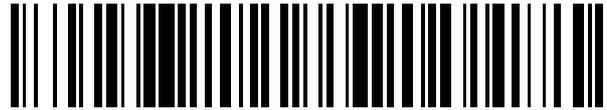


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 632 342**

51 Int. Cl.:

G06F 9/44 (2006.01)
H04N 21/214 (2011.01)
H04N 21/231 (2011.01)
H04N 21/239 (2011.01)
H04N 21/431 (2011.01)
H04N 21/436 (2011.01)
H04N 21/472 (2011.01)
H04N 21/482 (2011.01)
H04N 21/6543 (2011.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **14.01.2011** **E 11150981 (6)**

97 Fecha y número de publicación de la concesión europea: **12.04.2017** **EP 2357555**

54 Título: **Interfaz de usuario virtual**

30 Prioridad:

15.01.2010 US 688343

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

12.09.2017

73 Titular/es:

**GUEST TEK INTERACTIVE ENTERTAINMENT LTD. (100.0%)
Suite 240, 3030-3rd Avenue N.E.
Calgary, AB T2A 6TZ, CA**

72 Inventor/es:

**HULSE, DAVID y
THOMAS, JASON**

74 Agente/Representante:

CARPINTERO LÓPEZ, Mario

ES 2 632 342 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

Interfaz de usuario virtual

Antecedentes de la invención

5 La presente invención se refiere a proporcionar una interfaz de usuario virtual para un decodificador o dispositivo de cliente similar. Más en general, la presente invención permite la virtualización de la interfaz de usuario para cualquiera de una amplia variedad de aplicaciones y entornos informáticos.

10 Un enfoque convencional para facilitar la interacción del usuario con sistemas de información o entretenimiento que emplean decodificadores implica proporcionar una aplicación de interfaz de usuario que se ejecuta nativamente en el decodificador. Esta aplicación puede ser, por ejemplo, un reproductor Flash que presenta una jerarquía de menús. Desafortunadamente, debido a las capacidades típicas de un decodificador, la animación asociada con tales interfaces a menudo no se procesa de una manera satisfactoria, es decir, puede tomar el orden de segundos para que el reproductor proporcione cualquier nueva información que recibe (por ejemplo, representar una nueva pantalla de submenú desde una pantalla de menú anterior). Teniendo en cuenta lo que los usuarios se han acostumbrado a interactuar con interfaces de usuario en sistemas de ordenador de escritorio, este nivel de operación y la experiencia del usuario es inaceptable.

Debido a las limitaciones de rendimiento y a los costes de despliegue asociados con los decodificadores convencionales, es deseable proporcionar soluciones en las que los componentes de software que suministran los medios sean lo más independientes posible del dispositivo.

20 "The RFB Protocol Version 3.3", Richardson et al., páginas 1-26, XP002562572 describe el protocolo de memoria intermedia de trama remota (RFB) para acceso remoto a interfaces gráficas de usuario. El dispositivo de visualización es el cliente RFB y los cambios en la memoria intermedia de trama se originan en el servidor RFB. El protocolo RFB es un protocolo de cliente ligero y el cliente es sin estado. El cliente RFB y el servidor negocian un formato de píxeles y codificación. Los formatos de píxeles incluyen 24 bits o 16 bits de "color verdadero" y mapa de color de 8 bits. Los tipos de codificación incluyen en bruto, rectángulo de copia, RRE, RRE compacto y hextil. El cliente puede enviar un mensaje de formato *setpixel* al servidor que establece el formato en el que los valores de píxeles deben enviarse en mensajes de actualización de memoria intermedia de trama. El cliente también puede enviar un mensaje *setencodings* al servidor que establece los tipos de codificación en los que los datos de píxeles pueden ser enviados por el servidor.

30 El documento WO 99/17549 describe un decodificador que tiene una conexión de datos de Internet y que está controlado por entradas de usuario introducidas a través de un dispositivo de control remoto. Una GUI genera pantallas de menú que se superponen a las imágenes convencionales de vídeo de televisión, de modo que el usuario puede ver los gráficos del navegador generados por la GUI mientras visualiza imágenes de televisión en el fondo. La GUI está codificada en Java almacenado en el decodificador. El uso de Java permite que la GUI sea independiente del hardware. La GUI se proporciona en forma de panel de explorador que puede hacerse parcialmente transparente estableciendo apropiadamente el factor alfa correspondiente.

Sumario de la invención

De acuerdo con la presente invención, se proporcionan varios procedimientos, aparatos y productos de programas informáticos para permitir la virtualización de la interfaz de usuario.

De acuerdo con un primer aspecto de la invención, se proporciona un sistema como se define en la reivindicación 1.

40 De acuerdo con otros aspectos de la invención, se proporciona un procedimiento implementado por ordenador como se define en la reivindicación 9.

De acuerdo con un aspecto adicional de la invención, se proporciona un medio legible por ordenador, o medios, como se define en la reivindicación 15.

45 Una comprensión adicional de la naturaleza y de las ventajas de la presente invención se puede realizar haciendo referencia a las porciones restantes de la memoria descriptiva y los dibujos.

Breve descripción de los dibujos

La figura 1 es un diagrama simplificado que ilustra un ejemplo de un sistema en el que pueden implementarse realizaciones de la presente invención.

50 La figura 2 es un diagrama de bloques que ilustra varios componentes del sistema para una realización específica de la invención.

La figura 3 es un diagrama de flujo que ilustra la operación de una realización específica de la invención.

Descripción detallada de realizaciones específicas

A continuación, se hará referencia en detalle a realizaciones específicas de la invención que incluyen los mejores modos contemplados por los inventores para llevar a cabo la invención. Ejemplos de estas realizaciones específicas se ilustran en los dibujos adjuntos. Aunque la invención se describe conjuntamente con estas realizaciones específicas, se entenderá que no se pretende limitar la invención a las realizaciones descritas. Por el contrario, se pretende que cubra alternativas, modificaciones y equivalentes que puedan incluirse en el alcance de la invención como se define mediante las reivindicaciones adjuntas. En la siguiente descripción, se exponen detalles específicos para proporcionar una comprensión completa de la presente invención. La presente invención se puede poner en práctica sin comprender la presente invención. La presente invención puede practicarse sin algunos o todos estos detalles específicos. Además, características bien conocidas pueden no haberse descrito en detalle para evitar obscurecer innecesariamente la invención.

De acuerdo con diversas realizaciones de la invención, las interfaces de usuario virtuales que se ejecutan en uno o más servidores se presentan en dispositivos cliente remotos. Dichas realizaciones pueden caracterizarse como un servicio que agrupa, gestiona y limpia sesiones de interfaz de usuario virtual. Se describen realizaciones de la invención con referencia a interfaces de usuario asociadas con decodificadores que controlan la operación de televisores asociados (por ejemplo, jerarquías de menú). Sin embargo, debe observarse que las referencias a tales realizaciones no pretenden ser limitativas. Más bien, se contemplan realizaciones de la invención en las que se pueden presentar interfaces asociadas con cualquiera de una amplia variedad de aplicaciones de una manera que requiere muy poco en cuanto a recursos en el cliente y que puede configurarse para ser en gran medida independiente del hardware del dispositivo del cliente.

