

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 772 224**

51 Int. Cl.:

G11B 27/034 (2006.01)

G11B 27/34 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **06.04.2012** **PCT/US2012/032535**

87 Fecha y número de publicación internacional: **11.10.2012** **WO12139008**

96 Fecha de presentación y número de la solicitud europea: **06.04.2012** **E 12767666 (6)**

97 Fecha y número de publicación de la concesión europea: **11.12.2019** **EP 2695051**

54 Título: **Sistemas, medios de almacenamiento legibles por ordenador y métodos implementados por ordenador para compartir proyectos**

30 Prioridad:

08.04.2011 US 201161473544 P

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

07.07.2020

73 Titular/es:

LIEBMAN, ANDREW (100.0%)
140 Pleasant Street
Brookline, MA 02446

72 Inventor/es:

LIEBMAN, ANDREW

74 Agente/Representante:

ISERN JARA, Jorge

ES 2 772 224 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

Sistemas, medios de almacenamiento legibles por ordenador y métodos implementados por ordenador para compartir proyectos

Campo de la invención

La presente invención se refiere a sistemas de compartición de proyectos para habilitar la colaboración entre múltiples usuarios en proyectos compartidos. Más en concreto, algunas realizaciones de la presente invención se refieren al bloqueo de archivos contenedores, archivos de proyecto y otras colecciones de metadatos para proyectos de edición de vídeo no lineal.

Antecedentes de la invención

En los sistemas informáticos de compartición de proyectos existentes para compartir proyectos de edición de vídeo no lineal, archivos o contenedores que contienen metadatos (por ejemplo, referencias a clips, subclips, etc.) se comparten entre diversos usuarios y están ubicados en un servidor central al que pueden acceder los usuarios a través de un cliente de usuario, tal como en un ordenador personal. Algunos sistemas de este tipo no ofrecen tipo alguno de protección frente a que un usuario sobrescriba los cambios de otro. Por ejemplo, en algunos sistemas de este tipo, si dos usuarios abren el mismo archivo contenedor o de proyecto de forma simultánea, se puede proporcionar a ambos de los usuarios un acceso de escritura, de tal modo que la versión confirmada en última instancia en disco es la que se guarda en último lugar.

Otros sistemas de compartición de archivos crean una carpeta de usuario para cada usuario, y entonces establecen de forma automática tanto la propiedad como los permisos para archivos de acuerdo con la carpeta de usuario particular en la que reside un archivo. Proporcionar este tipo de control sobre el acceso a archivos habilita que cada usuario lea y escriba fácilmente archivos que pertenecen claramente a ese usuario, mientras se impide de forma simultánea que el usuario reciba acceso de escritura a la carpeta de usuario de cualquier otro usuario y los contenidos correspondientes.

Un sistema de este tipo, sin embargo, adolece de varios inconvenientes. La claridad global de la organización de archivos contenedores o de proyecto de un sistema de este tipo se ve obstaculizado por el hecho de que solo se logran unos permisos apropiados cuando un proyecto que consiste en muchos archivos contenedores y de proyecto está dispersado entre usuarios. Lograr unos permisos apropiados para archivos implica mover, de forma continua, archivos de carpeta a carpeta, lo que puede dificultar el seguimiento o la localización de un archivo particular. Esto es particularmente cierto a medida que crece el número de usuarios en un proyecto. Por ejemplo, se puede requerir a un usuario que está buscando un archivo en un proyecto particular en el que están trabajando 20 usuarios que busque en 19 carpetas de usuario diferentes antes de identificar el archivo deseado.

Además, dada la dispersión de archivos de un único proyecto por muchas carpetas diferentes, es difícil proporcionar una vista de pantalla única conveniente e integral de todos los archivos asociados con un único proyecto. Esto puede dificultar la gestión de los proyectos y el seguimiento de su progreso. También obstaculiza la capacidad de ver rápidamente la partición de trabajo entre diversos usuarios en un proyecto y comparar sus cargas de trabajo relativas, por ejemplo, para fines de gestión.

Existe una gran dificultad en asignar de forma automática tanto propiedades como permisos de una forma que habilita una transferencia conveniente de unos derechos de acceso de escritura exclusivo entre múltiples usuarios que están trabajando en un único conjunto de archivos. Otros intentos implican crear copias de archivos, lo que se puede volver costoso desde el punto de vista computacional. Otros intentos más implican crear estructuras de vínculo complejas, pero esto puede dar como resultado un riesgo de vínculos rotos cuando se mueven o se actualizan archivos. Ningún sistema existente habilita fácilmente la automatización de las diversas características deseables descritas en el presente documento.

La solicitud de patente publicada WO 2009/155578 A2, a nombre de Liebman Andrew [EE. UU.], de 23 de diciembre de 2009, muestra un editor de vídeo no lineal, en donde los archivos de medios no se modifican sino que se conservan, y solo se modifican metadatos que se refieren a vínculos a archivos de medios; dichos metadatos se almacenan realmente en archivos reales a nivel de sistema de archivos - "proyectos". Múltiples usuarios pueden colaborar en el proceso de edición, operando por lo tanto en proyectos compartidos. Se usa un mecanismo de control de acceso de la pluralidad de editores a partes de los metadatos, basándose en la propiedad y permisos a partes del proyecto compartido. Se renuncia a la propiedad, o se asume esta, al mover partes del proyecto a carpetas predeterminadas, con unos permisos pre-establecidos de lectura o de escritura. Por ejemplo, las partes de proyecto colocadas / movidas en una carpeta de usuario solo pueden ser escritas por el usuario respectivo, las partes de proyecto colocadas / movidas en una carpeta no protegida también pueden ser tomadas y escritas por otros usuarios, y las partes de proyecto colocadas / movidas en una carpeta de grupo pueden ser leídas por los usuarios del mismo grupo. La transferencia de archivos a través de las carpetas no protegidas da como resultado la entrega de la propiedad a todos los usuarios cuando se coloca allí un archivo, y que se asuma la propiedad cuando un archivo es arrastrado desde allí

al exterior.

Sumario

5 Existe la necesidad de un sistema de compartición de proyectos que proporcione un control automatizado sobre los derechos de acceso a archivos mientras se mantiene una estructura de directorios centralizada en la que los archivos contenedores y de proyecto, así como de que información en relación con los usuarios asociados particulares a los que pertenecen los archivos, se pueda localizar fácilmente y verse de forma integral. La presente invención se dirige a soluciones adicionales para abordar esta y otras necesidades, además de tener otras características deseables que
10 serán apreciadas por un experto en la materia tras la lectura de la presente memoria descriptiva.

De acuerdo con una realización ilustrativa de la presente invención, un sistema de compartición de proyectos implementado por ordenador incluye metadatos almacenados en un dispositivo de almacenamiento legible por ordenador no transitorio. Los metadatos pueden definir una parte de un proyecto de edición de vídeo no lineal. Los
15 permisos sobre los metadatos se pueden almacenar en un dispositivo de almacenamiento legible por ordenador no transitorio, y los permisos pueden habilitar que un grupo de uno o más usuarios editores lea los metadatos y habilitar que solo un propietario de los metadatos modifique o borre los metadatos. Un dispositivo de servidor se puede configurar para controlar el acceso de escritura a los metadatos al transferir, usando al menos un procesador, la propiedad de los metadatos a o desde un usuario de servidor que no es un editor. Si un primer usuario editor de los
20 uno o más usuarios editores tiene la propiedad de los metadatos, el primer usuario editor puede estar habilitado para renunciar a la propiedad de los metadatos al dar instrucciones al dispositivo de servidor para transferir la propiedad de los metadatos al usuario de servidor y se puede evitar que cada uno de los uno o más usuarios editores sin la propiedad de los metadatos dé instrucciones al dispositivo de servidor para transferir la propiedad de los metadatos. Si el usuario de servidor tiene la propiedad de los metadatos, uno cualquiera de los uno o más usuarios editores se
25 puede habilitar para asumir la propiedad de los metadatos al dar instrucciones al dispositivo de servidor para transferir la propiedad de los metadatos del usuario de servidor a ese uno de los uno o más usuarios editores.

De acuerdo con algunos aspectos ilustrativos de la presente invención, los metadatos se pueden almacenar en un archivo almacenado en un directorio, y los permisos sobre los metadatos pueden incluir permisos a nivel de directorio
30 y permisos a nivel de archivo. Los permisos a nivel de directorio pueden incluir un bit pegajoso. Alternativa o adicionalmente, los metadatos se pueden almacenar en una o más entradas de base de datos de una base de datos. Las instrucciones se pueden almacenar en al menos un dispositivo legible por ordenador no transitorio, y la ejecución de las instrucciones por al menos un procesador puede dar lugar a que una interfaz gráfica de usuario se presente en al menos un dispositivo de presentación. La interfaz gráfica de usuario puede habilitar que unas instrucciones para
35 transferir la propiedad de los metadatos se envíen al dispositivo de servidor. La interfaz gráfica de usuario puede incluir un indicador de estado actual para los metadatos, y el indicador de estado actual puede indicar, a uno de los uno o más usuarios editores, que los metadatos están actualmente disponibles para ser extraídos, son actualmente extraídos por ese uno de los uno o más usuarios editores o son actualmente extraídos por uno diferente de los uno o más usuarios editores. El dispositivo de servidor se puede ejecutar en un sistema operativo basado en Linux, uno basado en Unix o uno de tipo Unix.

De acuerdo con una realización ilustrativa de la presente invención, se proporciona un método implementado por ordenador. Se pueden establecer, usando al menos un procesador, unos permisos sobre metadatos almacenados en un dispositivo de almacenamiento legible por ordenador no transitorio. Los permisos pueden habilitar que un grupo de
45 uno o más usuarios editores lea los metadatos y pueden habilitar que solo un propietario de los metadatos modifique o borre los metadatos. El acceso de escritura a los metadatos se puede controlar al transferir, usando al menos un procesador y tras recibir una instrucción a través de al menos un dispositivo de entrada, la propiedad de los metadatos a o desde un usuario de servidor que no es un editor. Si un primer usuario editor de los uno o más usuarios editores tiene la propiedad de los metadatos, el primer usuario editor puede estar habilitado para renunciar a la propiedad de los metadatos al dar instrucciones al dispositivo de servidor para transferir la propiedad de los metadatos al usuario de servidor y se puede evitar que cada uno de los uno o más usuarios editores sin la propiedad de los metadatos dé instrucciones al dispositivo de servidor para transferir la propiedad de los metadatos. Si el usuario de servidor tiene la propiedad de los metadatos, uno cualquiera de los uno o más usuarios editores se puede habilitar para asumir la propiedad de los metadatos al dar instrucciones al dispositivo de servidor para transferir la propiedad de los metadatos
50 del usuario de servidor a ese uno de los uno o más usuarios editores.

