

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 557 295**

51 Int. Cl.:

G06F 15/16 (2006.01)

H04L 29/08 (2006.01)

H04L 29/06 (2006.01)

A63F 13/30 (2014.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **18.04.2006 E 06750476 (1)**

97 Fecha y número de publicación de la concesión europea: **28.10.2015 EP 1877919**

54 Título: **Sistema y método para la distribución selectiva de información**

30 Prioridad:

28.04.2005 US 675624 P

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

25.01.2016

73 Titular/es:

**ARENANET, LLC (100.0%)
3180 139th Ave SE 500
Bellvue, WA 98005, US**

72 Inventor/es:

**O'BRIEN, MIKE F.;
STRAIN, WILLIAM J. y
WYATT, PATRICK J.**

74 Agente/Representante:

VALLEJO LÓPEZ, Juan Pedro

ES 2 557 295 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 y método para la distribución selectiva de información

5 El género de los juegos en línea multijugador masivos (MMOG) es cada vez más popular.

Muchos usuarios disfrutan realmente de la idea de jugar estos juegos con y contra miles de otros usuarios. Los usuarios también tienden a apreciar el aspecto de "evasión" de muchos MMOG. Los desarrolladores de juegos pasan incontables horas creando enormes mundos en línea con niveles de interactividad y de detalle antes nunca vistos.

Aunque los mundos en línea que permiten a los usuarios evadirse son populares, estos juegos tienden a ser extraordinariamente complejos. Hay varios desafíos asociados a la creación y el equilibrio de una sociedad virtual para miles de jugadores. Y los desafíos no terminan una vez que el juego se ha desarrollado. Una vez desarrollados, muchos de los juegos todavía necesitan gestionarse de manera activa.

Aunque se dedica una gran cantidad de tiempo a hacer juegos mejores, más grandes, y más detallados, se dedica muy poco tiempo a abordar los retos asociados a la gestión real del juego de los juegos (por ejemplo, la distribución, la ejecución, el soporte y la actualización de los MMOG).

El documento US5898834 desvela un sistema de procesamiento diversificado y un método asociado para actualizar al menos una plataforma informática remota con una acción que se produce en una plataforma informática local. El sistema y el método desvelados minimizan la cantidad de información generada localmente que necesita transferirse a través de una red con el fin de mantener en una plataforma informática remota una copia de lo que se ha producido localmente. Para minimizar la información enviada entre las plataformas, cada plataforma puede contener unos elementos, denominados instancias, que responden a diversos tipos de comandos. Los tipos de instancias incluyen instancias master, proxy y free. Las instancias master y las instancias proxy se refieren al mismo elemento localizado en diferentes plataformas. Cada instancia master mantiene sus instancias proxy relacionadas actuales con su estado transmitiendo sus cambios entre las plataformas a sus instancias proxy relacionadas. Las instancias free responden a los cambios que se producen preferentemente en una sola plataforma. Al minimizar la cantidad de información que se necesita transmitir entre las plataformas, se reduce el ancho de banda consumido para actualizar diversas plataformas.

Avi Bar-Zeev, "Nomad Programming: Scenegraps: Past, Present and Future", <http://developer.nomadph.com/guest004.html> (consultado el 7 de febrero de 2012) desvela los conceptos básicos de los grafos de escena.

En consecuencia, en un primer aspecto, la presente invención proporciona un método de acuerdo con la reivindicación 1.

Breve descripción de los dibujos

Se apreciará que por motivos de simplicidad y claridad de la ilustración, los elementos ilustrados en las figuras no se han dibujado necesariamente a escala. Por ejemplo, las dimensiones de algunos de los elementos se exageran en relación con otros elementos. Las realizaciones que incorporan enseñanzas de la presente divulgación se muestran y se describen con respecto a los dibujos presentados en el presente documento, en los que:

La figura 1 representa un sistema que distribuye selectivamente información de acuerdo con las enseñanzas de la presente divulgación;

La figura 2 muestra una representación simplificada de un grafo acíclico dirigido que representa diversos elementos de programación y su interrelación de una manera que incorpora al menos una parte de las presentes enseñanzas; y

La figura 3 presenta un diagrama de flujo para una técnica de distribución selectiva que incorpora al menos alguna de las presentes enseñanzas.

El uso de los mismos símbolos de referencia en diferentes dibujos indica elementos similares o idénticos.

