

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 372 692**

51 Int. Cl.:  
**H04L 29/06** (2006.01)  
**H04N 7/24** (2011.01)  
**G06F 9/445** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 96 Número de solicitud europea: **03002723 .9**  
96 Fecha de presentación: **19.04.1995**  
97 Número de publicación de la solicitud: **1307025**  
97 Fecha de publicación de la solicitud: **02.05.2003**

54 Título: **SISTEMA INFORMÁTICO DISTRIBUIDO CLIENTE-SERVIDOR.**

30 Prioridad:  
**28.04.1994 US 233908**

45 Fecha de publicación de la mención BOPI:  
**25.01.2012**

45 Fecha de la publicación del folleto de la patente:  
**25.01.2012**

73 Titular/es:  
**OpenTV, Inc.**  
**275 Sacramento Street**  
**San Francisco, CA 94111, US**

72 Inventor/es:  
**Joseph, Kuriacose;**  
**Dureau, Vincent;**  
**Jessup, Ansley Wayne, Jr. y**  
**Delpuch, Alain**

74 Agente: **Tomas Gil, Tesifonte Enrique**

ES 2 372 692 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

Sistema informático distribuido cliente-servidor

[0001] Esta es una solicitud divisional de EP95105803.1 publicada como EP0680185.

5 [0002] La presente invención se refiere a un sistema informático distribuido cliente-servidor. Dicho sistema informático puede usarse en aplicaciones multimedia de difusión.

10 [0003] Los primeros sistemas informáticos fueron sistemas autónomos que generalmente consistían en grandes ordenadores. Posteriormente, varios sistemas informáticos de grandes ordenadores fueron conectados, o agrupados, para tratar tareas informáticas más grandes, tales como gran cantidad de usuarios de tiempo compartido. Con la llegada de los ordenadores personales, muchos sistemas informáticos autónomos de potencia relativamente baja fueron controlados directamente por los usuarios. Poco después, esta gran cantidad de ordenadores personales se acoplaron en redes de ordenadores, facilitando recursos compartidos y capacidades de comunicación a los usuarios de los ordenadores personales individuales y entre los usuarios y los grandes ordenadores preexistentes.

15 [0004] Un tipo de red así incluye un ordenador central, al que se denomina servidor, el cual, generalmente, incluye una gran cantidad de memoria masiva. Los programas usados por los usuarios de la red se almacenan centralmente en la memoria masiva en el servidor. Cuando un usuario desea ejecutar un programa, el ordenador del usuario solicita que se le envíe una copia de ese programa desde el servidor. Como respuesta a esta solicitud, el servidor transfiere una copia del programa desde su memoria masiva hasta la memoria principal del ordenador personal de ese usuario, y el programa se ejecuta en ese ordenador personal. Los datos también pueden almacenarse centralmente en el servidor y ser compartidos por todos los usuarios de la red. Los datos se almacenan en la memoria masiva del servidor a la que todos los usuarios de la red pueden acceder como respuesta a una petición. El servidor también sirve como concentrador para las comunicaciones de mensajes (correo electrónico) entre los usuarios de la red. El servidor de un sistema así maneja el almacenamiento y distribución de los programas, datos y mensajes, pero no aporta ninguna potencia de procesamiento a las tareas informáticas en sí de cualquiera de los usuarios. Es decir, un usuario no puede esperar que el ordenador servidor realice cualquiera de las tareas de procesamiento del programa que se ejecuta en el ordenador personal. Aunque esas redes desempeñan una función valiosa, no se trata de sistemas informáticos distribuidos, en los que los ordenadores interconectados colaboran para ejecutar una sola tarea informática.

20 [0005] A fin de mejorar dichas redes, podrá configurarse la red de tal manera que un usuario de la red pueda solicitar que el servidor, u otro ordenador personal conectado a la red, ejecute un programa. A esto se le denomina ejecución remota porque un ordenador (servidor u otro ordenador personal) que se halle a una distancia del solicitante ejecuta un programa como respuesta a la petición del solicitante. En un sistema así, el programa del cual se solicita una ejecución remota se envía desde el solicitante al ordenador remoto, o se recupera del servidor como respuesta a la solicitud realizada por el ordenador remoto. Cuando se recibe el programa, éste se ejecuta. De esta manera puede conseguirse varios ordenadores para que colaboren en la ejecución de una función informática.

35 [0006] Recientemente ha habido programas que distribuyen las tareas informáticas en sí necesarias para ejecutar una sola función informática. Por ejemplo, en un programa de base de datos así, en el que la base de datos se almacena en la memoria masiva del servidor, si un usuario desea realizar una consulta de la base de datos, la parte del programa de gestión de la base de datos del ordenador personal del usuario generará una solicitud de consulta, que se transmite al servidor. La parte del programa de gestión de la base de datos en el servidor lleva a cabo el procesamiento de la consulta, p.ej. analizando sintácticamente la solicitud de consulta, localizando dónde residen los datos especificados en la solicitud de consulta en su dispositivo de memoria masiva, accediendo a esos datos y devolviendo los resultados al ordenador personal que los ha solicitado por medio de la red. Entonces, la parte del programa de gestión de la base de datos del ordenador personal procesa los datos recibidos del servidor, p.ej. formateándolos y mostrándolos en la pantalla o imprimiéndolos en una impresora. Mientras el servidor procesa la solicitud de consulta, el ordenador personal puede ejecutar otro procesamiento, y mientras el ordenador personal está generando la solicitud de consulta y procesando los datos resultantes recibidos del servidor, el servidor puede procesar peticiones de consulta procedentes de otros ordenadores personales.

40 [0007] Otros tipos de programas también son útiles para este tipo de informática distribuida, a la que se denomina informática cliente-servidor. La compartición de las tareas de procesamiento entre el ordenador personal y el servidor mejora la eficacia informática total por la red. Dichos sistemas informáticos cliente-servidor, y las redes de ejecución remota, pueden denominarse sistemas informáticos distribuidos porque varios ordenadores (el servidor y/o los respectivos ordenadores periféricos) colaboran para realizar la función informática, p.ej. la gestión de la base de datos.

45 [0008] Recientemente se han propuesto los programas multimedia de difusión, más específicamente, los programas de televisión (TV) interactiva. Los programas de TV interactiva permitirán al espectador de un programa de televisión interactuar con ese programa. En un sistema de TV interactiva, la ubicación de la emisión central (red de TV, estudio local de TV, sistema por cable, etc.) tendrá un ordenador central, correspondiente al ordenador servidor, el cual producirá señales relacionadas con el programa de TV interactiva a transmitir simultáneamente con las señales de TV (vídeo y audio). Estas señales transportan datos que representan el programa de TV interactiva y pueden incluir órdenes, código de programa ejecutable y/o datos para el control de la interacción del espectador. Cada ubicación de espectador tendrá un ordenador, correspondiente al ordenador cliente, el cual recibirá las órdenes, código ejecutable y/o datos del ordenador central, ejecutará el código ejecutable, procesará los datos recibidos, aceptará entrada de datos del

usuario y facilitará datos al usuario por medio de la pantalla de TV. La entrada de datos del usuario puede devolverse al ordenador en la ubicación de emisión, permitiendo que el usuario interactúe con el programa de TV interactiva.

[0009] La patente estadounidense 4.965.825, SIGNAL PROCESSING APPARATUS AND METHODS (Aparatos y métodos de procesamiento de señales), expedida el 23 de octubre de 1990 por Harvey y otros, describe un sistema de TV interactiva en el cual una ubicación de emisión central incluye señales que transportan órdenes, código ejecutable y datos en, por ejemplo, el intervalo de borrado vertical de la señal de televisión para que las reciban los sistemas informáticos en las ubicaciones del espectador. Un ordenador que se halle en la ubicación del espectador extrae las órdenes, código ejecutable y datos, y ejecuta el código para procesar los datos e interactuar con el usuario. Un sistema así puede compararse con la función de ejecución remota de sistemas informáticos distribuidos, descritos anteriormente, en la que el ordenador del espectador se registra en el programa de TV interactiva y es controlado por la ubicación central.

[0010] EP 0 594 353 divulga un sistema de distribución de señal, tal como sistemas de televisión de cable interactiva, donde un servidor central comunica con terminales remotos donde el servidor asigna una dirección específica a cada terminal que se haya conectado para mensajes de control y sondeos. En el caso de un sistema de televisión de cable interactiva, tal terminal es la caja descodificadora o convertidor conectado entre el sistema de cable y el televisor. Periódicamente, el servidor emite un mensaje de control que contiene una dirección provisional y sondea la dirección provisional. Después de la recepción de dicho mensaje, un convertidor recién conectado almacena la dirección provisional y responde al sondeo con un mensaje de enlace ascendente. Después de la recepción del mensaje de enlace ascendente, el servidor inserta la dirección provisional en su lista de sondeo conforme al retraso de transmisión medido al enviar el sondeo a la recepción del mensaje de enlace ascendente y selecciona una nueva dirección provisional a usar en los mensajes de control de difusión posteriores. Los mensajes de enlace ascendente retumban por el servidor y son verificados por el convertidor. Si el convertidor no puede verificar un mensaje retumbado de enlace ascendente, lo que puede ocurrir si más de un convertidor recién conectado responde al mismo mensaje de control emitido, el convertidor repite su procedimiento de inicialización para adquirir una nueva dirección. Después de que la inicialización es completa, el software del sistema puede ser descargado desde el servidor al convertidor y el convertidor colocado en servicio.

[0011] En todos los sistemas anteriores, un ordenador central controla o responde a las solicitudes de ordenadores periféricos conectados a él a través de una red, es decir, el ordenador periférico (ordenador personal) solicita la ejecución remota de un programa, solicita un fichero o mensaje de otro ordenador, o manda una solicitud de pregunta a otro ordenador. Sólo en respuesta a una solicitud proporciona el otro ordenador una respuesta, por ejemplo ejecución remota, el solicitado fichero, mensaje o datos recuperados. Además, en general, es necesario que el ordenador periférico tenga todos los recursos necesarios para ejecutar completamente, o casi completamente, el programa deseado, con el servidor actuando sólo como otro mecanismo de almacenamiento o como mucho compartiendo una parte de las tareas informáticas.

[0012] Los inventores proponen en su solicitud copendiente 95105803.1, publicada como EP 0 680 185, un sistema informático distribuido en el que un servidor produce continuamente un flujo de datos. Este flujo de datos actúa como dispositivo de almacenamiento masivo para los ordenadores cliente que lo reciben. Este flujo de datos incluye repetitivamente datos que representan una aplicación informática distribuida en la que el ordenador de cliente puede participar, incluyendo código y datos ejecutables. Un mecanismo de transporte, incluyendo una vía de comunicación unidireccional de alta velocidad, lleva el flujo de datos del servidor al cliente. El cliente recibe el flujo de datos, extrae los datos informáticos representativos distribuidos y ejecuta la aplicación informática distribuida.

[0013] Dicho sistema informático distribuido comprende una fuente de flujo de datos continuo incluyendo repetitivamente la representación de una aplicación informática distribuida y un ordenador cliente, recibiendo el flujo de datos, para la extracción de los datos representativos del flujo de datos de la aplicación informática distribuida, y ejecutando la aplicación informática distribuida extraída.

