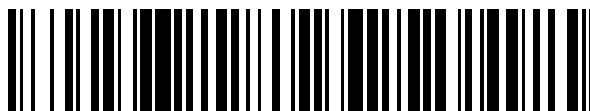


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 738 574**

51 Int. Cl.:

G06F 21/53 (2013.01)

G06F 17/22 (2006.01)

G06F 9/445 (2008.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **11.11.2009 PCT/US2009/064034**

87 Fecha y número de publicación internacional: **08.07.2010 WO10077443**

96 Fecha de presentación y número de la solicitud europea: **11.11.2009 E 09836588 (5)**

97 Fecha y número de publicación de la concesión europea: **19.06.2019 EP 2356582**

54 Título: **Aislamiento de aplicaciones hospedadas por código de complemento**

30 Prioridad:

09.12.2008 US 330528

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

23.01.2020

73 Titular/es:

**MICROSOFT TECHNOLOGY LICENSING, LLC
(100.0%)
One Microsoft Way
Redmond, WA 98052, US**

72 Inventor/es:

**GEORGE, SAM;
KAZA, AKHILESH y
HARSH, MICHAEL R.**

74 Agente/Representante:

CARPINTERO LÓPEZ, Mario

ES 2 738 574 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

Aislamiento de aplicaciones hospedadas por código de complemento

Antecedentes

- 5 Los navegadores actuales permiten que los complementos, que en general comprenden código de software hospedado que interactúa con el navegador/aplicación hospedados, proporcionen alguna funcionalidad deseada. Uno de tales complementos es Microsoft® Silverlight™, que proporciona una plataforma que permite el desarrollo y alojamiento de aplicaciones web ricas que a menudo incluyen animación, gráficos de vectores y/o reproducción de contenido de medios (por ejemplo, audio/vídeo). Un complemento puede hospedar una aplicación de terceros.
- 10 En general, esto se relaciona con una función significativa de desarrollo moderno de aplicaciones, en concreto componetización. Una idea general tras la componetización es dividir una aplicación grande en componentes menores que realizan porciones menores de la tarea mayor, son reutilizables y tienen una interfaz conocida. Con la componetización viene la capacidad de que las partes externas (por ejemplo, segundas o terceras) escriban componentes para una aplicación existente; por ejemplo, una encarnación moderna basada en la web de este concepto es una mezcla de web.
- 15 Sin embargo, cuando los componentes no de primeras partes se usan en una aplicación, tal como aplicaciones hospedadas por un complemento, el límite de confianza entre el componente y la aplicación se vuelve importante. Un componente no debería poder alterar la apariencia y funcionalidad del código de interfaz de usuario (UI), o supervisar datos de aplicación, y demás a menos que la aplicación de host se haya diseñado específicamente para esto y lo espere.
- 20 El documento US 2002/0007393 A1 describe un proxy mejorado para permitir hacer clic en una URL de anuncio en una página web dentro de un applet Java® que se ejecuta en una máquina de cliente. El proxy guarda información de cookies para el navegador, para soportar aplicaciones java que no permiten plantar cookies directamente. Cuando el usuario hace clic en la URL de anuncio, el proxy recupera la cookie relevante e información de enlace desde una base de datos del lado del host y usa la misma para ayudar al navegador a abrir la URL correcta asociada con el anuncio.
- 25 Los controles adicionales se proporcionan para el proxy incluyendo la capacidad de filtrar anuncios de manera apropiada para el applet, guardar anuncios desde servidores de terceros, supervisar el comportamiento del applet, guardar/pasar información de dirección IP del cliente y realizar tareas administrativas para el applet dentro del navegador para mejorar y facilitar suministro de anuncios entre anunciantes y sus audiencias objetivo. El proxy actúa así para extender el alcance y acceso de un applet convencional más allá de los recursos de un host de descarga asociado con tal applet. En otra realización, el applet controla un marco oculto dentro del navegador, marco oculto que, a diferencia del marco de applet, es capaz de descargar y plantar una cookie desde un servidor de anuncios dentro del navegador. Cuando el usuario hace clic en la URL de anuncio, el navegador usa la cookie desde el marco oculto y la pasa al Servidor de anuncios junto con la URL, por lo que el usuario experimenta una transferencia sin cortes y suave como ocurriría desde una etiqueta HTML convencional dentro de la página.
- 30 El documento US 5.974.549 desvela una técnica de crear una caja de arena segura dentro de la que una pluralidad de componentes de software descargados pueden ejecutarse de manera segura. Los componentes de software pueden ser de cualquier tipo, por ejemplo, Java, ActiveX, complemento Netscapre, etc. Un monitor de seguridad se implementa y se inyecta en el espacio de dirección de una aplicación supervisada arbitraria tal como navegador Web, por ejemplo, Internet Explorer, Netscape Navigator, etc. La aplicación supervisada se ejecuta entonces en modo seguro en el que cada componente de software descargado se ejecuta en una caja de arena segura. El monitor de seguridad detecta cuándo tal componente de software se descarga y es operativo para crear la caja de arena alrededor de este antes de poder ejecutarse. Si el componente de software intenta cometer una acción que infringe la seguridad, detiene la ejecución del componente de software y emite un aviso al usuario. El monitor de seguridad detecta violaciones de seguridad intentadas por el componente de software según una política de seguridad configurable por el usuario. Tal política puede incluir limitar el acceso a leer/escribir archivos, acceso a directorios, acceso a disco, creación y la lectura/escritura de conexiones de red, acceso a recursos del sistema y servicios y acceso a los espacios de dirección de otros procesos.
- 35
- 40
- 45

Sumario

La invención es lo que se especifica en las reivindicaciones independientes.

- 50 Las realizaciones preferentes se especifican en las reivindicaciones dependientes.

Este Sumario se proporciona para introducir una selección de conceptos representativos en forma simplificada que se describen adicionalmente a continuación en la Descripción Detallada. Este Sumario no pretende identificar aspectos clave o aspectos esenciales de la materia objeto reivindicada, ni pretende usarse de ninguna manera que pueda limitar el ámbito de la materia objeto reivindicada.

