



(19)



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 308 481**

(51) Int. Cl.:
H04L 29/08 (2006.01)

(12)

TRADUCCIÓN DE PATENTE EUROPEA

T3

(96) Número de solicitud europea: **05734189 .3**

(96) Fecha de presentación : **08.04.2005**

(97) Número de publicación de la solicitud: **1741271**

(97) Fecha de publicación de la solicitud: **10.01.2007**

(54) Título: **Métodos y aparato para compartir datos de pantalla gráficos de una manera adaptativa al ancho de banda.**

(30) Prioridad: **15.04.2004 US 709141**

(45) Fecha de publicación de la mención BOPI:
01.12.2008

(45) Fecha de la publicación del folleto de la patente:
01.12.2008

(73) Titular/es: **Citrix Systems, Inc.**
851 W. Cypress Creek Road
Fort Lauderdale, Florida 33309, US

(72) Inventor/es: **Schauser, Klaus, E.;**
Christiansen, Bernd;
Von Eicken, Thorsten;
Alexandrov, Albert y
Saavedra, Rafael, H.

(74) Agente: **Torner Lasalle, Elisabet**

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

Métodos y aparato para compartir datos de pantalla gráficos de una manera adaptativa al ancho de banda.

5 **Ámbito de la invención**

La presente invención se refiere generalmente a la sincronización de conjuntos de datos del nodo de origen y del nodo de consumidor, y más particularmente a técnicas y aparatos para sincronizar, de una manera adaptativa al ancho de banda, cada una de una pluralidad de representaciones de nodo de consumidor de un conjunto de datos dinámico con una representación del nodo de origen del conjunto de datos dinámico.

Antecedentes de la invención

La promesa de utilizar la red mundial de ordenadores, coloquialmente denominada Internet, para permitir que muchos individuos distintos de regiones geográficas y temporales distintas se comuniquen y colaboren en tiempo real o casi real aún permanece en gran medida sin cumplirse. Los diferentes anchos de banda de las diferentes conexiones resulta en dificultades para compartir información sensible al tiempo. El ejemplo más sencillo de este efecto es el hecho de “compartir pantalla”, es decir, actualizar las pantallas de múltiples clientes de tal manera que cada una refleje la pantalla de un servidor lo más exactamente posible. O bien las actualizaciones de pantalla quedan limitadas a la velocidad de la conexión más lenta, o bien los usuarios que se comunican sobre menor ancho de banda “se quedan atrás” respecto a los que tienen conexiones con mayor ancho de banda. Además, para que resulte útil, el sistema debe soportar simultáneamente varias fuentes de información y muchos consumidores de información, por ejemplo, una o más transmisiones de pantalla a cientos de usuarios. Asimismo, el sistema debería permitir consumidores de información que “llegan tarde”, es decir, que empiezan a recibir información de la fuente de información en un punto del tiempo posterior al del comienzo del flujo de información.

El documento US 2003/055896 describe un sistema en el que un servidor transmite a múltiples clientes parámetros de estado que definen el estado actual de la pantalla del servidor. Cada cliente compara sus parámetros de estado con los que ha recibido. Si se encuentra una discordancia, se piden al servidor nuevos datos de pantalla.

Algunos intentos de resolver este problema recurren a un servidor central que analiza el tráfico de información y enruta el tráfico entre fuentes de información y consumidores de información. Aunque estas técnicas alcanzan un éxito moderado, el uso de un servidor central para tomar decisiones en base al contenido de la información enrutada destruye la confidencialidad de la información enrutada, lo que es inaceptable.

Breve resumen de la invención

Según la presente invención se aportan un método y un sistema adaptativos al ancho de banda para sincronizar los datos de pantalla según se define en las reivindicaciones independientes que se adjuntan.

La presente invención permite una sincronización de datos de pantalla en múltiples nodos de consumidor que se adapta al ancho de banda disponible desechando los estados transitorios del conjunto de datos. El sistema utiliza plenamente el ancho de banda disponible para cada nodo de consumidor y es sencillo, eficaz y fiable. El sistema tiene también la capacidad de alojar múltiples sesiones de uno a muchos y permite que los nodos de consumidor ingresen en cualquier momento en una sesión de uno a muchos ya iniciada. El sistema también soporta el cifrado de datos de extremo a extremo.

En un aspecto la presente invención se refiere a un método adaptativo al ancho de banda para sincronizar datos de pantalla entre un nodo consumidor y un nodo origen. El nodo origen identifica un cambio en los datos de pantalla locales y crea al menos un paquete de datos que representa el cambio en los datos de pantalla locales. Desde el nodo origen se recibe información de metadatos que identifica una pluralidad de paquetes de datos que representan un estado de al menos una parte de los datos de pantalla. Al menos uno de los paquetes de datos identificados se recibe desde el nodo origen. Al menos uno de los paquetes de datos recibidos es seleccionado en respuesta a la información de metadatos recibida. La información de metadatos y el al menos un paquete de datos seleccionado son transmitidos a un nodo de consumidor.

En algunos ejemplos de realización, se recibe desde un nodo consumidor una solicitud del estado actual de los datos de pantalla cambiantes antes de que se haya seleccionado al menos uno de los paquetes de datos recibidos en respuesta a la información de metadatos recibida. En algunos de estos ejemplos, se reciben múltiples fragmentos de información de metadatos y múltiples paquetes de datos antes de que se reciba la petición del nodo consumidor. En algunos casos específicos de estos ejemplos de realización, uno de los fragmentos recibidos con información de metadatos y al menos uno de los paquetes de datos recibidos identificado por la información de metadatos seleccionada es transmitido al nodo consumidor.

En otros ejemplos de realización, se selecciona una pluralidad de paquetes de datos recibidos en respuesta a la información de metadatos recibida. En algunos de estos otros ejemplos de realización, cada paquete de la pluralidad de paquetes de datos seleccionada es transmitido al nodo consumidor. En aún otros ejemplos de realización, al menos

uno de los paquetes de datos identificados se recibe en forma cifrada, en forma comprimida o en forma cifrada y comprimida.

En otros ejemplos de realización, la información de metadatos recibida o el al menos un paquete de datos recibido se almacena en un dispositivo de memoria. En algunos de estos otros ejemplos de realización, se selecciona al menos uno de los paquetes de datos recibidos en respuesta a la información de metadatos recibida y se selecciona al menos uno de los paquetes de datos almacenados en respuesta a la información de metadatos recibida. En algunos de estos ejemplos de realización, el al menos un paquete seleccionado de los paquetes de datos recibidos y el al menos un paquete seleccionado de los paquetes de datos almacenados se transmite al nodo consumidor.

En todavía otros ejemplos de realización, la información transmitida al nodo consumidor que identifica el al menos un paquete de datos es almacenada en un elemento de memoria. En algunos de estos otros ejemplos de realización, se selecciona al menos uno de los paquetes de datos recibidos en respuesta a la información de metadatos recibida y la información almacenada que identifica el al menos un paquete de datos es transmitida al nodo consumidor.

En otro ejemplo de realización, un sistema adaptativo al ancho de banda para sincronizar datos de pantalla entre un nodo consumidor y un nodo origen incluye un nodo origen que transmite al menos un paquete de metadatos que identifica una pluralidad de paquetes de datos que representan el estado actual de los datos de pantalla cambiantes y que transmite al menos uno de los paquetes de datos identificados y un servicio de comunicaciones en comunicación con el nodo origen, donde el servicio de comunicaciones selecciona uno del al menos un paquete de metadatos y el al menos un paquete de datos para su transmisión a un primer nodo consumidor. En ciertos ejemplos de realización, el sistema incluye también un primer nodo consumidor, en el que el primer nodo consumidor le solicita el estado actual de los datos de pantalla cambiantes al servicio de comunicaciones.

En algunos de los ciertos ejemplos de realización, el servicio de comunicaciones selecciona uno del al menos un paquete de metadatos y el al menos un paquete de datos en respuesta a la solicitud enviada por el primer nodo consumidor. En otros de dichos ejemplos de realización, el sistema incluye un segundo nodo consumidor, en el que el segundo nodo consumidor le solicita al servicio de comunicaciones el estado actual del conjunto de datos cambiantes.

En otros ejemplos de realización, el sistema incluye un elemento de memoria, como un dispositivo de almacenamiento persistente. En algunos de estos otros ejemplos de realización, el servicio de comunicaciones almacena en el elemento de memoria el al menos un paquete de metadatos recibido. En otros de estos ejemplos de realización, el servicio de comunicaciones almacena en el elemento de memoria información relativa a la transmisión de paquetes a un nodo consumidor.

En aún otro aspecto, la presente invención se refiere a un servicio de comunicaciones que sincroniza representaciones de nodo consumidor con una representación de nodo origen de unos datos de pantalla cambiantes. El servicio de comunicaciones incluye un subsistema receptor, un motor de sincronización y un subsistema transmisor. El subsistema receptor recibe al menos un paquete de metadatos que identifica al menos un paquete de datos que representan el estado actual de los datos de pantalla cambiantes y al menos un paquete de datos identificado por el al menos un paquete de metadatos recibido. El motor de sincronización selecciona uno del al menos un paquete de metadatos y el al menos un paquete de datos. El subsistema de transmisión transmite el paquete seleccionado del al menos un paquete de metadatos y el al menos un paquete de datos. En algunos ejemplos de realización, el servicio de comunicaciones incluye también un elemento de memoria. En aún otros ejemplos de realización, el motor de sincronización selecciona uno del al menos un paquete de metadatos y el al menos un paquete de datos en respuesta a una petición recibida desde un nodo consumidor.

En todavía otro ejemplo de realización, la presente invención se refiere a un método adaptativo al ancho de banda para sincronizar datos de pantalla entre un nodo origen y una pluralidad de nodos consumidores. El nodo origen identifica un primer cambio en los datos de pantalla locales. Se recibe desde el nodo origen una primera información de metadatos que identifica un primer al menos un paquete de datos que representa un primer estado de datos de pantalla locales. El nodo origen identifica un segundo cambio en los datos de pantalla locales. Se recibe desde el nodo origen una segunda información de metadatos que identifica un segundo al menos un paquete de datos que representa un segundo estado de datos de pantalla locales. Se genera una tercera información de metadatos que representa la diferencia entre el primer al menos un paquete de datos identificado y el segundo al menos un paquete de datos identificado, donde la tercera información de metadatos identifica un tercer al menos un paquete de datos. La tercera información de metadatos y el tercer al menos un paquete de datos se transmiten a un nodo consumidor.