Descripción detallada de los dibujos

Como se ha indicado anteriormente, la siguiente descripción en combinación con las figuras se proporciona para ayudar a comprender las enseñanzas desveladas en el presente documento. La siguiente exposición se centrará en las implementaciones y realizaciones específicas de las enseñanzas. Este enfoque se proporciona para ayudar a describir las enseñanzas y no debe interpretarse como una limitación en el alcance o la aplicabilidad de las enseñanzas. Por ejemplo, gran parte de lo siguiente se centra en la distribución de actualizaciones y/o cualquier otra

información en relación con el funcionamiento de un MMOG. Aunque las enseñanzas pueden utilizarse sin duda en esta aplicación, las enseñanzas también pueden utilizarse en otras aplicaciones y con varios tipos de arquitecturas diferentes, tales como las arquitecturas de computación distribuida, las arquitecturas cliente/servidor o las arquitecturas servidor middleware.

5 En el contexto de los juegos, una técnica que incorpora al menos alguna de las presentes enseñanzas pueden incluir, por ejemplo, recibir una comunicación desde una aplicación cliente que se ejecuta en un dispositivo remoto. La comunicación puede indicar una condición de un personaje del juego con el que se está jugando en un juego en línea multijugador masivo (MMOG). El método también puede incluir desarrollar un manifiesto filtrado que incluye
10 una lista de archivos necesaria para la aplicación cliente y asociada a la condición. Una parte de la recopilación puede identificarse para la comunicación a la aplicación cliente, y puede iniciarse la comunicación de la parte.

15 En una realización, la información a comunicar puede incluir un archivo de geometría, un archivo de textura, un archivo de sonido y un archivo de animación. Los archivos pueden asociarse con alguna condición específica del personaje, tal como una localización de personaje actual, una localización de personaje prevista, y/o una interacción de personajes. En la práctica, un proceso de identificar qué archivos son necesarios para un usuario remoto puede facilitarse por el mantenimiento de un grafo de dependencia que mapea una pluralidad de elementos y/o localizaciones de juego con archivos de geometría, archivos de textura, archivos de sonido y archivos de animación asociados. En una implementación específica, el grafo de dependencia puede ser un grafo acíclico dirigido (DAG).

20 Como se ha indicado anteriormente, la figura 1 representa un sistema 100 que distribuye selectivamente información de acuerdo con las enseñanzas desveladas en el presente documento. Como se muestra, una plataforma informática 102 está configurada para servir un juego en línea a varios clientes remotos que se ejecutan en dispositivos informáticos remotos (representados en los ordenadores 104, 106, y 108). En la realización representada del sistema 100, una interfaz 110 está asociada a la plataforma informática 102 y está configurada para recibir una señal de actualización de uno o más de los clientes remotos. Como se muestra, la plataforma 102 también puede incluir un motor de actualización 112 que responde a una señal de actualización recibida. La señal de actualización puede ser, por ejemplo, una solicitud explícita de actualizaciones. La señal de actualización también puede ser una comunicación que indica una situación (por ejemplo, una condición de un personaje del juego) que se
25 interpreta y/o se reconoce que indica una necesidad de actualizaciones.

30 Durante el funcionamiento, el motor de actualización 112 puede acoplarse comunicativamente a un almacén de datos 114 que mantiene la información implicada durante el juego de un MMOG. Por ejemplo, un usuario en el ordenador 104 puede estar jugando con un personaje en un juego servido por la plataforma 102. El personaje del usuario puede trasladarse de una parte de un mundo en línea a otra parte. Una comunicación que indica esta transición puede representar la señal de actualización y puede indicar al motor de actualización 112 qué información adicional es o puede ser necesaria para el usuario en el ordenador 104.

35 Si el ordenador 104 no tiene la información que necesita para soportar la transición del personaje a una nueva parte del mundo en línea, puede reconocerse esta deficiencia y puede enviarse la información necesaria al ordenador 104. En la práctica, el motor de actualización 112 puede acceder al menos a una lista parcial de la información disponible para el MMOG determinado que se está jugando, puede seleccionar una parte de la información para su salida, y puede iniciar una salida de la parte al ordenador 104.

40 En función del detalle de implementación, el motor de actualización 112 puede implementarse, al menos parcialmente, por un procesador y un medio legible por ordenador manteniendo unas instrucciones operativas para dirigir el procesador para reconocer la señal de actualización, para acceder a la lista parcial, para seleccionar la parte, y para iniciar la salida. En una realización específica, el proceso de selección de la parte para su salida puede incluir un proceso de filtrado. Tal proceso podría implicar, por ejemplo, la comunicación de una parte de un grafo acíclico dirigido (DAG) al ordenador 104. La parte podría ser la parte de un DAG más grande que se asocia con una condición de un personaje con el que juega el usuario en el ordenador 104. El proceso de filtrado también puede implicar la recepción de una comunicación desde el ordenador 104 que indica al menos un archivo del DAG que necesita el usuario. En la práctica, los archivos necesitados realmente por el ordenador 104 pueden identificarse considerando una lista de la información pertinente y reduciendo la lista para eliminar una entrada ya almacenada en
45 el ordenador 104.