De acuerdo con algunos aspectos ilustrativos de la presente invención, el primer usuario puede tener la propiedad de los metadatos en el método implementado por ordenador, y el método puede incluir adicionalmente recibir, a través de al menos un dispositivo de entrada, una instrucción del primer editor usuario para renunciar a la propiedad de los metadatos. Adicionalmente, basándose en la instrucción recibida, la propiedad de los metadatos se puede transferir, usando el al menos un procesador, del primer usuario al usuario de servidor. Además, el usuario de servidor puede tener la propiedad de los metadatos, y el método puede incluir adicionalmente recibir, a través de al menos un dispositivo de entrada, una instrucción de uno de los uno o más editores usuarios para asumir la propiedad de los metadatos. Adicionalmente, basándose en la instrucción recibida, la propiedad de los metadatos se puede transferir, usando el al menos un procesador, del usuario de servidor al uno de los uno o más editores usuarios. Los metadatos se pueden almacenar en un archivo almacenado en un directorio, y en donde los permisos sobre los metadatos
65

comprenden permisos a nivel de directorio y permisos a nivel de archivo. Los permisos a nivel de directorio pueden incluir un bit pegajoso. Los metadatos se pueden almacenar en una o más entradas de base de datos de una base de datos. Las instrucciones se pueden almacenar en al menos un dispositivo legible por ordenador no transitorio, y la ejecución de las instrucciones por al menos un procesador puede dar lugar a que una interfaz gráfica de usuario se presente en al menos un dispositivo de presentación. La interfaz gráfica de usuario puede incluir un indicador de estado actual para los metadatos, y el indicador de estado actual puede indicar, a uno de los uno o más usuarios editores, que los metadatos están actualmente disponibles para ser extraídos, son actualmente extraídos por ese uno de los uno o más usuarios editores o son actualmente extraídos por uno diferente de los uno o más usuarios editores.

De acuerdo con una realización ilustrativa de la presente invención, se proporciona una visualización electrónica interactiva de información para al menos un primer usuario editor de un programa de edición de vídeo no lineal. El primer editor se asigna a un proyecto de edición de vídeo no lineal (NLE) que incluye una pluralidad de metadatos. La visualización interactiva incluye una pluralidad de indicadores de metadatos que representan la pluralidad de metadatos que definen el proyecto de edición de vídeo no lineal. Cada metadato de la pluralidad de metadatos puede ser propiedad de un usuario editor que representa un editor asignado al proyecto de NLE o de un usuario de servidor que no representa un editor asignado al proyecto de NLE. Para cada metadato de la pluralidad de metadatos, se puede incluir un indicador de estado asociado en la visualización interactiva. El indicador de estado asociado se puede seleccionar de entre el grupo que consiste en: un primer indicador de estado que indica eso de los metadatos por el usuario de servidor; un segundo indicador de estado que indica la propiedad de los metadatos por el primer usuario editor; y un tercer indicador de estado que indica la propiedad de los metadatos por un usuario editor que no es el primer usuario editor.

De acuerdo con algunos aspectos ilustrativos de la presente invención, al primer usuario editor solo se le puede permitir un acceso de escritura a los metadatos de la pluralidad de metadatos que son propiedad del primer editor. Se puede seleccionar cada uno del primer indicador de estado y el segundo indicador de estado. Para cada primer indicador de estado incluido en la visualización, la selección del primer indicador de estado puede dar lugar a que un dispositivo de servidor transfiera, del primer usuario al usuario de servidor, usando al menos un procesador, la propiedad de los metadatos asociados con el primer indicador seleccionado. Para cada segundo indicador de estado incluido en la visualización, la selección del segundo indicador de estado puede dar lugar a que un dispositivo de servidor transfiera, del usuario de servidor al primer usuario, usando al menos un procesador, la propiedad de los metadatos asociados con el segundo indicador seleccionado. Para cada metadato de la pluralidad de metadatos, el primer indicador de estado puede indicar que el primer editor usuario es propietario de los metadatos y ha extraído los metadatos, el segundo indicador de estado puede indicar que el primer usuario editor no tiene la propiedad de los metadatos y está habilitado para extraer los metadatos, y el tercer indicador de estado puede indicar que el primer usuario editor no tiene la propiedad de los metadatos y no está habilitado para extraer los metadatos.

Breve descripción de las figuras

Estas y otras características de la presente invención se entenderán más plenamente por referencia a la siguiente descripción detallada junto con los dibujos adjuntos, en los que:

la figura 1 es un diagrama ilustrativo de un sistema para controlar un acceso de escritura exclusivo al transferir la propiedad entre una pluralidad de usuarios editores, de acuerdo con una realización ilustrativa de la presente invención;

la figura 2 es una captura de pantalla que representa la visualización de un usuario editor que incluye un cliente de usuario ilustrativo que se ejecuta dentro del entorno de un programa de NLE, de acuerdo con algunos aspectos ilustrativos de la presente invención;

la figura 3A es una captura de pantalla del cliente de usuario de la figura 2, antes de una acción por un usuario editor, "jeff", de acuerdo con algunos aspectos ilustrativos de la presente invención;

la figura 3B es una captura de pantalla del cliente de usuario de la figura 3B, posteriormente a una acción por el usuario editor, "jeff", de acuerdo con algunos aspectos ilustrativos de la presente invención;

la figura 4A es una captura de pantalla de un cliente de usuario, antes de una acción por otro usuario editor, "jane", de acuerdo con algunos aspectos ilustrativos de la presente invención;

la figura 4B es una captura de pantalla del cliente de usuario de la figura 4A, posteriormente a una acción por el usuario editor, "jane", de acuerdo con algunos aspectos ilustrativos de la presente invención;

la figura 5A es un diagrama de flujo que representa un método para habilitar que un usuario editor asuma la propiedad de un archivo, de acuerdo con una realización ilustrativa de la presente invención;

la figura 5B es un diagrama de flujo que representa un método para habilitar que un usuario editor renuncie a la propiedad de un archivo, de acuerdo con una realización ilustrativa de la presente invención;

la figura 6 es un diagrama ilustrativo que representa un dispositivo informático ilustrativo para implementar algunas realizaciones de la presente invención;

5 la figura 7 es una captura de pantalla de unas propiedades y un permiso ilustrativos para un directorio "demo1" y los archivos contenidos en el mismo, como se ve en un servidor ilustrativo antes de que un usuario editor, jeff, solicite recibir un acceso de escritura y extraer el archivo "Acto_2.fcp", de acuerdo con algunos aspectos ilustrativos de la presente invención;

10 la figura 8 es una captura de pantalla de las propiedades y el permiso ilustrativos de la figura 7 para el directorio "demo1" y los archivos contenidos en el mismo, como se ve en el servidor ilustrativo posteriormente a que el usuario editor, jeff, solicite recibir un acceso de escritura y la propiedad del archivo "Acto_2.fcp", de acuerdo con algunos aspectos ilustrativos de la presente invención;

15 la figura 9 es una captura de pantalla de unas propiedades y un permiso ilustrativos para un directorio "Urgencias" y los archivos contenidos en el mismo, como se ve en un servidor ilustrativo en donde todos los archivos son propiedad del usuario de servidor, denominado en el presente documento "editar compartir", de acuerdo con algunos aspectos ilustrativos de la presente invención;

20 la figura 10 es una captura de pantalla de las propiedades y el permiso ilustrativos de la figura 9 para el directorio "Urgencias" y los archivos contenidos en el mismo, como se ve en el servidor ilustrativo posteriormente a que un usuario editor, "jane", solicite recibir un acceso de escritura al archivo "Urgencias_1.fcp" y reciba la propiedad del archivo, de acuerdo con algunos aspectos ilustrativos de la presente invención; y

25 la figura 11 es una captura de pantalla de una ventana de navegador que incluye diversos indicadores de estado de metadatos implementados para metadatos almacenados en una base de datos, de acuerdo con una realización ilustrativa de la presente invención.

Descripción detallada

30 Una realización ilustrativa de la presente invención se refiere a un método implementado por ordenador, sistema implementado por ordenador y medio de almacenamiento legible por ordenador para habilitar la compartición y el bloqueo de metadatos que forman parte o la totalidad de un proyecto en un programa de edición de vídeo no lineal (por ejemplo, Avid, Final Cut Pro, etc.). En la realización ilustrativa, el sistema está configurado para controlar el acceso de escritura de metadatos al transferir la propiedad de los metadatos entre diversos usuarios editores (es decir, editores humanos que están trabajando de forma colaborativa en un proyecto de edición de vídeo no lineal compartido). Cabe destacar que el sistema transfiere la propiedad entre diversos usuarios editores a través de un "intermediario" denominado en el presente documento "usuario de servidor". El usuario de servidor puede ser controlado por un dispositivo informático tal como un servidor. El usuario de servidor no representa, en el sistema, ninguno de los editores humanos asignados al proyecto de edición de vídeo no lineal. Las transferencias de propiedad de los metadatos pueden ser iniciadas por usuarios editores. Más específicamente, en la realización ilustrativa, cada usuario editor está habilitado para (a) asumir la propiedad de los metadatos propiedad del usuario de servidor; y (b) ceder la propiedad de los metadatos que sean de su propiedad al usuario de servidor. Además, en la realización ilustrativa, solo al propietario de los metadatos se le permite modificar esos metadatos. En consecuencia, el usuario de servidor proporciona un punto de transferencia de propiedad entre los usuarios editores, a través del cual los editores pueden transferir de forma segura el acceso de escritura de colecciones de metadatos sin crear copias innecesarias de metadatos, y similares.