Breve descripción de las ilustraciones

Estos y otros aspectos de esta invención serán fácilmente comprensibles a partir de la siguiente descripción detallada y las ilustraciones adjuntas, que pretenden ilustrar y no limitar la invención, y en las que:

La figura 1 es una vista esquemática de un ejemplo de realización de un sistema en red que tiene múltiples nodos consumidores en comunicación con un nodo origen;

las figuras 2A y 2B son diagramas de bloques que ilustran ejemplos de realización de ordenadores utilizables en relación con la presente invención;

la figura 3 es un diagrama de bloques que ilustra el flujo de paquetes en un ejemplo de realización de una arquitectura para sincronizar conjuntos de datos entre un nodo origen y una pluralidad de nodos consumidores de una manera adaptativa al ancho de banda;

5 la figura 4 es un diagrama de bloques de un ejemplo de realización de una arquitectura para sincronizar conjuntos de datos entre un nodo origen y una pluralidad de nodos consumidores de una manera adaptativa al ancho de banda;

la figura 5 es una vista esquemática de un sistema para compartir datos de pantalla; y

10 la figura 6 es una representación esquemática de una estructura de datos útil en un sistema para compartir datos de pantalla gráficos.

Descripción detallada de la invención

15 Con referencia a la figura 1, se representa un sistema en red que tiene un nodo origen 100 en comunicación con un número de nodos consumidores 150, 150', 150". Como se muestra en la figura 1, los nodos consumidores 150, 150', 150" pueden comunicar con el nodo origen 100 a través de redes con distinto ancho de banda. En el ejemplo de realización ilustrado en la figura 1, el nodo consumidor 150 comunica con el nodo origen 100 a través de una red 160 de elevado ancho de banda, como una red de área local (LAN). El nodo consumidor 150' comunica con el nodo origen 20 a través de una red 180 de pequeño ancho de banda, como una red inalámbrica. El nodo consumidor 150" comunica con el nodo origen a través de una red 170 que tiene un ancho de banda entre la red 180 de pequeño ancho de banda y la red 160 de elevado ancho de banda, como una conexión por línea de abonado digital (DSL). Aunque en el ejemplo de realización ilustrado en la figura 1 solo se muestran un nodo origen 100 y tres nodos consumidores 150, 150', 150", debe entenderse que el sistema puede aportar una pluralidad de cualquiera o de cada uno de dichos componentes. 25 Por ejemplo, en un ejemplo de realización, el sistema incluye múltiples nodos origen 100 agrupados lógicamente, cada uno de los cuales puede estar disponible para suministrar datos a un nodo consumidor 150, 150', 150". En estos ejemplos de realización, el grupo lógico de nodos origen 100 puede denominarse "granja de servidores" o "granja de contenidos". En otros ejemplos de realización, el nodo origen 100 es un servidor multiusuario que tiene un búfer de tramas virtual, es decir, un servidor de presentaciones.

30 Las conexiones de red 160, 170, 180 entre los nodos consumidores 150, 150', 150" y el nodo origen 100 pueden ser redes de área local (LAN), redes de área metropolitana (MAN) o una red de área amplia (WAN) como Internet. El nodo origen 100 y los nodos consumidores 150, 150', 150" pueden conectarse a las redes 160, 170, 180 mediante una diversidad de conexiones como líneas telefónicas estándar, enlaces LAN o WAN (por ejemplo, T1, T3, 56 Kb, X.25), conexiones de banda ancha (RDSI, Frame Relay, ATM) o conexiones inalámbricas. Las conexiones entre el 35 nodo origen 100 y los nodos consumidores 150, 150', 150" pueden utilizar una variedad de protocolos de comunicación de capas de enlace de datos (por ejemplo, TCP/IP, IPX, SPX, NetBIOS, NetBEUI, SMB, Ethernet, ARCNET, interfaz de datos distribuidos por fibra (FDDI), RS232, IEEE 802.11, IEEE 802.11a, IEEE 802.11b, IEEE 802.11g y conexiones asíncronas directas). Aunque en la figura 1 se muestran como redes separadas, las redes 160, 170, 180 40 pueden combinarse en una sola red física.

En muchos ejemplos de realización, el nodo origen 100 y los nodos consumidores 150, 150', 150" se aportan como ordenadores personales u ordenadores servidores, del tipo fabricado por la Hewlett-Packard Corporation de Palo Alto (California) o la Dell Corporation de Round Rock (Texas). Las figuras 2A y 2B muestran diagramas de bloques de un 45 ordenador 200 típico, utilizable como nodo origen 100 y como nodos consumidores 150, 150', 150". Como se muestra en las figuras 2A y 2B, cada ordenador 200 incluye una unidad de procesamiento central 202 y una unidad de memoria principal 204. Cada ordenador 200 puede incluir también otros elementos opcionales, como uno o más dispositivos de entrada y salida 230a-230n (indicados generalmente con el número de referencia 230) y una memoria caché 240 en comunicación con la unidad de procesamiento central 202.

50 La unidad de procesamiento central 202 es cualquier conjunto de circuitos lógicos que responde a instrucciones de la unidad de memoria principal 204 y las procesa. En muchos ejemplos de realización, la unidad de procesamiento central está constituida por una unidad de microprocesador, como el procesador 8088, el 80286, el 80386, el 80486, el Pentium, Pentium Pro, Pentium II, el Celeron o el Xeon, todos ellos producidos por la Intel Corporation de Mountain View (California); el procesador 68000, el 68010, el 68020, el 68030, el 68040, el PowerPC 601, el 55 PowerPC604, el PowerPC604e, el MPC603e, el MPC603ei, el MPC603ev, el MPC603r, el MPC603p, el MPC740, el MPC745, el MPC750, el MPC755, el MPC7400, el MPC7410, el MPC7441, el MPC7445, el MPC7447, el MPC7450, el MPC7451, el MPC7455, el MPC7457, todos ellos producidos por la Motorola Corporation de Schaumburg (Illinois); el procesador Crusoe TM5800, el Crusoe TM5600, el Crusoe TM5500, el Crusoe TM5400, el Efficeon TM8600, 60 el Efficeon TM8300 o el Efficeon TM8620, producidos por la Transmeta Corporation de Santa Clara (California); el procesador RS/6000, el RS64, el RS64 II, el P2SC, el POWER3, el RS64 III, el POWER3-II, el RS64 IV, el POWER4, el POWER4+, el POWER5 o el POWER6, todos ellos producidos por International Business Machines de White Plains (Nueva York); o el procesador AMD Opteron, el AMD Athalon 64 FX, el AMD Athalon o el AMD Duron, producidos por Advanced Micro Devices de Sunnyvale (California).

65 La unidad de memoria principal 204 puede estar constituida por uno o más chips de memoria capaces de almacenar datos y de permitir que cualquier dirección de almacenamiento sea directamente accesible para el microprocesador 202, como memoria estática de acceso aleatorio (SRAM), Burst SRAM o SynchBurst SRAM (BSRAM), memo-

ria dinámica de acceso aleatorio (DRAM), DRAM en modo de paginación rápida (FPM DRAM), DRAM mejorada (EDRAM), RAM con salida de datos extendida (EDO RAM), DRAM con salida de datos extendida (EDO DRAM), DRAM con salida de datos extendida en ráfagas (BEDO DRAM), DRAM mejorada (EDRAM), DRAM síncrona (SDRAM), JEDEC SRAM, PC100 SDRAM, SDRAM de doble velocidad de datos (DDR SDRAM), SDRAM mejorada (ESDRAM), SyncLink DRAM (SLDRAM), DRAM con rambus directo (DRDRAM) o RAM ferroeléctrica (FRAM).

En el ejemplo de realización que se muestra en la figura 2A, el procesador 202 comunica con la memoria principal 204 a través de un bus de sistema (que se describe después con más detalle). La figura 2B representa un ejemplo de realización de un sistema de ordenador 200 en el que el procesador comunica directamente con la memoria principal 204 a través de un puerto de memoria. Por ejemplo, en la figura 2B la memoria principal 204 puede ser DRDRAM.

Las figuras 2A y 2B representan ejemplos de realización en los que el procesador principal 202 comunica directamente con la memoria caché 240 a través de un bus secundario, a veces denominado un bus “trasero”. En otros ejemplos de realización, el procesador principal 202 comunica con la memoria caché 240 utilizando el bus de sistema 220. La memoria caché 240 típicamente tiene un tiempo de respuesta más rápido que la memoria principal 204 y típicamente consiste en SRAM, BSRAM o EDRAM.

En el ejemplo de realización ilustrado en la figura 2A, el procesador 202 comunica con varios dispositivos 230 de entrada/salida (I/O) a través de un bus de sistema local 220. Pueden utilizarse varios buses para conectar la unidad de procesamiento central 202 con los dispositivos I/O 230, tales como un bus VESA VL, un bus ISA, un bus EISA, un bus de arquitectura microcanal (MCA), un bus PCI, un bus PCI-X, un bus PCI-Express o un NuBus. En ejemplos de realización en los que el dispositivo I/O es una pantalla de vídeo, el procesador 202 puede utilizar un puerto de gráficos avanzado (AGP) para comunicarse con la pantalla. La figura 2B representa un ejemplo de realización de un sistema de ordenador 200 en el que el procesador principal 202 comunica directamente con el dispositivo I/O 230b a través de HyperTransport, Rapid I/O o InfiniBand. La figura 2B muestra también un ejemplo de realización que combina buses locales y comunicación directa: el procesador 202 comunica con el dispositivo I/O 230a utilizando un bus de conexión local mientras que se comunica directamente con el dispositivo I/O 230b.