[0014] En dicho sistema informático distribuido, no es necesario que el sistema informático de cliente incluya todos los recursos, en particular, memoria principal y dispositivo de almacenamiento masivo, necesarios para ejecutar el programa entero. En cambio, ningún dispositivo de almacenamiento masivo es requerido debido a que el flujo de datos proporciona la función del dispositivo de almacenamiento masivo, y el requisito de memoria principal es modesto porque sólo es necesario almacenar en la memoria la parte del programa que se ejecuta en ese momento. Cuando la parte ejecutada en ese momento se ha completado, su espacio de memoria es liberado, y la siguiente parte ejecutada es extraída del flujo de datos, almacenada en el espacio de memoria liberada, y dicha parte inicia la ejecución.

[0015] Además, tal sistema informático distribuido permite al usuario del ordenador cliente tener la opción de participar en la tarea informática distribuida. Si se desea participar, el ordenador cliente extrae los datos que representan la aplicación informática distribuida y ejecuta la aplicación informática distribuida, como se ha descrito anteriormente. Si no se desea participar, el flujo de datos es simplemente ignorado, y el procesamiento deseado por el usuario, o ninguno en absoluto, es realizado. Tal sistema informático distribuido también permite a cada ordenador cliente participante unirse a la función informática distribuida en cualquier momento y proceder a su propio ritmo a realizar su propia función informática.

[0016] Un sistema informático distribuido así es particularmente responsable de aplicaciones de TV interactivas porque permite a un espectador sintonizar un canal de TV interactiva en cualquier momento, incorporarse a la interactividad siempre que desee (o en absoluto), y permite a todos los espectadores proceder con sus diferentes ritmos. Esto es especialmente ventajoso en un entorno cuando un anuncio interactivo, con su propio código y datos ejecutables, se puede presentar dentro de un programa interactivo, o cuando el espectador desea cambiar de canal.

[0017] Según un primer aspecto de la presente invención se proporciona un sistema, según la reivindicación 1, que incluye: un ordenador servidor que incluye: un empaquetador de forma operacional para generar una primera corriente de paquete que incluye un módulo de código que representa una aplicación informática distribuida; una fuente de datos auxiliares para generar una fuente de paquete de datos auxiliares que incluye paquetes de vídeo; y un multiplexor acoplado al empaquetador para recibir la primera corriente de paquete, que incluye el módulo de código, y acoplado a la fuente de datos auxiliares para recibir la corriente de paquete de datos auxiliares, incluyendo los paquetes de vídeo; donde el empaquetador tiene que incluir repetitivamente el módulo de código dentro de la primera corriente de paquete, y el multiplexor tiene que multiplexar la primera corriente de paquete con la corriente de paquete de datos auxiliares para generar una segunda corriente de paquete.

[0018] Según una forma de realización preferida del primer aspecto el multiplexor es para medir el tiempo de multiplexado de la primera corriente de paquete con la corriente de paquete de datos auxiliares para generar la segunda corriente de paquete e incluye un constructor de flujo, acoplado al empaquetador, repetitivamente para presentar datos asociados con la aplicación informática distribuida al empaquetador para inclusión en la primera corriente de paquete y una fuente de aplicación, acoplada al constructor de flujo, para suministrar los datos asociados a la aplicación informática distribuida al constructor de flujo.

[0019] La fuente de aplicación incluye preferiblemente un compilador de aplicación para compilar los datos asociados a la aplicación informática distribuida y tiene que suministrar datos relacionados con transmisión de la aplicación informática distribuida al constructor de flujo, los datos relacionados con la transmisión de la aplicación informática distribuida que incluye un

índice de repetición deseado para el módulo de código en el primer flujo de datos.

[0020] El constructor de flujo es preferiblemente a constructo del módulo de código, utilizar los datos asociados a la aplicación suministrado de computación distribuida por la fuente de aplicación, y utilizando los datos asociados a la aplicación de computación distribuida, es a constructo de un módulo de datos con datos a ser procesado por código incluido en el módulo de código, y donde el empaquetador es de forma operacional a generar la primera corriente de paquete para incluir el módulo de datos.

[0021] El sistema puede incluir un ordenador de cliente, el ordenador de cliente incluyendo un receptor de flujo de datos para recibir un flujo de datos recibido y para extraer el módulo de datos del flujo de datos recibido y una memoria para memorizar al menos una parte de un primer módulo de datos de una pluralidad de módulos de datos, como extraído del flujo de datos recibido, antes de la recepción completamente del primer módulo de datos.

El ordenador de cliente tiene que ejecutar un primer módulo de código cuando el primer módulo de datos requerido por el primer módulo de código es recibido completamente y el receptor de flujo de datos incluye un extractor de datos de paquete para extraer paquetes de datos a partir dentro del flujo de datos recibido que incluía el módulo de datos y el extractor de paquete tiene que identificar los paquetes de datos que incluyen el módulo de datos que utiliza la información de identificación incluida en los paquetes de datos.

[0022] Preferiblemente el empaquetador tiene que incluir repetitivamente el módulo de datos en la primera corriente de paquete.

[0023] Según una forma de realización el constructor de flujo, que utiliza los datos asociados a la aplicación informática distribuida, tiene que incluir un módulo de directorio que incluye una lista que identifica el código y módulo de datos como incluido en la primera corriente de paquete, y donde el empaquetador es de forma operacional a generar la primera corriente de paquete para incluir el módulo de directorio y el empaquetador es repetitivamente para incluir el módulo de directorio en la primera corriente de paquete.

[0024] El sistema puede incluir un ordenador de cliente, el ordenador de cliente incluyendo un receptor de flujo de datos a recibir un flujo de datos recibido y para extraer el módulo de directorio a partir del flujo de datos recibido, incluyendo una memoria para memorizar al menos un primer módulo de directorio de un conjunto de módulos de directorio, como extraído del flujo de datos recibido, antes de recibir completamente el conjunto de módulos de directorio, el conjunto de módulos de directorio especificando un código asociado de identificación de directorio y módulo de datos incluido en el flujo de datos recibido y el ordenador de cliente tiene que pedir el código asociado y módulo de datos identificado por el directorio y el receptor de flujo de datos incluye un extractor de datos de paquete para extraer paquetes de datos desde dentro del flujo de datos recibido que incluía el módulo de directorio y el extractor de paquete tiene que identificar los paquetes de datos que incluían el módulo de directorio que utiliza la información de identificación incluida en los paquetes de datos.

[0025] Preferiblemente la fuente de datos auxiliares tiene que proporcionar los paquetes de vídeo para incluir datos codificados MPEG que representan una imagen televisiva y para proporcionar la corriente de paquete de datos auxiliares para incluir paquetes audio con datos audio digitalmente codificados.

- 5 [0026] En una alternativa el empaquetador, la fuente de datos auxiliares y el multiplexor comprenden una primera fuente, incluyendo el sistema una segunda fuente y un transporte multiplexor, el transporte multiplexor acoplado para recibir la segunda corriente de paquete desde la primera fuente y una tercera corriente de paquete de la segunda fuente, y para multiplexar la segunda corriente de paquete y la tercera corriente de paquete para transmisión por un mecanismo de transporte como una cuarta corriente de paquete y el multiplexor de transporte tiene que multiplexar por frecuencia la segunda y la tercera corrientes de paquete para transmisión sobre el mecanismo de transporte y el multiplexor de transporte tiene que multiplexar en el tiempo la segunda y la tercera corrientes de paquete para transmisión por el mecanismo de transporte.
- 10 [0027] Un reloj se puede acoplar para proporcionar datos de código de tiempo al empaquetador y la fuente de datos auxiliares, donde los datos de código de tiempo tienen que ser incluidos en cualquiera o en tanto la corriente de paquete de datos auxiliares y el primer flujo de datos de paquete.
- 15 [0028] El empaquetador puede incluir primeros paquetes en el primer flujo de datos para incluir información de identificación que indica que los primeros paquetes incluyen al menos una parte del módulo de código que representa la aplicación informática distribuida y la fuente de datos auxiliares puede incluir segundos paquetes en la corriente de paquete de datos auxiliares para incluir información de identificación que indica que los segundos paquetes incluyen datos de video.
- [0029] El sistema puede incluir transceptor de datos por el que los datos son recibidos en el servidor desde un ordenador de cliente por medio de una instalación de procesamiento central y el ordenador de cliente puede acceder el servidor por medio de la instalación de procesamiento central para recuperar el módulo de código.
- 20 [0030] La instalación de tratamiento central puede ser un sistema telefónico.
- [0031] El servidor puede recibir y procesar datos recibidos en el servidor desde el ordenador de cliente durante la ejecución de la aplicación informática distribuida por el ordenador de cliente y puede generar el módulo de código basado en los datos recibidos en el servidor desde el ordenador cliente o el módulo de datos basado en los datos recibidos en el servidor desde el ordenador de cliente.
- 25 [0032] El sistema puede incluir un ordenador de cliente que incluye un receptor de flujo de datos para recibir un flujo de datos recibido y para extraer el módulo de código que representa la aplicación de informática distribuida desde el flujo de datos recibido.
- [0033] El ordenador de cliente puede incluir una unidad de procesamiento, la unidad de procesamiento para ejecutar la aplicación informática distribuida y una memoria para almacenar al menos una parte de un primer módulo de código de una pluralidad de módulos de código, como extraídos del flujo de datos recibido, antes de recibir completamente el primer módulo de código.
- 30 [0034] Preferiblemente el ordenador de cliente tiene que ejecutar el primer módulo de código cuando completamente recibido y la memoria tiene que almacenar al menos el primer y un segundo módulo de código de la pluralidad de módulos de código, donde el ordenador de cliente tiene que conmutar entre ejecución de los primeros y los segundos módulos de código y el receptor de flujo de datos incluye un extractor de datos de paquete para extraer paquetes de datos, desde dentro del flujo de datos recibido, que incluye el módulo de código.
- 35 [0035] El extractor de paquete puede identificar los paquetes de datos que incluía el módulo de código que utiliza la información de identificación incluida en los paquetes de datos.
- [0036] Preferiblemente el receptor de flujo de datos incluye un extractor de datos auxiliares para extraer paquetes de video del flujo de datos recibido y un selector de corriente para seleccionar el flujo de datos recibido para recepción por el receptor de flujo de datos y un desmodulador para desmodular para recepción de una señal portadora modulado desde un enlace de satélite y conjunto de circuitos para demultiplexar en tiempo una pluralidad de flujos de datos portados por el portador de señal modulada.
- 40 [0037] Según un segundo aspecto de la presente invención es proporcionado un método, según la reivindicación 47, para distribuir una aplicación informática, el método incluyendo: la generación de una primera corriente de paquete que incluye repetitivamente un módulo de código que representa una aplicación informática distribuida; la generación de una corriente de paquete de datos auxiliares que incluye paquetes de video; y la multiplexación de la primera corriente de paquete con la corriente de paquete de datos auxiliares para generar una segunda corriente de paquete.
- 45 [0038] La FIGURA 1 es un diagrama en bloque de un sistema informático distribuido de acuerdo con la presente invención;
- 50 la FIGURA 2 es un diagrama en bloque de un ordenador servidor como se muestra en la FIGURA 1;
- la FIGURA 3 es un diagrama de temporización que muestra los flujos de datos producidos por un ordenador servidor en un sistema informático distribuido como se muestra en la FIGURA 1;
- la FIGURA 4 es un diagrama en bloque de un ordenador cliente como se muestra en la FIGURA 1
- 55 [0039] La FIGURA 1 es un diagrama en bloque de un sistema informático distribuido de acuerdo con la presente invención. En la FIGURA 1, un ordenador servidor 10, que puede incluir un gran sistema informático, se conecta a una pluralidad de ordenadores clientes 20 a través de un mecanismo de transporte 30. El ordenador servidor 10 puede