50 Por lo tanto, al usar una transferencia de propiedad para habilitar que los usuarios editores asignados a un proyecto de NLE asuman y cedan un acceso de escritura exclusivo a metadatos, los sistemas de la presente invención proporcionan un sistema de "inserción" y de "extracción" que no requiere la creación de copias o versiones de metadatos, y que no requiere mover la ubicación de los metadatos (o los archivos, objetos de base de datos, etc. en los que están contenidos los mismos). Además, a diferencia de los sistemas de inserción - extracción convencionales, algunas realizaciones de la presente invención proporcionan un "terreno neutral" extremadamente conveniente y no costoso desde el punto de vista computacional para metadatos que no son extraídos por usuario editor alguno: el usuario de servidor. En efecto, en los sistemas de acuerdo con algunas realizaciones de la presente invención, se "insertan" metadatos cuando un usuario editor cede la propiedad de los metadatos al usuario de servidor. Se "extraen" metadatos propiedad del usuario de servidor cuando un usuario editor asume la propiedad de los metadatos al quitar la propiedad de los metadatos al usuario de servidor. Solo el usuario editor que ha extraído metadatos está habilitado para modificar y guardar los cambios en los metadatos. Esto permite que los usuarios editores editen de forma segura unas porciones del proyecto de NLE en el que están trabajando los mismos sin riesgo de sobrescritura por otros usuarios editores. Por lo tanto, algunas realizaciones ilustrativas pueden proporcionar un sistema que habilita una compartición de archivos conveniente sin la necesidad de mover archivos, crear copias de archivos o hacer que los permisos se asignen de forma manual.

65 En consecuencia, los permisos para los metadatos se pueden asignar de una forma tal como para dotar a una

pluralidad de usuarios editores diferentes de un acceso de escritura exclusivo a metadatos que son de su propiedad. Además, este tipo de estructura de permisos se puede proporcionar a los usuarios editores sin sacrificar la capacidad de ninguno de los usuarios editores de crear metadatos nuevos dentro de un directorio cuyos contenidos (por ejemplo, archivos que contienen los metadatos, entradas de base de datos que contienen los metadatos, etc.) son propiedad de la pluralidad de usuarios editores diferentes. Por ejemplo, en los sistemas de acuerdo con algunas realizaciones de la presente invención que utilizan un sistema operativo de tipo Unix y para los cuales se almacenan metadatos en un archivo o archivos en una estructura de directorios, se puede dotar a los usuarios editores que tienen la propiedad de archivos que contienen metadatos que definen al menos un proyecto de NLE de un acceso de escritura a nivel de archivo a los archivos que son de su propiedad. Además, para cualquier archivo que no sea propiedad de uno o más usuarios editores pero para el cual se desea un permiso de lectura por los uno o más usuarios editores del archivo, se puede proporcionar a esos uno o más usuarios editores (por ejemplo, en virtud del hecho de estar asignados al proyecto de NLE al que pertenecen los metadatos) un acceso de lectura a los archivos a nivel de archivo y un acceso de escritura a los archivos a nivel de directorio (por ejemplo, al definir una "clase de grupos" con acceso de lectura - escritura e incluir tales uno o más usuarios editores en la clase de grupos, como sería apreciado por un experto en la materia tras la lectura de la presente memoria descriptiva). Además, en tales realizaciones ilustrativas, se puede aplicar un bit pegajoso a cualquier directorio de este tipo (al que se proporciona un acceso de escritura a los usuarios editores), con el fin de asegurar que solo el propietario de cada archivo en el directorio está habilitado para ejercer permisos de escritura a nivel de directorio sobre el archivo (por ejemplo, renombrar, borrar y mover ese archivo), mientras se sigue permitiendo que otros usuarios editores con acceso de escritura al directorio creen, en ese directorio, unos archivos nuevos que contienen metadatos que definen adicionalmente o contribuyen al proyecto de NLE. De esta forma, para aquellas realizaciones en las que los metadatos que definen un proyecto de NLE se almacenan en archivos, un acceso de escritura exclusivo a metadatos puede ser controlado por el sistema a través del uso de permisos a nivel de archivo y permisos a nivel de directorio específicamente asignados sobre los archivos que contienen los metadatos.

Además, todas las propiedades y los estados actuales de los metadatos se pueden presentar convenientemente en una vista única a cada usuario. Por ejemplo, se puede dotar a cada usuario editor de indicadores que indican si cada agrupación de metadatos (por ejemplo, cada archivo que contiene metadatos, etc.) (a) es extraída por ese usuario; (b) es extraída por otro usuario editor; o (c) está disponible para ser extraída. Esto puede habilitar que los usuarios vean fácil y convenientemente la propiedad y los estados actuales de todos los metadatos que forman un único proyecto de NLE dentro de una única ventana.

A continuación se describirá una realización ilustrativa en la que el sistema de bloqueo de metadatos se implementa al transferir la propiedad (y un acceso de escritura exclusivo) específicamente a archivos que contienen metadatos. Por ejemplo, tales archivos pueden incluir archivos contenedores producidos por software de NLE como Avid, archivos de proyecto "monolíticos" producidos por software de NLE como Final Cut Pro (por ejemplo, archivos .fcp), y similares. Sin embargo, la presente invención no se limita a estas realizaciones ilustrativas descritas en detalle en el presente documento. Se debería apreciar que el sistema descrito en el presente documento para transferir la propiedad de archivos se puede aplicar de forma similar a cualquier otro mecanismo o formato adecuado en el que se almacenen metadatos. Por ejemplo, en una realización alternativa descrita en mayor detalle en el presente documento, los metadatos se almacenan en objetos de base de datos, y por lo tanto, los permisos para los metadatos pueden ser manipulados al nivel de fragmentos individuales de metadatos. Un experto en la materia apreciará otras formas y mecanismos más para almacenar metadatos que permiten la transferencia de propiedad y la transferencia de un acceso de escritura exclusivo entre una pluralidad de usuarios editores a través del intermediario de un "usuario de servidor".

Se debería apreciar que "metadatos" en el presente documento se refiere tanto a una pluralidad de metadatos como a metadatos individuales (por ejemplo, un uso tanto singular como plural). La presente invención no se limita a aplicaciones específicas del esquema de bloqueo de metadatos novedoso o bien a una pluralidad de metadatos o bien a metadatos individuales. En su lugar, la propiedad de grupos de metadatos y de metadatos individuales (por ejemplo, almacenados en grupos o de forma individual en archivos, entradas de base de datos, etc.) se puede utilizar en el presente documento con algunas realizaciones de la presente invención.

Las figuras 1 a 11, en donde partes semejantes se designan de principio a fin por números de referencia semejantes, ilustran realizaciones ilustrativas de un sistema para compartir metadatos de acuerdo con la presente invención. Aunque la presente invención se describirá con referencia a las realizaciones ilustrativas ilustradas en las figuras, se debería entender que muchas formas alternativas pueden materializar la presente invención. Los expertos en la materia apreciarán adicionalmente formas diferentes de alterar los parámetros de las realizaciones divulgadas, de una forma que sigue manteniendo el espíritu y el alcance de la presente invención.

La figura 1 representa un sistema implementado por ordenador 100 ilustrativo para habilitar las funciones de la presente invención descritas en detalle en el presente documento, además de otras funciones que serán apreciadas fácilmente por un experto en la materia tras la lectura de la presente memoria descriptiva. El sistema 100 incluye un módulo de permisos 110, un usuario de servidor 112 y un generador de interfaz gráfica de usuario (GUI) 114. El sistema 100 también incluye al menos un directorio 116 que puede contener uno o más archivos 118 que contienen metadatos que definen uno o más proyectos de edición de vídeo no lineal, denominados en el presente documento "proyectos de NLE". Cada directorio 116 puede tener uno o más permisos 120 asignados al mismo por el módulo de

permisos 110. De forma similar, cada uno de los archivos 118 que contienen los metadatos puede tener uno o más permisos 122 asignados al mismo por el módulo de permisos 110. En ese sentido, los permisos sobre los archivos 118 imponen unos permisos sobre los metadatos que definen el proyecto o proyectos de NLE.

El sistema 100 se acopla a uno o más clientes de usuario 128, operado cada uno por un usuario editor 124 asignado para trabajar en el proyecto o proyectos de NLE contenidos en el directorio 116. Los uno o más clientes de usuario 128 se pueden conectar directamente al sistema 100. Como alternativa, los uno o más clientes de usuario 128 se pueden conectar al sistema 100 usando cualquier conexión en red de comunicación adecuada (no mostrada), incluyendo una conexión de Ethernet, una conexión a través de Internet, y cualquier otra conexión en red conocida.

Aunque el generador de GUI 114 es un componente del sistema 100 en la realización ilustrativa de la figura 1, en otras realizaciones ilustrativas de la presente invención, como alternativa, el generador de GUI 114 se ubica en cada uno de los clientes de usuario 128.

Los uno o más usuarios editores 124 forman un grupo de usuarios 126 que está asociado con el directorio 116. El grupo de usuarios 126 puede tener permisos específicos a nivel de grupo al directorio 116. Por ejemplo, en algunas realizaciones ilustrativas, el grupo de usuarios 126 define una "clase de grupos" de permisos, como sería apreciado por un experto en la materia. De forma opcional, el usuario de servidor 112 puede ser un miembro del grupo de usuarios 126, como sería apreciado por un experto en la materia tras la lectura de la presente memoria descriptiva.

Como sería apreciado por un experto en la materia tras la lectura de la presente invención, se pueden incluir múltiples directorios 116 en el sistema 100, de tal modo que cada directorio 116 está asociado con un grupo de usuarios 126 que contiene uno o más usuarios editores 124 asignados a uno o más proyectos de NLE contenidos en el directorio 116.

A través de sus clientes de usuario 128 respectivos, los usuarios editores 124 pueden acceder al directorio 116 como habilitados por los permisos 120 sobre el directorio 116. De forma similar, los usuarios editores 124 pueden acceder a los archivos 118 contenidos en el directorio 116 como habilitados por los permisos 122 sobre los archivos 118. En una realización ilustrativa, los permisos 120, 122 se definen de tal modo que: (a) de entre de los usuarios editores 124, solo el usuario editor 124 que tiene la propiedad de un archivo 118 está habilitado para modificar, borrar, mover o renombrar el archivo 118; (b) se da al grupo de usuarios 126 un acceso de lectura y un acceso de escritura al directorio 116 y, por lo tanto, está habilitado para crear unos archivos 118 nuevos en el directorio 116; y (c) no se concede acceso alguno a "otros" usuarios editores (no mostrados en la figura 1) que no son ni miembros del grupo de usuarios 126 ni un propietario del archivo 118.

Un experto en la materia apreciará una amplia diversidad de tipos, combinaciones y variaciones diferentes de permisos que se pueden utilizar tras la lectura de la presente memoria descriptiva. La totalidad de tales alternativas y modificaciones están contempladas dentro del alcance de la presente invención. Algunas realizaciones de la presente invención no se limitan a la selección particular de permisos que se describen en el presente documento.