- 55 Brevemente, diversos aspectos de la materia objeto descrita en el presente documento se dirigen a una tecnología por la que una aplicación que se ejecuta en un navegador puede hospedar otra aplicación de manera que evita que la

aplicación hospedada acceda a datos de aplicación host o los datos de cualquier otra aplicación hospedada. En un aspecto, la aplicación host es una aplicación de complemento de navegador (por ejemplo, Microsoft® Silverlight™) que instancia otro complemento que se oculta al navegador, y evita que la aplicación hospedada acceda a datos instanciando al otro complemento con acceso deshabilitado a datos de aplicación hospedada/navegador (por ejemplo, Elementos) u otros datos de aplicación.

En un aspecto, un elemento de etiqueta identifica la aplicación hospedada dentro del código (por ejemplo, XAML) asociado con la aplicación host. Un elemento host se añade al árbol de elemento de la aplicación host para representar la aplicación hospedada. El elemento de etiqueta puede especificar un área de renderización para la aplicación hospedada. El contenido desde la aplicación hospedada se renderiza junto con (por ejemplo, compuesto con) contenido desde la aplicación host.

En un aspecto, la aplicación hospedada puede ser un anuncio. La aplicación host puede proporcionar palabras clave a la aplicación hospedada, tal como correspondiente a un contenido de página, por lo que la aplicación hospedada puede seleccionar un anuncio relevante. La aplicación hospedada puede tener permiso para realizar algunas operaciones privilegiadas, como permitir que la aplicación hospedada abra una ventana del navegador para mostrar contenido de sitio web (por ejemplo, hacer aparecer un sitio web cuando el usuario hace clic en un anuncio en el área de renderización de la aplicación hospedada).

Otras ventajas pueden ser aparentes desde la siguiente descripción detallada cuando se toma junto con los dibujos.

Breve descripción de los dibujos

La presente invención se ilustra a modo de ejemplo y no se limita en las figuras adjuntas en las que los números de referencia similares indican elementos similares y en las que:

La figura 1 es un diagrama de bloques que muestra componentes de ejemplo para aislar una aplicación hospedada por una aplicación host (complemento de navegador).

La figura 2 es un diagrama de flujo que muestra etapas de ejemplo tomadas para aislar una aplicación host.

La figura 3 es un diagrama de bloques que muestra componentes de ejemplo para comunicación adicional entre una aplicación host y una aplicación de complemento aislada.

La figura 4 muestra un ejemplo ilustrativo de un entorno informático en el que pueden incorporarse diversos aspectos de la presente invención.

Descripción detallada

Diversos aspectos de la tecnología descrita en el presente documento se dirigen en general hacia un modelo de aislamiento en el que una primera aplicación (por ejemplo, complemento de navegador o similar) hospeda una segunda aplicación, (normalmente de terceros, no de confianza) de manera aislada. En general, esto se logra instanciando una instancia de complemento separada en memoria, que se desconoce por el navegador web, para aislar la segunda aplicación. Al mismo tiempo, la segunda aplicación (por ejemplo, anuncio) aparece para el usuario de ordenador ejecutándose como parte de la aplicación hospedada, debido a que, por ejemplo, su salida visible se renderiza junto con la salida visible de la aplicación de host. Como se entenderá, sin embargo, la aplicación hospedada no tiene acceso a información relacionada con la aplicación de host o el navegador web, a menos que la aplicación de host desee proporcionar alguna información.

Aunque Microsoft® Silverlight™ (un complemento multiplataforma, multinavegador) se usa como ejemplo de una aplicación de host, debería entenderse que cualquier ejemplo descrito en el presente documento es un ejemplo no limitante. En consecuencia, la presente invención no se limita a ninguna realización, aspectos, conceptos, estructuras, funcionalidades o ejemplos particulares descritos en el presente documento. Más bien, cualquiera de las realizaciones, aspectos, conceptos, estructuras, funcionalidades o ejemplos descritos en el presente documento no son limitantes, y la presente invención puede usarse de diversas formas que proporcionan beneficios y ventajas en computación y aplicaciones de host en general.

La figura 1 muestra diversos aspectos relacionados con aislamiento de aplicaciones como se describe en el presente documento. En general, un navegador 102 se carga con una aplicación 104 host, tal como el complemento Microsoft® Silverlight™. En una realización, la aplicación 104 host corresponde a un código 106 de aplicación de interfaz de usuario, como escrito en XAML (lenguaje extensible de formato para aplicaciones), que se analiza por un analizador 108 sintáctico en diversos elementos correspondientes a un árbol 110. Como es sabido, estos elementos se procesan para renderizarse en una salida visible en la página del navegador.

En una implementación, una aplicación 111 como un anuncio de terceros se hospeda por la aplicación 104 host como parte de un complemento 112 oculto (separado y no descubrible para el navegador) de forma que aísla de forma programática el complemento 112 oculto desde el navegador 102 y desde los datos de la aplicación de host, pero permite que la aplicación 111 hospedada participe en la renderización de la aplicación host, entrada y disposición. La renderización, entrada y disposición se gestionan por el complemento 112 oculto, por lo que, por ejemplo, la salida visible se limita a un área de interfaz de usuario que se reserva para la salida de la aplicación hospedada. Debe apreciarse que el complemento 112 oculto puede ser otro complemento Silverlight™, es decir, otra instancia que se

crea de forma privada (desconocida para el navegador) por un complemento Silverlight™ host. Además, debe apreciarse que pueden usarse otros tipos de complementos, siempre y cuando puedan instanciarse o configurarse de otra manera de forma que se evita que tengan cualquier comunicación directa con el navegador HTML, por ejemplo, el complemento se instancia de forma que no tiene acceso al Modelo de Objeto de Documento (DOM) del navegador.