La figura 1 muestra un diagrama simplificado de un entorno de red en el que pueden implementarse realizaciones de la presente invención. La red 102 puede representar cualquier combinación de redes locales o de área amplia privadas y/o públicas implementadas con cualquiera de una amplia variedad de tecnologías de red y protocolos de comunicación. Por ejemplo, la red 102 puede representar una o más redes IP, redes de cable, redes de telecomunicaciones, redes de televisión por satélite, redes por cable o inalámbricas, o cualquier combinación de las mismas. El hardware en varias ubicaciones de usuario 104 está conectado a través de la red 102 a los servidores 106, 108 y 110 (y almacenes de datos asociados) que pueden ser operados y/o controlados por uno o más proveedores de servicio. Se entenderá que cada representación de un servidor puede representar múltiples servidores sobre los cuales se pueden distribuir los diversos procesos y cargas de cálculo asociados con las realizaciones de la invención. En esta representación, los servidores 106, 108 y 110 pueden representar diversos aspectos de implementaciones particulares tales como, por ejemplo, servidores de interfaz de usuario virtual, servidores de medios, etc.

El hardware de localización de usuario puede incluir un dispositivo cliente, por ejemplo, un decodificador 112, y una pantalla asociada, por ejemplo, un televisor 114. Alternativamente, los dispositivos de visualización 116 pueden incluir suficientes recursos informáticos integrados para interactuar con los servidores remotos de acuerdo con la invención, por ejemplo, algunos televisores digitales de nueva generación. De acuerdo con algunas realizaciones, las ubicaciones de usuario podrían ser, por ejemplo, habitaciones de huéspedes en un contexto de hospitalidad, siendo la red intermedia parte de la infraestructura de la red privada del hotel. Alternativamente, las ubicaciones de usuario pueden ser residencias privadas y/o empresas que se suscriben a servicios de televisión digital y/o información proporcionados, por ejemplo, por operadores de sistemas por cable o por satélite. Según otra alternativa, las ubicaciones de usuarios podrían ser residencias privadas y/o empresas conectadas a una red pública, tal como Internet. Los expertos en la técnica apreciarán el rango completo de realizaciones posibles con referencia a estos ejemplos.

De acuerdo con una clase particular de realizaciones, la interfaz de usuario para un decodificador se ejecuta en un servidor de interfaz de usuario (UI) virtual que está alejado de los decodificadores en el sistema. Como se mencionó anteriormente, el servidor de interfaz de usuario virtual puede ser, en realidad, varios servidores. Las actualizaciones gráficas incrementales para la interfaz de usuario se envían al decodificador a través de un canal de comunicación dedicado utilizando cualquiera de una variedad de protocolos tales como, por ejemplo, el protocolo de computación de red virtual (VNC), el protocolo de escritorio remoto (RDP), o protocolo de escritorio remoto de Apple (ARD). De forma más general, las realizaciones de la presente invención en esta clase de realizaciones pueden implementarse utilizando cualquier protocolo conocido o propietario por el cual se puedan transmitir a un cliente actualizaciones gráficas incrementales de una interfaz y en las que la entrada de usuario recibida desde el cliente accione las actualizaciones gráficas.

Como se comprenderá, un sistema VNC convencional incluye típicamente un cliente, un servidor y un protocolo de comunicación relativamente simple denominado RFB (es decir, memoria intermedia de trama remota). Una memoria intermedia de trama es una memoria intermedia que contiene valores de color para cada píxel que compone una imagen de visualización en un momento dado en el tiempo. A medida que la memoria intermedia de trama se modifica mediante el software de aplicación, el servidor VNC envía rectángulos pequeños de la memoria intermedia de trama, correspondientes a las regiones que han cambiado desde la última actualización, al cliente VNC para su representación en una pantalla asociada. El cliente envía una entrada de usuario al servidor VNC para activar estas actualizaciones.

- De acuerdo con algunas realizaciones relativas a decodificadores, el decodificador gestiona tanto la reproducción de flujos de vídeo como la presentación de la interfaz de usuario. Como se describirá, la interfaz de usuario se ejecuta y se procesa en una memoria intermedia de trama en una plataforma remota que transmite actualizaciones gráficas de la interfaz al decodificador. De acuerdo con varias realizaciones de la invención, cuando las actualizaciones son recibidas por el decodificador se "mezclan" con el vídeo que está siendo reproducido por el decodificador. De acuerdo con algunas de estas realizaciones, esta operación de mezcla se facilita, identificando si las porciones de la información en las actualizaciones de la interfaz de usuario son opacas, transparentes o semitransparentes, de modo que los píxeles correspondientes de la presentación se pueden representar como gráficos, vídeo, o una combinación de ambos.
- A continuación, se describirá un ejemplo de una arquitectura particular para proporcionar interfaces de usuario virtuales de acuerdo con realizaciones de la invención con referencia a la figura 2. De acuerdo con algunas realizaciones, los componentes representados pueden desplegarse en entornos informáticos similares a los representados en la figura 1. Sin embargo, los diversos componentes de software se muestran sin referencia específica al hardware o plataformas informáticas en las que están operando los componentes de software porque los expertos en la técnica comprenden que tales componentes de software pueden desplegarse de una amplia variedad de maneras sin apartarse del alcance de la invención. Por ejemplo, los componentes del sistema en el lado del servidor pueden residir en la misma o en diferentes plataformas informáticas. Por lo tanto, al menos algunas de las referencias específicas a la localización o contexto de componentes específicos no deben considerarse como limitativas de la invención.
- Con referencia ahora a la figura 2 y al diagrama de flujo de la figura 3, cuando se enciende un dispositivo cliente, se inicia un cliente UI remoto 210 (301) e intenta establecer una conexión, por ejemplo, hace una petición de conexión TCP con un demonio 202 de interfaz de usuario remoto en un servidor de interfaz de usuario virtual. El demonio 202 de la interfaz de usuario remoto escucha estas peticiones de conexión de clientes (302), e inicia un proceso de fondo para cada una (304). Los procesos de fondo iniciados por el demonio 202 persisten mientras persisten sus correspondientes sesiones de interfaz de usuario virtual (descritas a continuación).
- El demonio 202 (que puede ser uno de los múltiples demonios que funcionan conjuntamente) es responsable de gestionar un conjunto de sesiones de interfaz de usuario simultáneas en el sistema. El demonio 202 mantiene una lista de las instancias 209 de servidor VNC disponibles que se ejecutan en el servidor, con memorias intermedias 206 de tramas virtuales asociadas. Cuando recibe una petición de conexión, el demonio 202 asigna una de las instancias de servidor VNC disponibles a la conexión (308) (eliminándola de la lista) y devuelve la información de sesión al cliente mediante la que se puede identificar la sesión (310). De acuerdo con una realización específica, la información de sesión incluye una dirección IP y un número de puerto con los que el cliente puede conectarse a la instancia de servidor VNC asignada.
- El demonio 202 también crea una instancia de una consola de interfaz de usuario 208 (312) que se ejecuta en el servidor de interfaz de usuario virtual. La consola de interfaz de usuario se crea y se procesa en la memoria intermedia de trama asociada con la instancia del servidor VNC asignado. De acuerdo con una realización específica, la consola es un reproductor de Adobe Flash que genera la interfaz de usuario. En contraste con un reproductor que se ejecuta de forma nativa en un decodificador convencional, la consola 208 vuelve la interfaz de usuario a la memoria intermedia de trama asociada con la instancia (314) de servidor VNC asignada en lugar de una memoria intermedia de trama en el decodificador. Debido a los mayores recursos de computación disponibles en el servidor, las tramas de la interfaz de usuario pueden reproducirse a una velocidad de tramas mucho mayor que en un decodificador convencional (casi 100 veces más rápido en una implementación particular).
- En la implementación basada en VNC representada, cada instancia 209 de servidor VNC es un servidor VNC acoplado con un servidor 211 X Windows (por ejemplo, Xvnc4). El servidor X Windows proporciona una API de representación para aplicaciones de cliente X Windows, como las aplicaciones (208) de interfaz de usuario, que representan cada aplicación en una memoria intermedia de tramas compartida con el servidor VNC acoplado. Cada instancia de servidor VNC 209 que opera en el servidor (que puede ser uno o más servidores) monitoriza continuamente su memoria intermedia asociada y transmite actualizaciones gráficas al cliente VNC al que se ha asignado la instancia de servidor VNC. De acuerdo con realizaciones específicas, estas actualizaciones se transmiten en respuesta a peticiones de actualización del cliente, y no corresponden necesariamente a las tramas que se representan en la memoria intermedia. Más bien son más típicamente alguna agregación de píxeles (típicamente una o más porciones rectangulares de la interfaz) que han cambiado desde que se envió la última actualización al cliente. De acuerdo con realizaciones específicas, corresponde al cliente solicitar el contenido completo de la memoria intermedia de trama o solo una actualización incremental. En este último caso, la instancia de servidor VNC recuerda el estado de la memoria intermedia de trama virtual cuando el cliente solicitó por última vez una actualización y transmite solamente la información que ha cambiado desde la última solicitud. Además, el servidor VNC no puede responder inmediatamente a la solicitud de actualización de un cliente. Es decir, el servidor VNC puede esperar brevemente para agregar actualizaciones para la transmisión al cliente para proporcionar una respuesta más eficiente de ancho de banda.
- Como se ha explicado anteriormente, cuando el dispositivo cliente se enciende, el cliente UI remoto 210 se inicia y establece una conexión con el demonio 202. El cliente UI remoto 210 inicia entonces una implementación de un