50 Por ejemplo, los archivos asociados a un árbol determinado en el mundo en línea pueden usarse en más de una parte del mundo (por ejemplo, el "mismo" árbol puede estar en el fondo de varias escenas diferentes). Como resultado, incluso si el motor de actualización 112 "sabe" que un personaje se está trasladando a una nueva zona y que la nueva zona incluye un árbol determinado, el motor de actualización 112 puede no "saber" que el ordenador 104 ya tiene archivos asociados al árbol, debido a que el mismo árbol apareció en una escena anterior. Para reducir descargas innecesarias, el ordenador 104 puede ayudar al motor de actualización 112 reduciendo una lista de archivos necesarios y/o desarrollando realmente la lista de los archivos necesarios.

65 Como se ha descrito anteriormente, se presentan a los usuarios varios MMOG modernos con un mundo virtual muy detallado y amplio. En la práctica, alguna parte del mundo virtual puede descargarse inicialmente a un usuario a

través de una red de área amplia 116. La descarga inicial puede incluir suficientes archivos para permitir jugar con el personaje en una parte inicial de un mundo virtual. La descarga inicial también podría ser un núcleo que es insuficiente por sí solo para permitir jugar con el personaje. Sin embargo, dicho núcleo puede facilitar una descarga posterior de la información necesaria para jugar.

5 En la práctica, para descargar el mundo completo para trabajar de manera eficaz, el mundo virtual puede necesitar ser relativamente pequeño. Para producir un mundo virtual lo suficientemente pequeño para descargar, los desarrolladores pueden necesitar tomar algunos atajos como sustituir los mapas de textura pintados a mano por unos procesados, usar más modelos procesados, reducir el número de efectos de sonido y/o cortar escenas de animación sobre la marcha.

10 En algunos casos estos cambios pueden reducir la calidad percibida del juego. Para mantener la calidad, algunos desarrolladores pueden elegir, en cambio, distribuir al menos una parte de un cliente de juego en un disco compacto (CD) o un disco de video digital (DVD) comprados. La utilización de un CD o un DVD puede permitir que un desarrollador comunique inicialmente cantidades más grandes de información. Sin embargo, dados los mapas de textura, los modelos 3D, los efectos de sonido, la voz grabada, etc., de los MMOG modernos, puede ser difícil y/o de un coste prohibitivo incluir los suficientes CD para representar completamente un gran mundo virtual.

15 Como tal, puede utilizarse un sistema como el representado en el sistema 100 para mejorar la capacidad de los desarrolladores para actualizar dinámicamente un juego añadiendo nuevos contenidos, modificando el código del lado del cliente, eliminando contenidos existentes, y/o cambiando contenidos. Utilizando el sistema 100, estas actualizaciones pueden producirse a través de enlaces de comunicación en oposición a o además de la distribución de nuevos CD.

20 Como se muestra en el sistema 100, un ordenador remoto como el ordenador 104 puede acoplarse comunicativamente a la plataforma 102, al menos parcialmente, a través de una red de área amplia 116 que puede ser, por ejemplo, la internet pública. Durante el funcionamiento, puede establecerse una sesión de red para facilitar la comunicación entre el cliente que se ejecuta en el ordenador 104 y la plataforma informática 102. El establecimiento de la sesión puede incluir uno o más protocolos de red tales como PPP, TCP/IP, UDP/IP, etc. Una vez establecida, una ruta de comunicación entre el ordenador 104 y la plataforma informática 102 puede permitir una transferencia continua de información. En algunas realizaciones, dicha ruta será capaz de comunicar la información identificada por el proceso de filtrado descrito anteriormente mientras que, de manera simultánea, sirve el juego en línea al ordenador 104. En otras realizaciones, la información filtrada puede comunicarse por fuera de un enlace de juego activo. Si la información de actualización se comunica dentro y/o fuera de un período de tiempo en el que un usuario está jugando, un diseñador de juegos puede querer utilizar un grafo de dependencia, tal como un grafo acíclico dirigido (DAG), para ayudar a identificar los archivos a descargar.

25 Como se ha indicado anteriormente en la Breve descripción de los dibujos, la figura 2 muestra una representación simplificada de un grafo acíclico dirigido 200 que representa diversos elementos de programación y su interrelación de una manera que incorpora al menos una parte de las presentes enseñanzas. Como se muestra, el DAG 200 contiene una parte de una representación de objeto de un modelo de mundo virtual.