- 5 Para esto, en una implementación de ejemplo, se proporciona un elemento de etiqueta (por ejemplo, en el código 106 de aplicación XAML) que identifica una fuente de la aplicación hospedada y representa un límite de aislamiento en el que instanciar el complemento 112 oculto para cargar la aplicación 111 hospedada (aislada):

```

10 <Canvas> <!--part of the host application -->
    <XapHost Source="http://www.advertisements.com/ad.xap" Height="300"
    Width="350" /> <!--the isolation boundary -->
    </Canvas>

```

El elemento de etiqueta se representa en la figura 1 por el elemento 114 XapHost. Entre otras operaciones, el elemento 114 XapHost es responsable de instanciar e iniciar el complemento 112 oculto en memoria, en una ubicación que no puede descubrirse desde el navegador 102; esto puede lograrse mediante API privadas o similares. La aplicación 111 hospedada no participa en el árbol 110 de la aplicación 104 host, excepto mediante el elemento 114 XapHost que actúa como un proxy para aislar el complemento 112 oculto. Por ejemplo, la aplicación 111 hospedada no puede atravesar el árbol aparte de hacia su propio nodo más superior, que corresponde al nodo raíz de su aplicación. Debe apreciarse que no tiene acceso al elemento XapHost, que reside en el árbol de aplicaciones de host. En su lugar, la aplicación 111 hospedada se limita a solo proporcionar la salida 116 para renderizar mediante el elemento 114 XapHost, (aunque el elemento 114 XapHost puede proporcionar datos adicionales como parámetros de inicialización y/o permitir algunos privilegios en la aplicación 111 hospedada como se describe a continuación en referencia a la figura 3). Estos y otros datos comunicados desde el elemento 114 XapHost al complemento 112 oculto/aplicación 111 hospedada se representan en la figura 1 como entrada 117.

Como se representa en general por las etapas de ejemplo de la figura 2, cuando el código 106 de aplicación XAML se analiza o cuando el elemento 114 XapHost se añade de forma programática al árbol 110 (etapa 202), en las etapas 204 y 206 el elemento 114 XapHost inicializa el complemento 112 oculto y carga la aplicación 111 hospedada correspondiente (por ejemplo, XAP) que se especifica por el atributo "Fuente", (que en el lenguaje XAML de ejemplo anterior se identifica como "http://www.advertisements.com/ad.xap"). La aplicación 111 hospedada no tiene acceso al modelo de objeto de dominio HTML del navegador (por ejemplo, la propiedad EnableHtmlAccess se ajusta a falsa) y se carga en su propio dominio de aplicación, por ejemplo, (Common Language Runtime, o CLR AppDomain), con su propio acceso a variables estáticas globales y demás; debe apreciarse que Silverlight™ contiene su propio mini CLR 118. Sin acceso al modelo de objeto de dominio (DomBridge), la aplicación 111 hospedada no tiene acceso a ningún otro contenido en la página, incluyendo la propia página u otros complementos.

Como se representa además en la figura 2 por las etapas 208 y 210, además de instanciar e inicializar el complemento 112 oculto y cargar la aplicación 111, el elemento 114 XapHost dirige notificaciones de entrada y disposición desde la aplicación 104 de host a la aplicación 111 hospedada. Adicionalmente, el elemento 114 XapHost procesa la salida de la aplicación 111 hospedada por lo que la salida puede componerse por el complemento de la aplicación de host; (debe apreciarse que cualquier otra salida como audio y/o salida táctil puede mezclarse de forma similar). La etapa 212 representa el derribo de la aplicación hospedada en un momento apropiado, por ejemplo, tras el apagado o cuando el conteo de referencia correspondiente en el elemento 114 XapHost va a cero (por ejemplo, cuando esa parte de la interfaz de usuario ya no va a renderizarse).

En un aspecto, representado en general en la figura 3, el elemento 114 XapHost puede incluir una o más interfaces (por ejemplo, API 330, entre las otras API privadas descritas antes) que permiten que la aplicación 104 host se comunique con la aplicación 111 hospedada, para pasar los parámetros 332 de inicialización. Por ejemplo, en un escenario de publicidad, los parámetros de inicialización pueden incluir una colección de palabras clave que están presentes en la página de host, para la que una aplicación de publicidad puede servir anuncios relevantes correspondientes a una o más palabras clave.

En otro aspecto también generalmente representado en la figura 3, mediante las API 330 el elemento 114 XapHost puede permitir que la aplicación hospedada realice ciertas operaciones privilegiadas, (o realizar las operaciones en su nombre tras una llamada 334 apropiada). Un ejemplo es abrir una nueva ventana del navegador en respuesta a una acción iniciada por el usuario, como hacer aparecer un sitio web correspondiente a un anuncio clicado.

Otra ventaja es la capacidad de controlar el consumo de potencia. Por ejemplo, los anuncios pueden ser una gran fuente de consumo de potencia en el uso de ordenadores modernos. Con el modelo de alojamiento de complementos (por ejemplo, XapHost), un modo de potencia reducida puede conservar potencia, tal como desactivando animaciones y otras operaciones de consumo de potencia.

ENTORNO OPERATIVO EJEMPLAR

La figura 4 ilustra un ejemplo de un entorno 400 de computación y conexión de redes adecuado en el que pueden implementarse los ejemplos de las figuras 1-3. El entorno 400 de sistema de computación es solo un ejemplo de un

entorno de computación adecuado y no pretende sugerir ninguna limitación en cuanto al ámbito de uso o funcionalidad de la invención. Tampoco el entorno 400 de computación debería interpretarse como con alguna dependencia o requisito referente a uno cualquiera o combinación de componentes ilustrados en el entorno 400 operativo ejemplar.