En el sistema de ordenador 200 puede haber presente una variedad de dispositivos I/O 230. Dispositivos de entrada pueden ser teclados, ratones, trackpads, trackballs, micrófonos y tabletas gráficas. Dispositivos de salida pueden ser pantallas de vídeo, altavoces, impresoras de chorro de tinta, impresoras láser e impresoras por sublimación. Un dispositivo I/O también puede aportar almacenamiento masivo para el sistema de ordenador 200, como un disco duro, una disquetera para recibir disquetes por ejemplo de 3,5 o de 5,25 pulgadas, discos ZIP, un lector CD-ROM, un lector grabador CD-R/RW, un lector DVD-ROM, unidades de cinta de distintos formatos y dispositivos de almacenamiento USB como la línea de dispositivos USB Flash Drive producida por Twintech Industries, Inc. de Los Alamitos (California).

En otros ejemplos de realización, un dispositivo I/O 230 puede ser un puente entre el bus de sistema 220 y un bus de comunicación externo, como un bus USB, un bus Apple Desktop, una conexión en serie RS-232, un bus SCSI, un bus FireWire, un bus FireWire 800, un bus Ethernet, un bus AppleTalk, un bus Gigabit Ethernet, un bus Modo de Transferencia Asíncrono, un bus HIPPI, un bus Super HIPPI, un bus SerialPlus, un bus SCI/LAMP, un bus FibreChannel o un bus de interfaz de sistema para pequeño ordenador conectado en serie.

Los ordenadores de sobremesa de propósito general del tipo ilustrado en las figuras 2A y 2B típicamente funcionan bajo el control de un sistema operativo, que controla la programación de tareas y el acceso a los recursos del sistema. Sistemas operativos típicos pueden ser: Microsoft Windows, producido por Microsoft Corp. De Redmond (Washington); MacOS, producido por Apple Computer de Cupertino (California); OS/2, producido por International Business Machines de Armonk (Nueva York); o Linux, un sistema operativo gratuito distribuido por Caldera Corp. De Salt Lake City (Utah), entre otros.

En algunos ejemplos de realización, el nodo consumidor 150, 150', 150'' es un dispositivo móvil, como un teléfono móvil o una agenda electrónica (PDA) provistos de JAVA, como los i50sx, i55sr, i58sr, i85s, i88s, i90c, i95c o im11000, todos ellos producidos por Motorola Corp. de Schaumburg (Illinois); el 6035 o el 7135, producidos por Kyocera de Kioto (Japón); o el i300 o el i330, producidos por Samsung Electronics Co., Ltd., de Seúl (Corea). En otros ejemplos de realización en los que el dispositivo cliente 140 es móvil, puede tratarse de una PDA como la Tungsten VV, la VII, la VIIx, la i705 o un dispositivo móvil que combina PDA y teléfono como los Treo 180, Treo 270 o Treo 600, todos ellos producidos por palmOne, Inc., de Milpitas (California).

En estos ejemplos de realización, los nodos consumidores 150, 150', 150'' conectan con el nodo origen 100 utilizando cualquiera de entre diversos protocolos conocidos de las familias GSM o CDMA, como el W-CDMA. Estos protocolos soportan servicios de comunicación inalámbrica comercial, y el W-CDMA en particular es el protocolo subyacente que soporta los servicios i-Mode y mMode, ofrecidos por NTT DoCoMo.

La figura 3 muestra un diagrama de bloques de un sistema para sincronizar un conjunto de datos entre el nodo origen 100 y una pluralidad de nodos consumidores 150, 150', 150'', así como el flujo de paquetes en un sistema durante el funcionamiento. Como se muestra en la figura 3, el sistema incluye un servicio de comunicaciones 300 con el que comunican el nodo origen 100 y la pluralidad de nodos consumidores 150, 150', 150''. El nodo origen 100 y los nodos consumidores 150, 150', 150'' pueden estar situados tras respectivos cortafuegos o firewalls (que no se muestran

en la figura 3). El nodo origen 100 y los nodos consumidores 150, 150', 150'' establecen conexiones de salida con el servicio de comunicación 300. En algunos ejemplos de realización, el servicio de comunicación 300 mantiene el estado para cada conexión de socket a un nodo consumidor 150, 150', 150'' en forma de descriptores de sockets. En otros ejemplos de realización, el nodo origen 100 y el servicio de comunicaciones 300 pueden aportarse como un mismo dispositivo físico. En estos ejemplos de realización, el nodo origen 100 y el servicio de comunicaciones 300 operan sobre el mismo hardware en un modo de tiempo compartido. Es posible compartir datos entre el nodo origen 100 y el servicio de comunicación 300 utilizando cualquiera de entre diversas técnicas, como objetos pipe o memoria compartida.

El nodo origen 100 codifica el estado actual de un conjunto de datos dinámicos, como datos de pantalla, en forma de un conjunto de paquetes de datos. En algunos ejemplos de realización, este proceso de codificación es directo. Por ejemplo, en el caso en que el conjunto de datos dinámicos es de datos de pantalla, pueden codificarse paquetes de datos almacenando en el paquete de datos los valores de píxel para una parte predeterminada de la pantalla. En algunos ejemplos de realización, el nodo origen 100 comprime los datos almacenados en los paquetes de datos. En otros ejemplos de realización, el nodo origen 100 cifra los datos almacenados en los paquetes de datos. En aún otros ejemplos de realización, el nodo origen 100 cifra y comprime los datos almacenados en los paquetes de datos. Conforme el conjunto de datos dinámico va cambiando, el nodo origen actualiza el conjunto de paquetes de datos que componen el estado actual del conjunto de datos.

El nodo origen 100 transmite el estado actual del conjunto de datos dinámico al servicio de comunicación 300 de una manera adaptativa al ancho de banda. En un ejemplo de realización, esto se consigue requiriendo que el nodo origen 100 posea un testigo de transmisión antes de iniciar la transmisión del estado actual del conjunto de datos. En este ejemplo de realización, el nodo origen 100 y el servicio de comunicación intercambian un número limitado de testigos de transmisión, por ejemplo cinco. En otros ejemplos de realización, el servicio de comunicación 300 transmite un mensaje al nodo origen 100 para indicar al nodo origen 100 cuándo puede enviar otra actualización del conjunto de datos.

Como se muestra en la figura 3, el servicio de comunicación 300 también puede incluir un elemento 310 para almacenamiento de datos, como una memoria de acceso aleatorio, una unidad de disco, un array de discos, una unidad óptica de lectura y escritura o cualquier otra forma de elemento de memoria que permita acceder a los datos almacenados. El elemento de almacenamiento 310 permite que el servicio de comunicación 300 almacene información sobre metadatos y paquetes de datos recibidos del nodo origen 100 entre peticiones de actualización desde distintos nodos consumidores 150, 150', 150''. Además, el elemento de almacenamiento 310 se puede utilizar para mantener un registro histórico de información de metadatos y paquetes de datos transmitidos desde el nodo origen 100. En otros ejemplos de realización, el elemento de almacenamiento 310 también puede almacenar los paquetes de datos transmitidos a los nodos consumidores respectivos 150, 150', 150''.

El nodo origen 100 crea información de metadatos que identifica cada uno de los paquetes de datos que representan el estado actual del conjunto de datos dinámicos. En el ejemplo de realización que se muestra en la figura 3, la información de metadatos comprende un paquete de metadatos 310, 320, 330. El paquete de metadatos 310 se crea en el momento de tiempo t1 e indica que el estado del conjunto de datos dinámicos en el tiempo t1 viene representado por el paquete de datos 0, el paquete de datos 1 y el paquete de datos 2. De forma semejante, el paquete de metadatos 330 indica que el estado del conjunto de datos dinámicos en el tiempo t2 viene representado por el paquete de datos 0, el paquete de datos 4 y el paquete de datos 5. En otros ejemplos de realización, en lugar de crear paquetes de metadatos que almacenan información de metadatos, la información de metadatos está incluida en los paquetes de datos. Por ejemplo, cada paquete de datos que comprende una actualización del conjunto de datos puede incluir un "encabezamiento de información de metadatos" que identifica el conjunto de actualización al que está asociado el paquete de datos.

Como se muestra en la figura 3, el nodo origen 100 transmite información de metadatos 310 al servicio de comunicación 300 seguida por los paquetes de datos identificados por la información de metadatos 310. Así, el nodo origen 100 transmite al servicio de comunicación 300 el paquete de datos 0 312, el paquete de datos 1 314 y el paquete de datos 2 316 a continuación del paquete de metadatos 310. En el momento de tiempo t2, el nodo origen 100 transmite al servicio de comunicación 300 el paquete de metadatos 320, que indica que el estado del conjunto de datos en el tiempo t2 viene representado por el paquete de datos 0, el paquete de datos 3 y el paquete de datos 4. El nodo origen 100 a continuación transmite el paquete de datos 3 322 y el paquete de datos 4 334 al servicio de comunicación 300. El nodo origen 100 no retransmite el paquete de datos 0 al servicio de comunicación 300 ya que ese paquete de datos se transmitió en relación con el primer paquete de metadatos 310. De forma semejante, en el tiempo t3 el nodo origen 100 transmite al servicio de comunicación 300 un paquete de metadatos 330 que indica que el estado actual del conjunto de datos está representado por el paquete de datos 0, el paquete de datos 4 y el paquete de datos 5. Como el nodo origen 100 ya transmitió el paquete de datos 0 al servicio de comunicación 300 después del primer paquete de metadatos 310 y el paquete de datos 4 después del segundo paquete de metadatos 320, el nodo origen 100 solo transmite el paquete de datos 5 332 después del tercer paquete de metadatos 330.

Como se ha descrito ya con relación al control de flujo entre el nodo origen 100 y el servicio de comunicación 300, el control de flujo entre los nodos consumidores 150, 150', 150'' y el servicio de comunicación 300 puede estar basado en testigos o estar basado en mensajes. Para facilitar la referencia, en el resto de la descripción se supondrá que el método para el control del flujo está basado en mensajes. Sin embargo, se pueden obtener las mismas ventajas de la invención con un sistema que utilice testigos de transmisión.