30 El DAG 200 representa un grafo dirigido sin ciclos. El DAG 200 puede ser una parte de árbol y una parte de grafo y puede incluir varios elementos de programa en sus nodos, que se indican en general en 202. El DAG 200 puede ayudar a un desarrollador de juegos de varias maneras. Problemas como la programación de descargas y el análisis de ruta crítica, por ejemplo, pueden simplificarse usando un DAG. Esto puede llegar a ser cada vez más importante en el contexto de la determinación de qué información es necesario descargar a un usuario con el fin de permitir que el usuario se beneficie de una parte "nueva" de un mundo en línea. El DAG puede ayudar a seleccionar la información a descargar en virtud de, por ejemplo, la capacidad del DAG para ordenarse topológicamente usando una primera búsqueda en profundidad.

35 Como se muestra, el DAG 200 se usa para representar un conjunto de archivos de programa cuya necesidad está en función de uno o más de otros archivos de programa. Los programas se representan, en general, en los nodos 202 en el grafo, y los enlaces representados, en general, en 204 identifican dependencias entre los programas. En la práctica, un mecanismo tal como el motor de actualización 112 de la figura 1 puede considerar el DAG 200 al determinar qué información se necesita descargar a un cliente. A continuación, puede presentarse la información identificada para su descarga. Si bien puede haber varias posibles variaciones, las estructuras de datos ejemplares para el DAG 200 podrían incluir matrices de adyacencia y listas de adyacencia.

40 De acuerdo con las enseñanzas desveladas en el presente documento, el nodo 1 del DAG 200 puede representar un nodo raíz para un mundo virtual, y los nodos 2, 3, y 4, pueden representar diferentes partes de ese mundo. Por ejemplo, los nodos 2, 3, y 4 pueden incluir varios archivos diferentes, tales como archivos de textura, archivos de animación, archivos de audio, etc., necesarios para representar las respectivas partes del mundo. Al bajar un nivel en el DAG 200, el nodo 5 puede representar una escena específica que forma la parte del mundo virtual representada por el nodo 2. Y el nodo 6 puede representar una escena específica que forma la parte del mundo virtual representada por el nodo 3.

Como se muestra, la parte del mundo representada por el nodo 4 puede incluir dos escenas representadas por los nodos 7 y 8. En la práctica, los nodos 7 y 8 pueden contener archivos que presentan una versión iluminada de una escena y una versión a oscuras de la misma escena, respectivamente. En una realización, un desarrollador de juegos puede querer actualizar el juego añadiendo algunos elementos nuevos al mundo. Los elementos nuevos pueden representarse, por ejemplo, por los nodos 9, 10, y 11. El nodo 9 puede ser un archivo de textura para un bosque, el nodo 10 puede ser un archivo de animación para los árboles y las hojas en el bosque, y el nodo 11 puede ser un archivo de audio que suena como el viento que sopla a través del bosque.

A medida que el personaje del jugador se mueve a través del mundo, el personaje puede acercarse a la versión a oscuras (nodo 8) de la escena mencionada anteriormente. Un servidor remoto que soporta el juego y lo sirve al jugador puede reconocer que el personaje se acerca la escena representada en el nodo 8 y optar por descargar el archivo del nodo 11, de manera que el personaje del jugador escuchará el sonido del viento que sopla a través del bosque a oscuras. Debido a que el personaje del jugador "se acercaba" a la parte actualizada del mundo virtual a través del nodo 8, un mundo a oscuras, el servidor "sabía" que el jugador solo necesitaba los archivos del nodo 11 en contraposición a todos los nuevos archivos contenidos en los nodos 9, 10, y 11.

Como tal, el servidor proporciona poco a poco de manera eficaz actualizaciones al jugador a medida que el jugador necesita las actualizaciones, permitiendo una actualización en línea del tipo justo a tiempo. Si el personaje del jugador se acerca más tarde a la escena representada por el nodo 6, el servidor puede reconocer que el programa cliente usado por el jugador necesita ahora todos los nuevos archivos contenidos en los nodos 9, 10, y 11. En algunos casos, el servidor puede generar un manifiesto de archivos para descargar, y el manifiesto puede incluir los archivos de los nodos 9, 10 y 11. A continuación, el manifiesto puede reducirse o filtrarse para determinar si el cliente realmente necesita todos los archivos. Como se ha indicado anteriormente, si el jugador ya ha recibido los archivos del nodo 11, estos archivos pueden eliminarse del manifiesto, y el tamaño de la descarga se hará más pequeño. Esta técnica de descarga razonada puede proporcionar un enfoque más eficiente para actualizar y/o descargar inicialmente las partes de un MMOG.