- conectarse a más de los tres ordenadores cliente 20 que se muestran en la Figura 1, y los ordenadores cliente 20 pueden dispersarse geográficamente de forma amplia. El ordenador cliente 22 se conecta bidireccionalmente a un ordenador local 40, a un sistema de procesamiento de datos auxiliares 50 y a una instalación de procesamiento central 60. La instalación de procesamiento central 60 se conecta bidireccionalmente al ordenador servidor 10. La instalación de procesamiento central 60 también puede conectarse a otras instalaciones diferentes del ordenador servidor 10 que se muestra en la FIGURA 1. El ordenador local 40 se conecta además bidireccionalmente a un dispositivo de almacenamiento masivo 70. El ordenador cliente 22 interactúa con un usuario 80 facilitando información al usuario a través de una pantalla visualizadora u otro dispositivo de salida (no se muestra) y aceptando información del usuario por medio de un teclado u otro dispositivo de entrada (tampoco se muestra).
- 5 [0040] Los ordenadores cliente 24 y 26 también interactúan con sus usuarios (no se muestra a fin de simplificar el dibujo). Además, los ordenadores cliente 24 y 26 se conectan bidireccionalmente a la instalación de procesamiento central 60. Sin embargo, dichos enlaces son opcionales. Los únicos requisitos para un ordenador cliente 20 es un modo de interactuar con un usuario y una conexión al mecanismo de transporte 30. Los enlaces a los ordenadores locales, sistemas de procesamiento de datos auxiliares, y la instalación de procesamiento central 60 son todos ellos opcionales y no necesariamente han de estar presentes en cada uno de los ordenadores cliente 20.
- 10 [0041] El mecanismo de transporte 30 incluye un enlace unidireccional de datos digitales de alta velocidad, tales como un enlace por satélite digital o de fibra óptica directo procedente de un servidor 10 a los ordenadores cliente 20. Los datos pueden transportarse por el sistema de transporte 30 mediante un sistema de datos por paquetes. En un sistema de este tipo, se transmite un flujo de paquetes de datos, cada uno incluyendo información de identificación que indique, entre otras cosas, el tipo de datos incluidos en el paquete y los datos en sí, por el enlace de datos. Un sistema de datos por paquetes así permite la multiplexión en el tiempo de varios flujos independientes de datos, cada uno identificado por la información de identificación en sus paquetes, dentro de un solo flujo de paquetes.
- 15 [0042] Además, es posible multiplexar una pluralidad de flujos de datos por paquetes así por los respectivos canales en el mismo medio físico (enlace radiofónico por satélite o fibra óptica) que forman el mecanismo de transporte 30. Por ejemplo, pueden modularse diferentes flujos de datos en señales portadoras que tengan diferentes frecuencias. Estas portadoras moduladas pueden transmitirse, por ejemplo, por medio de los respectivos transpondedores de un enlace por satélite. Además, si un transpondedor concreto tiene capacidad suficiente, es posible multiplexar en el tiempo varios flujos de datos de una sola portadora modulada.
- 20 [0043] Cada ordenador cliente 20 incluye un receptor de datos para seleccionar uno de los flujos de paquetes que se están transportando por el mecanismo de transporte 30, recibir el flujo de paquetes seleccionado y extraer los datos contenidos en ellos. Siguiendo con el ejemplo anterior, el receptor de datos puede incluir un demodulador sintonizable para recibir una de las respectivas portadoras moduladas del enlace por satélite. Además, el receptor de datos puede incluir un sistema de circuitos para la desmultiplexión en el tiempo de los respectivos flujos de datos que esa portadora modulada transporta.
- 25 [0044] Cuando funciona, el servidor 10 produce unos flujos de datos continuos en forma de flujo de paquetes para los ordenadores cliente 20. El servidor 10 inserta repetitivamente un paquete, o paquetes sucesivos, que contienen datos que representan la aplicación informática distribuida, incluyendo al menos un módulo de código ejecutable, en el flujo de datos. Este módulo de código contiene códigos ejecutables para los ordenadores cliente 20. Por ejemplo, el receptor de datos del ordenador cliente 22, controla continuamente los paquetes que hay en el flujo de datos del mecanismo de transporte 30. Cuando un paquete que incluye información de identificación indicando que contiene el módulo de código (o una parte del módulo de código) requerido por el ordenador cliente 22 se halla presente en el flujo de datos, el ordenador cliente 22 detecta su presencia, extrae el módulo de código (o parte del módulo de código) de ese paquete y lo almacena en la memoria principal. Cuando se ha recibido todo el módulo de código, el ordenador cliente 22 empieza a ejecutarlo.
- 30 [0045] Es posible que haya más de un módulo de código colocado en el flujo de datos continuos, cada uno de ellos incluyendo una parte diferente de la aplicación informática distribuida. Por ejemplo, es posible dividir la aplicación informática distribuida en pequeñas partes de tal manera que sólo deba ejecutarse una parte cada vez. La parte de la aplicación informática distribuida que se necesita ejecutar en la actualidad se carga en la memoria del ordenador cliente 22. Cuando esa parte completa su ejecución, entonces se extrae un módulo de código que incluye el código ejecutable para la próxima parte de la aplicación informática distribuida del flujo de datos, se almacena en memoria y se ejecuta. Cada parte se extrae del flujo de datos según necesidad. Si hay suficiente memoria en el ordenador cliente 22, es posible cargar varios módulos de código en la memoria y pasar del uno al otro sin extraerlos del flujo de datos, pero esto no es necesario. Estructurando una aplicación informática distribuida de esta forma, el tamaño de memoria requerido para el ordenador cliente 22 podrá minimizarse.
- 35 [0046] El servidor 10 también puede incluir repetitivamente un paquete o paquetes que contengan uno o más módulos de datos en el flujo de datos. Los módulos de datos contienen datos que el código ejecutable debe procesar en el módulo de código. Antes o durante la ejecución del código de un módulo de código extraído anteriormente, el ordenador cliente 22 puede necesitar acceder a los datos en el módulo o módulos de datos. Si es así, el ordenador cliente 22 controla el flujo de datos para el módulo o módulos de datos requeridos. Cuando los paquetes que contienen el módulo o módulos de datos (o partes del módulo o módulos de datos) se hallan presentes en el flujo de datos, se extraen y el contenido se almacena en la memoria principal del ordenador cliente 22. Cuando se han recibido completamente todos los módulos de datos requeridos, el ordenador cliente 22 empieza o continúa la ejecución del código del módulo de
- 40
- 45
- 50
- 55
- 60

código para procesar los datos del módulo o módulos de datos recibidos. Como ocurre con los módulos de código, es posible almacenar más de un módulo de datos en memoria, si existe suficiente memoria, en el ordenador cliente 22.

[0047] El servidor 10 puede también incluir repetitivamente en el flujo de datos un paquete o paquetes que incluyan un directorio de los módulos de código o de datos que se incluyen actualmente en el flujo de datos. El directorio incluye una lista de todos los módulos de código y datos que se hallan presentes en el flujo de datos, junto con información acerca de esos módulos. Si hay un directorio presente en el flujo de datos, entonces, antes de extraer cualquier módulo de código o de datos del flujo de datos, el ordenador cliente 22 controla el flujo de datos para el directorio. Cuando los paquetes que contiene el directorio (o partes del directorio) se hallan presentes en el flujo de datos, se extraen y sus datos se almacenan en la memoria principal del ordenador cliente 22. Cuando se ha recibido el directorio completamente, el ordenador cliente 22 evalúa las entradas contenidas en el directorio, luego solicita el primer módulo de código y/o de datos del flujo de datos y se continúa con la ejecución de la forma descrita anteriormente.

[0048] Cualquiera de los ordenadores cliente 20 puede incorporarse a la función informática distribuida representada por el flujo del paquete en cualquier momento, y cada uno de los ordenadores cliente 20 puede funcionar a su propia velocidad, generalmente como respuesta al usuario 80. Para permitir esto, el servidor 10 coloca repetitivamente el directorio y todos los módulos de código y de datos que los ordenadores cliente 20 requieran para realizar su parte de la función informática distribuida en el flujo de datos del mecanismo de transporte 30. Cuando uno de los ordenadores cliente 20 se incorpora a la función informática distribuida, éste controla el flujo de paquete seleccionado recientemente en el mecanismo de transporte 30 para el módulo de directorio, lo extrae y lo procesa como se describió anteriormente. Durante su ejecución, cuando uno de los ordenadores cliente 20 requiere un nuevo módulo de código y/o de datos, éste controla el flujo de datos del mecanismo de transporte 30 para el módulo de código y/o de datos requerido recientemente, lo extrae y lo ejecuta, si se trata de un módulo de código, o lo procesa, si se trata de un módulo de datos, de la forma descrita anteriormente.

[0049] El flujo de datos del paquete también puede incluir paquetes de datos auxiliares. El ordenador cliente 22 no requiere estos datos para la ejecución del código, aunque pueden estar relacionados con la ejecución puesto que el usuario 80 podría interactuar con el programa de ejecución del ordenador cliente 22 en base a los datos auxiliares recibidos. El receptor del flujo de datos del ordenador cliente 22 reconoce los paquetes de datos auxiliares en el flujo de datos del mecanismo de transporte 30 y los pasa directamente al procesador de datos auxiliares 50. El procesador de datos auxiliares 50 procesa sus paquetes independientemente del ordenador cliente 22. Si los datos auxiliares deben presentarse al usuario 80, el procesador de datos auxiliares 50 puede proporcionar su propio dispositivo de visualización (no se muestra) el cual puede ser compartido con el ordenador cliente 22, o el dispositivo de visualización (no se muestra) asociado con el ordenador cliente 22 puede ser compartido con el procesador de datos auxiliares 50, para facilitar una sola pantalla de información al usuario 80. El procesador de datos auxiliares 50 puede tener conexiones a otros elementos ilustrados (no se muestra), pero esto depende del tipo de datos.

[0050] Por ejemplo, en un sistema de TV interactiva, los datos auxiliares incluyen las partes vídeo y audio de la señal de televisión subyacente. Por ejemplo, los datos auxiliares incluirían paquetes de vídeo que contengan datos codificados MPEG, o del estilo MPEG, que representen los paquetes de imagen y vídeo de la televisión que contienen audio codificado digitalmente. Además, podría haber varios flujos de paquete audio diferentes que transporten los respectivos canales audio para estéreo, segundo programa de audio (SAP) o capacidad multilingüe. En un procesador de datos auxiliares 50 de un sistema de este tipo, los paquetes de vídeo se suministrarán a un decodificador MPEG (o parecido) conocido (no se muestra) que generaría señales vídeo estándar, y que se suministrarían al receptor o monitor vídeo (no se muestra) de televisión. Los paquetes audio se suministrarían a un decodificador audio conocido (no se muestra) que generaría señales audio estándar para el receptor o altavoces (no se muestra) de televisión.

[0051] En un sistema de TV interactiva de este tipo, el ordenador cliente 22 podría, como respuesta a la ejecución del módulo de código ejecutable, generar visualizaciones gráficas para ofrecer información al usuario 80. Estas visualizaciones gráficas pueden combinarse con la señal vídeo estándar desde el decodificador MPEG de manera conocida, y la imagen combinada puede visualizarse en el receptor o monitor vídeo de televisión. El ordenador cliente 22 también puede generar sonidos para ofrecer otra información al espectador. Los sonidos generados pueden combinarse, de manera conocida, con las señales audio estándar procedentes del decodificador audio, y el sonido combinado puede reproducirse por el receptor o los altavoces de televisión.