5 La invención es operativa con otros diversos entornos o configuraciones de sistema de computación de fin especial o fin general. Los ejemplos de sistemas de computación bien conocidos, entornos y/o configuraciones que pueden ser adecuados para su uso con la invención incluyen, pero no se limitan a: ordenadores personales, ordenadores de servidor, dispositivos portátiles o de mano, dispositivos de tableta, sistemas de multiprocesador, sistemas basados en multiprocesador, descodificadores, aparatos electrónicos de consumidor programables, PC de red, miniordenadores, ordenadores de marco principal, entornos de computación distribuida que incluyen cualquiera de los sistemas o dispositivos anteriores y similares.

10 La invención puede describirse en el contexto general de instrucciones ejecutables a ordenador, tal como módulos de programa, que se ejecutan por un ordenador. Generalmente, los módulos de programa incluyen rutinas, programas, objetos, componentes, estructuras de datos y demás, que realizan tareas particulares o implementan tipos de datos abstractos particulares. La invención también puede practicarse en entornos de computación distribuida donde las tareas se realizan por dispositivos de procesamiento remoto que se vinculan mediante una red de comunicaciones. En un entorno de computación distribuida, los módulos de programa pueden ubicarse en medios de almacenamiento informático remoto y/o local incluyendo dispositivos de almacenamiento de memoria.

15 Con referencia a la figura 4, un sistema ejemplar para implementar diversos aspectos de la invención puede incluir un dispositivo de computación de fin general en la forma de un ordenador 410. Los componentes del ordenador 410 pueden incluir, pero no se limitan a, una unidad 420 de procesamiento, una memoria 430 del sistema y un bus 421 del sistema que acopla diversos componentes del sistema incluyendo la memoria del sistema a la unidad 420 de procesamiento. El bus 421 del sistema puede ser de varios tipos de estructuras de bus incluyendo un bus de memoria o controlador de memoria, un bus periférico y un bus local usando cualquiera de una variedad de arquitecturas de bus. A modo de ejemplo, y no de limitación, tales arquitecturas incluyen bus de Arquitectura Estándar de la Industria (ISA), bus de Arquitectura de Microcanal (MCA), bus ISA Mejorado (EISA), bus local de Asociación de Estándares de Electrónica de Vídeo (VESA) y bus de Interconexión de Componente Periférico (PCI) también conocido como bus Mezzanine.

20 El ordenador 410 incluye normalmente una variedad de medios legibles a ordenador. Los medios legibles a ordenador pueden ser cualquier medio disponible al que se pueda acceder por el ordenador 410 e incluye tanto medios volátiles como no volátiles y medios removibles y no removibles. A modo de ejemplo, y no de limitación, los medios legibles a ordenador pueden comprender medios de almacenamiento informático y medios de comunicación. Los medios de almacenamiento informático incluyen medios volátiles y no volátiles, removibles y no removibles implementados en cualquier procedimiento o tecnología para almacenar información como instrucciones legibles a ordenador, estructuras de datos, módulos de programa u otros datos. Los medios de almacenamiento informático incluyen, pero no se limitan a, RAM, ROM, EEPROM, memoria flash u otra tecnología de memoria, CD ROM, discos versátiles digitales (DVD) u otro almacenamiento de disco óptico, casetes magnéticos, cinta magnética, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético o cualquier otro medio que pueda usarse para almacenar la información deseada y al que se pueda acceder por el ordenador 410. Los medios de comunicación incorporan normalmente instrucciones legibles a ordenador, estructuras de datos, módulos de programa u otros datos en una señal de datos modulada como una onda portadora u otro mecanismo de transporte e incluye cualquier medio de suministro de información. El término "señal de datos modulada" significa una señal que tiene una o más de sus características establecidas o cambiadas de manera que se codifica información en la señal. A modo de ejemplo, y no de limitación, los medios de comunicación incluyen medios alámbricos como una red alámbrica o conexión cableada directa, y medios inalámbricos como medios acústicos, RF, infrarrojos y otros medios inalámbricos. Las combinaciones de cualquiera de los anteriores también pueden incluirse dentro del ámbito de los medios legibles a ordenador.

25 La memoria 430 del sistema incluye medios de almacenamiento informático en la forma de memoria volátil y/o no volátil como memoria 431 de solo lectura (ROM) y memoria 432 de acceso aleatorio (RAM). Un sistema 433 básico de entrada/salida (BIOS), que contiene las rutinas básicas que ayudan a transferir información entre elementos dentro del ordenador 410, como durante el inicio, se almacena normalmente en la ROM 431. La RAM 432 contiene normalmente datos y/o módulos de programa que son accesibles inmediatamente para y/o se operan en la actualidad por la unidad 420 de procesamiento. A modo de ejemplo, y no de limitación, la figura 4 ilustra el sistema 434 operativo, programas 435 de aplicación, otros módulos 436 de programa y datos 437 de programa.

30 El ordenador 410 puede además incluir otros medios removibles/no removibles, volátiles/no volátiles de almacenamiento informático. A modo de ejemplo solo, La figura 4 ilustra una unidad 441 de disco duro que lee desde o escribe en los medios no removibles, no volátiles magnéticos, una unidad 451 de disco magnético que lee desde o escribe en un disco 452 magnético removible no volátil y una unidad 455 de disco óptico que lee desde o escribe en un disco 456 óptico removible no volátil como un CD ROM u otro medio óptico. Otros medios removibles/no removibles, volátiles/no volátiles de almacenamiento informático que se pueden usar en el entorno operativo ejemplar incluyen, pero no se limitan a, casetes de cinta magnética, tarjetas de memoria flash, discos versátiles digitales, cinta de vídeo digital, RAM de estado sólido, ROM de estado sólido y similares. La unidad 441 de disco duro se conecta normalmente al bus 421 del sistema mediante una interfaz de memoria no removible como la interfaz 440, y la unidad 451 de disco

magnético y la unidad 455 de disco óptico se conectan normalmente al bus 421 del sistema por una interfaz de memoria removible, como la interfaz 450.