Como se ha indicado anteriormente en la Breve descripción de los dibujos, la figura 3 presenta un diagrama de flujo para una técnica de distribución selectiva 300 que incorpora al menos alguna de las presentes enseñanzas. En la etapa 312, un desarrollador de juegos puede comenzar el proceso de diseño de un nuevo mundo virtual. El nuevo mundo puede ser el escenario para un nuevo MMOG que el desarrollador espera comercializar para miles de jugadores en línea.

En la práctica, el MMOG y/o el mundo que actúa como escenario para el MMOG pueden "convertirse" en la etapa 314 en una representación DAG, en la que cada nodo puede representar un bloque básico de código. Cada bloque básico o nodo puede incluir una lista de códigos que termina en algún punto final después de comenzar en algún otro punto. Cada borde o enlace puede representar una relación o próxima acción adoptada si así se indica en el punto final de un nodo determinado.

En la etapa 316, puede seleccionarse una parte inicial del mundo para su distribución, y en la etapa 318 puede distribuirse la parte inicial. En la práctica, la parte inicial puede distribuirse a través de una descarga en línea, como un disco puesto a la venta en una tienda, como un disco distribuido gratuitamente, y/o algún otro mecanismo de distribución apropiado. Una vez distribuida, un usuario con la intención de jugar el juego puede recibir la parte inicial y cargarla en su ordenador. La parte inicial puede establecer una aplicación cliente en el ordenador del usuario y facilitar la capacidad del usuario para comenzar a jugar el nuevo juego.

En la etapa 320, puede recaudarse un pago del usuario por la parte inicial y/o por el servicio de servir el juego al usuario. En la etapa 322, puede activarse la cuenta del usuario, y el usuario puede establecer un personaje para jugar el nuevo juego. En la etapa 324, una plataforma informática remota puede servir el juego al usuario. Aunque el juego puede servirse al usuario desde un dispositivo remoto, la acción de jugar el juego también puede hacer uso de los recursos informáticos locales para el usuario. Por ejemplo, un cliente que se ejecuta en el ordenador del usuario puede soportar una parte significativa de la carga de cálculo asociada al juego del usuario del MMOG.

En la etapa 326, puede recibirse una comunicación que indica que el cliente del usuario necesita una actualización y/o archivos adicionales para experimentar de manera más plena el juego. En la etapa 328, pueden identificarse los archivos que deben descargarse al cliente. El proceso de identificación de estos archivos puede implicar, por ejemplo, la consideración de un DAG que representa alguna parte del MMOG. Una vez identificados, los archivos pueden descargarse al cliente en un momento posterior, casi en tiempo real, sobre una base justo-a-tiempo, y/o de alguna otra manera. En la etapa 330, los archivos pueden haberse descargado de tal manera que el personaje del usuario sea capaz de experimentar los elementos que estaban representados en los archivos.

En la etapa 332, puede determinarse que el cliente del usuario necesita actualizarse. En la etapa 334, puede crearse un manifiesto de los nuevos archivos necesarios para el cliente. Una vez más, puede considerarse un DAG para ayudar en la creación del manifiesto. En función del detalle de implementación, el ordenador del usuario puede encargarse de considerar el DAG para ayudar a desarrollar el manifiesto y/o una plataforma informática alejada del usuario puede encargarse de este proceso. En la etapa 336, el manifiesto puede filtrarse con respecto a una lista de

los archivos que ya están en el ordenador del usuario. Si un archivo ya reside en el ordenador del usuario, el archivo puede reducirse con respecto al manifiesto para evitar su descarga innecesaria. Al igual que con el manifiesto original, el ordenador del usuario puede encargarse del proceso de reducción y/o una plataforma informática alejada del usuario puede encargarse de este proceso.

5 En la etapa 338, pueden descargarse los archivos necesarios para el cliente del usuario, y en la etapa 340 el personaje del usuario puede ser capaz de experimentar las partes del juego actualizadas. En la etapa 342, una plataforma informática remota puede servir el juego a miles de usuarios mientras que ayuda simultáneamente en el proceso de proporcionar archivos de actualización y/u otros archivos adicionales. El proceso puede hacerse avanzar hasta detenerse en la etapa 344.

10 Aunque la técnica 300 se ha descrito en una serie de etapas secuenciales, la secuencia de las etapas y la parte que ejecuta las etapas pueden cambiarse libremente sin alejarse del alcance de las enseñanzas. Pueden añadirse, eliminarse y/o modificarse etapas de varias maneras. Del mismo modo, las etapas pueden ordenarse y/o enlazarse de nuevo.