[0052] Además, pueden incluirse datos de código de tiempo en el flujo de datos del paquete auxiliar de televisión o en el flujo de datos del paquete que representa la aplicación de TV interactiva, o en ambos flujos. Esto permite la sincronización de cualquier imagen gráfica o sonidos generados por el ordenador cliente 22 con la señal de televisión procedente de los datos auxiliares. En este caso, el ordenador cliente 22 tendría acceso a los datos del código de tiempo, y controlaría la generación de la imagen gráfica y/o sonido para que tuviera lugar en un momento deseado, tal como lo indiquen los datos del código de tiempo.

[0053] En un sistema de TV interactiva de este tipo, tanto el ordenador cliente 22 como el procesador de datos auxiliares 50 pueden incluirse en un solo recinto, tal como un receptor de televisión o una caja decodificadora set-top de televisión. Un receptor de televisión, o una caja decodificadora, incluirían conectores para conectar a un ordenador local u otro equipo.

[0054] El usuario 80 aporta información al programa en funcionamiento en el ordenador cliente 22 durante su ejecución. El servidor 10 podría requerir estos datos a fin de realizar la función informática distribuida. Por ejemplo, en un sistema

de TV interactiva el usuario 80 puede ofrecer información al ordenador cliente a través de una unidad de control remoto manual.

[0055] Los datos del usuario se transfieren al ordenador servidor 10 por medio de la instalación de procesamiento central 60. En una representación, se envían datos de los ordenadores cliente 20 al ordenador servidor 10 por medio de módems a través del sistema telefónico que actúa como la instalación de procesamiento central 60. El ordenador servidor 10 recibe y procesa los datos recibidos desde los ordenadores cliente 20 durante la ejecución de su parte de función informática distribuida.

[0056] El ordenador servidor 10 puede generar nuevos módulos de código y/o datos, o modificar módulos existentes, en el flujo de datos del mecanismo de transporte 30, de la manera descrita anteriormente, en base a los datos recibidos. Alternativamente, el ordenador servidor 10 puede devolver información inmediatamente a los ordenadores cliente 20 en la otra dirección por medio de la instalación de procesamiento central 60. Todos los ordenadores cliente 20 que participen en la función informática distribuida procesan la información de los módulos de código y/o de datos generados recientemente, mientras que la información transmitida del ordenador servidor 10 a los ordenadores cliente 20 a través de la instalación de procesamiento central 60 se relaciona específicamente al ordenador cliente (22, 24 ó 26) al que se ha enviado la información.

[0057] En otra representación, la instalación de procesamiento central 60 puede incluir su propio sistema informático, conectado separadamente por medio de un módem a los ordenadores cliente 20 y al ordenador servidor 10 por medio del sistema telefónico. En cualquiera de las representaciones anteriores, la instalación de procesamiento central 60 ofrece acceso a otros ordenadores o instalaciones de procesamiento (no se muestra) por medio del sistema telefónico. De este modo, si es necesaria la información de otros sistemas informáticos para realizar la función informática distribuida, tanto los ordenadores cliente 20 como el ordenador servidor 10 podrán acceder a esos sistemas informáticos a través de un módem por medio del sistema telefónico.

[0058] Un puerto de entrada/salida (E/S) del ordenador cliente 22 se conecta a un puerto correspondiente del ordenador local 40. El ordenador local 40 se da en combinación con el ordenador cliente 22. El ordenador local 40 puede ser un ordenador personal usado por el usuario 80 del ordenador cliente 22, o puede ser un ordenador más grande, o una red informática ubicada en el mismo emplazamiento que el ordenador cliente 22. Esto permite que el ordenador cliente 22 acceda a los datos de la memoria masiva acoplada 70 del ordenador personal o un ordenador en la red ubicado en el emplazamiento del ordenador cliente 22. Además, el ordenador cliente 22 podría usar la memoria masiva 70 del ordenador local 40 para almacenar datos a recuperar posteriormente. Es posible que el ordenador local 40 incluya tanto un dispositivo de salida (no se muestra), como podría ser el monitor de un ordenador, como un dispositivo de entrada (tampoco se muestra), como podría ser el teclado de un ordenador. El ordenador cliente 22 y/o el procesador de datos auxiliares 50 pueden compartir ambos dispositivos, tal como se describió anteriormente.

[0059] Por ejemplo, el sistema informático distribuido que se muestra en la FIGURA 1 puede formar parte de un sistema informático corporativo muy extendido, y el servidor 10 puede encontrarse en una ubicación central de esa corporación. El ordenador cliente 22 puede ubicarse en un lugar remoto, y el ordenador local 40 puede conectarse a la red del ordenador personal en esa ubicación. Los trabajadores de esa ubicación pueden almacenar datos compartidos (p.ej. información financiera) en el servidor conectado a esa red. La función informática distribuida puede incluir la recopilación de datos financieros locales de los ordenadores cliente en las ubicaciones remotas, el procesamiento de esos datos financieros y el retorno de los resultados financieros totales a los ordenadores cliente. En una aplicación de este tipo, el código ejecutable en el ordenador cliente 22 accede a los datos del ordenador local 40 (bien desde su memoria masiva acoplada 70 o bien a través de la red) a través del puerto E/S y los envía al ordenador servidor 10 a través de la instalación de procesamiento central 60. El ordenador servidor 10 continúa su procesamiento en base a la información recibida del ordenador cliente 22 (y otros ordenadores cliente 20), y devuelve los resultados de ese procesamiento a los ordenadores cliente 20 bien a través de la instalación de procesamiento central 60 o bien a través del flujo de datos del mecanismo de transporte 30.

[0060] En otro ejemplo, el sistema informático distribuido puede ser un sistema de televisión interactiva que emita un programa de televenta como la aplicación informática distribuida. En un caso así, los datos auxiliares transportan la parte vídeo y audio de la señal de televisión, los cuales pueden mostrar y describir los artículos que se ofrecen a la venta, y pueden incluir tanto actores en directo como gráficos sobrepuestos generados en los estudios centrales. Los módulos de código y datos que forman la aplicación de televisión interactiva pueden incluir datos sobre los productos que se ofrecerán a la venta durante este programa, o parte del programa, y el código ejecutable a interactuar con el usuario de la manera descrita a continuación.

[0061] Cuando un espectador desea pedir un artículo, se pulsa un botón en el mando a distancia del televisor. Este botón emite una señal al ordenador cliente 22 para que muestre una serie de instrucciones y menús necesarios para solicitar la información necesaria para hacer el pedido, p.ej. el número del artículo, nombre y dirección del espectador, el método de pago, el número de la tarjeta de crédito (si es necesario), etc. Estas instrucciones se generan en el ordenador cliente como gráficos que se superponen en la imagen de vídeo del televisor. También es posible generar una voz generada por ordenador que se combina con el audio del televisor bien como voz en off, o reemplazando el audio de televisión. El espectador responde a la instrucción facilitando la información solicitada a través del mando a distancia del televisor. Cuando la información solicitada por la visualización en pantalla y/o las instrucciones vocales han sido ingresadas por el espectador, se envían a un ordenador central a través del módem del ordenador cliente. Puede enviarse una confirmación del pedido en la otra dirección desde el ordenador central.

[0062] También es posible que el espectador pueda ingresar una vez por adelantado la información permanente sobre el espectador (es decir, el nombre, dirección, forma de pago y número de tarjeta de crédito), de manera que no es necesario solicitar esa información cada vez que se haga un pedido. La información se almacena en la memoria permanente del ordenador cliente. En un caso así, cuando se hace un pedido, esa información se recupera de la memoria permanente, se anexa al número de artículo y se transmite al ordenador central. También es posible que, por medio de códigos de tiempo u otras señales de control introducidos en el flujo de datos, el ordenador cliente sepa qué artículo se está ofreciendo a la venta en ese momento. En tal caso, el espectador podrá pedirlo con sólo pulsar un botón en el mando a distancia del televisor. Como respuesta, el ordenador cliente puede combinar la información recibida anteriormente relacionada con el artículo que se ofrece a la venta en ese momento con la información personal almacenada anteriormente relacionada con el espectador, transmitir el pedido al ordenador central y recibir a cambio la confirmación.

[0063] Puesto que los módulos de código y de datos relacionados con el programa de teletienda se introducen repetitivamente en el flujo de datos, un espectador puede sintonizar con el programa en cualquier momento y participar interactivamente. De igual forma, no es necesario que el espectador participe interactivamente, sino que puede ignorar simplemente la parte interactiva del programa.

[0064] También es posible que el ordenador cliente 22 reciba información de control procedente del ordenador local 40. Por ejemplo, el usuario 80, usando el ordenador local 40, podría controlar el ordenador cliente 22 a través del puerto de E/S para seleccionar un flujo de datos deseado del mecanismo de transporte 30, y procesar el programa que se emita en ese momento en el flujo de datos, con interacción con el usuario 80 a través de los dispositivos de entrada y salida (no se muestran) conectados al ordenador local 40.

[0065] También es posible que el usuario 80 haga que el ordenador cliente 22 acceda al ordenador servidor 10 a través de la instalación de procesamiento central 60, en vez de hacerlo a través del flujo de datos del mecanismo de transporte 30, y reciba los módulos de código y de datos a través de este enlace bidireccional.

[0066] La FIGURA 2 es un diagrama en bloque que muestra un ordenador servidor 10 como se muestra en la Figura 1. En la Figura 2, una fuente de código de aplicación informática distribuida y datos 101 incluye un compilador de aplicación y un módulo de gestión de software (no se muestra) y tiene un terminal de salida conectado a un terminal de entrada de un generador de flujo 102. Un terminal de salida del generador de flujo 102 se conecta a un terminal de entrada de un empaquetador de transporte 104. Un terminal de salida del empaquetador de transporte 104 se conecta a un primer terminal de entrada de un multiplexor de paquetes 106. Un terminal de salida del multiplexor de paquetes 106 se conecta a un terminal de entrada de un multiplexor de transporte 110. Un terminal de salida del multiplexor de transporte 110 se conecta al medio físico que forma el mecanismo de transporte 30 (de la FIGURA 1). Un segundo terminal de entrada del multiplexor de paquetes 106 se conecta a una fuente de paquetes de datos auxiliares 107. Un reloj 109 tiene los respectivos terminales de salida conectados a los correspondientes terminales de entrada del empaquetador de transporte 104 y fuente de datos auxiliares 107. Un transceptor de datos 103 tiene un primer terminal bidireccional conectado a la instalación de procesamiento central 60 (de la FIGURA 1) y un segundo dato bidireccional conectado al código de aplicación y fuente de datos 101.

[0067] El código de aplicación y la fuente de datos 101, el generador de flujo 102, el empaquetador de transporte 104, la fuente de datos auxiliares 107, el reloj 109 y el multiplexor de paquetes 106, combinados, forman una fuente de canal IOS para el mecanismo de transporte, ilustrado por un recuadro en líneas discontinuas. Otras fuentes de canal, incluyendo componentes parecidos a los mostrados en la fuente de canal 108 pero que no se muestran en la FIGURA 1, se representan con otro recuadro en líneas discontinuas 108a. Las otras fuentes de canal (108a) tienen terminales de salida conectados a otros terminales de entrada del multiplexor de transporte 110, y pueden tener terminales de entrada conectados a las instalaciones de procesamiento central a través de transceptores de datos.