cliente VNC 212 (por ejemplo, DirectVNC) en el dispositivo cliente, por ejemplo, el decodificador, que es la aplicación que solicita las actualizaciones de la interfaz desde la instancia de servidor VNC asignada. Utilizando la información de sesión, el cliente VNC 212 establece una conexión VNC directa con la instancia de servidor VNC asignada al cliente (316) sobre la que se pasan posteriormente las actualizaciones de interfaz.

- 5 Debe observarse de nuevo que, si bien el dispositivo cliente puede ser, en algunas realizaciones, un decodificador, se contemplan implementaciones en las que las funcionalidades descritas en el presente documento están integradas en el hardware que en última instancia reproduce la interfaz, por ejemplo, un televisor digital con suficientes recursos de procesamiento y memoria. Alternativamente, al menos algunas de las funcionalidades descritas en este documento podrían implementarse en un dispositivo autónomo separado que opera con un
10 televisor, un decodificador o ambos.

Una vez establecida la conexión VNC, el cliente VNC 212 solicita actualizaciones desde la instancia del servidor VNC (318) correspondiente, que se reproduce en el decodificador. Esto se hace a través de una conexión entre el cliente VNC 212 y el hardware 214 de procesamiento del dispositivo cliente.

- 15 De acuerdo con una clase particular de realizaciones, se proporcionan extensiones al Protocolo VNC para gestionar información de píxeles transparente. Como es bien sabido, dentro de los sistemas gráficos, una memoria intermedia de trama es un área de memoria comúnmente utilizada para contener los valores de color de píxeles que forman una imagen de visualización. La forma en la que los píxeles se almacenan dentro de una memoria intermedia de trama puede tomar muchos formatos diferentes dependiendo del propósito de la memoria intermedia de trama. Por ejemplo, una memoria intermedia de trama utilizada para el almacenamiento de tramas de vídeo puede almacenar
20 píxeles en un formato YUV420 plano, mientras que una memoria intermedia de trama utilizada para accionar un dispositivo de visualización puede almacenar píxeles en un formato RGB empaquetado. Estos diferentes formatos están determinados por el espacio de color de la memoria intermedia de trama y una representación binaria predeterminada de un píxel dentro de ese espacio de color. Por ejemplo, un espacio de color RGB podría almacenar píxeles dentro de una palabra de 16 bits, con 5 bits asignados a los componentes rojo y azul y 6 bits asignados al
25 componente verde. Alternativamente, pueden utilizarse 24 bits con 8 bits asignados a cada uno de los tres componentes de color. Sin embargo, puesto que 24 bits (3 bytes) es inconveniente para trabajar con sistemas informáticos con un tamaño de palabra nativa de 32 bits, es común almacenar píxeles RGB en una palabra de 32 bits, con 8 bits asignados a cada componente de color, dejando 8 bits adicionales sin usar. El número de bits utilizados para almacenar un píxel se denomina a veces profundidad de color de la memoria intermedia de trama.

- 30 En algunos sistemas gráficos, particularmente aquéllos en los que se contempla que las imágenes de visualización estarán compuestas con otras fuentes de imagen (por ejemplo, tramas de vídeo), la memoria intermedia de trama también puede almacenar información de transparencia junto con cada píxel. Esta información de transparencia a veces se conoce como el "canal alfa" o "componente alfa". El componente alfa de un píxel determina el grado de transparencia del píxel, que va desde completamente opaco a completamente invisible. Usando reglas de composición de imagen estándar bien conocidas por los expertos en la técnica, se puede usar una memoria intermedia de trama que contiene datos de píxeles junto con información alfa para combinar múltiples imágenes. Una posible aplicación de esto sería "mezclar" tramas de vídeo con una capa gráfica, de manera que una interfaz gráfica de usuario (representada en la capa gráfica) pudiera superponerse sobre el vídeo de una manera translúcida.

- 40 En una memoria intermedia de trama RGB, donde la información de píxeles se almacena usando 24 bits dentro de una palabra de 32 bits, los 8 bits no utilizados se utilizan comúnmente para almacenar la información de canal alfa para cada píxel. Dicho formato de píxel se denomina a veces ARGB:8888, aunque también son posibles otros formatos de píxeles que incluyen un canal alfa.