En consecuencia, los permisos a nivel de directorio se pueden establecer para limitar todo acceso, incluso un acceso de solo lectura, a solo los miembros del grupo de usuarios de espacio de proyectos, en contraposición a permitir que la pertenencia de un usuario en un grupo de usuarios "editores" general (por ejemplo, el grupo de usuarios 126) determine el acceso. En algunas realizaciones ilustrativas, los permisos 120 sobre el directorio 116 son diferentes de los permisos 122 en los archivos 118 contenidos en el directorio 116. Por ejemplo, en algunas realizaciones implementadas para sistemas operativos de tipo Unix, se puede utilizar un bit pegajoso para imponer que el propietario de un archivo 118 es el único de los usuarios editores 124 en el grupo de usuarios 126 al que se le permite modificar, renombrar, mover y / o borrar el archivo 118. En contraposición, sin el uso de un bit pegajoso, al usuario o propietario de grupo del directorio 116 se le permite borrar, renombrar o mover unos archivos 118 contenidos en el directorio 116. Se puede establecer que el directorio 116 sea propiedad del servidor, el usuario de servidor 112, u otro propietario "genérico" no editor, por ejemplo, para evitar que los usuarios editores 124 reciban la autoridad para cambiar los permisos 120 o las propiedades sobre el directorio 116.

En consecuencia, como se ha descrito previamente en el presente documento, para implementaciones del sistema 100 implementadas en un sistema operativo de tipo Unix (por ejemplo, un sistema operativo basado en Linux, un sistema operativo basado en Unix, etc.), los permisos 120, 122 pueden incluir uno o más bits pegajosos. En la realización ilustrativa provista en el presente documento, el bit pegajoso se puede añadir a los permisos de directorio, por ejemplo, para dotar a cualquier usuario editor 124 particular de la capacidad exclusiva de modificar, borrar, renombrar o mover los archivos 118 que son propiedad del usuario editor 124 particular (protegiendo de ese modo el trabajo del usuario frente a la sobrescritura por otro usuario editor 124) sin poner en peligro o afectar de forma negativa a la capacidad de otros usuarios editores 124 asignados al proyecto de NLE asociado con el directorio 116 de crear, dentro del directorio 116, unos archivos 118 que contienen metadatos adicionales que adicionalmente definen o contribuyen al proyecto o proyectos de NLE asociados con el directorio 116. Además, en algunas realizaciones, se puede incluir un bit o bits pegajosos adicionales, por ejemplo, aplicado cada uno a un directorio de nivel secundario contenido dentro del directorio 116.

El módulo de permisos 110 se puede configurar para asignar de forma automática los permisos 120, 122 basándose

en ajustes de permiso predeterminados, que se pueden almacenar en el módulo de permisos 110. Como ejemplos, uno o más de los conjuntos de ajustes de permiso predeterminados se pueden preprogramar en el módulo de permisos 110; pueden ser seleccionados por un usuario editor 124, administrador de sistemas o usuario de nivel raíz tras la creación de los mismos por el usuario editor 124, administrador de sistemas o usuario de nivel raíz; se pueden basar en los directorios 116 previamente creados; se pueden basar en unos archivos 118 previamente creados; se pueden predeterminar de alguna otra forma; o puede ser cualquier combinación de los mismos. El generador de GUI 114 se puede configurar para crear de forma automática una interfaz de usuario en todos o algunos de los clientes de usuario 128 que dota a los usuarios editores 124 de información en forma de uno o más indicadores de visualización con respecto a la propiedad, permisos, ubicaciones, información de extracción, historia de extracción, cualquier otra información, o cualquier combinación de los mismos,

El usuario de servidor 112 puede ser una entidad usuario plenamente automatizada y puede ser controlado, por ejemplo, por un dispositivo informático tal como un dispositivo de servidor (incluyendo un dispositivo de servidor que comprende un medio legible por ordenador e implementado de acuerdo con el dispositivo informático 600 representado en la figura 6 y descrito en detalle en el presente documento). De forma similar a los usuarios editores 124, el usuario de servidor 112 también puede recibir la propiedad de los archivos 118 y / o los directorios 116. En la realización ilustrativa implementada para programas de NLE, el usuario de servidor 112 proporciona un mecanismo mediante el cual los usuarios editores 124 están habilitados para transferir la propiedad de un archivo 118 entre sí mismos. Más específicamente, los usuarios editores 124 están habilitados para dar instrucciones al sistema 100 para transferir la propiedad a y desde el usuario de servidor 112, permitiendo de ese modo que los usuarios editores 124 pasen la propiedad de los archivos ente sí a través del intermediario del usuario de servidor 112. Además, puede haber una provisión para la implementación de un número limitado de usuarios administrativos a los que se concede el privilegio específico de modificar la propiedad de archivos (por ejemplo, quitarle la propiedad a un usuario editor 124 particular). De forma similar, puede haber una provisión para la implementación de "usuarios con derechos restringidos" para los usuarios editores 124 que no pueden tomar el control o la propiedad de cualquier archivo 118, y a los que no se les permite escribir ningún archivo 118 en ningún directorio 116. En consecuencia, se debería apreciar que los permisos 120, 122 se pueden establecer para proporcionar excepciones a los derechos de acceso de los usuarios editores 124 particulares, y / o para proporcionar niveles de acceso diferentes para tipos diferentes de usuarios editores 124.

Los usuarios editores 124 están habilitados para usar sus clientes de usuario 128 respectivos para dar instrucciones al módulo de permisos 110 para cambiar la propiedad de un directorio 116 y / o un archivo 118. En la realización ilustrativa descrita en el presente documento, cualquier usuario editor 124 dado está habilitado para dar instrucciones al módulo de permisos 110 solo para (a) cambiar la propiedad de un archivo 118 de ese usuario editor 124 particular al usuario de servidor 112, y (b) cambiar la propiedad de un archivo 118 del usuario de servidor 112 a ese usuario editor 124 particular. De esta forma, se puede prohibir que los usuarios editores 124 den instrucciones al módulo de permisos 110 para cambiar las propiedades de un archivo 118 directamente de cualquier primer usuario editor 124 a cualquier segundo usuario editor 124.

En consecuencia, en general, el módulo de permisos 110 se puede equipar con la capacidad de cambiar la propiedad de cualquier directorio 116 y cualquier archivo 118, y más específicamente se puede equipar con la capacidad de transferir la propiedad de cualquier archivo 118 entre los usuarios editores 124 y el usuario de servidor 112 (en ambos sentidos).

Las capacidades de transferencia específicas proporcionadas a los usuarios editores 124 y descritas en el presente documento, así como otras interacciones con el sistema 100, son habilitadas por los uno o más clientes de usuario 128. El cliente de usuario 128 se puede operar en una estación de trabajo de usuario, que puede ser cualquier dispositivo informático adecuado capaz de operar el cliente de usuario 128, incluyendo, a modo de ejemplo, un "portátil", un "escritorio", un "dispositivo de mano", un "dispositivo móvil", un "ordenador de tipo tableta", un "servidor informático" adicional, un "transceptor portátil", un "adaptador multimedia" (por ejemplo, para TV por Internet), y cualquier otro dispositivo informático. En consecuencia, cualquiera de las características, componentes y funciones del sistema 100 descrito en el presente documento (incluyendo la GUI / visualización) se puede personalizar y / o alterar para dar cabida a la conexión, el funcionamiento y el tipo específico de la estación de trabajo con la que se está comunicando el sistema 100.

Más específicamente, los usuarios editores 124 pueden interactuar con una o más GUI creadas por el generador de GUI 114 y proporcionadas a los usuarios editores 124 como visualizaciones en sus clientes de usuario 128 respectivos en una estación de trabajo de usuario. El generador de GUI 114 puede actualizar de forma automática la información presentada en las GUI en los clientes de usuario 128 para reflejar los cambios en las propiedades y / o el estado de un archivo 118. Específicamente, el generador de GUI 114 actualiza de forma automática los indicadores de propiedad y los indicadores de estado en los clientes de usuario posteriormente a los cambios en los mismos. El generador de GUI 114 puede generar visualizaciones que incluyen cualquiera de las características de visualización ilustrativas descritas en el presente documento y representadas en las figuras 2 a 4B.

La figura 2 representa una captura de pantalla de un cliente de usuario 138 ilustrativo implementado como una ventana de navegador 130 que se ejecuta dentro del entorno de un programa de NLE 132. En la realización ilustrativa representada en la figura 2, tras su lanzamiento, la ventana de navegador 130 se reanuda de forma automática con el

mismo tamaño y posición en el visualizador del usuario y / o con respecto al entorno del programa de NLE 132 que la ventana de navegador 130 ocupada al final de la sesión más reciente del usuario editor. En el ejemplo descrito en el presente documento, la ventana de navegador 130 se representa para un usuario editor 124 que se denominará en el presente documento "jeff".

La figura 3A representa una captura de pantalla de primer plano de la ventana de navegador 130 representada en la esquina del programa de NLE 132 de jeff en la figura 2. La ventana de navegador 130 puede ser cualquier visualización electrónica interactiva de información adecuada. En la realización ilustrativa de las figuras 3A y 3B, los metadatos se almacenan en los archivos 118 (por ejemplo, archivos contenedores, archivos de proyecto, etc.) almacenados en el directorio 116. En consecuencia, la ventana de navegador 130 presenta la estructura de directorios que incluye el directorio 116 con los archivos 118 contenidos en el mismo. Como se representa en las figuras 3A y 3B, el directorio 116 (etiquetado "demo1" en este ejemplo) es indicado en la ventana de navegador 130 por un icono de carpeta y los archivos que contienen los metadatos son indicados por iconos de archivo. En consecuencia, cada icono de carpeta indica un directorio en un dispositivo de almacenamiento legible por ordenador no transitorio que está asociado con el proyecto de NLE y en el que se ubican los archivos 118.

Una pluralidad de indicadores de metadatos 152 que representan los metadatos que definen el proyecto de NLE se incluyen en la ventana de navegador 130. Más específicamente, dado que los metadatos se almacenan en los archivos 118 en la realización ilustrativa de las figuras 3A y 3B, los indicadores de metadatos 152 más específicamente pueden comprender indicadores de archivo, cada uno de los cuales representa un archivo que contiene metadatos que definen o contribuyen al proyecto de NLE. Adicionalmente, la ventana de navegador 130 incluye numerosos indicadores de estado de metadatos que proporcionan una indicación del estado de los metadatos que definen o contribuyen al proyecto de NLE. En consecuencia, en la realización ilustrativa de las figuras 3A y 3B, los indicadores de estado de metadatos incluidos en la ventana de navegador 130 más específicamente pueden comprender indicadores de estado de archivo.