Las unidades y sus medios de almacenamiento informático asociados, antes descritos e ilustrados en la figura 4, proporcionan almacenamiento de instrucciones legibles a ordenador, estructuras de datos, módulos de programa y otros datos para el ordenador 410. En la figura 4, por ejemplo, la unidad 441 de disco duro se ilustra como almacenando el sistema 444 operativo, programas 445 de aplicación, otros módulos 446 de programa y datos 447 de programa. Debe apreciarse que estos componentes pueden ser iguales o diferentes del sistema 434 operativo, programas 435 de aplicación, otros módulos 436 de programa y datos 437 de programa. El sistema 444 operativo, programas 445 de aplicación, otros módulos 446 de programa y datos 447 de programa reciben números diferentes en el presente documento para ilustrar que, como mínimo, son copias diferentes. Un usuario puede introducir comandos e información en el ordenador 410 mediante dispositivos de entrada como una tableta o un digitalizador electrónico, 464, un micrófono 463, un teclado 462 y un dispositivo 461 de apuntado, normalmente llamado ratón, bola rastreadora o panel táctil. Otros dispositivos de entrada no mostrados en la figura 4 pueden incluir una palanca de mando, mando de juegos, antena parabólica, escáner o similar. Estos y otros dispositivos de entrada se conectan a menudo a la unidad 420 de procesamiento mediante una interfaz 460 de entrada de usuario que se acopla al bus del sistema, pero puede conectarse por otras estructuras de interfaz y bus, como un puerto paralelo, puerto de juegos o un bus en serie universal (USB). Un monitor 491 u otro tipo de dispositivo de representación también se conecta al bus 421 del sistema mediante una interfaz, como una interfaz 490 de vídeo. El monitor 491 también puede integrarse con un panel de pantalla táctil o similar. Debe apreciarse que el monitor y/o panel de pantalla táctil pueden acoplarse físicamente a una carcasa en la que se incorpora el dispositivo 410 de computación, como en un ordenador personal de tipo tableta. Además, los ordenadores como el dispositivo 410 de computación pueden incluir además otros dispositivos de salida periféricos como altavoces 495 e impresora 496, que pueden conectarse mediante una interfaz 494 periférica de salida o similar.

El ordenador 410 puede operar en un entorno en red usando conexiones lógicas a uno o más ordenadores remotos, como un ordenador 480 remoto. El ordenador 480 remoto puede ser un ordenador personal, un servidor, un enrutador, un PC de red, un dispositivo de pares u otro nodo de red común, y normalmente incluye muchos o todos los elementos descritos antes en relación con el ordenador 410, aunque solo se ha ilustrado un dispositivo 481 de almacenamiento de memoria en la figura 4. Las conexiones lógicas representadas en la figura 4 incluyen una o más redes 471 de área local (LAN) y una o más redes 473 de área amplia (WAN), pero pueden incluir también otras redes. Tales entornos en red son comunes en oficinas, redes informáticas en empresas, intranets e Internet.

Cuando se usa en un entorno en red LAN, el ordenador 410 se conecta a la LAN 471 mediante una interfaz de red o adaptador 470. Cuando se usa en un entorno en red WAN, el ordenador 410 incluye normalmente un módem 472 u otro medio para establecer comunicaciones sobre la WAN 473, tal como Internet. El módem 472, que puede ser interno o externo, puede conectarse al bus 421 del sistema mediante la interfaz 460 de entrada de usuario u otro mecanismo apropiado. Un componente 474 en red inalámbrico que comprende una interfaz y una antena puede acoplarse mediante un dispositivo adecuado como un punto de acceso u ordenador de pares a una WAN o LAN. En un entorno en red, los módulos de programa representados en relación con el ordenador 410 o porciones del mismo, pueden almacenarse en el dispositivo de almacenamiento de memoria remoto. A modo de ejemplo, y no de limitación, La figura 4 ilustra programas 485 de aplicación remotos que residen en un dispositivo 481 de memoria. Puede apreciarse que las conexiones de red mostradas son ejemplares y pueden usarse otros medios de establecer un enlace de comunicaciones entre los ordenadores.

Un subsistema 499 auxiliar (por ejemplo, para representación auxiliar de contenido) puede conectarse mediante la interfaz 460 de usuario para permitir que los datos como contenido de programa, estado del sistema y notificaciones de eventos se proporcionen al usuario, incluso si las porciones principales del sistema informático están en un estado de baja potencia. El subsistema 499 auxiliar puede conectarse al módem 472 y/o interfaz 470 de red para permitir la comunicación entre estos sistemas mientras la unidad 420 de procesamiento principal está en un estado de baja potencia.

CONCLUSIÓN

Aunque la invención es susceptible de tener diversas modificaciones y construcciones alternativas, ciertas realizaciones ilustradas de la misma se muestran en los dibujos y se han descrito antes en detalle. Debería entenderse, sin embargo, que no existe intención de limitar la invención a las formas específicas desveladas, sino al contrario, la intención es cubrir todas las modificaciones, construcciones alternativas y equivalentes que caen dentro del ámbito de la invención.

REIVINDICACIONES

1. Un procedimiento, en un entorno de computación, para aislar una aplicación (111) hospedada por una aplicación (104) host, comprendiendo el procedimiento:

5 hospedar (202) la aplicación (111) hospedada dentro de la aplicación host correspondiente a un complemento (104) que se ejecuta en un navegador (102), incluyendo evitar que la aplicación hospedada acceda a datos del navegador o datos de cualquier otra aplicación hospedada; comprendiendo el procedimiento, además:

10 proporcionar un elemento (114) de etiqueta para identificar la aplicación hospedada dentro de código (106) asociado con la aplicación host, en el que el elemento de etiqueta identifica una fuente de la aplicación hospedada y un área de renderización para la aplicación hospedada; detectar (202) el elemento de etiqueta, y en respuesta, instanciar (204) un complemento oculto y cargar (206) la aplicación hospedada en el complemento oculto,

15 en el que evitar que la aplicación hospedada acceda a datos comprende instanciar el complemento oculto con acceso deshabilitado a datos del navegador o datos de cualquier otra aplicación que se ejecuta en el navegador.