- 45 Según varias realizaciones de la presente invención, el servidor X Windows está configurado para reproducir una memoria intermedia de trama almacenando píxeles en el formato RGB, utilizando 24 bits para retener la información RGB dentro de una palabra de 32 bits. Las aplicaciones cliente de Standard X Windows se procesarán a través de la API de X Windows en esta memoria intermedia de trama RGB. Sin embargo, aplicaciones especializadas, como aquellas que requieren representación de información de transparencia, pueden acceder directamente a la memoria intermedia de tramas de X Windows (utilizando la extensión XSHM estándar en X Windows). Estas aplicaciones pueden utilizar los 8 bits no utilizados dentro de cada píxel de 32 bits para almacenar el componente alfa para cada
50 píxel. Desafortunadamente, los clientes de estándar X Windows que no toman este enfoque no necesariamente establecerán correctamente el componente alfa en cada píxel (ya que X Windows no lo admite directamente) y así "heredarán" cualquier componente alfa preexistente para cada píxel. Por esta razón, es necesario modificar el servidor estándar X Windows para que las aplicaciones cliente estándar representen un componente alfa para cada píxel, dando como resultado píxeles completamente opacos. Este cambio garantiza que las aplicaciones cliente estándar de X Windows se reproducirán de manera opaca, mientras que las aplicaciones cliente más especializadas pueden representarse de forma translúcida al establecer directamente el componente alfa para cada píxel.
55

- 60 Así, las realizaciones de la presente invención emplean un servidor X Windows, modificado como se ha descrito anteriormente, acoplado con un servidor VNC. El servidor VNC monitoriza la memoria intermedia de trama compartida con el servidor X Windows y proporciona actualizaciones gráficas a los clientes VNC en otras partes de la red utilizando el protocolo VNC. Una parte inherente del protocolo VNC es la negociación entre el cliente y el

servidor del formato de píxel que las actualizaciones gráficas van a utilizar. El protocolo VNC define varios formatos de píxeles diferentes, ninguno de los cuales incluye un componente alfa, es decir, el protocolo VNC estándar no admite información de transparencia. Por lo tanto, estas realizaciones de la invención extienden el protocolo VNC para soportar formatos de píxeles adicionales.

5 Las extensiones al protocolo VNC pueden implementarse creando los denominados formatos de pseudo-codificación. Un cliente entonces indicará que soporta una extensión dada solicitando una pseudo-codificación correspondiente a esa extensión. Un servidor que no soporta esa extensión simplemente ignorará la solicitud, mientras que un servidor adecuadamente mejorado responderá enviando al cliente un mensaje de acuse de recibo de propietario asociado con la extensión.

10 De acuerdo con estas realizaciones de la presente invención, el protocolo VNC se ha ampliado mediante la adición de un mensaje de pseudo-codificación conocido como "Preservar Canal Alfa". Cuando un cliente solicita a un servidor que cumpla con esta pseudo-codificación, el servidor responde enviando al cliente un mensaje de acuse de recibo propietario. El cliente solicita entonces un formato de píxeles RGB utilizando 24 bits para almacenar los datos RGB dentro de una palabra de 32 bits y el servidor garantiza que todas las actualizaciones gráficas conserven los
15 datos del componente alfa utilizando los 8 bits restantes dentro de la palabra de 32 bits para transferir el componente alfa de cada píxel. En particular, el procedimiento "ZRLE" de envío de actualizaciones gráficas no comprimirá píxeles de 32 bits en 24 bits, lo que daría lugar a la pérdida de los datos del componente alfa.

Con referencia de nuevo a la figura 3, y de acuerdo con algunas realizaciones, el cliente VNC 212 también envía
20 varios tipos de entrada recibidos desde el decodificador (por ejemplo, interacción del usuario con la interfaz virtual a través de un control remoto, puntero, ratón o teclado) a la instancia del servidor VNC (320) que los pasa a la consola 208 de interfaz de usuario a través de la API del servidor X Windows. Tras la recepción de estos eventos de entrada de usuario, la aplicación 208 de interfaz de usuario reacciona a los eventos, lo que puede dar lugar a que se realicen actualizaciones gráficas en la interfaz de usuario y, por lo tanto, se convierten en la memoria intermedia de trama asociada al servidor X Windows asignado.

25 Mientras el dispositivo cliente está operativo, la consola de interfaz de usuario y la instancia del servidor VNC asignados al codificador se mantienen, incluso mientras el codificador está reproduciendo otros medios y el usuario no está viendo o interactuando con la interfaz de usuario virtual. La conexión, por ejemplo, el conector TCP, que se establece para iniciar la sesión de interfaz de usuario, también se mantiene durante la sesión para indicar que los recursos asignados a ese cliente (por ejemplo, instancia del servidor VNC) siguen siendo utilizados o no se está
30 transmitiendo nada a través de la conexión). Cuando termina la sesión (por ejemplo, el dispositivo cliente se apaga), se termina la conexión TCP (322) y el demonio remoto de interfaz de usuario inicia una limpieza en el lado del servidor que implica la terminación del intérprete de interfaz de usuario (324) y la recuperación de la instancia del servidor VNC (326).

35 Cabe destacar que en las realizaciones implementadas usando decodificadores, el contenido real de los medios seleccionados por el usuario en la interfaz de usuario y reproducido por el decodificador en el televisor asociado se entrega típicamente al decodificador a través de otro canal (por ejemplo, una conexión TCP o UDP entre un servidor de medios remoto y el software intermedio correspondiente que opera en el decodificador) no mostrado en la figura 2.

40 De acuerdo con una clase específica de realizaciones, el movimiento de la funcionalidad de consola de interfaz de usuario desde el decodificador a un servidor final implica una bifurcación de la funcionalidad en la consola original que era nativa en el decodificador. Es decir, en soluciones anteriores en las que una interfaz de usuario nativa operaba en el decodificador, interactuaba con el hardware de representación del decodificador (es decir, para atraer la interfaz a la memoria de trama), así como un software intermedio residente (por ejemplo, 215) para controlar la reproducción de los medios. De acuerdo con una realización particular de la presente invención y como se ilustra en
45 la figura 2, la interfaz de usuario se mueve al extremo posterior, la interfaz al hardware de representación se reemplaza eficazmente con el cliente VNC y la interfaz al software intermedio se sustituye por un "proxy" intermedio que actúa en el decodificador que gestiona la comunicación con el software intermedio 215. Esta "Capa de abstracción remota de software intermedio" (por ejemplo, proxy de red de software intermedio 216) recibe comunicaciones de la interfaz de usuario mediante una conexión de red (por ejemplo, un conector TCP) y las
50 convierte en comunicaciones entre procesos reconocidas por el software intermedio, por ejemplo, comunicaciones a través de clavijas Unix que se establecen y persisten mientras el decodificador está operativo. El conector entre la consola y la capa de abstracción puede establecerse, por ejemplo, utilizando la dirección IP del cliente pasada desde el demonio 202 a la consola 208.

