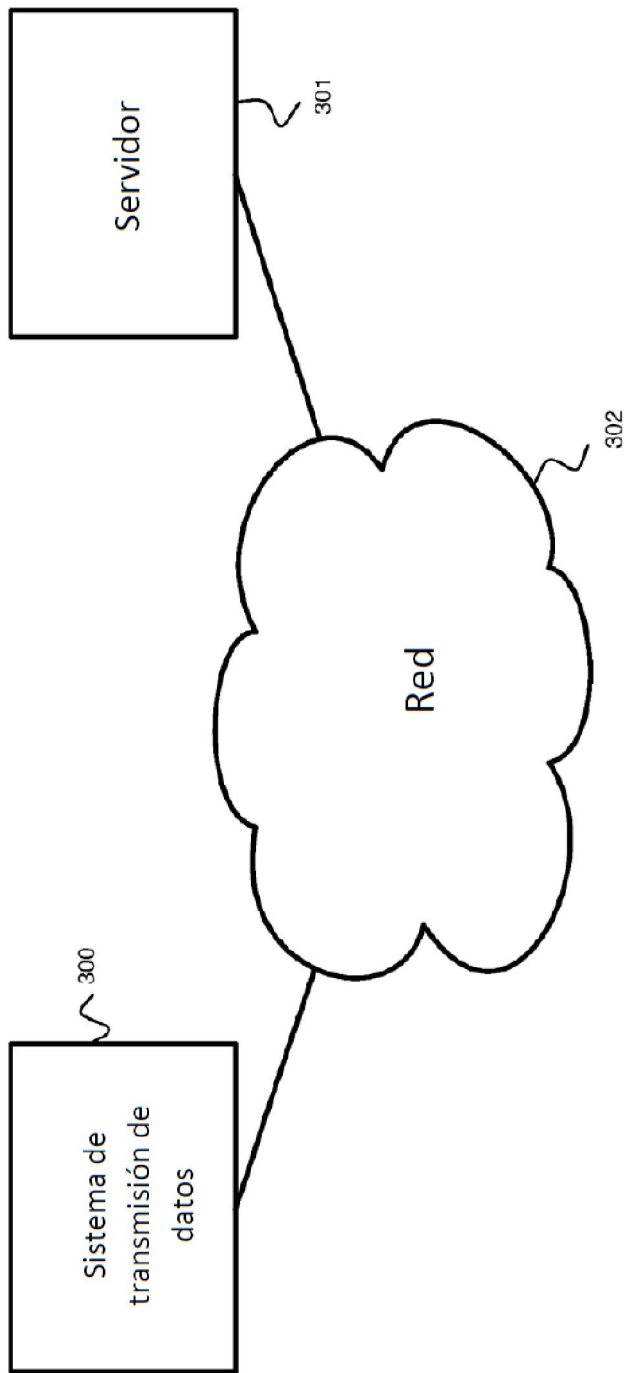


Figura 3