La figura 3 muestra un ejemplo de realización de un sistema en el que un nodo consumidor 150 comunica con el servicio de comunicación 300 mediante una conexión de gran ancho de banda. En este caso, el nodo consumidor 150 solicita actualizaciones del conjunto de datos con la frecuencia suficiente para que el servicio de comunicación 300 transmita al nodo consumidor 150 una corriente de información de metadatos y paquetes de datos idéntica a la corriente de información de metadatos y paquetes recibida por el servicio de comunicación 300 desde el nodo origen 100. Como también se muestra en la figura 3, el nodo consumidor 150", que comunica con el servicio de comunicación 300 mediante una conexión de escaso ancho de banda, solicita actualizaciones del conjunto de datos con menos frecuencia y, por tanto, recibe del servicio de comunicación 300 una corriente de paquetes diferente de la que el servicio de comunicación 300 recibe del nodo origen 100. Como se muestra en la figura 3, el servicio de comunicación 300 transmite el primer paquete de metadatos 310 y los paquetes de datos 0-3, 312, 314, 316 al nodo consumidor 150". El siguiente paquete de metadatos recibido por el nodo consumidor 150" es el tercer paquete de metadatos 330, que indica que el estado del conjunto de datos dinámicos está representado por el paquete de datos 0, el paquete de datos 4 y el paquete de datos 5. Como el nodo consumidor 150" aún no ha recibido el paquete de datos 4 ni el paquete de datos 5, el servicio de comunicación 300 transmite estos paquetes de datos al nodo consumidor 150".

La figura 3 representa también el flujo de paquetes enviado a un nodo consumidor que "se suma tarde". Como se muestra en la figura 3, un consumidor que se añade en el tiempo t3 recibe el tercer paquete de metadatos 330 así como los paquetes de datos identificados por el tercer paquete de metadatos. Los paquetes de datos transmitidos al nodo consumidor 150, 150' 150" por el servicio de comunicación 300 pueden ser recuperados del elemento de almacenamiento 310, recién recibidos del nodo origen 100 o alguna combinación de ambas cosas.

La entrega de actualizaciones del conjunto de datos desde el servicio de comunicación 300 puede realizarse utilizando un modelo "push", un modelo "pull" o un modelo de "push atómico". En los modelos "push", el servicio de comunicación 300 transmite información de metadatos y paquetes de datos a los nodos consumidores 150, 150', 150". La diferencia entre el modelo "push" y el modelo "push atómico" es que, en el modelo de "push atómico", el servicio de comunicación 300 se compromete a transmitir todos los paquetes de datos identificados mediante la información de metadatos transmitida antes de empezar la transmisión de otro conjunto de datos. Este compromiso no existe en el modelo "push", lo cual significa que los paquetes de datos que no se transmitieron con éxito en una actualización anterior del conjunto de datos pueden no ser enviados nunca al nodo consumidor 150, 150', 150". En el modelo "pull", el nodo consumidor 150, 150', 150" recibe del servicio de comunicación 300 la información de metadatos y a continuación solicita paquetes de datos específicos al servicio de comunicación 300.

En ciertos ejemplos de realización, la información contenida en los paquetes de metadatos se cifra incrementalmente. En estos ejemplos de realización, las representaciones de "cable" de los paquetes de metadatos pueden diferir a pesar de que cifran la misma información. Un breve ejemplo muestra por qué es así. A lo largo del tiempo, el nodo origen 100 envía tres paquetes de metadatos al servicio de comunicación 300. El contenido de los paquetes de metadatos consiste en conjuntos de números de paquetes de datos (1, 2, 3), (2, 3, 4) y (3, 4, 5). En el "cable", cada conjunto se representa como una delta del conjunto anterior. Así, el nodo origen 100 transmite los siguientes paquetes de metadatos al servicio de comunicación 300: (1, 2, 3), (-1, +4) y (-2, +5), donde "-" y "+" indican la supresión o la adición de un número de paquete del/al conjunto anterior. Si un nodo consumidor 150 se salta el contenido del segundo paquete de metadatos, recibe información que describe los conjuntos (1, 2, 3) y (3, 4, 5). En el "cable" estos dos conjuntos se representan incrementalmente como (1, 2, 3) y (-1, +4, -2, +5). Mientras que el nodo origen 100 transmitió el contenido del segundo paquete de metadatos al servicio de comunicación 300 como (-2, +5), el servicio de comunicación 300 transmitió la misma información al nodo consumidor 150 como (-1, +4, -2, +5).

La figura 4 ilustra otro ejemplo de realización de un sistema para sincronizar un conjunto de datos entre un nodo origen 100 y uno o más nodos consumidores 150, 150', 150" que incluye múltiples servicios de comunicación 300, 300', 300" (denominados en general 300). Como se muestra en la figura 4, el nodo origen comunica con más de un servicio de comunicación 300. De forma semejante, cada nodo consumidor 150, 150', 150" también puede comunicar con uno o más servicios de comunicación 300. Los servicios de comunicación 300 también comunican entre sí de manera *peer-to-peer*.

En este ejemplo de realización, cada par de servicios de comunicación 300 determina mutuamente una dirección para el flujo de datos. Por ejemplo, el servicio de comunicación 300 y el servicio de comunicación 300" pueden acordar entre sí que, para los fines de su enlace punto a punto, el servicio de comunicación 300 es el "emisor" y el servicio de comunicación 300" es el "receptor", lo que quiere decir que el "emisor" cumplirá la función del servicio de comunicación 300 descrito en relación a la figura 3 y el "receptor" cumplirá la función del nodo consumidor 150 descrito en relación a la figura 3. El servidor de comunicación 300', sin embargo, cumplirá la función de "emisor" cuando se comunique con los nodos consumidores 150, 150', 150".

La presente invención puede aportarse como uno o más programas legibles por un ordenador incorporados en uno o más artículos manufacturados. El artículo manufacturado puede ser un disquete, un disco duro, un disco compacto, un disco digital versátil, una tarjeta de memoria flash, una PROM, una RAM, una ROM o una cinta magnética. En general, los programas legibles por ordenador pueden implementarse en cualquier lenguaje de programación. Ejemplos de lenguajes susceptibles de ser utilizados pueden ser C, C++, C# o Java. Los programas de software pueden almacenarse en uno o más artículos manufacturados como código objeto.

Ejemplos

Los siguientes ejemplos de sistemas para compartir contenido se aportan para ilustrar las diversas maneras en que los sistemas y métodos descritos pueden utilizarse y en modo alguno para limitar el alcance de la invención descrita.

Ejemplo 1

Los sistemas y métodos descritos pueden utilizarse para implementar un sistema para compartir datos de pantalla que permite a varias máquinas cliente presentar los datos de pantalla de un solo servidor. El sistema es útil en una variedad de contextos de difusión o “multicast” y, en particular, es útil en un contexto de conferencia para permitir que múltiples individuos visualicen los mismos datos gráficos durante la conferencia.

La figura 5 ilustra el diagrama de un sistema para compartir datos de pantalla. Como se muestra en la figura 5, un servidor host 100 monitoriza su estado de pantalla. En el ejemplo de realización que se ilustra en la figura 5, el servidor host 100 subdivide su pantalla en 12 cuadros, aunque puede utilizarse cualquier número de cuadros para representar completamente la pantalla del servidor host 100. En algunos ejemplos de realización, los cuadros son todos del mismo tamaño, es decir, cada cuadro representa el mismo número de píxeles de pantalla. En otros ejemplos de realización, como el que se ilustra en la figura 5, algunos de los cuadros tienen un tamaño distinto al de otros cuadros. En aún otros ejemplos de realización, un cuadro puede solaparse con otro cuadro o, como se muestra en la figura 5, puede haber cuadros que no se solapan.

Como se muestra en la figura 5, la pantalla anterior 500 del servidor host es representada por un primer conjunto de cuadros (que no se muestra), codificados como un primer conjunto de paquetes de datos: 13, 14, 3, 4, 15, 6, 7, 8, 17, 10, 11 y 12. Si el servidor host 100 posee un testigo de transmisión, transmite estos doce paquetes de datos al servidor de comunicaciones 200, como se ha descrito antes.

En un segundo instante de tiempo, la pantalla 510 del servidor host ha cambiado. El servidor host 100 identifica los cuadros en particular que han cambiado de estado y crea un paquete codificado para cada cuadro que ha cambiado, es decir, los paquetes de datos 19, 20, 21 y 22. Si el servidor host 100 no poseía un testigo de transmisión pero ahora recibe uno, el servidor host 100 transmitirá los doce paquetes de datos actualizados al servidor de comunicación 200, es decir, los paquetes de datos 13, 14, 3, 4, 15, 19, 20, 17, 21, 22 y 12. Si el servidor host ya ha transmitido los paquetes de datos que representan el estado de la pantalla 500, a continuación el servidor host solo necesita transmitir al servidor de comunicaciones 200 los paquetes de datos 19, 20, 21 y 22. De esta manera, la transmisión de datos de pantalla entre el servidor host 100 y el servidor de comunicaciones 200 se realiza de una manera adaptativa al ancho de banda.

En algunos ejemplos de realización, el servidor host 100 cifra los paquetes de datos transmitidos al servidor de comunicaciones 200. En otros ejemplos de realización, el servidor host 100 comprime los paquetes de datos transmitidos al servidor de comunicaciones 200. En aún otros ejemplos de realización el servidor host 100 cifra y comprime a la vez los paquetes de datos.

En muchos ejemplos de realización, el servidor de comunicaciones 200 mantiene una copia de cada cuadro que comprende el estado más reciente de la pantalla del nodo servidor. En algunos ejemplos de realización, cada cuadro se asocia con una marca de tiempo (*timestamp*) cuando se transmite al servicio de comunicaciones 200. En otros ejemplos de realización, cada cuadro se asocia con un número que se incrementa de manera monótona con cada nuevo cuadro que se transmite al servicio de comunicación 200.