55 De acuerdo con una realización específica, la capa de abstracción remota de software intermedio puede caracterizarse como un sistema de entrega y recuperación de eventos que es efectivamente una capa de abstracción para las capacidades de medios de la plataforma subyacente, es decir, proporciona la traducción necesaria entre la plataforma de hardware subyacente y la lógica que lo controla en la interfaz de usuario remota. Las llamadas gestionadas por la capa de abstracción remota de software intermedio incluyen, por ejemplo, información que determina los cambios que se deben realizar al estado del dispositivo de reproducción de medios
60 debido a la interacción del usuario, por ejemplo, sintonización a un nuevo canal de televisión. Por ejemplo, cuando

un usuario presiona "Canal siguiente" en su mando a distancia, esto produce un evento de pulsación de tecla remota que es consumido por la consola de interfaz de usuario que se ejecuta en el servidor remoto. La lógica en la consola de interfaz de usuario determina entonces cuál es el siguiente canal de la guía y presenta los detalles correspondientes al usuario. Tras la selección de ese canal, la interfaz de usuario transmite un mensaje a través del conector de llamada de software intermedio con instrucciones para comenzar la reproducción del canal seleccionado. Por lo tanto, la capa de abstracción remota de software intermedio solo necesita proporcionar una interfaz genérica que pueda presentarse de la misma forma o similar en cualquier pieza de hardware, exponiendo así las capacidades de reproducción de medios de la plataforma. Los eventos de notificación desde el software intermedio a la consola de interfaz de usuario pueden incluir cualquier tipo de hardware o actualizaciones de estado de reproducción. Ejemplos de tales eventos incluyen una notificación de que el clip actual ha terminado, la posición de tiempo en el clip actual, información de estado de error o cualquier elemento relacionado con el estado dinámico de las capacidades de medios o hardware de la plataforma de destino.

De acuerdo con un conjunto específico de realizaciones, el mecanismo de abstracción proporcionado por la capa de abstracción remota de software intermedio emplea un conjunto de puertos locales y a través de estos puertos puede proporcionar servicios de punto final a piezas de software independientes locales. De acuerdo con una de tales realizaciones, la capa de abstracción remota de software intermedio mantiene dos conectores de dominio Unix conectados al conector local de devolución de llamada de software intermedio y al conector de entrega de eventos de software intermedio respectivamente. La capa de abstracción remota de software intermedio también mantiene dos puertos de red para que los servicios remotos puedan acceder al sistema como si fueran un proceso local a través de un protocolo remoto. De acuerdo con algunas realizaciones, la capa de abstracción remota de software intermedio también se puede emplear para facilitar el control remoto del decodificador, por ejemplo, para fines de prueba o depuración.

Debe entenderse que la capa de abstracción remota de software intermedio no es un componente necesario para todas las implementaciones de interfaces de usuario virtuales de acuerdo con la presente invención. Es decir, como se ha descrito anteriormente, la capa de abstracción remota de software intermedio se proporciona como parte de una solución en la que la mayor parte de la funcionalidad del software originalmente diseñado para ejecutarse en el decodificador se mueve a un servidor de etapa final. Proporcionar esta capa de abstracción permite esta solución sin tener que modificar el software intermedio existente en el decodificador. Sin embargo, la presentación de otros tipos de interfaces de usuario virtuales de acuerdo con las diversas realizaciones de la presente invención, dependiendo de su funcionalidad, puede no requerir este proxy intermedio, por ejemplo, aplicaciones que no requieren control sobre la reproducción de medios o no deberían tener estas capacidades de control.

Más generalmente, se contemplan realizaciones de la invención en las que la consola de interfaz de usuario que funciona en el servidor UI virtual puede ser reemplazada por cualquier aplicación arbitraria configurada para convertirse en una memoria intermedia de trama a través de un servidor de X Windows. De acuerdo con una clase particular de realizaciones, tales aplicaciones preferiblemente no incluyen mucho si cualquier contenido de vídeo en que la representación de vídeo en un cliente remoto podría no ser enteramente satisfactoria si se suministra a través de una conexión VNC. Es decir, la velocidad de tramas requerida para representar correctamente el vídeo, generalmente no coincide con la velocidad de las actualizaciones incrementales realizadas con un enfoque similar al VNC, dando lugar de esta manera a varios artefactos de vídeo no deseados, tales como "desgarro" o tramas de vídeo parcialmente reproducidas. Una alternativa adecuada para suministrar dicho contenido de vídeo se describe en la solicitud de patente US n.º 12/473.086 para SERVICIOS DE ESCRITORIO VIRTUAL, presentada el 27 de mayo de 2009, publicada como publicación n.º US 2010-0269135.

A la vista de lo anterior, se contemplan realizaciones en las que interfaces de usuario virtuales generadas de acuerdo con la presente invención incluyen contenido de audio. De acuerdo con algunas de estas realizaciones, un flujo de audio asociado con la interfaz se transmite al cliente además de las actualizaciones gráficas de la interfaz descrita anteriormente. Esto puede hacerse, por ejemplo, utilizando una extensión propietaria del protocolo VNC. Alternativamente, puede proporcionarse un canal de audio secundario mediante un protocolo separado (por ejemplo, audio de pulsos).

La capacidad de ejecutar aplicaciones arbitrarias de forma remota de acuerdo con algunas realizaciones de la invención convierte eficazmente los dispositivos cliente incorporados en plataformas abiertas que pueden disfrutar de un rango mucho más amplio de desarrollo de terceros de aplicaciones de software. Es decir, la naturaleza misma de los entornos de computación propietarios integrados de decodificadores asociados con sistemas de televisión por cable o satélite, o con otros sistemas de televisión digital propietarios, presenta barreras para el desarrollo de aplicaciones para dichas plataformas por parte de desarrolladores de terceros. Por el contrario, las realizaciones de la invención son capaces de crear tales oportunidades permitiendo la operación remota y la entrega de aplicaciones arbitrarias en plataformas de servidor más estándar, permitiendo así que las aplicaciones que tradicionalmente solo están disponibles en sistemas informáticos de escritorio estén disponibles en plataformas informáticas integradas.

Se pueden emplear realizaciones de la invención para mover una amplia gama de funcionalidades (por ejemplo, rasterización, representación de fuentes, procesamiento de gráficos, etc.) a un servicio de extremo final. Por lo tanto, el papel del dispositivo cliente puede reducirse significativamente, por ejemplo, reenviando la entrada del usuario, proporcionando actualizaciones visuales y exponiendo una API para interactuar con sus capacidades de

reproducción de medios. La capacidad de respuesta, las capacidades de representación y las capacidades lógicas para cualquier aplicación dada pueden hacerse neutrales al dispositivo. También se pueden eliminar las dependencias que transfieren a otras plataformas, eliminando una dependencia de un sistema de representación gráfica y reemplazándola por un cliente ligero con una base de origen pequeña. El cliente ligero puede configurarse para que no requiera bibliotecas de soporte y solo funciones estándar mínimas, por ejemplo, conexiones de red y una memoria intermedia de trama para actualizar. Las nuevas funciones y componentes de software que son neutrales para la plataforma de destino también pueden integrarse de forma rápida y sencilla ya que la arquitectura del servidor (por ejemplo, una plataforma x86) ejecutará normalmente un sistema operativo con todas las bibliotecas necesarias para ejecutar cualquier pieza moderna de software. De este modo, se contemplan realizaciones de la invención que eliminan completamente la necesidad de proporcionar puertos, proporcionar bibliotecas para, o sintonizar prácticamente cualquier software que se desee proporcionar a un usuario final en prácticamente cualquier dispositivo cliente, incluyendo entornos de decodificadores propietarios y similares.