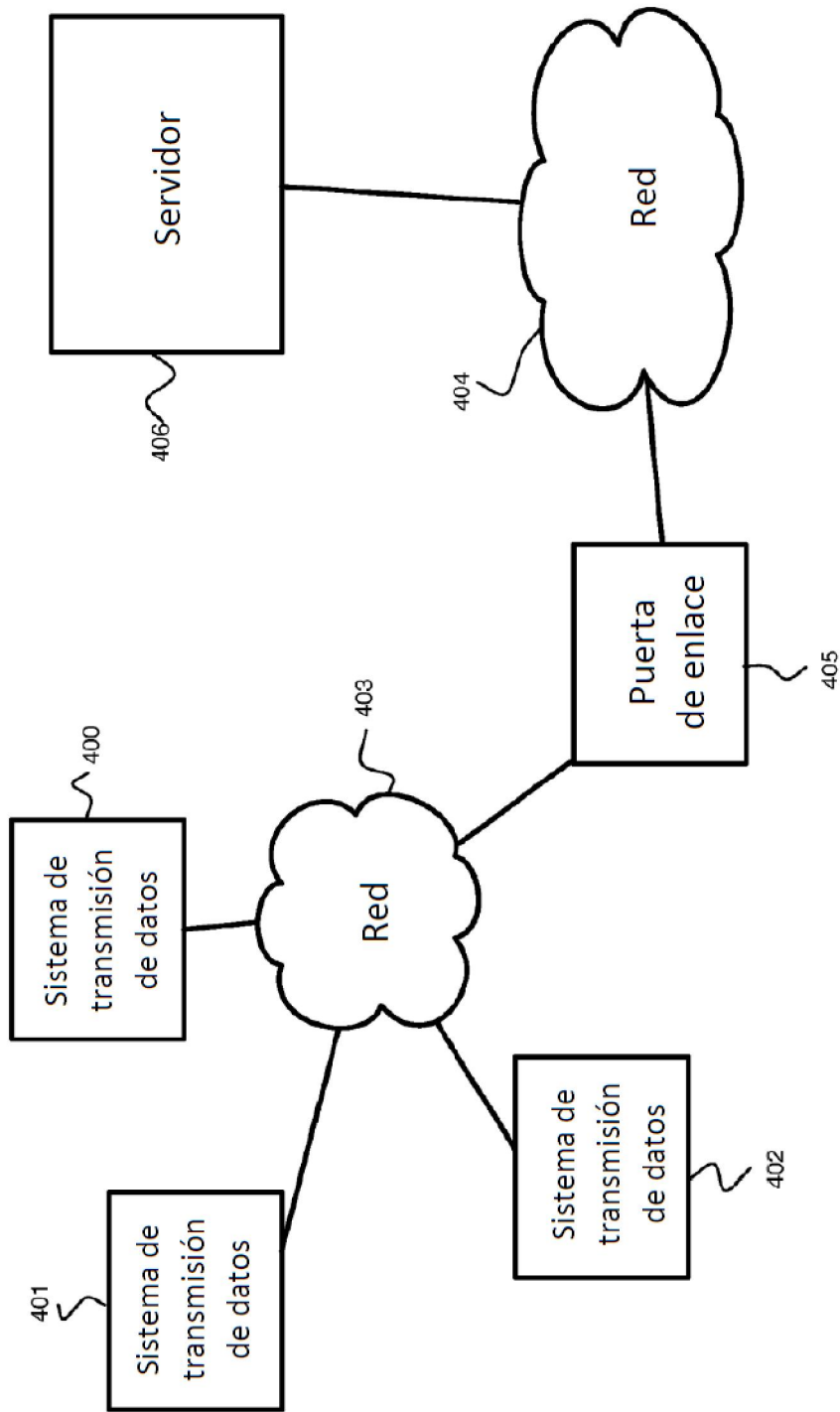


Figura 4

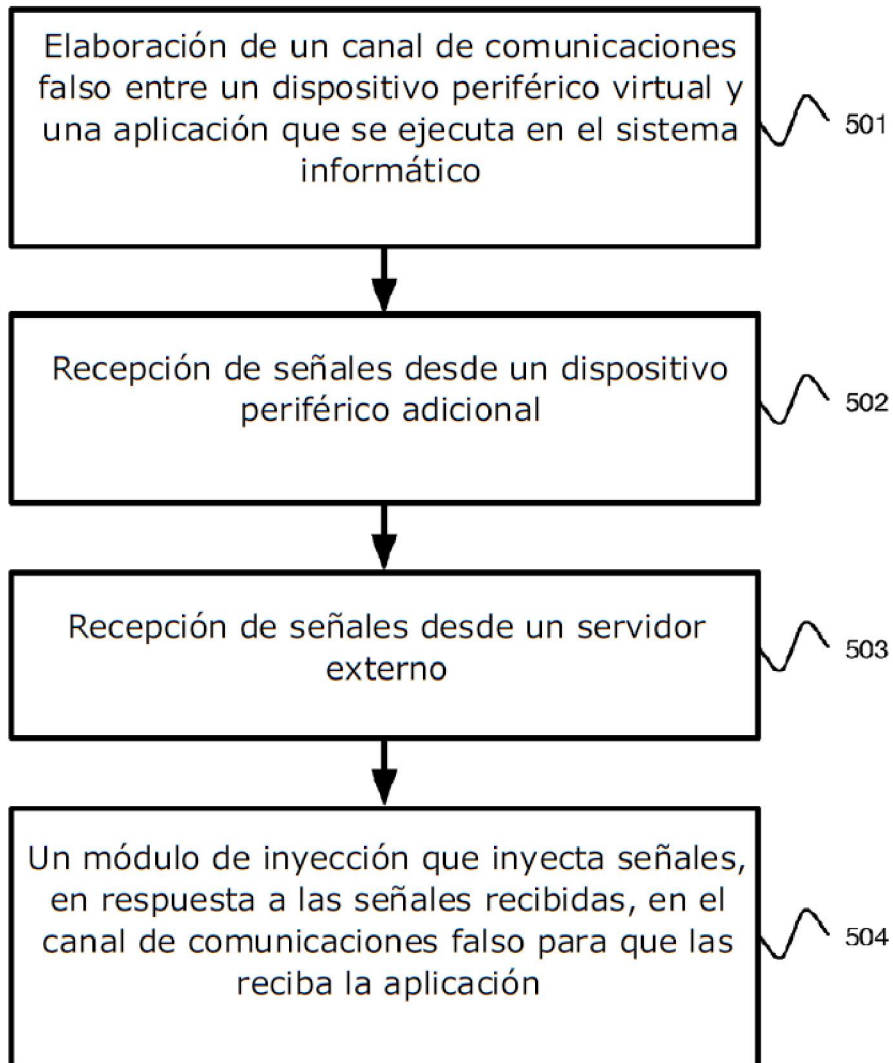