15 El objeto desvelado anteriormente debe considerarse ilustrativo, y no restrictivo, y se pretende que las reivindicaciones adjuntas cubran todas estas modificaciones, mejoras, y otras realizaciones que caigan dentro del verdadero alcance de la presente invención. Por lo tanto, en la máxima medida permitida por la ley, el alcance de la presente invención debe determinarse por la interpretación más amplia admisible de las siguientes reivindicaciones y sus equivalentes, y no podrá restringirse o limitarse por la descripción detallada anterior.

20

REIVINDICACIONES

1. Un método para la distribución selectiva de información a un dispositivo de juego remoto, que comprende además:
- 5 recibir una comunicación desde una aplicación cliente que se ejecuta en un dispositivo remoto (104, 106, 108), indicando la comunicación una condición de un personaje del juego; desarrollar un manifiesto filtrado que comprende una recopilación de información necesaria para la aplicación cliente y asociada a la condición, desarrollándose el manifiesto filtrado mediante:
- 10 la identificación de un archivo como ya residente en el dispositivo remoto; y la exclusión del archivo del manifiesto filtrado;
- 15 identificar una parte de la recopilación para comunicarla a la aplicación cliente basándose en el manifiesto filtrado; e iniciar la comunicación de la parte.
2. El método de la reivindicación 1, en el que la recopilación de información comprende un archivo de geometría, un archivo de textura, un archivo de sonido y un archivo de animación.
- 20 3. El método de la reivindicación 1, en el que la condición se selecciona a partir de un grupo que consiste en una localización de personaje actual, una localización de personaje prevista y una interacción de personajes.
4. El método de la reivindicación 1, que comprende además:
- 25 recibir del dispositivo remoto una solicitud de un grafo de dependencia que mapea una pluralidad de elementos de juego con archivos asociados; comunicar al menos una parte del grafo de dependencia en respuesta a la solicitud; y recibir del dispositivo remoto información que representa el manifiesto filtrado.
- 30 5. El método de la reivindicación 1, que comprende además mantener un grafo de dependencia que mapea una pluralidad de elementos de juego con los archivos de geometría, archivos de textura, archivos de sonido y archivos de animación asociados.
- 35 6. El método de la reivindicación 5, en el que el grafo de dependencia es un grafo acíclico dirigido DAG, que comprende además proporcionar al menos una parte del DAG a la aplicación cliente.
7. Un sistema para distribuir selectivamente información a un cliente remoto que comprende:
- 40 una plataforma informática (102) configurada para servir un juego en línea a una pluralidad de clientes remotos (104, 106, 108); una interfaz (110) configurada para recibir una señal de actualización para al menos uno de la pluralidad de clientes remotos, indicando la señal de actualización una condición de un personaje del juego; un motor de actualización (112), que responde a la señal de actualización, y que se puede hacer funcionar para
- 45 acceder a al menos una lista parcial de la información disponible, seleccionar, para su salida, una parte de la información disponible asociada a la condición identificando un archivo ya residente en un dispositivo remoto y excluyendo el archivo de la parte, e iniciar una salida de la parte; y una memoria (114) que mantiene la lista parcial.
- 50 8. El sistema de la reivindicación 7, en el que el motor de actualización (112) selecciona la parte en respuesta a un proceso de filtrado que comprende:
- 55 comunicar una parte de un grafo acíclico dirigido DAG a un primer cliente remoto, la parte asociada a una condición de un personaje con el que juega el primer cliente remoto; y recibir una comunicación del primer cliente remoto que indica al menos un archivo del DAG que necesita el primer cliente remoto.
- 60 9. El sistema de la reivindicación 7, que comprende además un enlace de comunicación acoplado comunicativamente a la plataforma informática y que se puede hacer funcionar para interconectar, al menos parcialmente, la plataforma informática y el al menos uno de la pluralidad de clientes remotos.