[0068] Cuando funciona, se suministran los datos que representan el programa de aplicación informática distribuida, y los datos relacionados con la transmisión del programa por el mecanismo de transporte 30 al generador de flujo 102 desde la fuente de aplicación 101. Estos datos pueden suministrarse en forma de ficheros que contienen datos representando los módulos de código y de datos, o mediante secuencias de comandos que facilitan información sobre cómo construir los módulos de código y de datos, u otra información de este tipo. Los módulos de código y datos pueden ser constantes o pueden cambiar dinámicamente, en base a información recibida desde los ordenadores cliente 20 por medio de la instalación informática central 60 y/u otras fuentes. Los ficheros de módulos de código y datos ejecutables pueden ser generados por un programador de aplicaciones mediante un compilador, intérprete o ensamblador de forma conocida como respuesta a la programación del idioma fuente. El fichero de datos relacionado con la transmisión de los módulos incluye información tal como: las tasas de repetición deseadas para el directorio y los módulos de código y datos a incluir en el flujo de datos; el tamaño de la memoria principal en los ordenadores cliente 20 necesaria para almacenar cada módulo y ejecutar completamente el programa de aplicación; un nivel prioritario para el módulo, si se trata de un módulo de código, etc.

[0069] El generador de flujo 102 procesa los datos de la fuente de aplicación 101. Como respuesta, el generador de flujo 102 construye un módulo de directorio, dando una imagen conjunta del programa de aplicación. La información del módulo de directorio incluye, por ejemplo, la identificación de todos los módulos de código y de datos que se están transmitiendo repetitivamente en el flujo de datos, su tamaño y posiblemente otra información relacionada con esos módulos. Entonces, los datos representativos del programa de aplicación se procesan para generar los módulos de código y de datos. Los módulos de directorio, código y datos así contruidos se formatean añadiendo encabezamientos

de módulos y códigos de detección y/o de corrección de errores a cada módulo. También se genera un programa de transmisión. Una vez completado este procesamiento, los datos que representan el módulo de directorio y los módulos de código y datos se presentan repetitivamente al empaquetador de transporte 104 de acuerdo con el programa generado anteriormente.

5 [0070] El empaquetador de transporte 104 genera un flujo de paquetes que representan el módulo de directorio y los módulos de código y datos conforme se emiten desde el generador de flujo 102. Cada paquete tiene una longitud predeterminada constante, y se genera dividiendo el flujo de datos procedente del generador de flujo en grupos de bits, y añadiendo un encabezamiento de paquete con información que identifique la información contenida en el paquete, y un código de detección y/o de corrección de errores, etc., en cada grupo, de tal manera que cada paquete tiene la misma longitud predeterminada. (Si no se tienen suficientes datos del generador de flujo 102 para llenar completamente un paquete, el paquete se rellena con datos nulos). Estos paquetes se multiplexan en el tiempo con los paquetes de datos auxiliares, de forma conocida, para formar un solo flujo de paquetes en el multiplexor de paquetes 106. También es posible que los paquetes generados tengan varias longitudes. En este caso, el encabezamiento del paquete para cada paquete incluirá la longitud de ese paquete. Además, los paquetes de datos del código de tiempo se colocan en los paquetes del flujo de datos y/o en los paquetes de datos auxiliares en base a los datos recibidos del reloj 109.

[0071] Los flujos de paquetes procedentes de todas las fuentes de canal (108 y 108a) se multiplexan en un solo canal de transporte, que se transmite por el mecanismo de transporte 30. Como se describió anteriormente, los flujos de paquetes pueden ser multiplexadas en frecuencias haciendo que cada flujo de paquetes module una señal portadora a una frecuencia diferente, con todas las portadoras transportadas por un enlace satélite a los ordenadores cliente 20, de forma conocida. Además, si hay suficiente capacidad en un canal portador, se podrán multiplexar en el tiempo varios flujos de paquetes y usarse para modular una sola portadora, también de forma conocida. Por ejemplo, se ha propuesto temporizar múltiples de hasta ocho flujos de datos de televisión interactiva a través de un solo enlace por satélite.

[0072] Los datos procedentes de los ordenadores cliente 20 a través de la instalación de procesamiento central 60 (de la Figura 1) se reciben en el ordenador servidor 10 por medio del transceptor de datos 103, el cual puede incluir su propio procesador (no se muestra). Si se genera una respuesta inmediata, el procesador del transceptor 103 devuelve esa respuesta a través de la instalación de procesamiento central 60 a un ordenador cliente específico (22-26), un juego específico de ordenadores cliente 20 o a todos los ordenadores cliente 20 por turno. No obstante, si se desea una respuesta común a todos los ordenadores cliente 20, el programador de aplicaciones puede enmendar los ficheros de código y de datos en la fuente de código y datos de la aplicación 101 usando el compilador de aplicaciones. Luego, el generador de flujo vuelve a procesar estos ficheros rectificadas para generar otro flujo. También es posible que los ficheros de código y datos de la fuente de aplicación puedan rectificarse automática y dinámicamente (es decir, en tiempo real) como respuesta a los datos recibidos desde el transceptor 103, y que el flujo sea actualizado conforme se reciben los datos desde los ordenadores cliente 20.

[0073] La FIGURA 3 es un diagrama de temporización que muestra los flujos de datos producidos por el ordenador servidor 10 en un sistema informático distribuido como se muestra en la FIGURA 1. En la FIGURA 3 se muestra el ordenador servidor 10 produciendo simultáneamente una pluralidad de flujos de paquetes 32-38. Cada flujo de paquetes (32-38) se muestra como una banda horizontal dividida en paquetes que tienen la misma duración y número de bits. Tal como se describió arriba, es posible que el tamaño de los paquetes de un flujo de paquetes varíe con la cantidad de datos a transferir. En la FIGURA 3 puede verse que los tiempos iniciales de los paquetes no están sincronizados. Es posible sincronizar los paquetes, pero no es necesario. En la FIGURA 3, a los paquetes con datos que representan directorios se les designa DIR, a los paquetes que transmiten datos que representan los módulos de código se les designa CM, a los paquetes con datos que representan datos de módulos se les designa DM y a los paquetes con datos auxiliares se les designa AUX.

[0074] En la serie superior de paquetes 32, el paquete situado más a la izquierda contiene datos que representan un módulo de código, CM. A esto le siguen tres paquetes que contienen datos auxiliares, AUX, seguidos por otro paquete que contiene datos que representan el módulo de código, CM. En la serie de paquetes 32 puede observarse que el módulo de código se produce repetitivamente. Es posible que haya más o menos paquetes entre las repeticiones sucesivas de los paquetes de módulo de código CM. El programador puede especificar la tasa de repetición cuando se programe la aplicación, y puede variarse durante la ejecución de la aplicación.

50 [0075] En la siguiente serie de paquetes 34, el paquete situado más a la izquierda contiene datos auxiliares, AUX. Los dos paquetes siguientes contienen las partes respectivas de un módulo de código (CMI y CM2). El último paquete contiene datos auxiliares, AUX. En la serie de paquetes 34 puede observarse que si un módulo de código es demasiado grande como para ser incluido en un solo paquete, éste puede ser transferido por más de uno, en cuyo caso cada paquete incluirá una parte del módulo de código. Aunque en la serie de paquetes 34 se muestran los dos paquetes como si incluyeran el módulo de código (CMI y CM2), puede usarse cualquier cantidad de paquetes para transferir el módulo de código, según sea su tamaño. Los dos paquetes que transfieren el módulo de código (CMI y CM2) se transmiten repetitivamente (no se muestra) en la serie de paquetes 34, como se describió anteriormente.

[0076] En la serie de paquetes 36, el paquete situado más a la izquierda contiene datos que representan un módulo de código (CM). El siguiente paquete (DMI) es un primer paquete que contiene datos que representan un módulo de datos. El siguiente paquete contiene datos auxiliares, AUX. El siguiente paquete (DM2) es un segundo paquete que contiene el resto de datos que representan el módulo de datos. En la serie de paquetes 36 puede observarse que también puede incluirse un módulo de datos (DMI y DM2), asociado con el módulo de código (CM), en el flujo de datos. Tanto el módulo

de código (CM) como el módulo de datos (DMI y DM2) se transmiten repetitivamente (no se muestra) en la serie de paquetes 36. La tasa de repetición del módulo de código (CM) puede ser diferente de la del módulo de datos (DMI y DM2), y ambas tasas pueden ser especificadas por el programador de la aplicación y variarse durante la ejecución de la aplicación.

5 [0077] También puede observarse que si el módulo de datos es demasiado grande como para ser incluido en un solo paquete, éste puede ser transferido por más de un paquete, en cuyo caso cada paquete incluirá una parte del módulo de datos. Aunque en la serie de paquetes 36 se muestran los dos paquetes como si incluyeran el módulo de datos (DM1 y DM2), puede usarse cualquier cantidad de paquetes para transferir el módulo de datos, según sea su tamaño. También puede observarse que los paquetes que transfieren el módulo de datos no precisan transmitirse  
10 secuencialmente, sino que pueden tener paquetes intermedios en el flujo de paquetes. Lo mismo ocurre para los paquetes múltiples que transfieren un módulo de código o módulo de directorio (no se muestra).

[0078] En la serie más inferior de paquetes 38, el paquete situado más a la izquierda contiene datos que representan el directorio (DIR). El siguiente paquete contiene datos que representan un módulo de código (CM), seguido por un paquete que contiene datos auxiliares (AUX) y un paquete que contiene datos que representan un módulo de datos (DM). En la serie del paquete 38 puede observarse todo el módulo de un directorio (DIR), un módulo de código (CM) y un módulo de datos (DM) en una soel flujo de paquetes. Las respectivas tasas de repetición de estos tres módulos pueden ser diferentes, según lo especifique el programador de la aplicación, y pueden variarse durante la ejecución de la aplicación.

[0079] La FIGURA 4 es un diagrama en bloque de un ordenador cliente 22 como se muestra en la FIGURA 1. En la FIGURA 4, el mecanismo de transporte 30 (de la FIGURA 1) se conecta a un terminal de entrada de un selector de flujo 202. Un terminal de salida de un selector de flujo 202 se conecta a los respectivos terminales de entrada de un extractor de datos auxiliares 204 y a un extractor de datos del paquete 206. Un terminal de salida del extractor de datos auxiliares 204 se conecta al procesador de datos auxiliares 50 (de la FIGURA 1). Un terminal bidireccional del extractor de datos del paquete 206 se conecta a un terminal correspondiente de un adaptador de E/S del flujo 208. Un terminal de salida de control del adaptador de E/S del flujo se conecta a un terminal correspondiente de entrada de control del selector de flujo 202. La combinación del selector de flujo 202, el extractor de datos auxiliares 204 y el extractor de datos del paquete 206 forma un receptor de flujo de datos 207 para el ordenador cliente 22, que se muestra mediante la línea discontinua de la FIGURA 4.