2. El procedimiento de la reivindicación 1 en el que detectar (202) el elemento de etiqueta comprende:

20 detectar cuándo el elemento de etiqueta correspondiente a la aplicación hospedada se añade de forma programática a un árbol de elementos de la aplicación host, en el que el árbol se obtiene desde el análisis del código (106) asociado con la aplicación host en elementos, o detectar el elemento de etiqueta al analizar el código (106) asociado con la aplicación host.

3. El procedimiento de la reivindicación 1 que comprende adicionalmente, componer la salida desde la aplicación host con la salida desde la aplicación hospedada.

25 4. El procedimiento de la reivindicación 1 que comprende adicionalmente comunicar parámetros desde la aplicación host a la aplicación hospedada, o controlar acciones de la aplicación hospedada para intentar controlar el consumo de potencia, o tanto comunicar parámetros desde la aplicación host a la aplicación hospedada como controlar acciones de la aplicación hospedada para intentar controlar el consumo de potencia.

5. El procedimiento de la reivindicación 4 en el que comunicar los parámetros comprende proporcionar información por la que la aplicación hospedada puede seleccionar al menos un anuncio relevante.

30 6. El procedimiento de la reivindicación 1 que comprende adicionalmente permitir que la aplicación hospedada realice al menos una operación privilegiada.

7. El procedimiento de la reivindicación 1 en el que el código (106) asociado con la aplicación host es un código de aplicación de interfaz de usuario o en el que el código (106) asociado con la aplicación host es código XAML.

8. El procedimiento de la reivindicación 1 en el que unas API privadas se asocian con la aplicación host, en el que el complemento oculto se instancia por la aplicación host y se comunica con la aplicación host mediante las API privadas.

35 9. El procedimiento de la reivindicación 1 en el que la aplicación hospedada proporciona elementos de interfaz de usuario, comprendiendo el procedimiento además renderizar (210) salida visible procesando elementos de interfaz de usuario de la aplicación host y elementos de interfaz de usuario de la aplicación hospedada.

40 10. El procedimiento de la reivindicación 1 en el que la aplicación hospedada corresponde a un anuncio, comprendiendo además el procedimiento proporcionar una o más palabras clave correspondientes a contenido de página a la aplicación hospedada.

11. El procedimiento de la reivindicación 1 que comprende adicionalmente permitir que la aplicación hospedada abra una ventana del navegador para mostrar contenido de sitio web.

12. Un sistema, en un entorno de computación, configurado para aislar una aplicación (111) hospedada por una aplicación (104) host, estando el sistema adaptado para:

45 hospedar la aplicación (111) hospedada dentro de la aplicación host correspondiente a un complemento (104) que se ejecuta en un navegador (102), incluyendo evitar que la aplicación hospedada acceda a datos del navegador o datos de cualquier otra aplicación hospedada; estando el sistema adaptado además para:

50 proporcionar un elemento (114) de etiqueta para identificar la aplicación hospedada dentro de código (106) asociado con la aplicación host, en el que el elemento de etiqueta se configura para identificar una fuente de la aplicación hospedada y un área de renderización para la aplicación hospedada;

detectar el elemento de etiqueta, y
en respuesta, instanciar un complemento (112) oculto y cargar la aplicación hospedada en el complemento oculto,

5 en el que evitar que la aplicación hospedada acceda a datos comprende instanciar el complemento oculto con acceso deshabilitado a datos del navegador o datos de cualquier otra aplicación que se ejecuta en el navegador.

13. El sistema de la reivindicación 12 que se adapta además para realizar el procedimiento de una cualquiera de las reivindicaciones 2 a 11.

14. Uno o más medios legibles a ordenador que tienen instrucciones ejecutables a ordenador, que cuando se ejecutan realizan etapas para aislar una aplicación (111) hospedada por una aplicación (104) host, comprendiendo las etapas:

10 hospedar la aplicación (111) hospedada dentro de la aplicación host correspondiente a un complemento (104) que se ejecuta en un navegador (102), incluyendo evitar que la aplicación hospedada acceda a datos del navegador o datos de cualquier otra aplicación hospedada;
comprendiendo las etapas, además:

15 proporcionar un elemento (114) de etiqueta para identificar la aplicación hospedada dentro de código (106) asociado con la aplicación host, en el que el elemento de etiqueta identifica una fuente de la aplicación hospedada y un área de renderización para la aplicación hospedada;
detectar el elemento de etiqueta, y
en respuesta, instanciar (204) un complemento oculto y cargar (206) la aplicación hospedada en el complemento oculto,

20 en el que evitar que la aplicación hospedada acceda a datos comprende instanciar el complemento oculto con acceso deshabilitado a datos del navegador o datos de cualquier otra aplicación que se ejecuta en el navegador.

15. Los uno o más medios legibles a ordenador de la reivindicación 14 que tienen instrucciones ejecutables a ordenador adicionales, que cuando se ejecutan realizan el procedimiento de una cualquiera de las reivindicaciones 2 a 11.