En la realización ilustrativa de las figuras 3A y 3B, se utilizan tres indicadores de estado de archivo diferentes, incluyendo un indicador de archivo bloqueado 144, un indicador de archivo desbloqueado 136 y un indicador de archivo reclamable 143. En la realización ilustrativa de las figuras 3A y 3B, el indicador de archivo bloqueado 144 indica que un archivo ha sido extraído por otro usuario editor aparte de jeff, es propiedad de otro usuario editor aparte de jeff, y es de solo lectura para jeff. El indicador de archivo reclamable 143 indica que un archivo está disponible para ser extraído y, por lo tanto, es "reclamable" por jeff. El indicador de archivo desbloqueado 136 indica que un archivo es extraído por jeff, propiedad de jeff, y escribible solo por jeff.

En consecuencia, un archivo que se "extrae" es un archivo para el cual se ha asignado la propiedad a uno de los usuarios editores 124. Un archivo que está "disponible para ser extraído" es un archivo para el cual se ha asignado la propiedad al usuario de servidor 112.

En la realización ilustrativa de las figuras 3A a 4B, el indicador de archivo bloqueado 144 incluye un icono sólido y cerrado de un candado. El indicador de archivo desbloqueado 136 incluye un icono abierto de un candado. El indicador de archivo reclamable 143 incluye un icono hueco y cerrado de un candado. Además, los indicadores de archivo 136, 143, 144 se pueden codificar con colores, para proporcionar una indicación adicional a los usuarios del estado de los archivos asociados. Por ejemplo, la totalidad de los indicadores de archivo bloqueado 144 se pueden colorear de rojo, la totalidad de los indicadores de archivo reclamable 143 se pueden colorear de amarillo, y la totalidad de los indicadores de archivo desbloqueado 136 se pueden colorear de verde.

La ventana de navegador 130 también puede incluir unos indicadores de propietario 140 que indican qué usuario editor 124 particular es propietario de un archivo particular que se extrae. Por ejemplo, los indicadores de propietario 140 pueden incluir una cadena de texto que representa un nombre de un propietario, tal como "andy". En la ventana de navegador 130 de la realización ilustrativa de las figuras 3A a 4B, los indicadores de propietario 140 se colocan bajo una columna de "Propietario" 142. En la realización ilustrativa de las figuras 3A a 3B, solo se presenta un indicador de propietario 140 para archivos que son propiedad de los usuarios editores 124 que no son jeff. Si un archivo se ha asignado a jeff o al usuario de servidor 112, entonces no hay indicador de propietario correspondiente alguno bajo la columna de "Propietario" 142.

La figura 3B representa una ventana de navegador 130 resultante posteriormente a una acción por jeff de asumir la propiedad de un archivo titulado "Acto_2.fcp" al dar instrucciones al módulo de permisos 110 para transferir la propiedad (y, por lo tanto, un acceso de escritura) de "Acto_2.fcp" del usuario de servidor 112 a jeff. En consecuencia, en la figura 3A, el archivo "Acto_2.fcp" se presenta con un indicador de archivo reclamable 143, debido a que, en ese momento, el usuario de servidor 112 es propietario del archivo. Por otro lado, en la figura 3B, el archivo "Acto_2.fcp" se presenta con un indicador de archivo bloqueado 136, debido a que, en ese momento, jeff es propietario del archivo. La instrucción para reclamar la propiedad de un archivo actualmente propiedad del usuario de servidor 112 se puede implementar usando cualquier orden adecuada, incluyendo seleccionar (por ejemplo, al hacer clic sobre) el indicador de archivo reclamable 143 que indica que un archivo está sin asignar y está disponible para ser extraído.

De forma similar, las figuras 4A y 4B representan la ventana de navegador 130 de jeff antes y después

(respectivamente) de una acción por otro usuario editor 124 llamado "jane" de extraer y recibir la propiedad del archivo "Urgencias_1.fcp" al dar instrucciones al módulo de permisos 110 para transferir la propiedad del usuario de servidor 112 a jane. En la figura 4A, el archivo "Urgencias_1.fcp" está acompañado por un indicador de archivo reclamable 143, que indica que el archivo es propiedad del usuario de servidor 112 y tiene la capacidad de que el archivo sea extraído por jeff. En la figura 4B, por otro lado, la ventana de navegador 130 se ha actualizado de tal modo que, en su lugar, el archivo "Urgencias_1.fcp" está acompañado por un indicador de archivo bloqueado 144, que indica a jeff que otro usuario (jane) ha extraído el archivo "Urgencias_1.fcp". Además de actualizar de forma automática los indicadores de estado de archivo, el generador de GUI 114 también actualiza de forma automática la ventana de navegador 130 para incluir los indicadores de propietario 140 apropiados, por ejemplo, al añadir el texto "jane" junto al texto "Urgencias_1.fcp".

En consecuencia, usando un sistema de este tipo de indicadores de estado de archivo, un estado personalizado de cualquier archivo se puede presentar a cualquier usuario editor 124 desde la perspectiva de ese usuario de la propiedad de sí mismo o la propiedad por otros usuarios editores 124, como se describe en el presente documento. De forma similar, los indicadores de propiedad pueden proporcionar una información personalizada a cada usuario con respecto a quién es propietario de archivos que se extraen.

La figura 5A representa un método mediante el cual el sistema 100 permite que un usuario editor 124 asuma la propiedad, de acuerdo con algunas realizaciones de la presente invención. La propiedad de un archivo 118 almacenado en el directorio 116 se puede asignar inicialmente, usando al menos un procesador, a un propietario (la etapa 510). Esto puede tener lugar, por ejemplo, tras crear o importar un archivo 118. Se puede asignar de forma automática que el propietario inicial sea el usuario editor particular que creó el archivo 118. Como alternativa, el propietario inicial puede ser el usuario de servidor 112, o cualquier otro usuario adecuado. En una realización ilustrativa, la propiedad de los archivos 118 en el directorio 116 es transferible al menos entre uno cualquiera de los miembros del grupo de usuarios 126 y el usuario de servidor 112. El módulo de permisos 110 recibe, a través de al menos un dispositivo de entrada, una instrucción de uno cualquiera de los uno o más usuarios editores 124 para asumir la propiedad de un archivo 118 (la etapa 512). Más específicamente, la instrucción recibida en la etapa 512 es una instrucción para transferir la propiedad del usuario de servidor 112 al usuario editor 124 desde el que se recibe la instrucción en la etapa 512. Tras recibir la instrucción, el módulo de permisos 110 transfiere, usando al menos un procesador, la propiedad del archivo 118 solicitado del usuario de servidor 112 al usuario que envió la instrucción (la etapa 514).

Basándose en el cambio de la propiedad del archivo 118, el módulo de permisos 110 puede dar instrucciones al generador de GUI 114 para actualizar, usando al menos un procesador, los indicadores de propiedad correspondientes y los indicadores de estado de archivo correspondientes para cada cliente de usuario 128 (la etapa 516), de una forma tal que dota a cada usuario editor 124 de una perspectiva personalizada de la propiedad de y de la disponibilidad para extraer los archivos 118. Dependiendo de los detalles de los cambios en la propiedad y el estado de archivo, la etapa 516 puede incluir cualquiera de: indicar a cualquier usuario editor que no tiene la propiedad del archivo 118 de que el archivo 118 está bloqueado y se extrae, indicar a cualquier usuario editor que sí tiene la propiedad del archivo 118 de que el archivo 118 está desbloqueado y se extrae, e indicar el nombre de un usuario editor con la propiedad del archivo 118 a cualquier usuario editor que no tiene la propiedad del archivo 118.

La figura 5B representa un método mediante el cual el sistema 100 permite que un usuario editor 124 renuncie a la propiedad, de acuerdo con algunas realizaciones de la presente invención. La propiedad de un archivo 118 almacenado en el directorio 116 se puede asignar inicialmente, usando al menos un procesador, a un propietario (la etapa 518), como se ha descrito anteriormente con referencia a la etapa 510 a partir de la figura 5A. El sistema 100 recibe entonces, a través de al menos un dispositivo de entrada, una instrucción electrónica de uno cualquiera de los uno o más usuarios editores 124 para renunciar a la propiedad de un archivo 118 (la etapa 520). Más específicamente, la instrucción recibida en la etapa 520 es una instrucción para transferir la propiedad al usuario de servidor 112 desde el usuario editor 124 desde el que se recibe la instrucción en la etapa 520. Tras recibir la instrucción, el módulo de permisos 110 transfiere, usando al menos un procesador, la propiedad del archivo 118 solicitado del usuario de servidor 112 al usuario que envió la instrucción (la etapa 522). Basándose en el cambio de la propiedad del archivo 118, el módulo de permisos 110 puede dar instrucciones al generador de GUI 114 para actualizar, usando al menos un procesador, los indicadores de propiedad correspondientes y los indicadores de estado de archivo correspondientes para cada cliente de usuario 128 (la etapa 524), como se ha descrito previamente en el presente documento con referencia a la etapa 516 de la figura 5A. Por ejemplo, la etapa 524 puede incluir indicar a la totalidad de los usuarios que ningún usuario editor tiene la propiedad del archivo 118 y el archivo 118 está disponible para ser extraído.

Además, se puede dotar a los usuarios editores 124 de la capacidad de crear unos archivos 118 nuevos dentro de los directorios existentes para los cuales se les concede acceso, así como la capacidad de crear directorios y / o subdirectorios nuevos. Los permisos 122 para un archivo 118 nuevo se pueden generar a partir de un conjunto predeterminado de permisos y se pueden asignar de forma automática en cualquier número de formas. Por ejemplo, los permisos 122 en un archivo 118 nuevo se pueden asignar de forma automática basándose en ajustes preprogramados en el módulo de permisos 110, se pueden basar en selecciones por el usuario editor 124 que crea el archivo 118, se pueden basar en selecciones de un administrador de sistemas o usuario de nivel raíz, se pueden basar en el directorio 116 en el que se creó el mismo, se pueden basar en otros archivos 118 dentro del directorio 116 en el

que se creó el mismo, se pueden basar en otros factores, o se pueden basar en cualquier combinación adecuada de los mismos. En una realización ilustrativa, la propiedad de un archivo 118 nuevo se asigna de forma automática al usuario editor 124 que creó el archivo 118 nuevo.