[0080] El adaptador de E/S de flujo 208 forma parte de una unidad de procesamiento 224 del ordenador cliente 22, que se muestra mediante una línea discontinua en la FIGURA 4. Además del adaptador de E/S de flujo 208, la unidad de procesamiento 224 incluye un procesador 210, una memoria de lectura/escritura (RAM) 212 y una memoria de sólo lectura (ROM) 214 conectados todos ellos juntos de forma conocida a través de un bus del sistema 216. Pueden tenerse otras instalaciones de entrada y salida a través del puerto de E/S 218, que se halla conectado al procesador local 40 (de la FIGURA 1); del adaptador de E/S del usuario 220, para comunicarse con el usuario 80; y del módem 222, conectado a la instalación de procesamiento central 60 (de la FIGURA 1); todos ellos conectados también al bus del sistema 216 de forma conocida. Pueden conectarse otros adaptadores (no se muestran) al bus del sistema 216 para ofrecer otras posibilidades a la unidad de procesamiento 224.

[0081] Tal como se describe más arriba, el extractor de datos auxiliares 204, el puerto de E/S 218 y el módem 222 no son necesarios en el ordenador cliente 20 de acuerdo con la presente invención. Aparecen en la FIGURA 1 y FIGURA 4 para mostrar otra funcionalidad extra opcional.

[0082] Cuando funciona, el procesador 210 de la unidad de procesamiento 224 recupera las instrucciones del programa almacenadas permanentemente en la ROM 214, o almacenadas provisionalmente en la RAM 212, Y ejecuta las instrucciones recuperadas para leer los datos de la ROM 212 y/o RAM 214, escribir los datos en la RAM 212 y/o recibir datos de fuentes externas, o suministrar datos a fuentes externas, a través del puerto de E/S 218, adaptador de E/S del usuario 220 y/o módem 222, de forma conocida. Bajo control del programa, el procesador 210 también puede solicitar un módulo de código y/o de datos del flujo de datos suministrado al ordenador cliente 22 a través del mecanismo de transporte 30 (de la Figura 1). Para recuperar estos datos, el procesador 210 ordena primero al adaptador de E/S del flujo 208 que envíe una señal de control de selección al selector de flujo 202, posiblemente como respuesta a la información facilitada por el usuario desde el adaptador de E/S del usuario 220. Entonces, el procesador 210 solicita un módulo de código o datos específico al adaptador de E/S del flujo 208. El adaptador de E/S del flujo 208 transmite esta solicitud al extractor de datos del paquete 204.

[0083] El mecanismo de transporte 30 (de la FIGURA 1) aporta toda la pluralidad de flujos de paquete (32-38 de la FIGURA 3) que transfiere al selector de flujo 202, el cual sólo pasa el flujo de paquetes seleccionado. El extractor de datos auxiliares 204 controla el flujo de paquetes seleccionado, extrae los paquetes de datos auxiliares del mismo y los entrega directamente al procesador de datos auxiliares 50 (de la Figura 1). El extractor de datos del paquete 206 controla de igual forma el flujo de paquetes seleccionado, extrae los paquetes del módulo de directorio, código y/o datos solicitados por el adaptador de E/S del flujo 208 y los entrega al adaptador de E/S del flujo 208. Los datos de los paquetes devueltos al adaptador de E/S del flujo 208 se entregan a la RAM 212. Cuando se ha recuperado todo el módulo del flujo de paquetes (que podría necesitar varios paquetes, como se describió anteriormente), el adaptador de E/S del flujo 208 notifica al procesador 210 su recepción. Entonces, el procesador 210 puede continuar la ejecución de su programa.

[0084] El flujo de datos del sistema informático distribuido que se muestra en la FIGURA 1 es parecido a un sistema de almacenamiento masivo de los sistemas de la técnica anterior. Un programa de aplicación que se ejecute en el procesador 210 pide un módulo listado en el directorio de la misma manera que ese programa pediría un fichero que contenga un módulo de código o de datos anteriormente almacenado en un dispositivo de almacenamiento masivo en un sistema de la técnica anterior. El receptor del flujo de datos 207 es parecido a un dispositivo de almacenamiento masivo, y la E/S del flujo 208 actúa de forma parecida a un adaptador de almacenamiento masivo en un sistema de la técnica anterior localizando los datos deseados, transfiriéndolos a una ubicación predeterminada (memoria intermedia de E/S) en la memoria del sistema e informando al procesador de la realización de la recuperación. Sin embargo, el adaptador de E/S del flujo 208 sólo puede recuperar código y datos del flujo de datos. No pueden ponerse datos en el flujo de datos.

[0085] Como se describió anteriormente, la aplicación de informática distribuida puede dividirse en más de un módulo de código, cada uno de ellos incluyendo un código ejecutable para una parte diferente de la aplicación de informática distribuida. Cuando se desea un módulo de código concreto, el procesador 210 solicita ese módulo de código al adaptador de E/S del flujo 208. Cuando se ha completado la ejecución de ese módulo, el procesador 210 solicita el siguiente módulo de la E/S del flujo 208. Puesto que los módulos de código y de datos se transportan repetitivamente en el flujo de datos, puede borrarse un módulo de la RAM 212 cuando no se necesite actualmente sin la necesidad de almacenarse provisionalmente, puesto que si se requiere posteriormente, podría recuperarse de nuevo del flujo de datos cuando sea necesario. Sin embargo, si la RAM 212 tiene suficiente capacidad, el procesador 210 puede solicitar al adaptador de E/S del flujo que cargue simultáneamente varios módulos de código en la RAM 212. Si puede realizarse esto, entonces el procesador 210 puede pasar de un módulo de código a otro sin esperar que el adaptador de E/S del flujo 208 los extraiga del flujo de datos.

[0086] Tal como se describió anteriormente, pueden conectarse otros adaptadores de E/S al bus del sistema 216 de forma conocida. Por ejemplo, en un sistema de TV interactiva, puede conectarse un adaptador de gráficos al bus del sistema 216. El adaptador de gráficos genera señales que representan las imágenes gráficas, de forma conocida, como respuesta a las instrucciones recibidas del procesador 210. Además, estas señales pueden combinarse con la señal vídeo estándar producida por el decodificador del vídeo (descrito arriba) en el procesador de datos auxiliares SO de un sistema de TV interactiva. Cuando se combinan la señal representativa de la imagen gráfica y la señal vídeo estándar, la señal resultante representa una imagen en la que la imagen generada por el adaptador de gráficos se superpone a la imagen representada por la señal vídeo de emisión. También es posible combinar selectivamente estas dos señales representativas de imagen bajo el control del procesador 210.

[0087] Un sistema de TV interactiva también puede tener un adaptador acústico conectado al bus del sistema 216. El adaptador acústico genera una señal que representa un sonido generado por ordenador (como música, voz sintetizada u otro sonido), de forma conocida, como respuesta a las instrucciones del procesador 210. Además, estas señales pueden combinarse con la señal audio estándar producida por el decodificador audio (descrito anteriormente) en el procesador de datos auxiliares SO de un sistema de TV interactiva. Cuando se combinan la señal representativa de audio y la señal audio estándar, la señal resultante representa la combinación del sonido generado por el adaptador acústico y la señal audio de emisión. También es posible combinar selectivamente estas dos señales representativas de sonido bajo el control del procesador 210.

[0088] La temporización de la generación y visualización de las señales representativas de la imagen gráfica y del sonido puede controlarse mediante la recepción de los datos del código de tiempo del flujo de datos. Esto permite que un módulo de código ejecutable sincronice la visualización de la imagen generada por el procesador y la presentación del sonido generado por el procesador al vídeo y audio de emisión. También es posible sincronizar la operación de la aplicación de TV interactiva mediante la inserción de paquetes especializados en el flujo de datos, lo que provoca una interrupción del código que se ejecute actualmente en el procesador 210. La E/S de flujo 208 supervisa la aparición de dichos paquetes especializados en el flujo de datos, y genera una interrupción, de forma conocida, para el procesador 210. El procesador 210 responde a esa interrupción, también de forma conocida, ejecutando una rutina de servicio de interrupción (ISR). Esta ISR puede usarse para sincronizar la aplicación de TV interactiva, o para otros fines.

[0089] Un ordenador cliente 22 de un sistema informático distribuido como el que se muestra en la FIGURA 1 no necesita un dispositivo de almacenamiento masivo, ni una gran cantidad de RAM 212. Un sistema así reduce el coste de un ordenador cliente, y aumenta la funcionalidad de los ordenadores cliente de coste más bajo. Además, un ordenador cliente de este tipo tiene la opción de participar en una función informática distribuida, puede incorporarse a la función informática distribuida en cualquier momento (o puede salirse y volver más tarde), y puede participar a su propio ritmo.

**REIVINDICACIONES**

1. Un sistema que incluye:  
un ordenador servidor (10) que incluye:  
5 un empaquetador (104) para generar una primera corriente de paquete que incluye un módulo de código (CM1;CM2) que representa una aplicación informática distribuida;  
una fuente de señales auxiliares (107) para generar un paquete de datos auxiliares; y  
un multiplexor (106) conectado al empaquetador (104) para recibir el primer flujo de paquete, incluyendo el módulo de código (CM1;CM2), y conectado a la fuente de datos auxiliares (107) para recibir el flujo de paquete  
10 de datos auxiliares, incluyendo los paquetes de video;  
teniendo el empaquetador (104) que incluir repetitivamente el módulo de código (CM1;CM2) dentro del primer flujo de paquete, y el multiplexor (106) tiene que multiplexar el primer flujo de paquete con el flujo de paquete de datos auxiliares para generar un segundo flujo de paquete.
2. Sistema según la reivindicación 1, donde el multiplexor (106) tiene que multiplexar en el tiempo el primer flujo de paquete con el flujo de paquete de datos auxiliares para generar el segundo flujo de paquete.
3. Sistema según cualquiera de las reivindicaciones 1-2, que incluye un constructor de flujo (102), conectado al empaquetador (104), repetitivamente para presentar datos asociados con la aplicación informática distribuida al empaquetador (104) para inclusión dentro del primer flujo de paquete.
4. Sistema según la reivindicación 3, que incluye una fuente de aplicación (101), conectado al constructor de flujo (102), para suministrar los datos asociados a la aplicación de computación distribuida al constructor de flujo (102).
5. Sistema según la reivindicación 4, donde la fuente de aplicación (101) incluye un compilador de aplicación para compilar los datos asociados a la aplicación informática distribuida.
6. Sistema según cualquiera de las reivindicaciones 4-5, donde la fuente de aplicación (101) tiene que suministrar datos relacionados con la transmisión de la aplicación informática distribuida al constructor de flujo (102), incluyendo los datos relacionados con la transmisión de la aplicación informática distribuida un índice de repetición deseado para el módulo de código (CM1;CM2) en el primer flujo de datos.
7. El sistema según la reivindicación 3, donde el constructor de flujo (102) tiene que construir el módulo de código (CM1;CM2), que utiliza los datos asociados a la aplicación informática suministrados por la fuente de aplicación (101).
8. Sistema según la reivindicación 3, donde el constructor de flujo (102), utilizando los datos asociados a la aplicación informática distribuida, tiene que construir un módulo de datos que contiene datos por ser procesados por el código incluido en el módulo de código (CM1;CM2), y teniendo el empaquetador (104) que generar el primer flujo de paquete para incluir el módulo de datos.
9. Sistema según la reivindicación 8, donde el empaquetador (104) tiene que incluir repetitivamente el módulo de datos en el primer flujo de paquete.
10. Sistema según la reivindicación 8, donde el constructor de flujo (102), utilizando los datos asociados a la aplicación informática distribuida, tiene que construir un módulo de directorio que incluye una lista que identifica el código y módulo de datos como incluidos en el primer flujo de paquete, y teniendo el empaquetador (104) que generar la primera corriente de paquete para incluir el módulo de directorio.
11. Sistema según la reivindicación 10, teniendo el empaquetador (104) que incluir repetitivamente el módulo de directorio en el primer flujo de paquete.
12. Sistema según cualquiera de las reivindicaciones precedentes, donde la fuente de datos auxiliares (107) tiene que proporcionar los paquetes de video para incluir datos codificados MPEG que representan una imagen televisiva.
13. Sistema según cualquiera de las reivindicaciones precedentes, donde la fuente auxiliar (107) tiene que proporcionar el flujo de paquete de datos auxiliares para incluir paquetes audio con datos audio digitalmente codificados.
14. Sistema según cualquiera de las reivindicaciones precedentes, donde el empaquetador (104), la fuente de datos auxiliares (107) y el multiplexor (106) comprenden una primera fuente (108), incluyendo el sistema una segunda fuente (108A) y un multiplexor de transporte (110), el multiplexor de transporte (110) conectado para recibir el segundo flujo de paquete desde la primera fuente (108) y un tercer flujo de paquete de la segunda fuente (108A), y para multiplexar el segundo flujo de paquete y el tercer flujo de paquete para transmisión por un mecanismo de transporte (30) como un cuarto flujo de paquete.