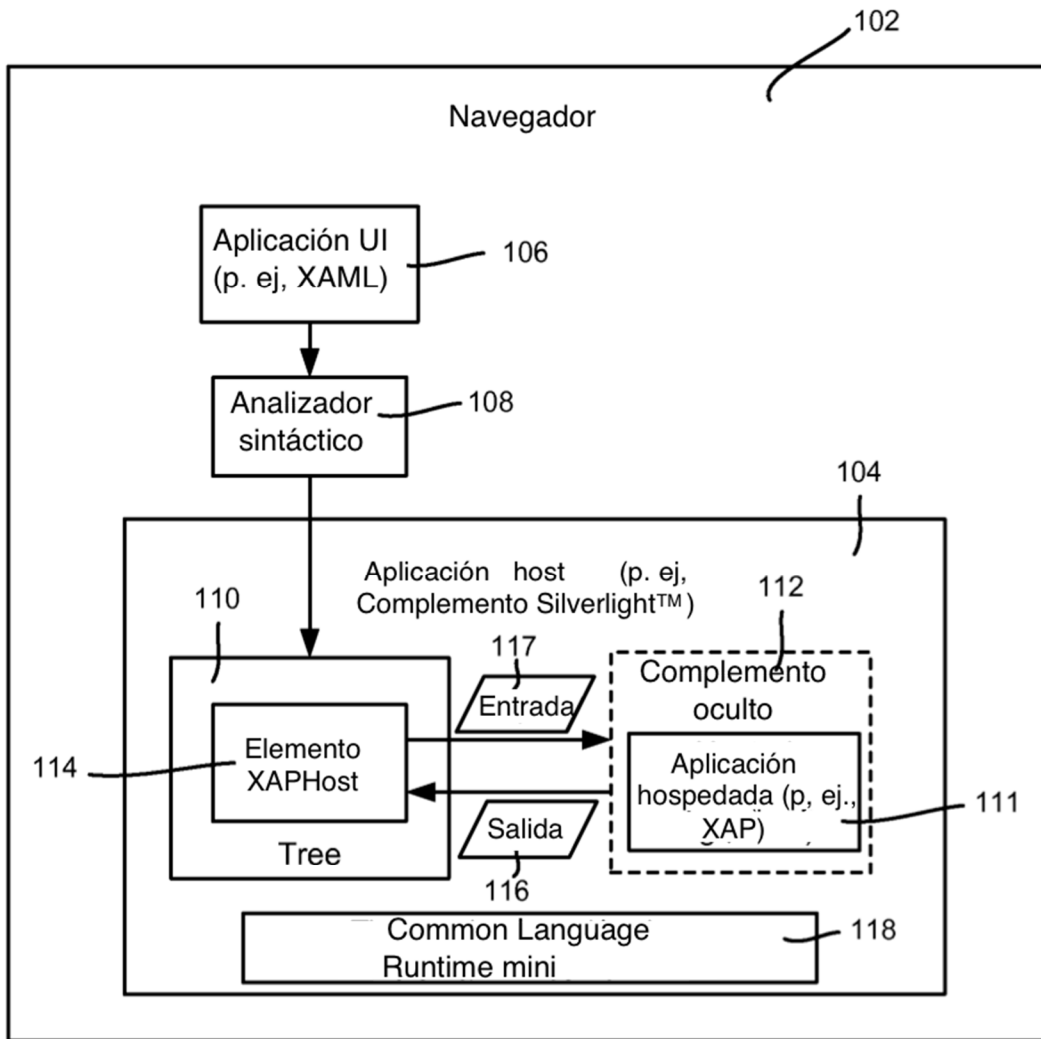


FIG. 1

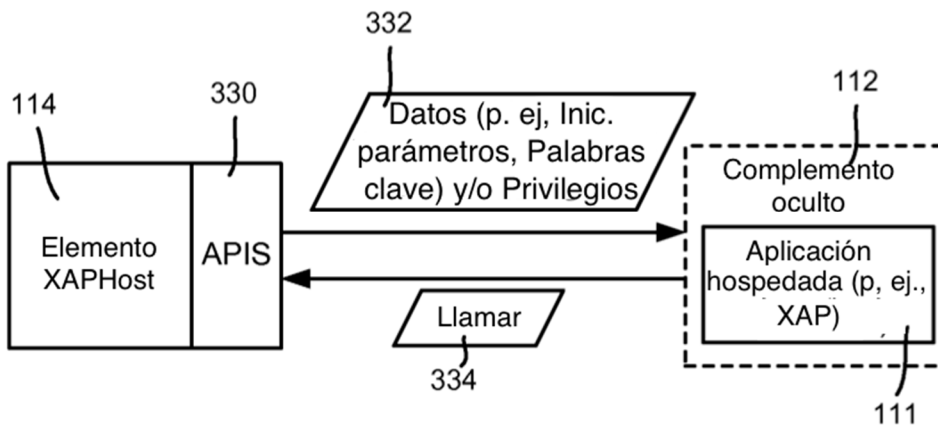
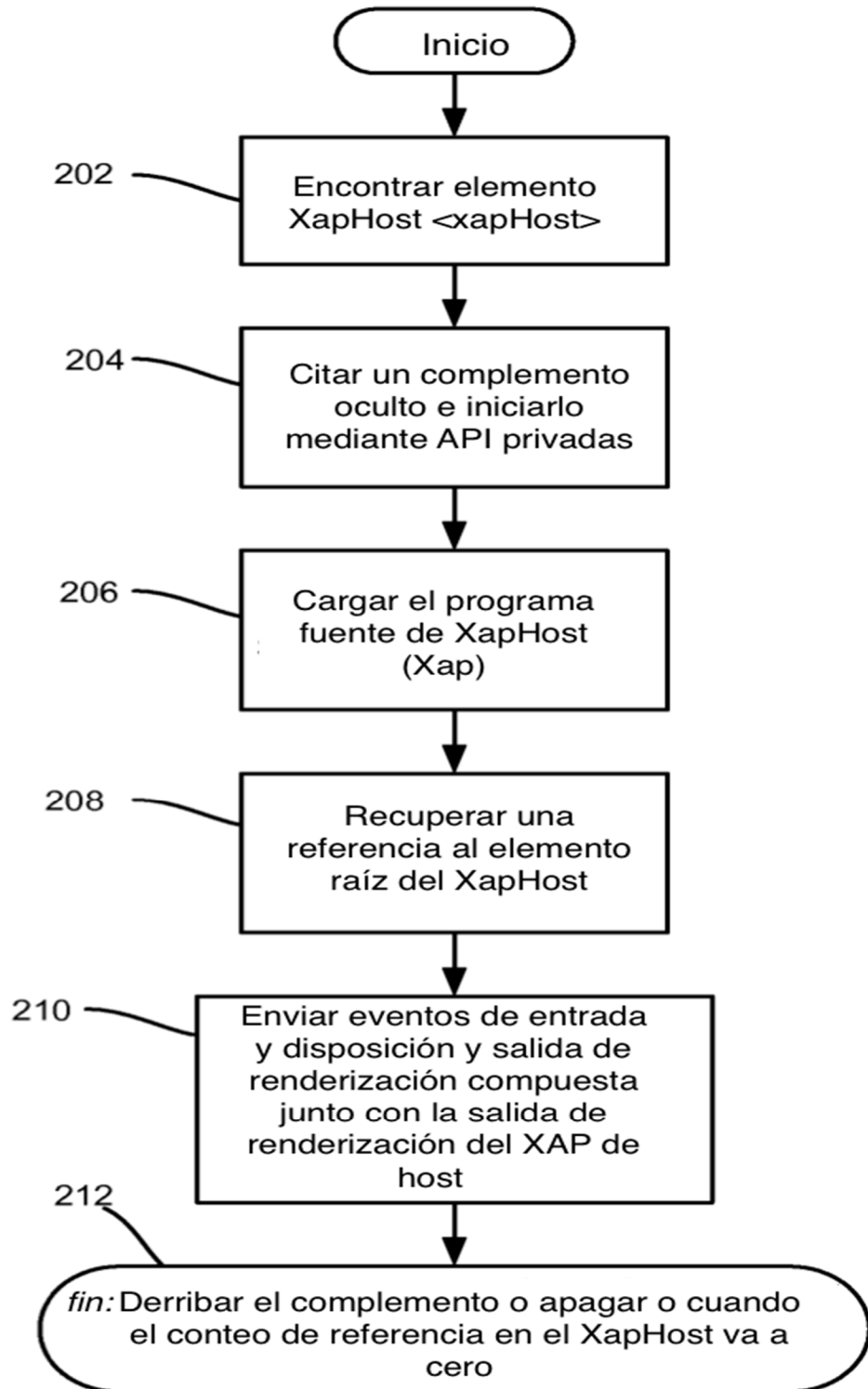