Aunque la invención se ha mostrado y descrito particularmente con referencia a realizaciones específicas de la misma, se entenderá por los expertos en la técnica que pueden realizarse varios cambios en la forma y los detalles de las realizaciones divulgadas sin apartarse del alcance de la invención. Además, aunque se han descrito aquí varias ventajas, aspectos y objetos de la presente invención con referencia a diversas realizaciones, se entenderá que el ámbito de la invención no debe estar limitado por referencia a tales ventajas, aspectos y objetos. Por el contrario, el alcance de la invención debe determinarse con referencia a las reivindicaciones adjuntas.

REIVINDICACIONES

1. Un sistema para proporcionar interfaces de usuario virtuales a través de una red (102), que comprende:
 - una pluralidad de clientes remotos (104) con pantallas (114) asociadas; y
 - uno o más servidores (106, 108, 110) configurados para proporcionar una interfaz de usuario virtual y desplegarse en la red (102) y cada servidor estando configurado para:
 - gestionar una pluralidad de aplicaciones de interfaz de usuario (208) teniendo, cada una, una memoria intermedia (206) de trama virtual asociada;
 - en respuesta a una solicitud de conexión desde un cliente remoto (104), asociar una aplicación de interfaz de usuario (208) con una asignada de las memorias intermedias de trama virtual (206), estando configurada la aplicación de interfaz de usuario (208) para reproducir una interfaz gráfica de usuario en la memoria intermedia de trama virtual asignada;
 - establecer (316) una primera conexión bidireccional persistente dedicada al cliente remoto (104) para conducir una sesión de interfaz de usuario virtual;
 - enviar (318) actualizaciones de la interfaz gráfica de usuario desde la memoria intermedia (206) de trama virtual asignada al cliente remoto (104) a través de la primera conexión, incluyendo al menos algunas de las actualizaciones información de transparencia para utilizar en la fusión de la interfaz gráfica de usuario con vídeo en el cliente remoto;
 - recibir (320) la entrada del usuario desde el cliente remoto a través de la primera conexión y proporcionar la entrada del usuario a la aplicación de interfaz de usuario, representando la entrada de usuario la interacción de un usuario asociado con el cliente remoto con una representación de la interfaz gráfica de usuario, generar instrucciones de reproducción para controlar la reproducción de vídeo en el cliente remoto de acuerdo con la entrada del usuario y enviar las instrucciones de reproducción al cliente remoto a través de una segunda conexión a través de la red;
 - estando configurado cada cliente (104) para recibir las actualizaciones gráficas de la interfaz de usuario desde la memoria intermedia de trama virtual correspondiente a través de la primera conexión y para reproducir la representación de la interfaz gráfica de usuario correspondiente con el vídeo en la pantalla (114) asociada, utilizando las actualizaciones y la información de transparencia, estando cada cliente configurado además para transmitir la entrada de usuario al uno o más servidores a través de la primera conexión y en el que cada cliente está configurado además para proporcionar funciones de control de reproducción de medios usando software intermedio (215) e incluyendo cada cliente una capa de abstracción de software intermedio (216) configurada para facilitar la comunicación entre la aplicación de interfaz de usuario correspondiente en el servidor y el software intermedio a través de la segunda conexión a través de la red y para controlar el vídeo en la pantalla asociada de acuerdo con las instrucciones de reproducción recibidas sobre la segunda conexión, en el que la segunda conexión está separada de la primera conexión, y en el que cada cliente está también configurado para recibir vídeo a través de una tercera conexión a través de la red entre un servidor de medios remoto y el software intermedio, y en el que la tercera conexión está separada de la primera conexión y de la segunda conexión.
2. El sistema de la reivindicación 1, en el que uno o más de los clientes (104) se implementa en un decodificador (112) configurado para proporcionar servicios de televisión interactiva en la pantalla (114) asociada.
3. El sistema de la reivindicación 2, en el que la interfaz gráfica de usuario comprende un menú interactivo para navegar por los servicios de televisión interactiva.
4. El sistema de la reivindicación 1, en el que uno o más de los clientes (104) está integrado en un único dispositivo (116) con la pantalla asociada.
5. El sistema de la reivindicación 1, en el que la primera conexión comprende una conexión de ordenador de red virtual, VNC, una conexión de protocolo de escritorio remoto, una conexión RDP, o una conexión de protocolo de escritorio remoto de Apple, ARD.
6. El sistema de la reivindicación 1, en el que la primera conexión comprende una conexión de computación de red virtual, VNC, y en el que el uno o más servidores están configurados para incluir la información de transparencia en al menos algunas de las actualizaciones en respuesta a solicitudes de mensaje de pseudo-codificación recibidas desde el cliente correspondiente.
7. El sistema de la reivindicación 1, estando el uno o más servidores configurados además para establecer la segunda conexión con cada uno de los clientes a través de la red en respuesta a solicitudes de conexión correspondientes de los clientes, y para terminar la sesión de interfaz virtual correspondiente en respuesta a la terminación de la segunda conexión.

8. El sistema de la reivindicación 1, en el que cada actualización comprende solamente una o más porciones de la interfaz gráfica de usuario correspondiente que ha cambiado desde una actualización previa.

9. Un procedimiento implementado por ordenador para interactuar con una interfaz de usuario virtual a través de una red (102), que comprende:

5 transmitir una solicitud de conexión desde un cliente remoto a un servidor a través de la red, estando configurado el servidor para gestionar la operación de una pluralidad de aplicaciones de interfaz de usuario, cada una de las cuales tiene una memoria intermedia de trama virtual asociada y, en respuesta a la solicitud de conexión, asociar una aplicación de interfaz de usuario que opera en el servidor con una asignada de las memorias intermedias de trama virtual, estando configurada la aplicación de interfaz de usuario para reproducir una interfaz gráfica de usuario en la memoria intermedia de trama virtual asignada;

10 establecer una primera conexión bidireccional persistente dedicada al servidor a través de la red para conducir una sesión de interfaz de usuario virtual;

15 recibir actualizaciones de la interfaz gráfica de usuario desde la memoria intermedia a través de la primera conexión, incluyendo al menos algunas de las actualizaciones información de transparencia para utilizar en la fusión de la interfaz gráfica de usuario con vídeo;

reproducir una representación de la interfaz gráfica de usuario junto con el vídeo en una pantalla asociada con el dispositivo remoto utilizando las actualizaciones y la información de transparencia y utilizando software intermedio en el dispositivo remoto;

20 transmitir la entrada del usuario al servidor a través de la primera conexión para presentación al usuario a la aplicación de interfaz de usuario, representando la entrada de usuario la interacción de un usuario con la representación de la interfaz gráfica de usuario, generando instrucciones de reproducción para controlar la reproducción de vídeo en el cliente remoto de acuerdo con la entrada del usuario y enviando las instrucciones de reproducción al cliente remoto a través de una segunda conexión a través de la red;

25 facilitando el software intermedio la comunicación con la aplicación de interfaz de usuario en el servidor a través de la segunda conexión y controlando el vídeo en la pantalla de acuerdo con las instrucciones de reproducción recibidas sobre la segunda conexión desde el servidor; y

recibiendo el cliente remoto vídeo a través de una tercera conexión a través de la red entre un servidor de medios remoto y el software intermedio, en el que la segunda conexión está separada de la primera conexión y en el que la tercera conexión está separada de la primera conexión y de la segunda conexión.