El servidor de comunicación 200 compone una actualización para un nodo visualizador 150 con tanta frecuencia como lo permite el ancho de banda de la red que conecta el nodo visualizador 150 con el servidor de comunicaciones 200. Como se muestra en la figura 5, la pantalla 520 del visualizador presenta datos de pantalla de un instante de tiempo previo a la pantalla anterior del host 500. Es decir, los datos de pantalla del servidor host han cambiado dos veces (representado por la pantalla 500 y la pantalla 510) desde la última vez que el nodo visualizador 150 solicitó una actualización. El array de paquetes de datos 570 muestra los paquetes de datos que comprenden los datos de pantalla presentados en el momento actual por el nodo visualizador 150. El array de paquetes de datos 590 representa los paquetes de datos que el servidor de comunicaciones 200 debe transmitir al nodo visualizador 150 para actualizar la pantalla 520 del visualizador al estado de la pantalla 510 del host. Como se ha descrito antes, el servidor de comunicaciones 200 transmite al nodo visualizador 150 información de metadatos que identifica ocho paquetes de datos: los paquetes de datos 13, 14, 15, 19, 20, 17, 21 y 22. En algunos ejemplos de realización, la información de metadatos identifica explícitamente qué cuadro sustituye a qué otro cuadro, quizá describiendo la posición del nuevo cuadro. El servidor de comunicaciones 200 a continuación transmite al nodo visualizador los paquetes que representan los nuevos cuadros.

En otro ejemplo de realización, el servicio de comunicación 200 responde a una petición de actualización del nodo visualizador 150 enviando al nodo visualizador 150 todos los paquetes de datos que tienen una marca de tiempo más reciente que la marca de tiempo de la pantalla del visualizador. En algunos de estos ejemplos de realización, el servicio de comunicaciones 200 no recibe y almacena completamente un conjunto de paquetes de datos que componen una actualización de pantalla antes de enviar la actualización al nodo visualizador 150. En estos ejemplos de realización,

el servicio de comunicación 300 marca con el mismo valor de marca de tiempo todos los paquetes cuya información de metadatos identifica como parte de la actualización de pantalla. Después, a medida que van llegando paquetes de datos, el servidor de comunicación los envía al nodo visualizador 150.

5 En un ejemplo de realización particular, la información de metadatos se formatea en paquetes y los paquetes de metadatos se asocian con números monótonamente crecientes. Como se ha descrito antes, cada paquete de metadatos describe el conjunto de cuadros que representa el estado actual de la pantalla. En este ejemplo de realización, el servicio de comunicación 300 almacena, para cada nodo consumidor 150, el número del último paquete de metadatos que se ha transmitido a ese nodo consumidor 150, así como el conjunto de todos los paquetes de datos que se han
10 enviado al nodo consumidor. Cuando el servicio de comunicación 300 determina que ha llegado el momento de enviar una actualización a un nodo consumidor 150, o bien al recibir una petición de actualización de pantalla desde un nodo consumidor 150, el servicio de comunicación determina primero si el último paquete de metadatos (es decir, el paquete de metadatos que tiene asociado el número más alto) se ha transmitido ya al nodo consumidor 150. Si no es así, el servicio de comunicación 300 transmite al nodo consumidor 150 el paquete de metadatos más reciente. El servicio
15 de comunicación 300 transmite también el conjunto de paquetes de datos identificados por el paquete de metadatos, a menos que un paquete de datos en particular ya se haya transmitido antes al nodo consumidor 150.

En otro ejemplo de realización, el conjunto de cuadros (es decir, paquetes de datos) que debe transmitirse a un nodo consumidor se computa asociando cada cuadro con una marca de tiempo e identificando todos los cuadros visibles
20 cuyas marcas de tiempo sean más recientes que el cuadro más reciente ya recibido por el nodo consumidor 150. La figura 6 es un diagrama que representa una estructura de datos que permite la sustitución eficaz de un cuadro de pantalla por el servicio de comunicación 300 y, dada una marca de tiempo, la identificación de los cuadros que están desfasados con respecto a un nodo consumidor 150 determinado. Todos los cuadros que componen una pantalla 610 se almacenan en una lista 620 doblemente enlazada, ordenada por la marca de tiempo e indexada por la posición del
25 cuadro en la pantalla. Como se muestra en la figura 6, cuando un cuadro nuevo 29 sobrescribe un cuadro viejo 19, el cuadro 19 es borrado de la lista y el cuadro 29 se inserta en la cabeza de la lista. Cuando un nodo visualizador 150 solicita una actualización de pantalla, el servicio de comunicación 300 recorre iterativamente la lista de cuadros 620 y transmite cuadros al nodo visualizador hasta que encuentra un cuadro con una marca de tiempo más antigua que el cuadro más reciente que hay en la pantalla del nodo visualizador.

30 Ejemplo 2

En otro ejemplo, los sistemas y métodos de sincronización descritos se utilizan para implementar un sistema de chat. En este sistema, un participante en el chat añade texto u otro contenido a una sesión en curso e identifica el
35 contenido añadido como un paquete de datos. En un ejemplo de realización, el participante también asocia una marca de tiempo al contenido añadido. A continuación el participante transmite información de metadatos que identifica el estado actual del chat. En un ejemplo de realización, la información de metadatos identifica el estado actual del chat como el paquete recién añadido junto con todos los anteriores paquetes de datos añadidos al chat.

40 El participante transmite la información de metadatos juntamente con un paquete de datos que representa el contenido recién añadido. En un ejemplo de realización, la información de metadatos y el paquete de datos se envían a un nodo receptor, como se ha descrito antes. Los receptores de la información de metadatos y el paquete de datos combinan el paquete de datos recibido con los paquetes de datos del chat anteriormente recibidos en el orden en que se reciben los paquetes de datos. En otro ejemplo de realización, el receptor combina los paquetes de datos del chat
45 en base al momento de tiempo en que se envió el paquete de datos. En aún otro ejemplo de realización, el receptor combina los paquetes de datos en base a la marca de tiempo asociada con el paquete de datos.

Un usuario “atrasado” que se incorpore tarde a la sesión de chat recibirá información de metadatos que identifica todos los paquetes de datos que representan la sesión de chat. El usuario atrasado solicitará (es decir, “pull”) o bien se
50 le enviarán (es decir, “push”) todos los paquetes de datos identificados por la información de metadatos y los presentará en pantalla según el orden de las marcas de tiempo.

Ejemplo 3

55 En otro ejemplo, los sistemas y métodos de sincronización descritos pueden utilizarse para implementar un sistema de presentación remota. En este ejemplo, un presentador convierte una presentación de diapositivas en una serie de imágenes página por página. Cuando el presentador muestra una diapositiva, la imagen de la página que representa esa diapositiva se transmite a todos los nodos visualizadores. En muchos ejemplos de realización, cada diapositiva es representada por múltiples paquetes de datos.

60 En este ejemplo, el presentador envía por “push atómico” la diapositiva actualmente presentada enviando por push atómico información de metadatos que identifica cada paquete de datos que representa la diapositiva y transmitiendo en modo push todos los paquetes de datos aún no enviados al receptor. El presentador también puede enviar en modo push la diapositiva anterior y la siguiente. En otros ejemplos de realización, los nodos visualizadores pueden recibir
65 en modo “pull” otras páginas de la presentación, si se dispone de ancho de banda adicional. La información que puede insertarse en un paquete de datos de puntero láser incluye la coordenada x, la coordenada y, el tiempo, el documento o el número de página.

ES 2 308 481 T3

Además de transmitir vía multicast la presentación de diapositivas, este ejemplo de realización de la invención puede utilizarse para compartir otros documentos basados en páginas. Este ejemplo de realización también puede incorporar una característica de “puntero láser” por la cual la posición de un “puntero láser” del presentador también es transmitida a todos los receptores, permitiendo que el presentador dirija la atención de los receptores hacia áreas de interés en el documento.

Ejemplo 4

En aún otro ejemplo, los sistemas y métodos de sincronización antes descritos pueden utilizarse para implementar un sistema que permite a múltiples usuarios hacer anotaciones en un documento. En este ejemplo, cada anotación es representada por un paquete de datos. Los paquetes de datos de cada anotación pueden incluir información relativa al momento en que se hizo la anotación y quién la hizo. Otra información del paquete de datos de la anotación puede incluir el documento en el que se hizo la anotación, el número de página del documento en el que se hizo la anotación, la pluma utilizada para hacer la anotación, la coordenada x de la anotación o la coordenada y de la anotación.

En este ejemplo, la información de metadatos identifica todos los paquetes de datos de la anotación. De esta manera, alguien que “llegue tarde” recibirá todas las anotaciones hechas en el documento.

Aunque la invención se ha ilustrado y descrito con referencia a ejemplos de realización preferentes y específicos, los expertos en la técnica comprenderán que es posible introducir en la misma diversos cambios de forma y de detalle sin apartarse del ámbito de la invención según se describe en las reivindicaciones adjuntas.