FIG. 3

FIG. 2



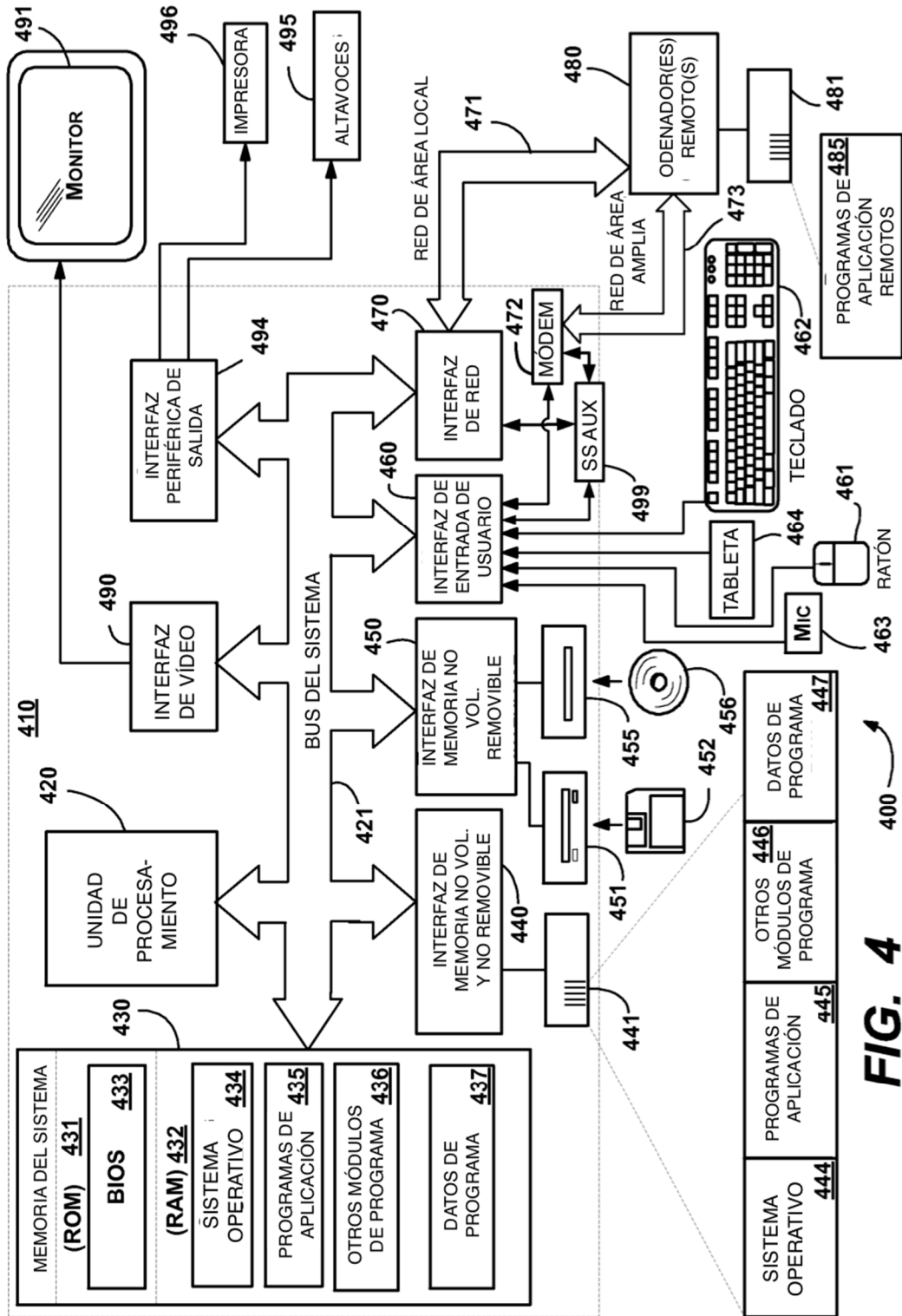


FIG. 4