5 Cuando se crea un directorio 116 nuevo, un usuario editor 124 se puede habilitar para seleccionar uno o más usuarios editores 124 que se van a incluir en el grupo de usuarios 126 asociado con el directorio 116 nuevo. Como alternativa, la pertenencia de los usuarios editores 124 en un directorio 116 nuevo se puede basar en, por ejemplo, una asignación de un grupo específico de los usuarios editores 124 a un espacio de proyectos con el que está contenido el directorio 116. Por ejemplo, en algunas realizaciones ilustrativas, el espacio de proyectos es un directorio del nivel más alto al
10 que se asignan una pluralidad de usuarios editores 124 y que contiene el directorio 116, de tal modo que cualquier directorio 116 nuevo creado por los usuarios editores 124 en el espacio de proyectos se asigna de forma automática al mismo grupo de los usuarios editores 124 asignados al espacio de proyectos. Además, los permisos 120 para el directorio 116 nuevo se pueden generar a partir de un conjunto predeterminado de permisos 120 y se pueden asignar en cualquier número de formas, por ejemplo, para lograr la funcionalidad descrita previamente en el presente documento. Por ejemplo, los permisos 120 sobre los directorios 116 nuevos se pueden asignar de forma automática basándose en ajustes preprogramados en el módulo de permisos 110, se pueden basar en selecciones por el usuario editor 124 que crea el directorio, se pueden basar en selecciones de un administrador de sistemas o usuario de nivel raíz, se pueden basar en otros permisos 120 de los directorios 116 existentes, se pueden basar en los permisos establecidos para el espacio de proyectos, se pueden basar en otros factores, o se pueden basar en cualquier
20 combinación adecuada de los mismos. En una realización ilustrativa, la propiedad de un archivo 118 nuevo se asigna de forma automática al usuario editor 124 que creó el archivo 118 nuevo.

Algunas realizaciones de la presente invención se pueden implementar para proyectos compartidos de programas de NLE, que requieren múltiples editores para trabajar de forma colaborativa juntos sobre los mismos metadatos que
25 definen, por ejemplo, un único proyecto. Específicamente, esto puede incluir aplicaciones de software de NLE que guardan metadatos como un archivo de proyecto (por ejemplo, en archivos "monolíticos") tales como FINAL CUT PRO®, fabricado y comercializado por la empresa Apple, Inc. De forma similar, algunas realizaciones de la presente invención se pueden implementar adicionalmente en otros productos y software de NLE que almacenan metadatos en subunidades discretas, o archivos contenedores, tales como las aplicaciones de software de NLE proporcionadas por
30 Avid Technology, Inc., de Burlington, Massachusetts.

Se pueden implementar características adicionales de acuerdo con la presente invención. Por ejemplo, en el lado de cliente del sistema, si un usuario editor 124 selecciona un archivo 118 particular (por ejemplo, al hacer clic sobre un nombre de archivo o en un icono de visualizador correspondiente), el archivo 118 se abrirá en una aplicación de NLE
35 correspondiente. Esta capacidad no está actualmente disponible en varias aplicaciones de NLE convencionales. Es decir, en ausencia de la implementación de la presente invención, en muchas aplicaciones de NLE, un usuario no puede navegar a una ubicación de archivo en, por ejemplo, Windows® Explorer (fabricado y comercializado por Microsoft Corporation, con sede en Redmond, Washington), y seleccionar el archivo que se va a abrir sin recibir un mensaje de error.

Además, cuando un usuario abre un archivo 118 a través de una aplicación de cliente implementada de acuerdo con la presente invención, si el archivo 118 es propiedad de otro usuario editor (por ejemplo, el usuario editor 124, lo que da como resultado un acceso de solo lectura), la aplicación de cliente puede presentar un mensaje de advertencia, tal como "Estáis abriendo un Archivo de Proyecto con permisos de Solo Lectura. No se puede guardar ningún cambio
45 que hagáis. ¿Qué quieres hacer?" La aplicación de cliente puede presentar entonces varias opciones al usuario, tales como Cancelar (cancelar la acción de abrir el archivo 118), Proceder - Solo Lectura (abrir el archivo 118 en un modo de solo lectura), o similares.

De forma similar, cuando un usuario abre un archivo 118 a través de una aplicación de cliente implementada de acuerdo con la presente invención, si el archivo 118 es propiedad del usuario de servidor 112 y, por lo tanto, el archivo 118 está disponible para que el usuario tome la propiedad del mismo, la aplicación de cliente puede proporcionar un mensaje de advertencia, tal como, "No eres el propietario de este Archivo de Proyecto. Tus cambios no se guardarán a menos que tomes la propiedad. ¿Quieres tomar la propiedad?" La aplicación de cliente puede presentar entonces
50 varias opciones al usuario, tales como Cancelar (cancelar la acción de abrir el archivo 118), Proceder - Solo Lectura (abrir el archivo 118 en un modo de solo lectura), Tomar la Propiedad (enviar una instrucción al sistema 100 para asumir la propiedad del archivo 118), Abrir, y / o similares. En consecuencia, en tales situaciones, los usuarios pueden tener la capacidad de asumir la propiedad tras abrir el archivo 118, sin tener en primer lugar que hacer clic sobre el indicador de archivo reclamable 143 correspondiente.

En consecuencia, el sistema 100 habilita que un grupo de los usuarios editores 124 asignados a un proyecto de NLE intercambie un acceso de escritura exclusivo a metadatos al asumir y renunciar a la propiedad de archivos que contienen los metadatos. La propiedad se transfiere a través del intermediario del usuario de servidor 112. Esto se puede realizar sin mover la ubicación de unos archivos 118 que contienen los metadatos y sin crear duplicados, copias o versiones de los archivos 118. Además, cada usuario editor 124 está habilitado para ver una GUI personalizada que
65 presenta los archivos 118 que forman el proyecto de NLE en uno de tres estados: bloqueado (extraído para un acceso de escritura y, por lo tanto, propiedad de otro usuario editor 124); desbloqueado (extraído para un acceso de escritura

y, por lo tanto, propiedad del usuario editor 124 para el que se personaliza la GUI); y reclamable (propiedad del usuario de servidor 112 y capaz de ser extraído).

La figura 6 ilustra un dispositivo informático 600 ilustrativo dentro de un entorno operativo ilustrativo para implementar métodos y sistemas ilustrativos de la presente invención. El dispositivo informático 600 es meramente un ejemplo ilustrativo de un entorno informático adecuado y no limita en modo alguno el alcance de la presente invención. Un "dispositivo informático", como es representado por la figura 6, puede incluir una "estación de trabajo", un "servidor", un "portátil", un "escritorio", un "dispositivo de mano", "dispositivo móvil", un "ordenador de tipo tableta", u otros dispositivos informáticos, como sería entendido por los expertos en la materia. Dado que el dispositivo informático 600 se representa para fines ilustrativos, algunas realizaciones de la presente invención pueden utilizar cualquier número de dispositivos informáticos 600 en cualquier número de formas diferentes con el fin de implementar una única realización de la presente invención. En consecuencia, algunas realizaciones de la presente invención no se limitan a un único dispositivo informático 600, como sería apreciado por un experto en la materia, ni las mismas se limitan a un único tipo de implementación o configuración del dispositivo informático 600 ilustrativo.

El dispositivo informático 600 puede incluir un bus 610 que se puede acoplar a uno o más de los siguientes componentes ilustrativos, directa o indirectamente: una memoria 612, uno o más procesadores 614, uno o más componentes de presentación 616, unos puertos de entrada / salida 618, unos componentes de entrada / salida 620 y una fuente de alimentación 624. Un experto en la materia apreciará que el bus 610 puede incluir uno o más buses, tales como un bus de direcciones, un bus de datos, o cualquier combinación de los mismos. Un experto en la materia apreciará adicionalmente que, dependiendo de las aplicaciones y usos previstos de una realización particular, una multitud de estos componentes se pueden implementar mediante un único dispositivo. De forma similar, en algunos casos, un único componente se puede implementar mediante múltiples dispositivos. En ese sentido, la figura 6 es meramente ilustrativa de un dispositivo informático ilustrativo que se puede usar para implementar una o más realizaciones de la presente invención, y no limita en modo alguno la invención.

El dispositivo informático 600 puede incluir o interaccionar con una diversidad de medios legibles por ordenador. Por ejemplo, los medios legibles por ordenador pueden incluir una Memoria de Acceso Aleatorio (RAM); memoria de solo lectura (ROM); memoria de solo lectura programable borrable electrónicamente (EEPROM); memoria flash u otras tecnologías de memoria; CDROM, discos versátiles digitales (DVD) u otros medios ópticos u holográficos; casetes magnéticos, cinta magnética, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnético que se pueden usar para codificar información y a los que puede acceder el dispositivo informático 200.

La memoria 612 puede incluir medios de almacenamiento informático en forma de memoria volátil y / o no volátil. La memoria 612 puede ser extraíble, no extraíble, o cualquier combinación de las mismas. Dispositivos tales como unidades de disco duro, memoria de estado sólido, unidades de disco óptico, y similares, son dispositivos de hardware ilustrativos. El dispositivo informático 600 puede incluir uno o más procesadores que leen datos a partir de componentes tales como la memoria 612, los diversos componentes de E / S 616, etc. El componente o componentes de presentación 616 presentan indicaciones de datos a un usuario o a otro dispositivo. Los componentes de presentación ilustrativos incluyen un dispositivo de visualización, altavoz, componente de impresión, componente de vibración, etc.

Los puertos de E / S 618 pueden permitir que el dispositivo informático 600 se acople lógicamente a otros dispositivos, tales como unos componentes de E / S 620. Algunos de los componentes de E / S 620 se pueden integrar en el dispositivo informático 600. Los ejemplos de tales componentes de E / S 620 incluyen un micrófono, palanca de mando, dispositivo de registro, controlador de juegos, antena parabólica, escáner, impresora, dispositivo inalámbrico, dispositivo de interconexión de redes, y similares.