15. Sistema según la reivindicación 14, donde el multiplexor de transporte (110) tiene que multiplexar en frecuencia el segundo y tercer flujo de paquete para transmisión por el mecanismo de transporte (30).
16. Sistema según la reivindicación 14, donde el multiplexor de transporte (110) tiene que multiplexar en el tiempo el segundo y tercer flujo de paquete para transmisión por el mecanismo de transporte (30).
- 5 17. Sistema según cualquiera de las reivindicaciones precedentes, que incluye un reloj (109) conectado para proporcionar datos de código de tiempo al empaquetador (104) y la fuente de datos auxiliares (107), teniendo los datos de código de tiempo que ser incluidos en uno o ambos flujos de paquete de datos auxiliares y el primer flujo de datos de paquete.
- 10 18. Sistema según cualquiera de las reivindicaciones precedentes, donde el empaquetador (104) tiene que incluir primeros paquetes en el primer flujo de datos para incluir información de identificación que indique que los primeros paquetes incluyen por lo menos una parte del módulo de código (CM1;CM2) que representa la aplicación informática distribuida.
- 15 19. Sistema según cualquiera de las reivindicaciones precedentes, donde la fuente de datos auxiliares (107) tiene que incluir segundos paquetes en la corriente de paquete de datos auxiliares para incluir información de identificación indicando que los segundos paquetes incluyen datos de video.
- 20 20. Sistema según cualquiera de las reivindicaciones precedentes, que incluye un transceptor de datos (103) por el que los datos son recibidos en el servidor (10) desde un ordenador de cliente (22) por medio de una instalación de procesamiento central (60).
- 20 21. Sistema según la reivindicación 20, donde el ordenador de cliente (22) tiene que acceder el servidor (10) por medio de la instalación de tratamiento central (60) para recuperar el módulo de código (CM1;CM2).
22. Sistema según la reivindicación 20, donde la instalación de tratamiento central (60) es un sistema telefónico.
- 25 23. Sistema según cualquiera de las reivindicaciones 20-22, en el que el servidor (10) recibe y procesa los datos recibidos en el servidor (10) del ordenador de cliente (22) durante ejecución de la aplicación informática distribuida por el ordenador de cliente (22).
24. Sistema según cualquiera de las reivindicaciones 20-23, en el que el servidor (10) genera el módulo de código (CM1 CM2) basado en los datos recibidos en el servidor (10) desde el ordenador de cliente (22).
25. Sistema según cualquiera de las reivindicaciones 20-24, donde el servidor (10) genera el módulo de datos en base a los datos recibidos en el servidor (10) desde el ordenador de cliente (22).
- 30 26. Sistema según cualquiera de las reivindicaciones precedentes, que incluye un ordenador de cliente (22), el ordenador de cliente (22) incluyendo un receptor de flujo de datos (207) para recibir un flujo de datos recibido y para extraer el módulo de código (CM1;CM2) que representa la aplicación informática distribuida desde el flujo de datos recibido.
- 35 27. Sistema según la reivindicación 26, en el que el ordenador de cliente (22) incluye una unidad de procesamiento (224), la unidad de procesamiento (224) para ejecutar la aplicación informática distribuida.
28. Sistema según cualquiera de las reivindicaciones 26-27, en el que el ordenador de cliente (22) incluye una memoria para almacenar al menos una parte de un primer módulo de código (CM1;CM2) de una pluralidad de módulo de código (CM1;CM2)s, como extraídos del flujo de datos recibido, antes de recibir completamente el primer módulo de código (CM1;CM2).
- 40 29. Sistema según la reivindicación 28, en el que el ordenador de cliente (22) tiene que ejecutar el primer módulo de código (CM1;CM2) cuando recibido completamente.
30. Sistema según la reivindicación 28, en el que la memoria tiene que almacenar al menos el primer y un segundo módulo de código (CM1;CM2) de la pluralidad de módulo de código (CM1;CM2)s, en el que el ordenador de cliente (22) tiene cambiar entre la ejecución del primer y segundo módulo de código (CM1;CM2)s.
- 45 31. Sistema según la reivindicación 26, donde el receptor de flujo de datos (207) incluye un extractor de datos de paquete (206) a extraer paquetes de datos, de en el flujo de datos recibido, que incluyen el módulo de código (CM1;CM2).
32. Sistema según la reivindicación 31, donde el extractor de paquete (206) es a identificar los paquetes de datos que incluido el módulo de código (CM1;CM2) utilización de información de identificación incluido en los paquetes de datos.
- 50 33. Sistema según la reivindicación 8, incluir un ordenador de cliente (22), el ordenador de cliente (22) incluyendo un receptor de flujo de datos (207) a recibir un flujo de datos recibido y a extraer el módulo de datos del flujo de datos recibido.
34. Sistema según la reivindicación 33, donde el ordenador de cliente (22) incluye una memoria para almacenar al

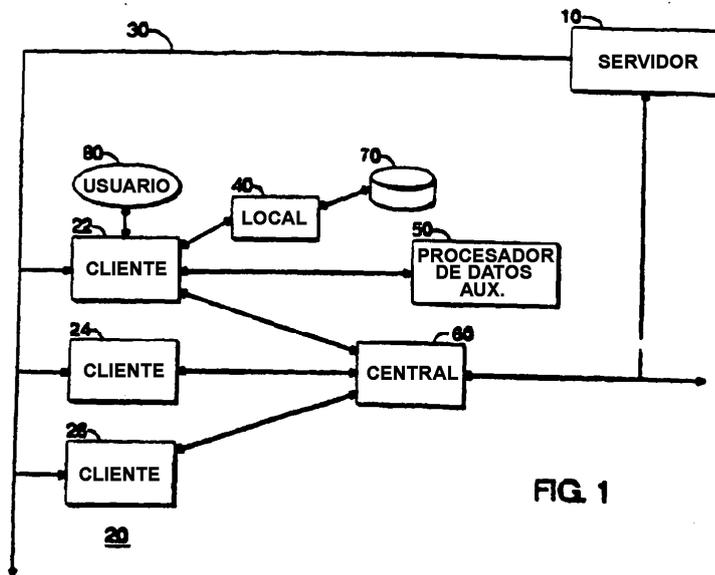
menos una parte de un primer módulo de datos de una pluralidad de módulos de datos, como extraídos del flujo de datos recibido, antes de recibir completamente el primer módulo de datos.

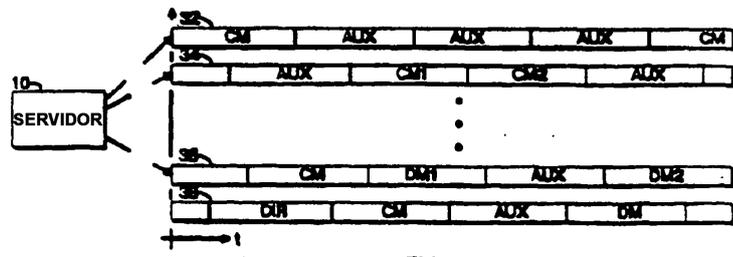
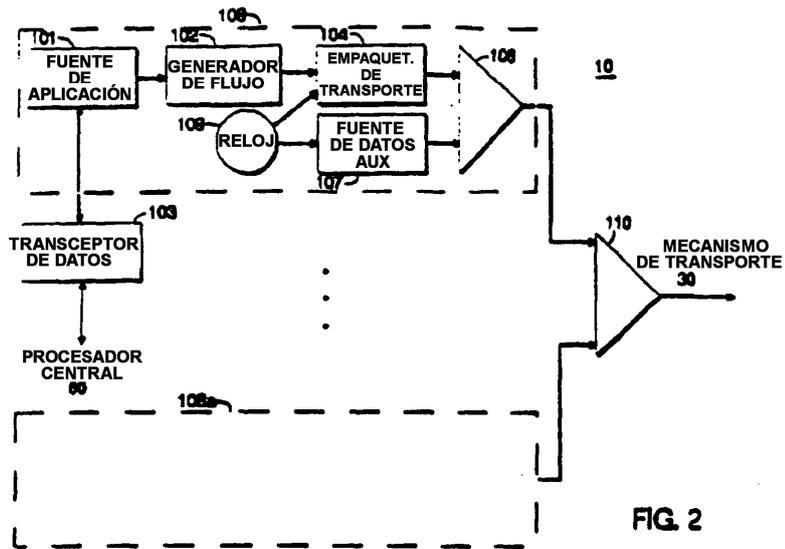
- 5 35. Sistema según la reivindicación 34, donde el ordenador de cliente (22) tiene que ejecutar un primer módulo de código (CM1;CM2) cuando el primer módulo de datos requerido por el primer módulo de código (CM1;CM2) es recibido completamente.
36. Sistema según la reivindicación 33, donde el receptor de flujo de datos (207) incluye un extractor de datos de paquete (206) para extraer paquetes de datos desde dentro del flujo de datos recibido que incluía el módulo de datos.
- 10 37. Sistema según la reivindicación 36, en el que el extractor de paquete (206) tiene que identificar los paquetes de datos que incluyen el módulo de datos que utiliza información de identificación incluida en los paquetes de datos.
38. Sistema según la reivindicación 10, que incluye un ordenador de cliente (22), incluyendo el ordenador de cliente (22) un receptor de flujo de datos (207) para recibir un flujo de datos recibido y para extraer el módulo de directorio del flujo de datos recibido.
- 15 39. Sistema según la reivindicación 38, donde el ordenador de cliente (22) incluye una memoria para almacenar al menos un primer módulo de directorio de un conjunto de módulos de directorio, como extraídos del flujo de datos recibido, antes de recibir completamente el conjunto de módulos de directorio, el conjunto de módulos de directorio especificando un código asociado de identificación de directorio y módulo de datos incluido en el flujo de datos recibido.
- 20 40. Sistema según la reivindicación 39, donde el ordenador de cliente (22) tiene que pedir el código asociado y módulo de datos identificado por el directorio.
41. Sistema según la reivindicación 38, donde el receptor de flujo de datos (207) incluye un extractor de datos de paquete (206) para extraer paquetes de datos desde dentro del flujo de datos recibido que incluía el módulo de directorio.
- 25 42. Sistema según la reivindicación 41, en el que el extractor de paquete (206) tiene que identificar los paquetes de datos que incluían el módulo de directorio que utiliza la información de identificación incluida en los paquetes de datos.
43. Sistema según cualquiera de las reivindicaciones 26-42, en el que el receptor de flujo de datos (207) incluye un extractor de datos auxiliares (204) para extraer paquetes de video del flujo de datos recibido.
- 30 44. Sistema según cualquiera de las reivindicaciones 26-43, donde el receptor de flujo de datos (207) incluye un selector de flujo (202) para seleccionar el flujo de datos recibido para recepción del receptor de flujo de datos (207).
45. Sistema según la reivindicación 44, donde el selector de flujo de datos (202) comprende un demodulador para demodular la recepción de una señal portadora modulada de un enlace de satélite.
- 35 46. Sistema según la reivindicación 45, donde el selector de flujo de datos (202) comprende un conjunto de circuitos para demultiplexar en el tiempo una pluralidad de flujos de datos portados por la señal portadora modulada.
47. Método para distribuir una aplicación informática, el método incluyendo:  
 40 generar una primera corriente de paquete que incluye repetitivamente un módulo de código (CM1;CM2) que representa una aplicación informática distribuida;  
 generar un flujo de paquetes de datos auxiliares que incluye paquetes de vídeo; y  
 multiplexar el primer flujo de paquete con el segundo flujo de paquete de datos auxiliares para generar un segundo flujo de paquete.
- 45 48. Método según la reivindicación 47, que incluye multiplexar en el tiempo la primera corriente de paquete con la corriente de paquete de datos auxiliares para generar la segunda corriente de paquete.
49. Método según cualquiera de las reivindicaciones 47-48, que incluye presentar repetitivamente datos asociados a la aplicación informática distribuida para inclusión en la primera corriente de paquete.
50. Método según la reivindicación 49, que incluye la compilación de datos asociados a la aplicación informática distribuida.
- 50 51. Método según cualquiera de las reivindicaciones 47-50, que incluye el suministro de datos relacionados con la transmisión de la aplicación informática distribuida, los datos relacionados con la transmisión de la aplicación informática distribuida incluyendo un índice de repetición deseado para el módulo de código (CM1;CM2) en el primer flujo de datos.
52. Método según la reivindicación 49, que incluye la construcción de un módulo de código (CM1;CM2) utilizando