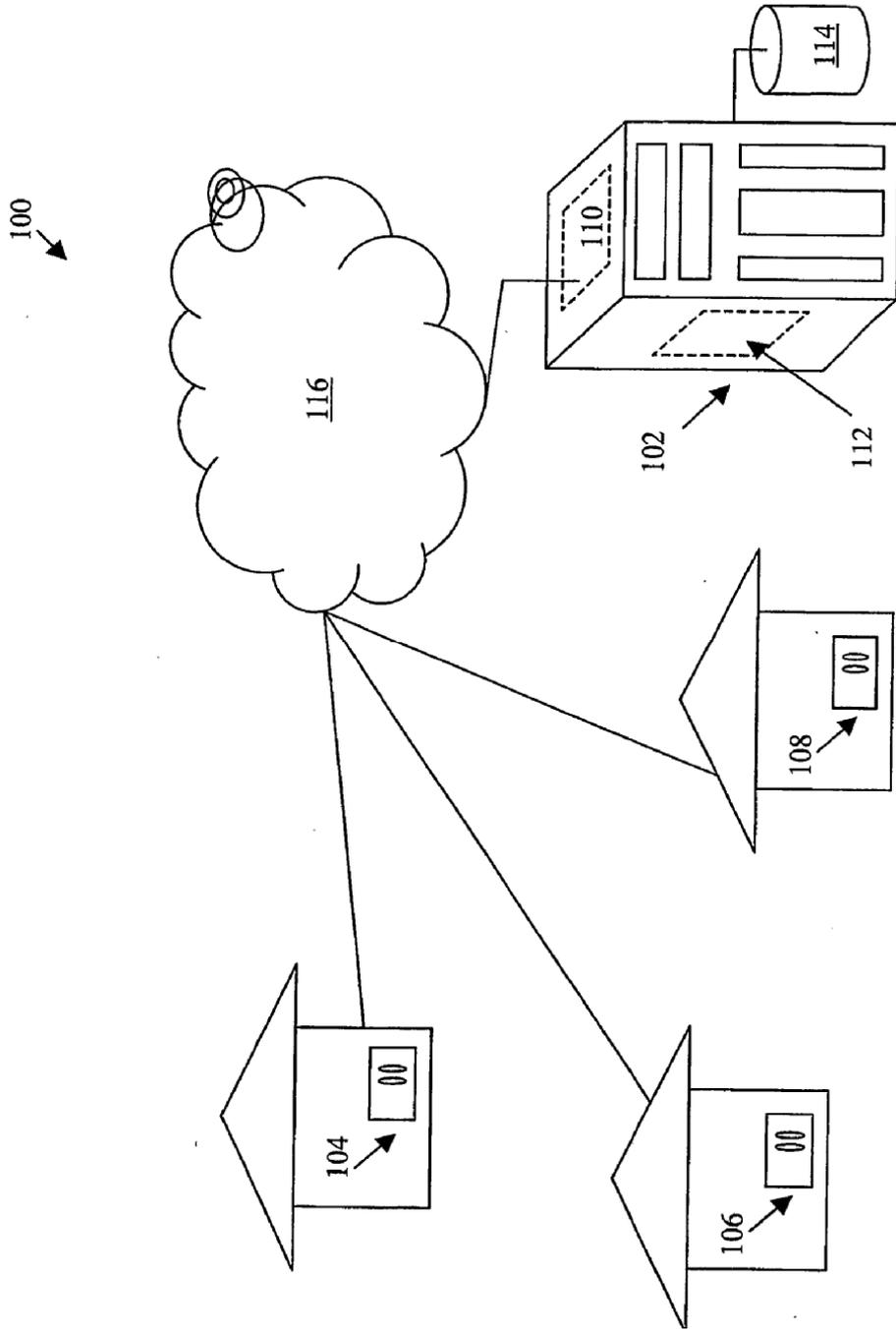


FIG. 1

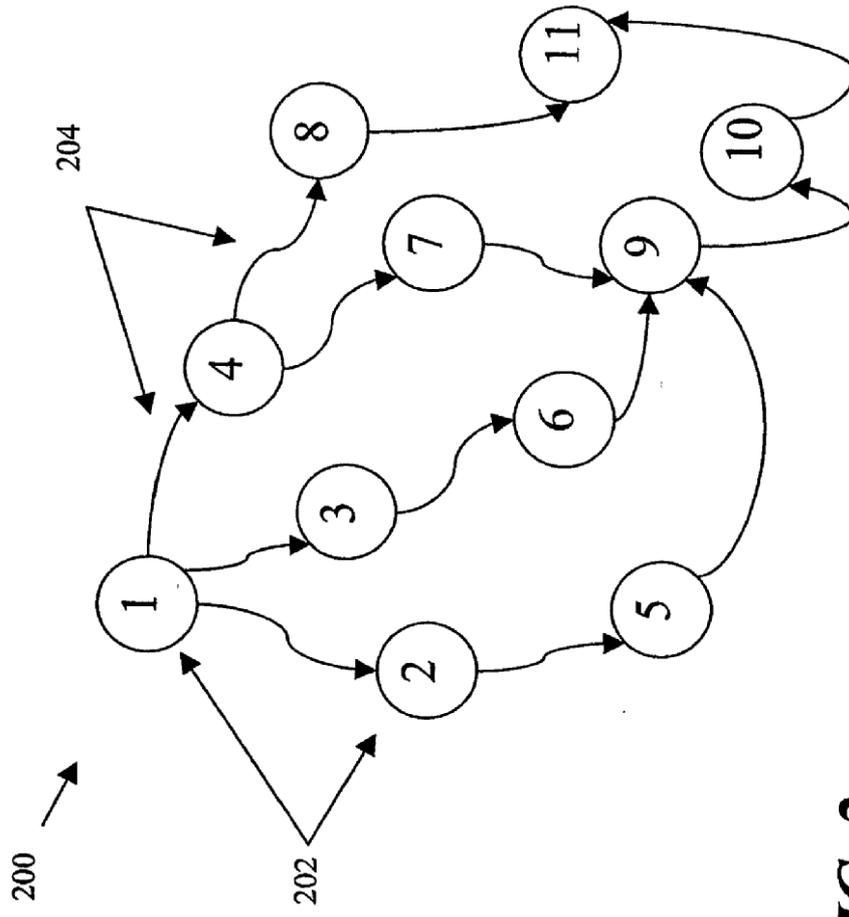