REIVINDICACIONES

1. Un método adaptativo al ancho de banda para sincronizar datos de pantalla entre un nodo origen (100) y una pluralidad de nodos consumidores (150, 150', 150''), cuyo método comprende los pasos de:
 - (a) identificar, por parte del nodo origen (100), un cambio en los datos de pantalla locales;
 - (b) crear, por parte del nodo origen (100), al menos un paquete de datos que represente el cambio en los datos de pantalla locales;
 - (c) recibir, por un servicio de comunicación (300) desde el nodo origen (100), información de metadatos que identifique al menos un paquete de datos que represente un estado actual de los datos de pantalla locales después del cambio de los datos de pantalla locales;
 - (d) recibir, por dicho servicio de comunicación (300) desde el nodo origen (100), al menos uno de los paquetes de datos identificados;
 - (e) seleccionar por dicho servicio de comunicación (300) al menos uno de los paquetes de datos recibidos en respuesta a la información de metadatos recibida;
 - (f) transmitir, por dicho servicio de comunicación (300) a un nodo consumidor (150, 150', 150''), la información de metadatos; y
 - (g) transmitir, por dicho servicio de comunicación (300) al nodo consumidor (150, 150', 150''), el al menos un paquete de datos seleccionado.
2. El método de la reivindicación 1, comprendiendo además el paso de, antes del paso (e), recibir una petición del estado actual de los datos de pantalla locales del nodo origen (100) por parte de un nodo consumidor (150, 150', 150'').
3. El método de la reivindicación 2, comprendiendo además el paso de repetir los pasos de (a) a (d) hasta que se recibe la petición del estado actual de los datos de pantalla locales del nodo origen por parte de un nodo consumidor (150, 150', 150'').
4. El método de la reivindicación 3, en el que el paso (e) comprende los pasos de: (e-a) seleccionar una de la información de metadatos recibida; y (e-b) seleccionar al menos uno de los paquetes de datos recibidos identificado por la información de metadatos seleccionada.
5. El método de la reivindicación 1, en el que el paso (e) comprende seleccionar una pluralidad de los paquetes de datos recibidos en respuesta a la información de metadatos recibida.
6. El método de la reivindicación 5, en el que el paso (f) comprende transmitir a un nodo consumidor (150, 150', 150'') cada uno de la pluralidad de paquetes de datos seleccionados.
7. El método de la reivindicación 1, en el que el paso (d) comprende recibir del nodo origen (100) al menos uno de los paquetes de datos identificados en forma cifrada.
8. El método de la reivindicación 1, comprendiendo además el paso de almacenar en un dispositivo de memoria la información de metadatos recibida.
9. El método de la reivindicación 1, comprendiendo además el paso de almacenar en un dispositivo de memoria el al menos un paquete de datos recibido.
10. El método de la reivindicación 9, en el que el paso (e) comprende: (e-a) seleccionar al menos uno de los paquetes de datos recibidos en base a la información de metadatos recibida; y (e-b) seleccionar al menos uno de los paquetes de datos almacenados en base a la información de metadatos recibida.
11. El método de la reivindicación 10, en el que el paso (g) comprende: (g-a) transmitir al nodo consumidor (150, 150', 150'') el al menos un paquete de datos seleccionado de los paquetes de datos recibidos; y (g-b) transmitir al nodo consumidor (150, 150', 150'') el al menos un paquete de datos seleccionado de los paquetes de datos almacenados.
12. El método de la reivindicación 1, comprendiendo además el paso de almacenar en un dispositivo de memoria información para identificar el al menos un paquete de datos transmitido al nodo consumidor (150, 150', 150'').
13. El método de la reivindicación 12, comprendiendo además el paso de seleccionar al menos uno de los paquetes de datos recibidos en base a la información de metadatos recibida y la información almacenada que identifica al menos un paquete de datos transmitido al nodo consumidor (150, 150', 150'').

ES 2 308 481 T3

14. Un sistema adaptativo al ancho de banda para sincronizar datos de pantalla entre un nodo origen (100) y una pluralidad de nodos consumidores (150, 150', 150''), cuyo sistema comprende:

el nodo origen (100) configurado para identificar un cambio en los datos de pantalla locales, crear al menos un paquete de datos que represente el cambio, crear al menos un paquete de metadatos que identifique el al menos un paquete de datos que representa el cambio en los datos de pantalla locales y transmitir el al menos un paquete de metadatos y el al menos uno de los paquetes de datos identificados a un servicio de comunicación (300); y

el servicio de comunicación (300) configurado para seleccionar uno del al menos un paquete de metadatos y el al menos un paquete de datos para su transmisión a un primer nodo consumidor (150, 150', 150'').

15. El sistema de la reivindicación 14, comprendiendo además un primer nodo consumidor (150, 150', 150'') configurado para solicitar al servicio de comunicación (300) el estado actual de los datos de pantalla locales del nodo origen.

16. El sistema de la reivindicación 15, en el que el servicio de comunicación (300) está configurado además para seleccionar uno de los al menos un paquete de metadatos y el al menos un paquete de datos en respuesta a la petición realizada por el primer nodo consumidor (150, 150', 150'').

17. El sistema de la reivindicación 15, comprendiendo además un segundo nodo consumidor (150, 150', 150'') configurado para solicitar al servicio de comunicación (300) el estado actual de los datos de pantalla locales del nodo origen.

18. El sistema de la reivindicación 17, en el que el nodo origen (100) está configurado además para transmitir una pluralidad de paquetes de metadatos, donde cada uno de los paquetes de la pluralidad de paquetes de metadatos representa un estado de los datos de pantalla locales del nodo origen.

19. El sistema de la reivindicación 18, en el que el servicio de comunicación (300) está configurado además para seleccionar un primer paquete de metadatos para su transmisión al primer nodo consumidor (150, 150', 150'') y un segundo paquete de metadatos para su transmisión al segundo nodo consumidor (150, 150', 150'').

20. El sistema de la reivindicación 14, en el que el servicio de comunicación (300) comprende además un elemento de memoria.

21. El sistema de la reivindicación 20, en el que el elemento de memoria es un dispositivo de almacenamiento permanente.

22. El sistema de la reivindicación 20, en el que el servicio de comunicación (300) está configurado además para almacenar el al menos un paquete de metadatos recibido en el elemento de memoria.

23. El sistema de la reivindicación 20, en el que el servicio de comunicación (300) está configurado además para almacenar el al menos un paquete de datos recibido en el elemento de memoria.

24. El sistema de la reivindicación 20, en el que el servicio de comunicación (300) está configurado además para almacenar en el elemento de memoria información relativa a la transmisión de paquetes a un nodo consumidor (150, 150', 150'').

25. El sistema de la reivindicación 14, en el que el nodo origen (100) está configurado además para cifrar el al menos un paquete de datos antes de su transmisión al nodo consumidor (150, 150', 150'').

26. Un método adaptativo al ancho de banda para sincronizar datos de pantalla entre un nodo origen (100) y una pluralidad de nodos consumidores (150, 150', 150''), cuyo método comprende los pasos de:

(a) identificar, por parte del nodo origen (100), un primer cambio en los datos de pantalla locales;

(b) recibir, por un servicio de comunicación (300) desde el nodo origen (100), una primera información de metadatos que identifique al menos un primer paquete de datos que represente un primer estado de los datos de pantalla locales después del primer cambio de los datos de pantalla locales;

(c) identificar, por parte del nodo origen (100), un segundo cambio en los datos de pantalla locales;

(d) recibir, por parte del servicio de comunicación (300) desde el nodo origen (100), una segunda información de metadatos que identifique al menos un segundo paquete de datos que represente un segundo estado de los datos de pantalla locales después del segundo cambio de los datos de pantalla locales;

(e) generar, por parte del servicio de comunicación (300), una tercera información de metadatos que represente la diferencia entre el primer al menos un paquete de datos identificado y el segundo al menos un paquete de datos identificado, donde la tercera información de metadatos identifica al menos un tercer paquete de datos;

ES 2 308 481 T3

(f) transmitir, por dicho servicio de comunicación (300) a un nodo consumidor, la tercera información de metadatos; y

5 (g) transmitir, por dicho servicio de comunicación (300) al nodo consumidor (150, 150', 150''), el tercer al menos un paquete de datos.