Figura 5

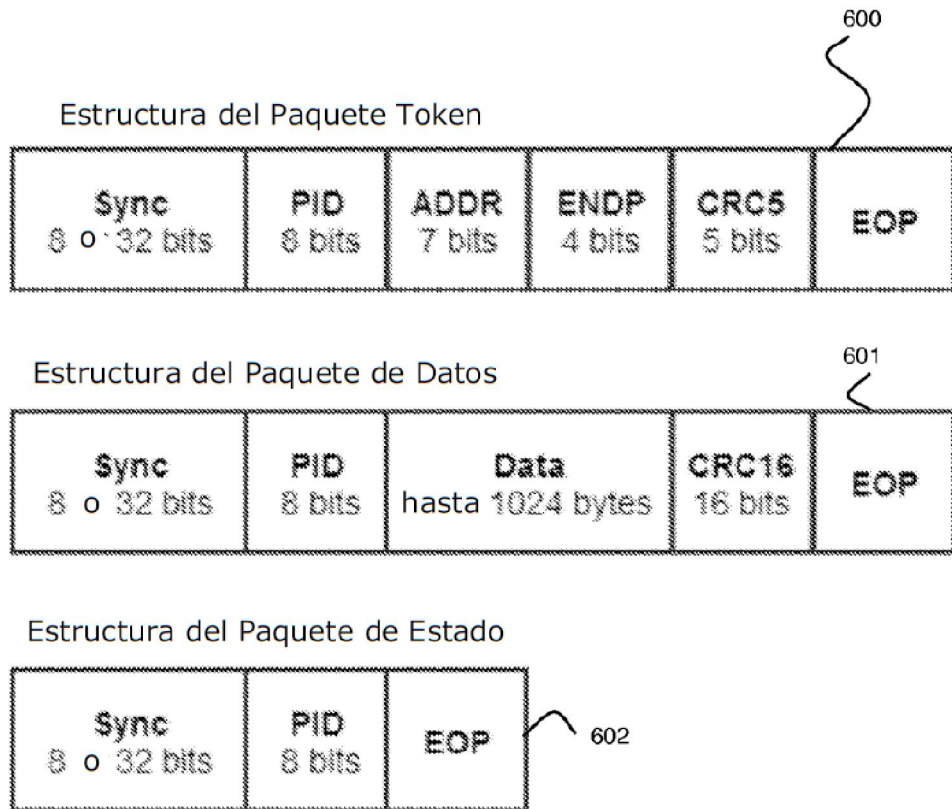