FIG. 2

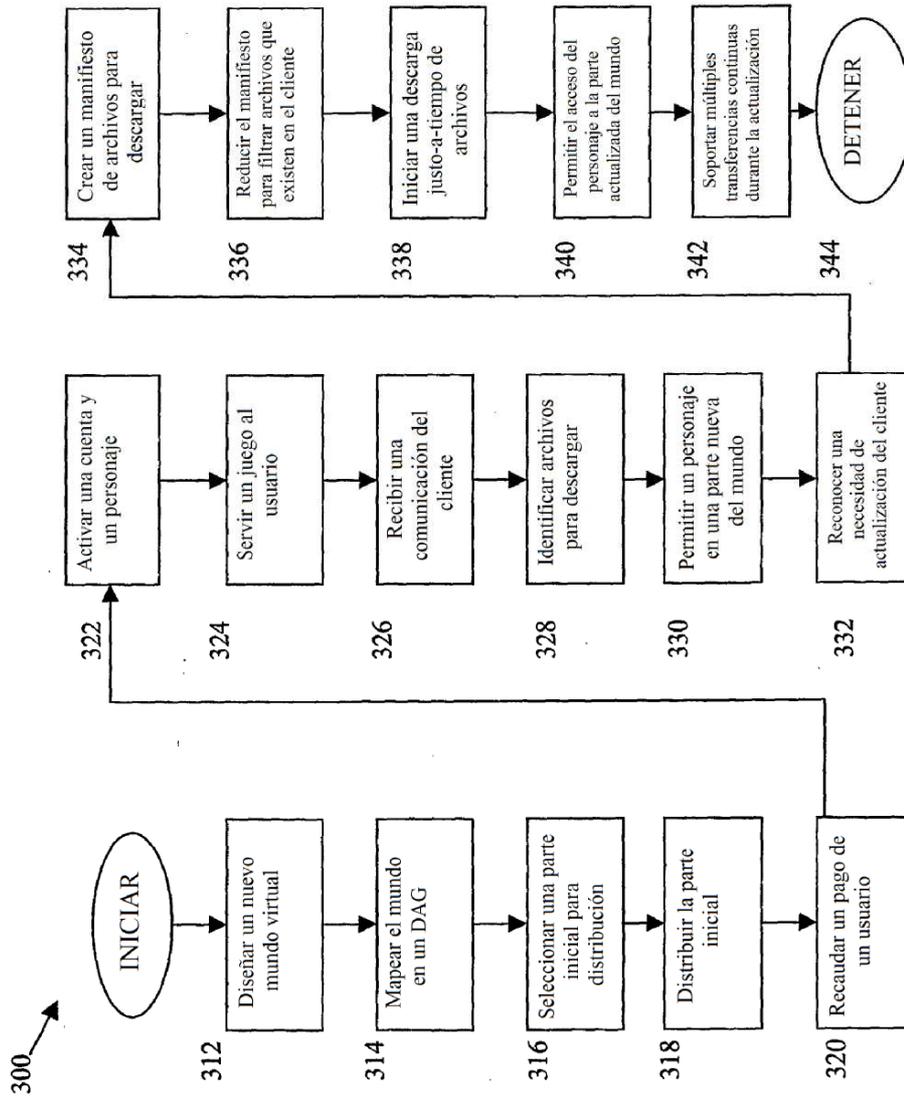


FIG. 3