10

15

20

25

30

35

40

45

50

55

60

65

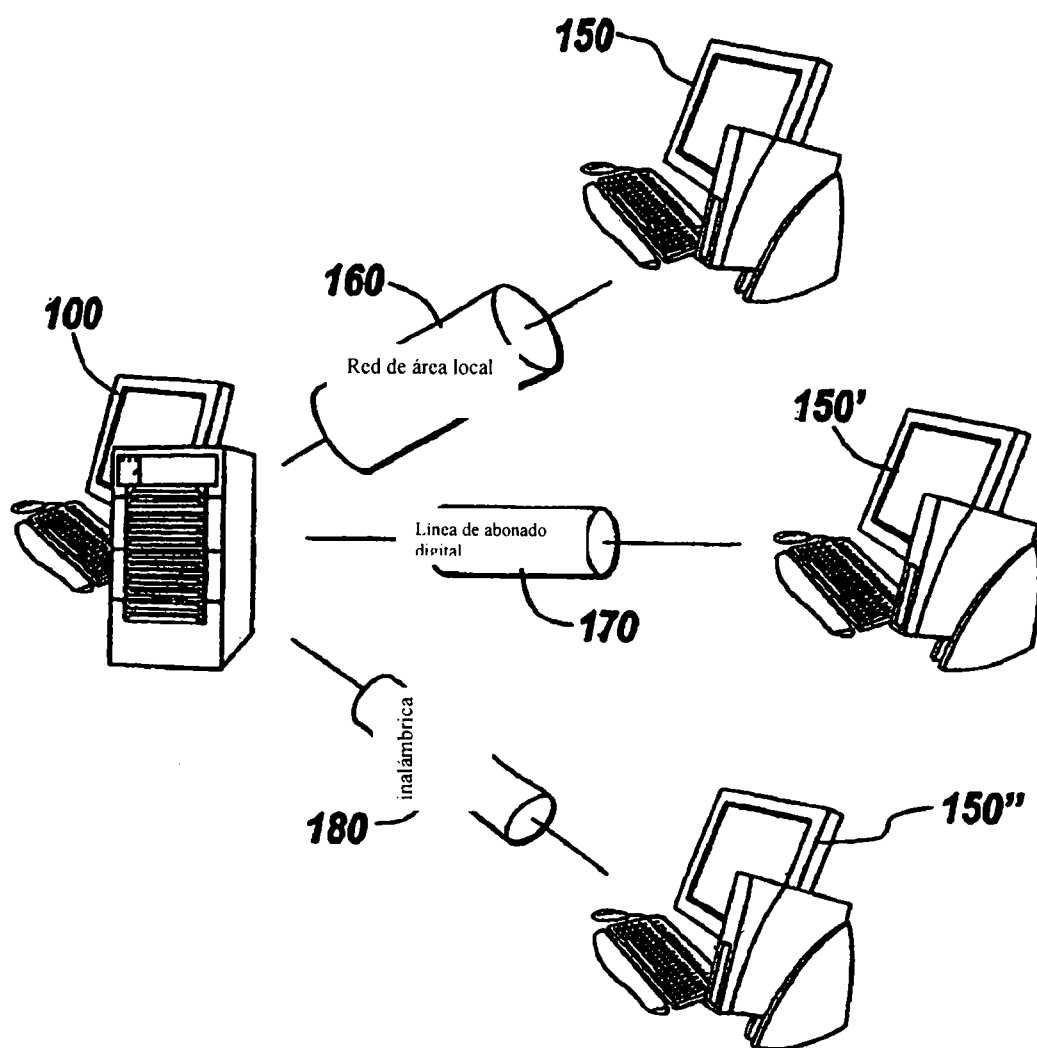


Fig. 1

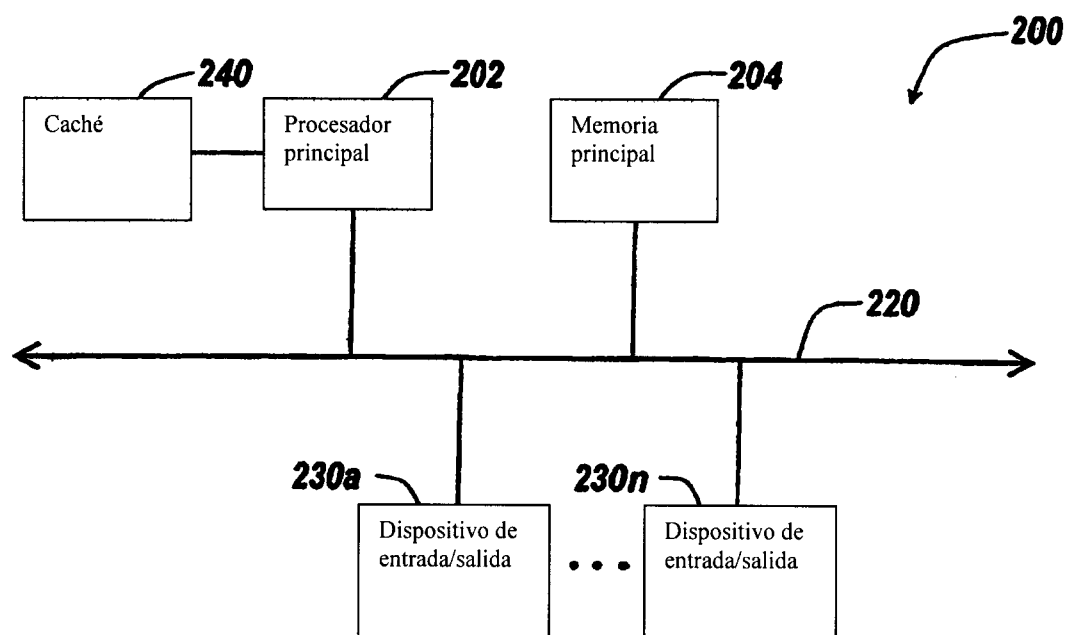


Fig. 2A

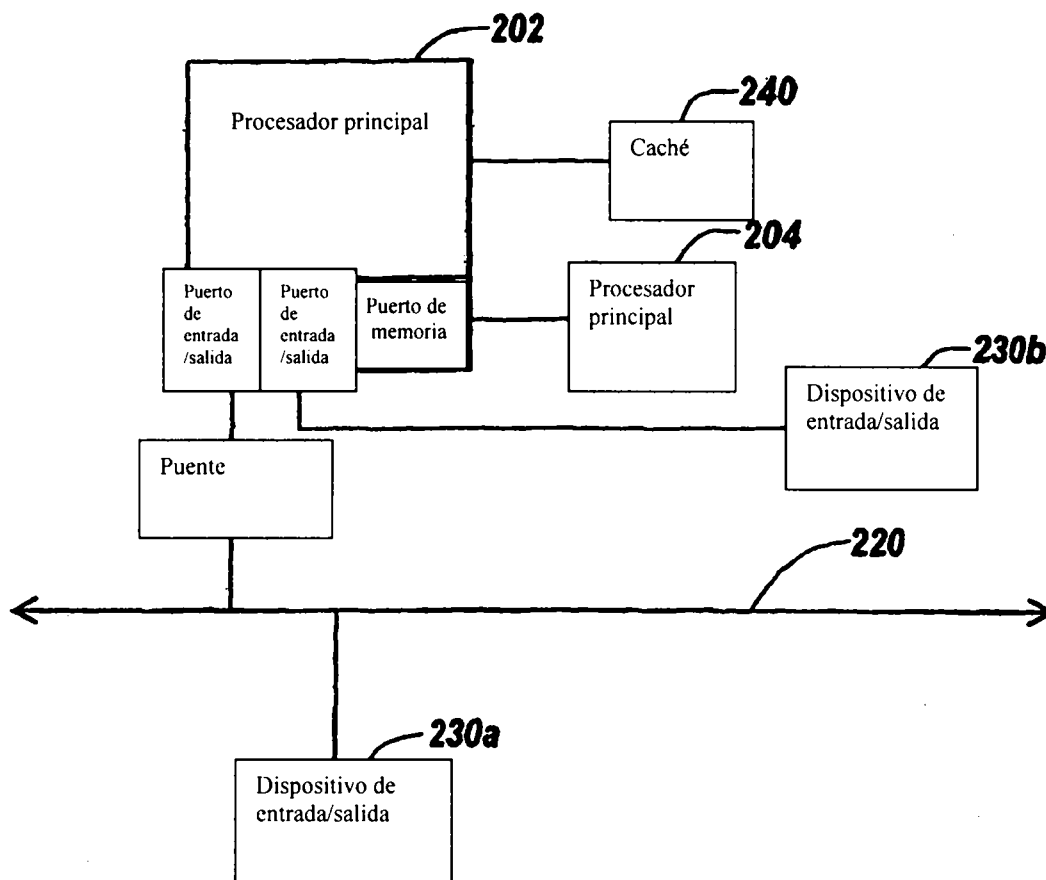


Fig. 2B