30 10. El procedimiento implementado por ordenador de la reivindicación 9, en el que la primera conexión comprende una conexión VNC, y en el que la información de transparencia está incluida en al menos algunas de las actualizaciones en respuesta a una solicitud de formato de pseudo-codificación recibida desde el cliente remoto.

35 11. El procedimiento implementado por ordenador de la reivindicación 9, que comprende además establecer la segunda conexión con el cliente remoto sobre la red en respuesta a una solicitud de conexión del cliente y finalizar la sesión de interfaz virtual en respuesta a la terminación de la segunda conexión.

40 12. El procedimiento implementado por ordenador de la reivindicación 9, en el que cada actualización comprende solamente una o más porciones de la interfaz gráfica de usuario que ha cambiado desde una actualización anterior, comprendiendo el procedimiento además enviar (318) las actualizaciones ya sea en respuesta a un cambio detectado en la representación gráfica Interfaz de usuario o en respuesta a un disparador programable no relacionado con la detección de cambios en la interfaz gráfica de usuario.

13. El procedimiento implementado por ordenador de la reivindicación 9, en el que la interfaz gráfica de usuario comprende un menú interactivo para seleccionar servicios de televisión interactiva.

45 14. El procedimiento implementado por ordenador de la reivindicación 9, en el que la primera conexión comprende una conexión de ordenador de red virtual, VNC, una conexión de protocolo de escritorio remoto, una conexión RDP, o una conexión de protocolo de escritorio remoto de Apple, ARD.

15. Medio legible por ordenador que almacena códigos de programa de ordenador ejecutables mediante dispositivos de procesamiento de datos para realizar el procedimiento implementado por ordenador de cualquiera de las reivindicaciones 9 a 14.