Figura 6

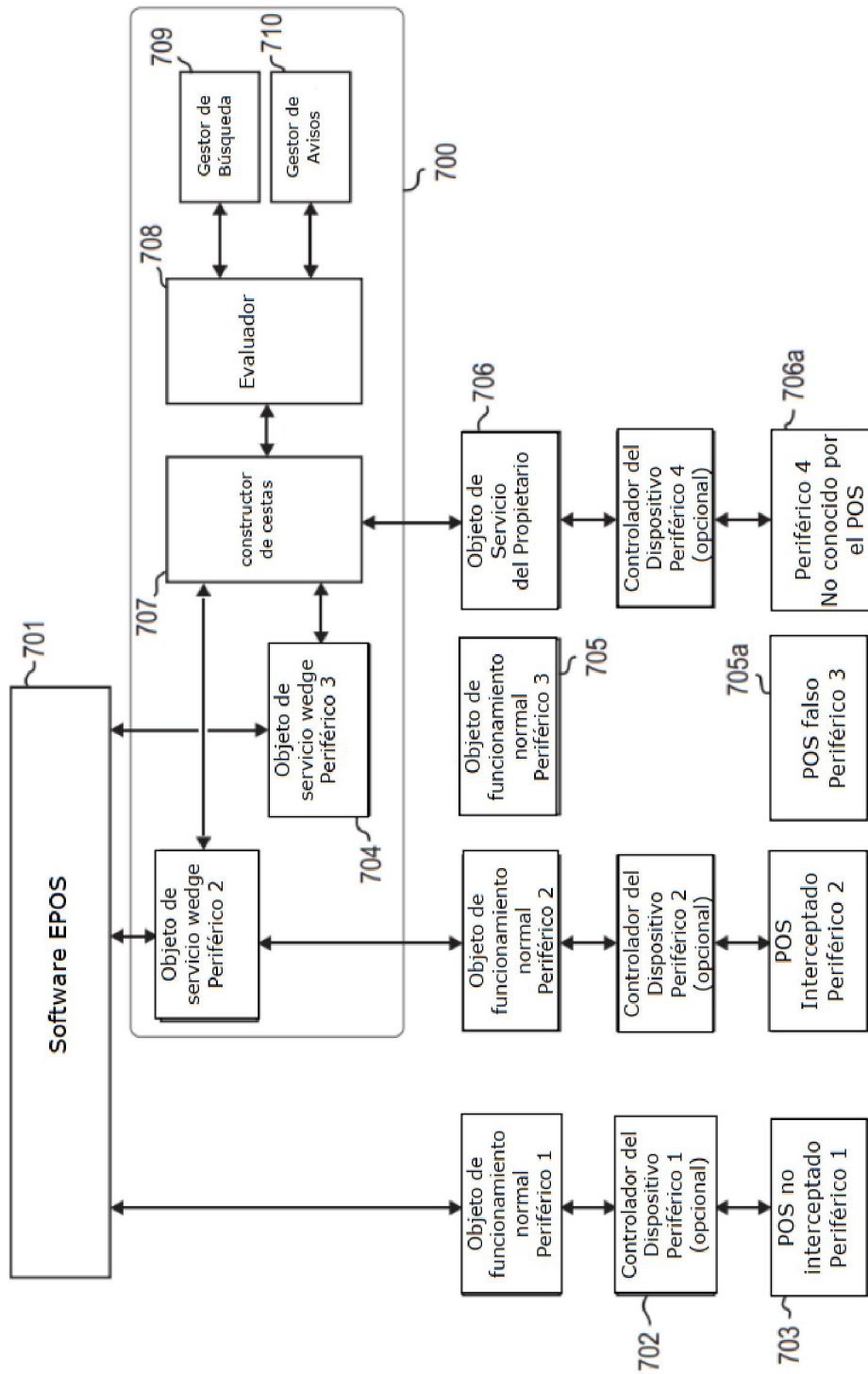