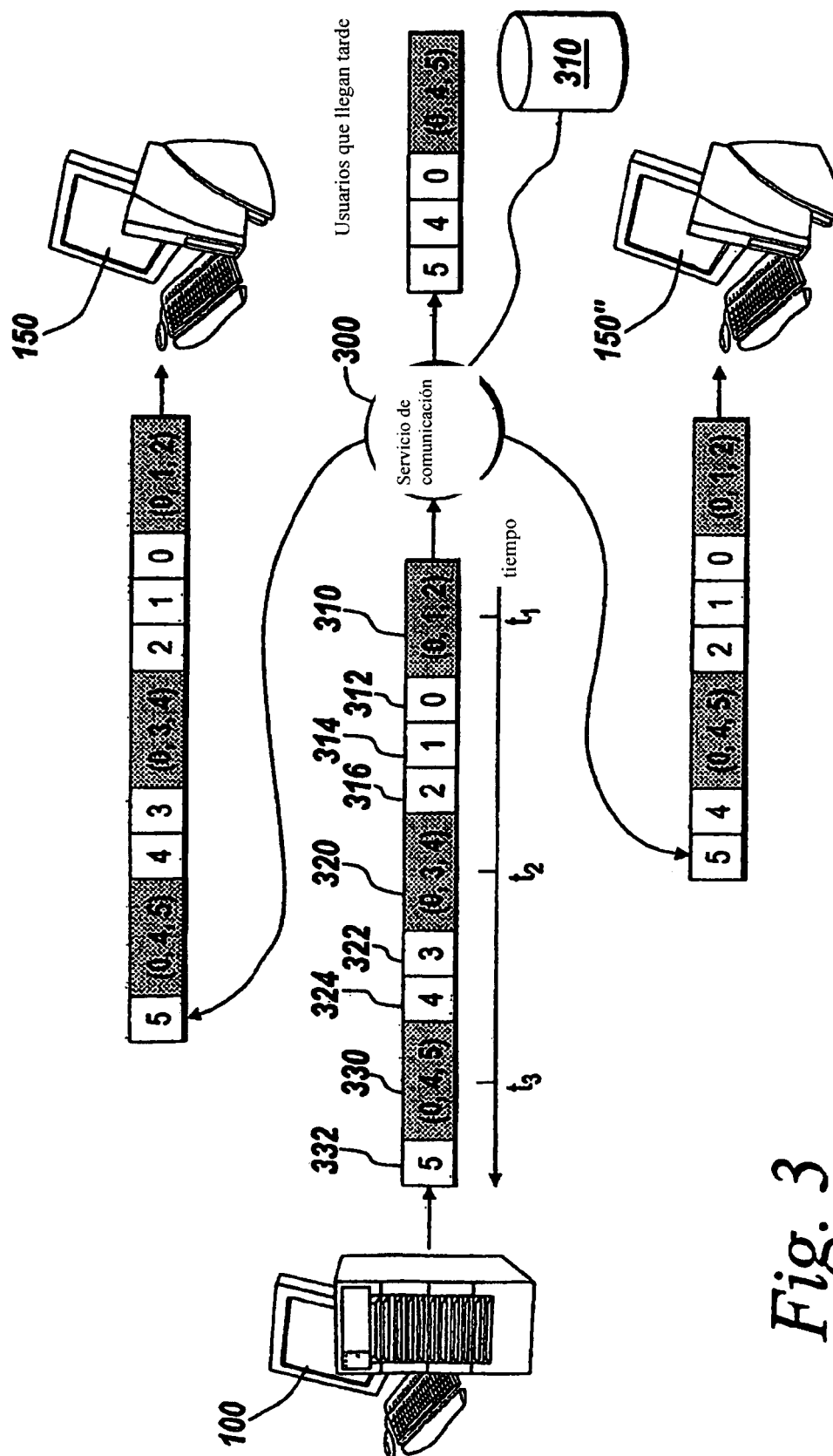


Fig. 3

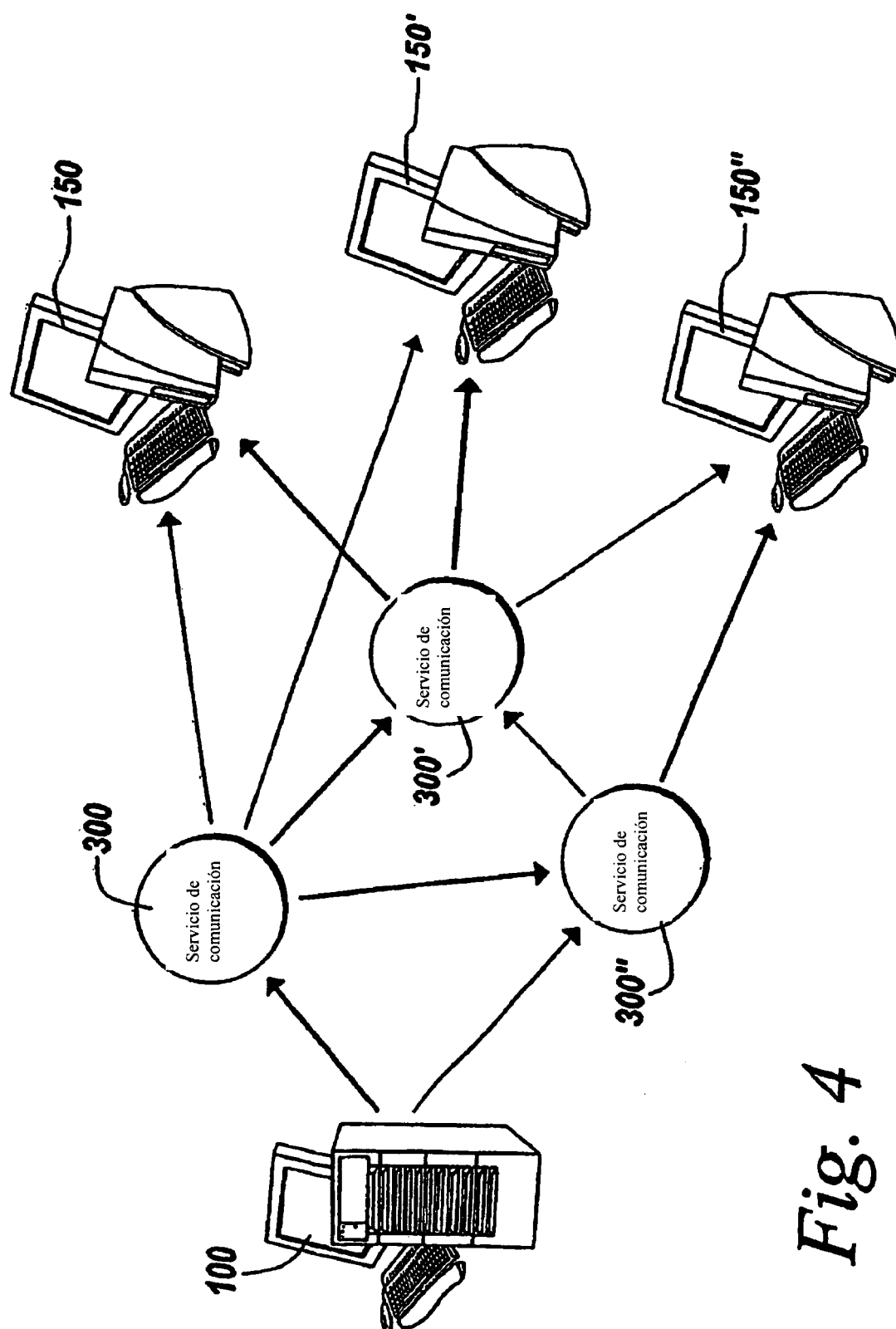


Fig. 4

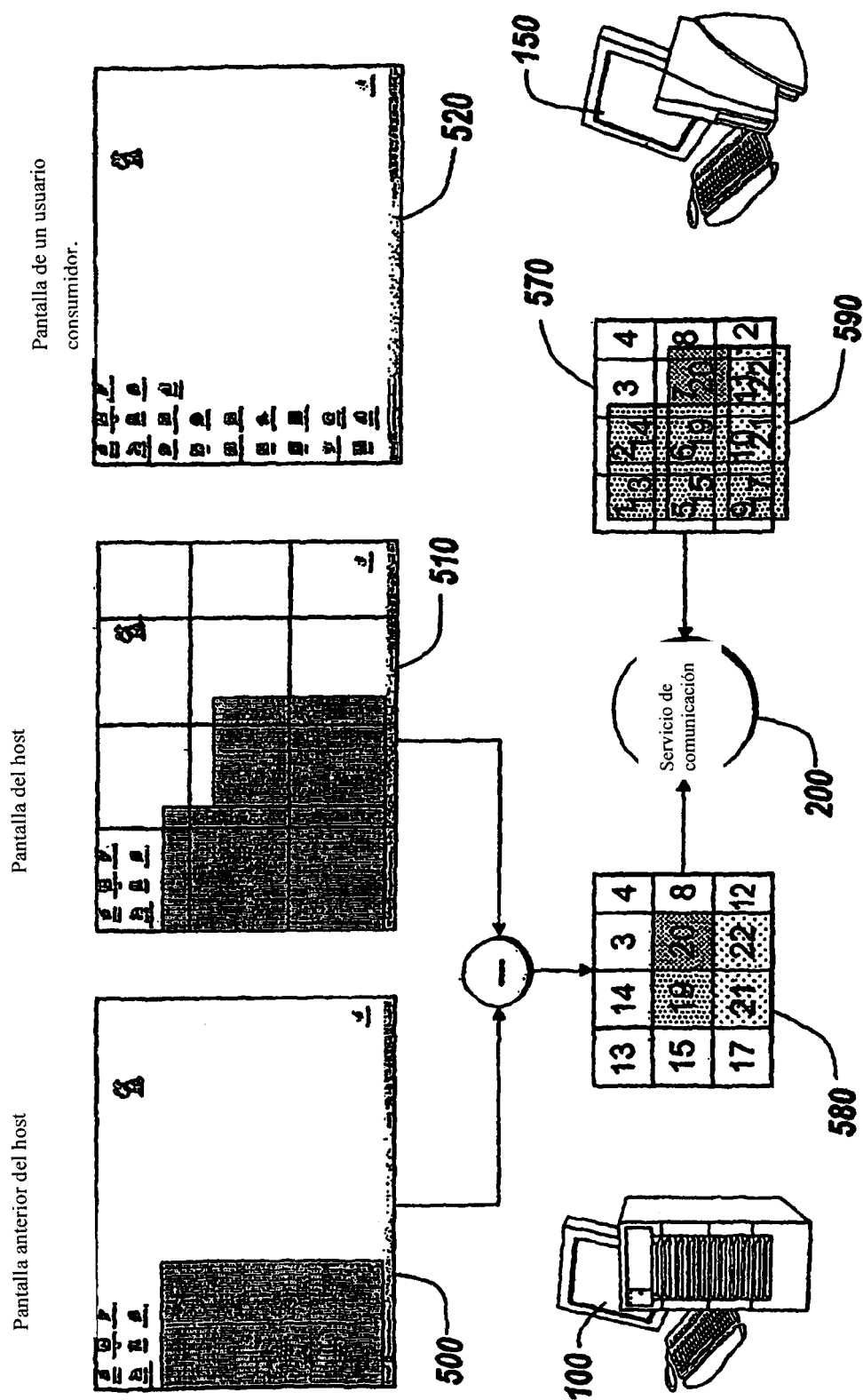


Fig. 5

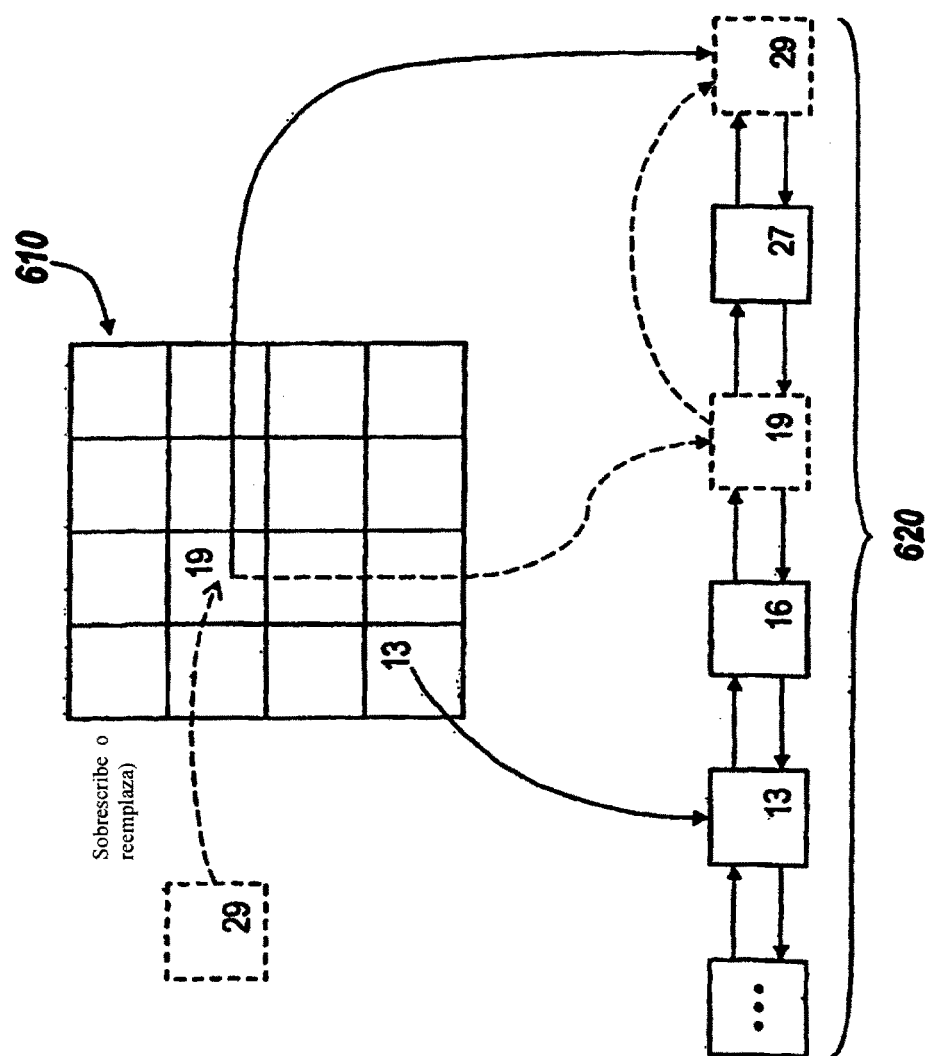


Fig. 6