Las figuras 7 a 10 son unas capturas de pantalla que representan propiedades y permisos, por ejemplo, como se vería desde un terminal de servidor. En particular, las figuras 7 a 10 representan cambios en las propiedades y los permisos que son realizados por el módulo de permisos 110 basándose en recibir diversas instrucciones de los usuarios editores 124. En la realización ilustrativa de las figuras 7 a 10, se dota al usuario de servidor 112 del nombre de usuario "editar compartir". Los nombres de usuario "andy" y "jeff" representan dos usuarios editores 124 que se incluyen en un grupo de usuarios 126 titulado "editores". Las figuras 7 y 8 representan cambios en las propiedades y los permisos para un directorio ilustrativo titulado "demo1", así como para los archivos contenidos en el directorio ilustrativo "demo1". Como se observa en la figura 7, las propiedades y los permisos indican que el archivo "Acto_2.fcp" es propiedad del usuario de servidor ("editar compartir"). La figura 8 representa la propiedad y los permisos del archivo "Acto_2.fcp" después de que el usuario editor 124 llamado jeff haya enviado al sistema 100 una instrucción para asumir la propiedad (y, por lo tanto, un acceso de escritura) del archivo "Acto_2.fcp". Como se observa en la figura 8, posteriormente a la presentación de la instrucción de jeff para asumir la propiedad, el archivo "Acto_2.fcp" es propiedad de jeff. Si posteriormente a recibir la propiedad del archivo "Acto_2.fcp", jeff fuera a presentar una instrucción para ceder la propiedad del archivo "Acto_2.fcp" al sistema 100, entonces los permisos y las propiedades volverían al estado representado en la figura 7.

De forma similar, las figuras 9 y 10 representan los permisos y las propiedades para un subdirectorio "Urgencias" del directorio "demo1", así como los contenidos de archivo en el subdirectorio "Urgencias". Como se representa en la

figura 9, el archivo "Urgencias 1.fcp" es inicialmente propiedad del usuario de servidor 112. La figura 10 representa la propiedad y los permisos para el subdirectorio "Urgencias" y sus contenidos posteriormente a que un usuario editor llamado "jane" envíe una instrucción al sistema 100 para asumir la propiedad de (y, por lo tanto, un acceso de escritura a) el archivo "Urgencias_1.fcp".

Algunas realizaciones de la presente invención proporcionan numerosos beneficios frente a los sistemas existentes. Por ejemplo, se puede dotar a los usuarios editores de un control automatizado sobre los derechos de acceso a archivos mientras se mantiene un cliente de usuario conveniente en el que los archivos, los estados de archivos y propietarios, se pueden localizar fácilmente y verse de forma integral. Adicionalmente, algunas realizaciones de la presente invención pueden habilitar una compartición de archivos conveniente sin la necesidad de mover archivos, crear copias de archivos o hacer que los permisos sean asignados de forma manual, por ejemplo, por un administrador de sistemas no automatizado. Otros beneficios serán apreciados por un experto en la materia tras la lectura de la presente memoria descriptiva.

La presente invención se ha descrito en el presente documento con referencia a implementaciones ilustrativas para aplicaciones de software de NLE que almacenan metadatos que definen un proyecto de NLE en forma de archivos (por ejemplo, archivos contenedores, archivos de proyecto "monolíticos", etc.). Sin embargo, en otras realizaciones, los metadatos se almacenan en formatos diferentes. Por ejemplo, en algunas realizaciones alternativas e ilustrativas de acuerdo con la presente invención, los metadatos que definen uno o más proyecto(s) de NLE se almacenan de forma colectiva en una base de datos (por ejemplo, todos ellos en un único archivo de base de datos). En estas realizaciones alternativas, cada elemento de metadatos se almacena como uno o más objetos / entradas de base de datos, como sería apreciado por un experto en la materia tras la lectura de la presente memoria descriptiva. Al igual que con los archivos 118 descritos previamente en el presente documento, el sistema 100 puede habilitar un intercambio de permisos de escritura exclusiva a metadatos almacenados en una base de datos al transferir la propiedad de los objetos de base de datos en los que se almacenan los metadatos. Al igual que con las realizaciones de las figuras 1 a 10, se puede dotar a cada usuario editor 124 de una GUI que presenta los objetos de base de datos que forman el proyecto de NLE en uno de tres estados: bloqueado (extraído para un acceso de escritura y, por lo tanto, propiedad de otro usuario editor 124); desbloqueado (extraído para un acceso de escritura y, por lo tanto, propiedad del usuario editor 124 para el que se personaliza la GUI); y reclamable (propiedad del usuario de servidor 112 y capaz de ser extraído).

En tales realizaciones en las que se almacenan metadatos en una base de datos (por ejemplo, una base de datos MySQL proporcionada por Oracle Corporation con sede en Redwood City, California, o una base de datos PostgreSQL proporcionada por el Grupo de Desarrollo Global de PostgreSQL, como sería apreciado por los expertos en la materia), los metadatos en la base de datos se pueden organizar en "directorios virtuales" a través del uso de etiquetas específicas y / u otros campos. Más específicamente, cada uno de los objetos de base de datos que forman los metadatos se puede etiquetar, marcar, etc. como que tiene una ubicación específica en una estructura de directorios ficticia. En consecuencia, las etiquetas, marcas, etc. habilitan que los metadatos se manipulen (por ejemplo, se organicen, clasifiquen, filtren, sean propiedad de, etc.) de una forma similar a las estructuras de directorios tradicional. Como un ejemplo ilustrativo y no limitante, cada elemento de metadatos se puede marcar con una ruta de archivo apropiada en un directorio jerárquico ficticio, permitiendo de ese modo que el elemento de metadatos sea manipulado de acuerdo con la misma estructura organizativa que habilita un directorio jerárquico convencional.

Tras la lectura de la presente memoria descriptiva, un experto en la materia apreciará otras formas adecuadas más para almacenar y organizar metadatos que definen uno o más proyectos de NLE. La totalidad de tales alternativas y modificaciones se contemplan dentro del alcance de la presente invención, en la medida en la que tales formatos y mecanismos de almacenamiento alternativos habiliten la transferencia de propiedad entre los usuarios editores 124 a través del intermediario del usuario de servidor 112. La presente invención no se limita de forma exclusiva a las realizaciones ilustrativas y a modo de ejemplo descritas en el presente documento.

Adicionalmente, algunas realizaciones de la presente invención se pueden implementar con numerosas características adicionales. Como un ejemplo no limitante, en algunas realizaciones, una vez que un usuario editor 124 cierra sesión de la aplicación de software de NLE (que el usuario editor 124 usa para modificar y guardar los metadatos propiedad del usuario editor 124), el sistema 100 transfiere de forma automática la propiedad de cualesquiera metadatos propiedad en ese momento de ese usuario editor 124 al usuario de servidor 112. En otras realizaciones, una vez que un usuario editor 124 cierra sesión de la aplicación de software de NLE, el sistema 100 permite que el usuario editor 124 conserve la propiedad de cualesquiera metadatos propiedad de ese usuario editor 124 en ese momento. En otras realizaciones más, el sistema 100 dota a los usuarios editores 124 de ambas de las opciones anteriores (por ejemplo, la primera opción para "insertar" de forma automática cualesquiera metadatos extraídos por el usuario editor 124 en ese momento en el que el usuario editor 124 cierra sesión de la aplicación de software de NLE, y la segunda opción para conservar la propiedad de cualesquiera metadatos extraídos por el usuario editor en el momento de cerrar sesión de la aplicación de software de NLE). Por ejemplo, el sistema 100 se puede configurar para presentar ambas opciones a los usuarios editores 124 y para permitir que algunos o la totalidad de los usuarios editores 124 seleccionen una opción preferida para servir como el comportamiento por defecto por el sistema 100.

Por ejemplo, la figura 11 representa otra realización de la ventana de navegador 130 adaptada para un sistema 100

en el que los metadatos que definen o contribuyen a un proyecto de NLE se almacenan, todos ellos, en una o más bases de datos. En consecuencia, cada metadato se almacena en forma de una o más entradas de base de datos y / o como uno o más objetos de base de datos. En tales realizaciones, como se ha descrito previamente en el presente documento, el control sobre un acceso de escritura exclusivo a los diversos metadatos se logra al transferir la propiedad de los objetos / entradas de base de datos específicos que contienen los metadatos.

Al igual que con la realización de las figuras 3A a 4B, la ventana de navegador 130 de la figura 11 visualiza los estados de la pluralidad de metadatos que definen un proyecto de NLE a un usuario editor particular asignado al proyecto de NLE. En general, la ventana de navegador 130 de la figura 11 puede ser cualquier visualización interactiva electrónica de información adecuada. Como se ha descrito previamente en el presente documento, la ventana de navegador 130 se personaliza para un usuario editor particular, y visualiza el estado de cada metadato (por ejemplo, capaz de ser extraído por ti, propiedad de otro usuario editor que no seas tú, etc.) desde la perspectiva del usuario editor particular. En general, la ventana de navegador 130 puede ser generada por el generador de GUI 114) para cada usuario editor que se asigna a un proyecto de NLE, de tal modo que se dota a cada usuario editor de una presentación personalizada del estado de los metadatos para los proyectos de NLE a los que esté asignado el mismo.

En la realización ilustrativa de la figura 11, la ventana de navegador 130 incluye una o más carpetas de proyecto 150 que indican, cada una, un "directorio virtual" (como se ha descrito previamente en el presente documento) para metadatos almacenados en una base de datos almacenada en un dispositivo de almacenamiento legible por ordenador no transitorio. Cada carpeta de proyecto 150 está asociada con un proyecto de NLE específico, y cada carpeta de proyecto 150 define (por ejemplo, a través del uso de etiquetas, marcadores u otros campos, de base de datos) un conjunto específico de metadatos almacenados en la base de datos y asociados con el proyecto de NLE. En las carpetas de proyecto están contenidos unos indicadores de carpeta 151 adicionales que indican "subdirectorios virtuales" de forma jerárquica por debajo del directorio virtual indicado por la carpeta de proyecto 150. Además, la ventana de navegador también incluye los indicadores de metadatos 152 que representan los metadatos que forman el proyecto de NLE.

Al igual que con la realización de las figuras 3A a 4B, la ventana de navegador 130 incluye un indicador de estado de metadatos para cada metadato representado por uno de los indicadores de metadatos 152. Además, como las realizaciones ilustrativas de las figuras 3A a 4B, cada uno de los indicadores de estado de metadatos es uno de: un indicador de metadatos desbloqueados 154 (por ejemplo, comparable con el indicador de archivo desbloqueado 136), un indicador de metadatos reclamables 156 (por ejemplo, comparable con el indicador de archivo reclamable 143) o un indicador de metadatos bloqueados 158 (por ejemplo, comparable con el indicador de archivo bloqueado 144). El indicador de metadatos bloqueados 158 puede indicar que los metadatos particulares han sido extraídos por otro usuario editor aparte del usuario editor para el cual se personaliza la ventana de navegador 130, es propiedad de otro usuario editor aparte del usuario editor para el cual se personaliza la ventana de navegador 130, y es de solo lectura para el usuario editor para el cual se personaliza la ventana de navegador 130. El indicador de metadatos reclamables 156 puede indicar que los metadatos particulares están disponibles para ser extraídos y, por lo tanto, son "reclamables" por el usuario editor para el cual se personaliza la ventana de navegador 130. El indicador de metadatos desbloqueados 154 puede indicar que los metadatos particulares son extraídos por el usuario editor para el cual se personaliza la ventana de navegador 130, propiedad del usuario editor para el cual se personaliza la ventana de navegador 130, y escribibles solo por el usuario editor para el cual se personaliza la ventana de navegador 130 (por ejemplo, con la excepción de los súper usuarios, usuarios de nivel raíz, usuarios con privilegios especiales, etc.).