Figura 7

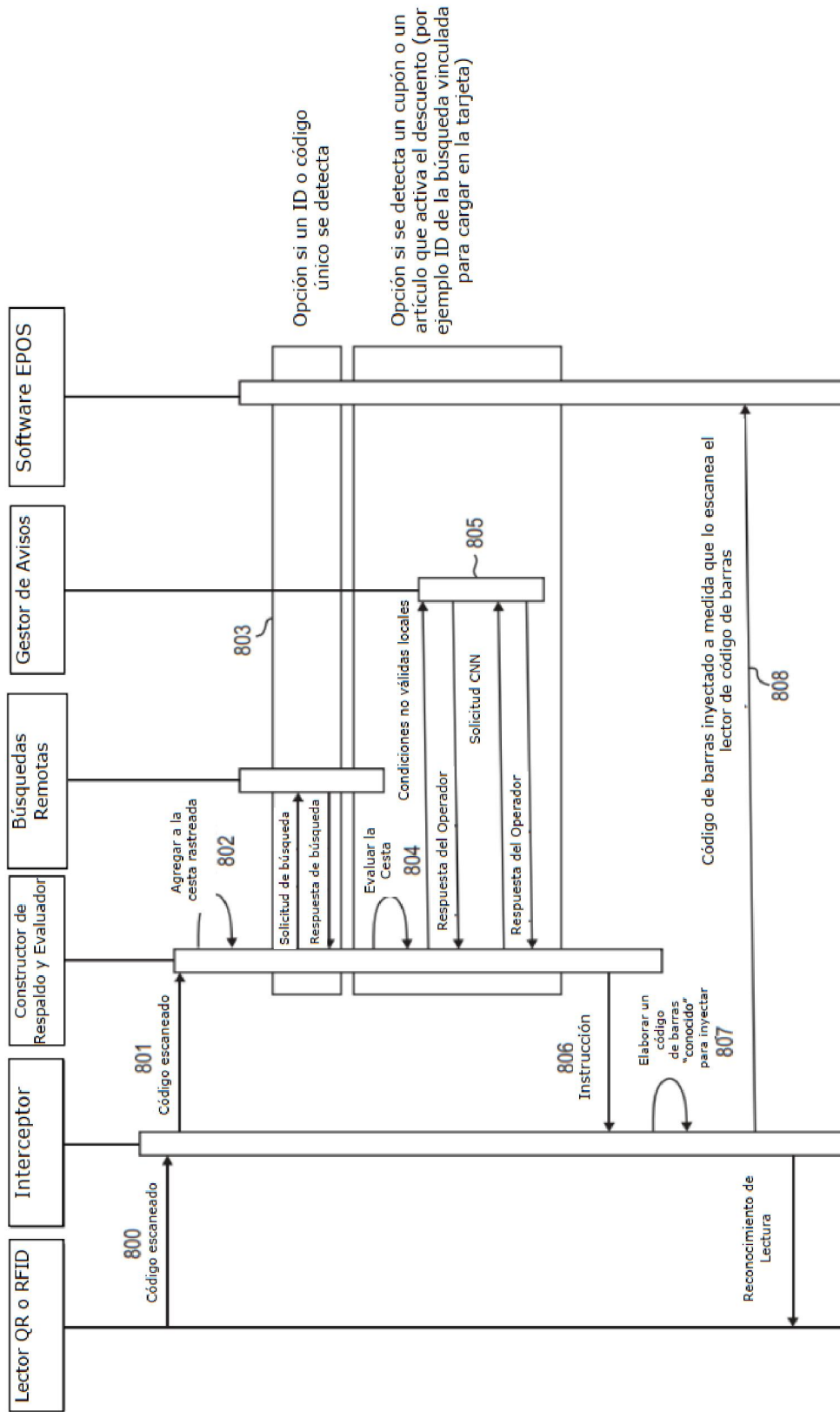


Figura 8