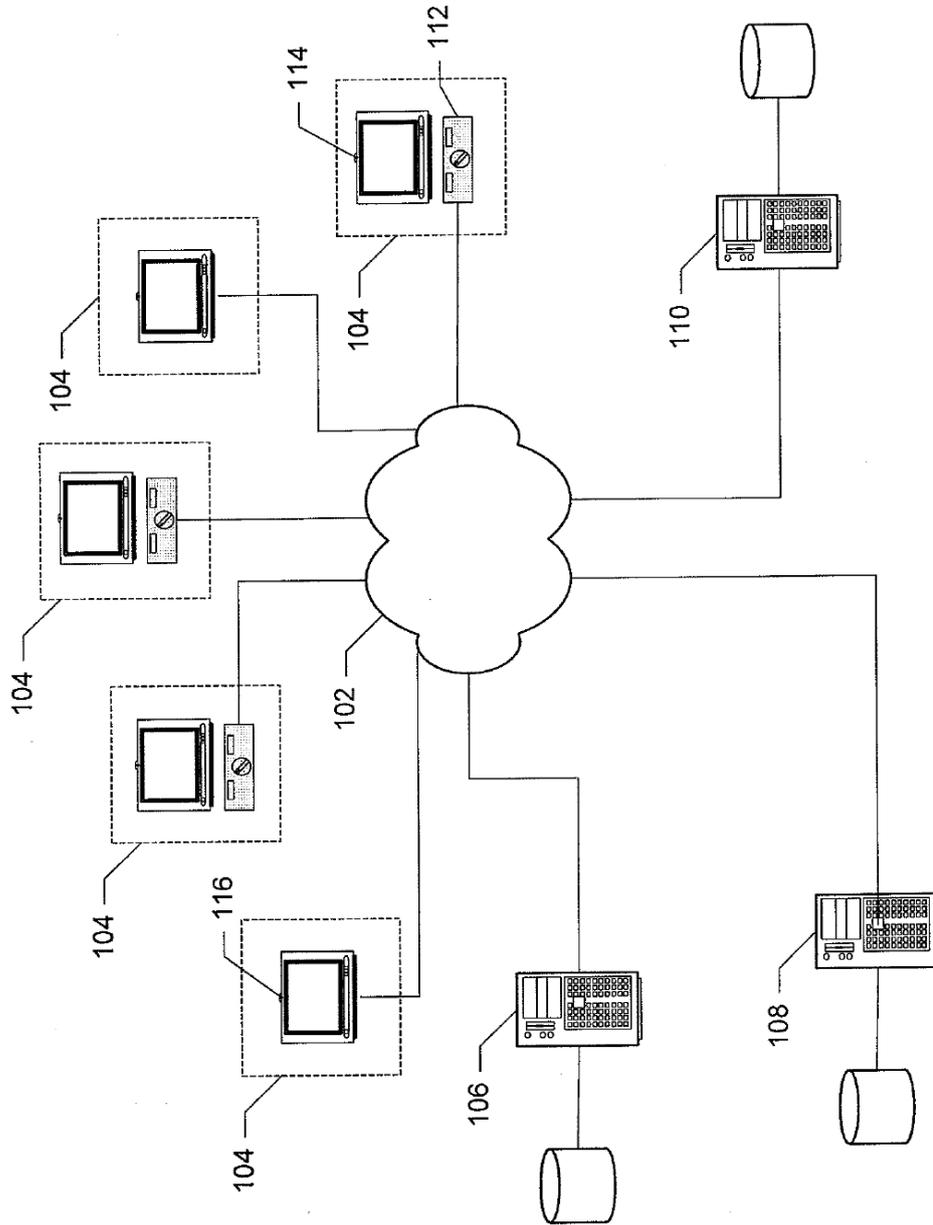


FIG. 1

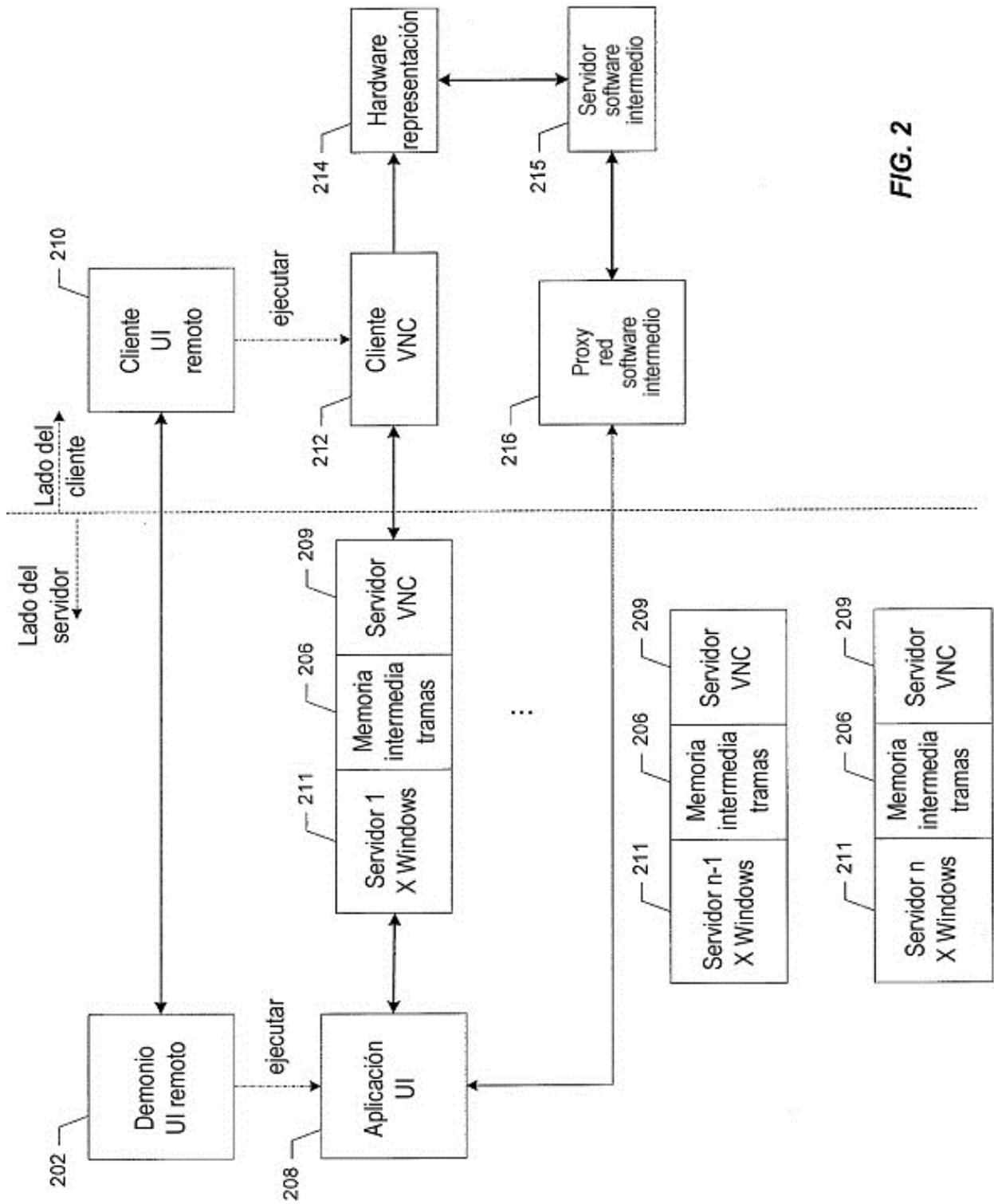


FIG. 2

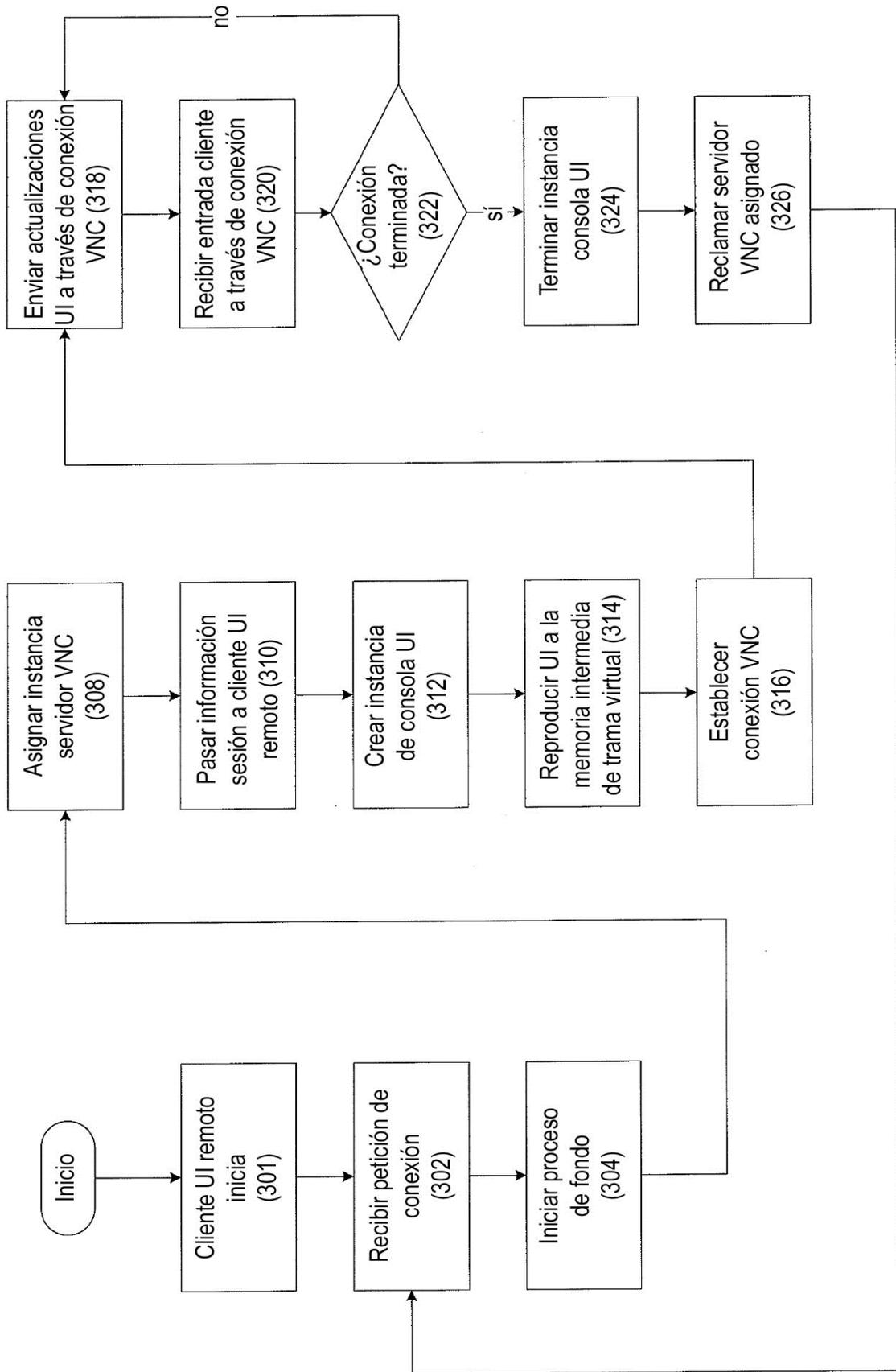


FIG. 3