los datos asociados a la aplicación informática distribuida.

53. Método según la reivindicación 49, que incluye la utilización de datos asociados a la aplicación informática distribuida para construir un módulo de datos con datos por ser procesados por código incluido en el módulo de código (CM1;CM2), y la generación del primer flujo de paquete para incluir el módulo de datos.
- 5 54. Método según la reivindicación 53, que incluye incluir repetitivamente el módulo de datos en el primer flujo de paquete.
55. Método según la reivindicación 53, que incluye la utilización de datos asociados a la aplicación informática distribuida para construir un módulo de directorio que incluye una lista de módulos de datos y código incluidos en el primer flujo de paquete, y la generación del primer flujo de paquete para incluir el módulo de directorio.
- 10 56. Método según la reivindicación 55, que incluye incluir repetitivamente el módulo de directorio en el primer flujo de paquete.
57. Método según cualquiera de las reivindicaciones 47-56, que incluye el suministro de paquetes de video para incluir datos codificados MPEG que representan una imagen televisiva.
- 15 58. Método según cualquiera de las reivindicaciones 47-57, que incluye el suministro de flujo de paquete de datos auxiliares para incluir paquetes audio con datos audio digitalmente codificados.
59. Método según cualquiera de las reivindicaciones 47-48, en el que una primera fuente (108) genera el segundo flujo de paquete y una segunda fuente (108A) genera un tercer flujo de paquete, el método incluyendo la recepción del segundo flujo de paquete desde la primera fuente (108), la recepción del tercer flujo de paquete desde la segunda fuente (108A), y multiplexión del segundo flujo de paquete y el tercer flujo de paquete para transmisión por un mecanismo de transporte (30) como un cuarto flujo de paquete.
- 20 60. Método según la reivindicación 59, que incluye la multiplexión en frecuencias del segundo y tercer flujos de paquete para transmisión por el mecanismo de transporte (30).
61. Método según la reivindicación 59, que incluye la multiplexión en el tiempo del segundo y tercer flujos de paquete para transmisión por el mecanismo de transporte (30).
- 25 62. Método según cualquiera de las reivindicaciones 47-61, que incluye el suministro de datos de código de tiempo en cualquiera de o los dos flujos de paquete de datos auxiliares y el primer flujo de datos de paquete.
63. Método según cualquiera de las reivindicaciones 47-62, que incluye el suministro de primeros paquetes en el primer flujo de datos para incluir información de identificación que indica que los primeros paquetes incluyen por lo menos una parte del módulo de código (CM1;CM2) que representan la aplicación informática distribuida.
- 30 64. Método según cualquiera de las reivindicaciones 47-63, que incluye el suministro de segundos paquetes en el flujo de paquete de datos auxiliares para incluir información de identificación que indica que los segundos paquetes incluyen paquetes de video.
65. Método según cualquiera de las reivindicaciones 47-64, que incluye la recepción de datos en un servidor (10) desde un ordenador de cliente (22) por medio de una instalación de tratamiento central (60).
- 35 66. Método según la reivindicación 65, que incluye el acceso al servidor (10) utilizando el ordenador de cliente (22), por medio de la instalación de tratamiento central (60), para recuperar el módulo de código (CM1;CM2).
67. Método de cualquiera de las reivindicaciones 65-66, donde la instalación de tratamiento central (60) es un sistema telefónico.
- 40 68. Método según la reivindicación 65, que incluye el tratamiento de datos recibidos en el servidor (10) desde el ordenador de cliente (22) durante ejecución de la aplicación informática distribuida por el ordenador de cliente (22).
69. Método según la reivindicación 65, que incluye la generación del módulo de código (CM1;CM2) en el servidor (10) basado en los datos recibidos en el servidor (10) desde el ordenador de cliente (22).
- 45 70. Método según la reivindicación 65, que incluye la generación del módulo de datos en el servidor basado en los datos recibido en el servidor (10) desde el ordenador de cliente (22).
71. Método según la reivindicación 47, que incluye la recepción de un flujo de datos recibido en un ordenador de cliente (22) y la extracción del módulo de código (CM1;CM2) que representa la aplicación informática distribuida del flujo de datos recibido.
- 50 72. Método según la reivindicación 71, que incluye la ejecución de la aplicación informática distribuida en el ordenador de cliente (22).
73. Método según la reivindicación 72 donde el ordenador de cliente (22) incluye una memoria, incluyendo el método el almacenamiento de al menos una parte del módulo de código (CM1;CM2) en la memoria, como extraído del flujo de datos recibido, antes de recibir completamente el módulo de código (CM1;CM2).

74. Método según la reivindicación 73, que incluye la ejecución del módulo de código (CM1;CM2) cuando completamente recibido y el ordenador de cliente (22).
- 5 75. Método según la reivindicación 73, donde la memoria tiene que almacenar al menos un primer y un segundo módulo de código (CM1;CM2), el método incluyendo el intercambio entre la ejecución del primer y segundo módulo de código (CM1;CM2)s en el ordenador de cliente (22).
76. Método según la reivindicación 71, que incluye la utilización de un extractor de datos de paquete (206) para extraer paquetes de datos desde dentro del flujo de datos recibido que incluye el módulo de código (CM1;CM2).
- 10 77. Método según la reivindicación 76, que incluye la identificación de los paquetes de datos que incluían el módulo de código (CM1;CM2) utilizando la información de identificación incluida en los paquetes de datos.
78. Método según la reivindicación 53, que incluye la recepción de un flujo de datos recibido en un ordenador de cliente (22) y la extracción del módulo de datos del flujo de datos recibido.
- 15 79. Método según la reivindicación 78, en el que el ordenador de cliente (22) incluye una memoria, incluyendo el método el almacenamiento de al menos una parte del módulo de datos, como extraído del flujo de datos recibido, antes de recibir completamente el módulo de datos.
80. Método según la reivindicación 79, que incluye la ejecución de un primer módulo de código (CM1;CM2) cuando el módulo de datos requerido por el primer módulo de código (CM1;CM2) es recibido completamente.
81. Método según la reivindicación 78, que incluye la utilización de un extractor de datos de paquete (206) para extraer paquetes de datos desde dentro del flujo de datos recibido que incluía el módulo de datos.
- 20 82. Método según la reivindicación 81, que incluye la identificación de los paquetes de datos que incluyen el módulo de datos que utilizan la información de identificación incluida en los paquetes de datos.
83. Método según la reivindicación 55, que incluye la recepción de un flujo de datos recibido en un ordenador de cliente (22) y la extracción del módulo de directorio desde el flujo de datos recibido.
- 25 84. Método según la reivindicación 83 en el que el ordenador de cliente (22) incluye una memoria, el método incluyendo el almacenamiento de al menos un primer módulo de directorio de un conjunto de módulos de directorio, como extraído del flujo de datos recibido, antes de recibir completamente el conjunto de módulos de directorio, incluyendo el conjunto de módulos de directorio un código asociado de identificación de directorio y módulos de datos incluidos en el flujo de datos recibido.
- 30 85. Método según la reivindicación 84, que incluye la petición del código asociado y módulos de datos identificados por el directorio que utiliza el ordenador de cliente (22).
86. Método según la reivindicación 83, que incluye la extracción de paquetes de datos desde dentro del flujo de datos recibido que incluía el módulo de directorio.
87. Método según la reivindicación 86, que incluye la identificación de los paquetes de datos que incluían el módulo de directorio que utiliza la información de identificación incluida en los paquetes de datos.
- 35 88. Método según cualquiera de las reivindicaciones 47-87, que incluyen la extracción de paquetes de video del flujo de datos recibido.
89. Método según la reivindicación 71, que incluyen la selección del flujo de datos recibido para recepción por un ordenador de cliente (22).
- 40 90. Método según la reivindicación 89, en el que la selección incluye demodulación de una señal portadora modulada recibida desde un enlace satelital.
91. Método según la reivindicación 89, en el que la selección incluye la demultiplexión en el tiempo de una pluralidad de flujos de datos portados por la señal portadora modulada.





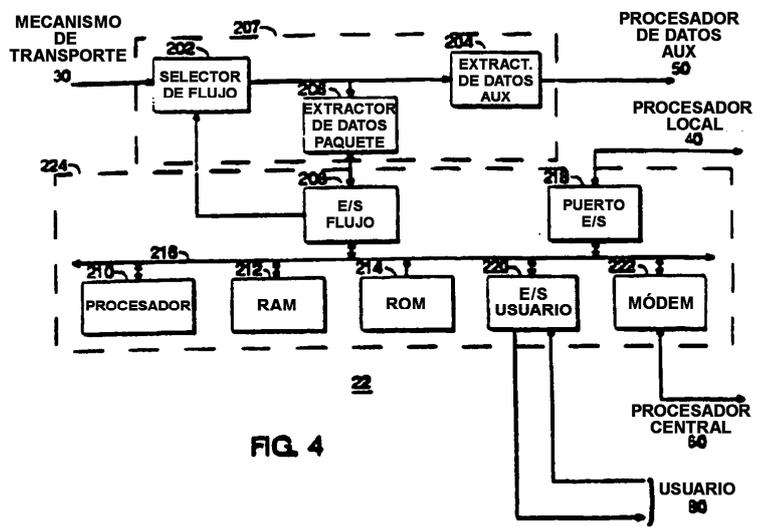


FIG. 4