Al igual que con las realizaciones ilustrativas de las figuras 3A a 4B, en la realización ilustrativa de la figura 11, el indicador de metadatos bloqueados 158 incluye un icono sólido y cerrado de un candado. El indicador de metadatos desbloqueados 154 incluye un icono abierto de un candado. El indicador de archivo reclamable 143 incluye un icono hueco y cerrado de un candado. Alternativa o adicionalmente, los indicadores de estado de metadatos 136, 143, 144 se pueden codificar con colores, para proporcionar una indicación similar a los usuarios del estado de los archivos asociados. La ventana de navegador 130 de las figuras 11 también puede incluir los indicadores de propietario 140 (no mostrados en la figura 11) que indican qué usuario editor 124 particular es propietario de un archivo particular que se extrae.

Los metadatos descritos en el presente documento incluyen cualesquiera metadatos que definen, contribuyen a, o están asociados de otro modo con un proyecto de NLE. En consecuencia, los metadatos pueden incluir referencias a clips, códigos de tiempo, decisiones de edición, notas, y cualquier otro metadato. Tras la lectura de la presente memoria descriptiva, un experto en la materia apreciará una amplia diversidad de otros tipos de metadatos asociados con proyectos de NLE que se pueden editar, borrar, mover o modificar de otro modo. La totalidad de tales metadatos se contempla dentro del alcance de la presente invención y se puede implementar con algunas realizaciones de la presente invención.

Numerosas modificaciones y realizaciones alternativas de la presente invención serán evidentes a los expertos en la materia a la vista de la descripción anterior. Por ejemplo, en algunas realizaciones alternativas de la presente invención, los usuarios editores 124 están habilitados para transferir la propiedad de los metadatos entre sí directamente, sin la etapa de pasar la propiedad de los metadatos al intermediario del usuario de servidor 112. En consecuencia, en tales realizaciones alternativas, los usuarios editores 124 están habilitados para dar instrucciones al

sistema 100 para renunciar a o ceder la propiedad de los metadatos (por ejemplo, almacenados en un archivo 118, almacenados en un objeto de base de datos, etc.) directamente a otro usuario editor 124. En consecuencia, esta descripción se ha de interpretar como ilustrativa únicamente y es para el fin de enseñar a los expertos en la materia el mejor modo para llevar a cabo la presente invención. Se tiene por objeto que la presente invención esté limitada solo en la medida en la que lo requieran las reivindicaciones adjuntas y las normas jurídicas aplicables.

5

REIVINDICACIONES

1. Un sistema de compartición de proyectos implementado por ordenador, que comprende:

- 5 uno o más archivos o una o más entradas de base de datos almacenados en un dispositivo de almacenamiento legible por ordenador no transitorio, para un proyecto de edición de vídeo no lineal; unos metadatos contenidos en los uno o más archivos o las una o más entradas de base de datos almacenados en un dispositivo de almacenamiento legible por ordenador no transitorio, definiendo los metadatos una parte del proyecto de edición de vídeo no lineal;
- 10 unos permisos sobre los metadatos almacenados en un dispositivo de almacenamiento legible por ordenador no transitorio, habilitando los permisos que un grupo de uno o más usuarios editores lea los metadatos y habilitando que solo un propietario de los metadatos modifique o borre los metadatos;
- un dispositivo de servidor configurado para controlar el acceso de escritura a los metadatos al transferir, usando al menos un procesador, la propiedad de los metadatos a o desde un usuario de servidor que no es un editor;
- 15 en donde, si un primer usuario editor de los uno o más usuarios editores tiene la propiedad de los metadatos, el primer usuario editor está habilitado para renunciar a la propiedad de los metadatos al dar instrucciones al dispositivo de servidor para transferir la propiedad de los metadatos al usuario de servidor y cada uno de los uno o más usuarios editores sin la propiedad de los metadatos no está habilitado para dar instrucciones al dispositivo de servidor para transferir la propiedad de los metadatos;
- 20 en donde, si el usuario de servidor tiene la propiedad de los metadatos, uno cualquiera de los uno o más usuarios editores está habilitado para asumir la propiedad de los metadatos al dar instrucciones al dispositivo de servidor para transferir la propiedad de los metadatos del usuario de servidor a ese uno de los uno o más usuarios editores;
- y
- 25 en donde la propiedad de los metadatos se transfiere sin mover los uno o más archivos o las una o más entradas de base de datos.

2. El sistema de la reivindicación 1, en donde los metadatos se almacenan en los uno o más archivos almacenados en un directorio, y en donde los permisos sobre los metadatos comprenden permisos a nivel de directorio y permisos a nivel de archivo.

3. El sistema de la reivindicación 2, en donde los permisos a nivel de directorio incluyen un bit pegajoso.

4. El sistema de la reivindicación 1, en donde los metadatos se almacenan en una o más entradas de base de datos de una base de datos.

5. El sistema de la reivindicación 1, que comprende adicionalmente unas instrucciones almacenadas en al menos un dispositivo legible por ordenador no transitorio, en donde la ejecución de las instrucciones por al menos un procesador da lugar a que una interfaz gráfica de usuario se presente en al menos un dispositivo de presentación, habilitando la interfaz gráfica de usuario que unas instrucciones para transferir la propiedad de los metadatos se envíen al dispositivo de servidor.

6. El sistema de la reivindicación 5, en donde la interfaz gráfica de usuario comprende un indicador de estado actual para los metadatos, indicando el indicador de estado actual, a uno de los uno o más usuarios editores, que los metadatos están actualmente disponibles para ser extraídos, son actualmente extraídos por ese uno de los uno o más usuarios editores o son actualmente extraídos por uno diferente de los uno o más usuarios editores.

7. El sistema de la reivindicación 1, en donde el dispositivo de servidor se ejecuta en un sistema operativo basado en Linux, uno basado en Unix o uno de tipo Unix.

8. Un método implementado por ordenador, que comprende:

- establecer, usando al menos un procesador, unos permisos sobre unos metadatos contenidos en uno o más archivos o una o más entradas de base de datos almacenados en un dispositivo de almacenamiento legible por ordenador no transitorio, habilitando los permisos que un grupo de uno o más usuarios editores lea los metadatos y habilitando que solo un propietario de los metadatos modifique o borre los metadatos;
- 55 controlar el acceso de escritura a los metadatos al transferir, usando al menos un procesador y tras recibir una instrucción a través de al menos un dispositivo de entrada, la propiedad de los metadatos a o desde un usuario de servidor que no es un editor;
- 60 en donde, si un primer usuario editor de los uno o más usuarios editores tiene la propiedad de los metadatos, el primer usuario editor está habilitado para renunciar a la propiedad de los metadatos al dar instrucciones al dispositivo de servidor para transferir la propiedad de los metadatos al usuario de servidor y cada uno de los uno o más usuarios editores sin la propiedad de los metadatos no está habilitado para dar instrucciones al dispositivo de servidor para transferir la propiedad de los metadatos;
- 65 en donde, si el usuario de servidor tiene la propiedad de los metadatos, uno cualquiera de los uno o más usuarios editores está habilitado para asumir la propiedad de los metadatos al dar instrucciones al dispositivo de servidor

para transferir la propiedad de los metadatos del usuario de servidor a ese uno de los uno o más usuarios editores; y en donde los metadatos definen una parte de un proyecto de edición de vídeo no lineal y la propiedad de los metadatos se transfiere sin mover los uno o más archivos o las una o más entradas de base de datos.

5 9. El método implementado por ordenador de la reivindicación 8, en donde el primer usuario tiene la propiedad de los metadatos, y en donde el método comprende adicionalmente:

recibir, a través de al menos un dispositivo de entrada, una instrucción del primer editor usuario para renunciar a la propiedad de los metadatos; y

10 basándose en la instrucción recibida, transferir, usando el al menos un procesador, la propiedad de los metadatos del primer usuario al usuario de servidor.

10. El método implementado por ordenador de la reivindicación 8, en donde el usuario de servidor tiene la propiedad de los metadatos, y en donde el método comprende adicionalmente:

15 recibir, a través de al menos un dispositivo de entrada, una instrucción de uno de los uno o más editores usuarios para asumir la propiedad de los metadatos; y, basándose en la instrucción recibida, transferir, usando el al menos un procesador, la propiedad de los metadatos del usuario de servidor al uno de los uno o más editores usuarios.

20 11. El método implementado por ordenador de la reivindicación 8, en donde los metadatos se almacenan en los uno o más archivos almacenados en un directorio, y en donde los permisos sobre los metadatos comprenden permisos a nivel de directorio y permisos a nivel de archivo.

25 12. El método implementado por ordenador de la reivindicación 11, en donde los permisos a nivel de directorio incluyen un bit pegajoso.

13. El método implementado por ordenador de la reivindicación 8, en donde los metadatos se almacenan en una o más entradas de base de datos de una base de datos.

30 14. El método implementado por ordenador de la reivindicación 8, que comprende adicionalmente unas instrucciones almacenadas en al menos un dispositivo legible por ordenador no transitorio, en donde la ejecución de las instrucciones por al menos un procesador da lugar a que una interfaz gráfica de usuario se presente en al menos un dispositivo de presentación.

35 15. El método implementado por ordenador de la reivindicación 14, en donde la interfaz gráfica de usuario comprende un indicador de estado actual para los metadatos, indicando el indicador de estado actual, a uno de los uno o más usuarios editores, que los metadatos están actualmente disponibles para ser extraídos, son actualmente extraídos por ese uno de los uno o más usuarios editores o son actualmente extraídos por uno diferente de los uno o más usuarios editores.

40 16. Un aparato configurado para generar una visualización electrónica interactiva de información para un primer usuario editor de un programa de edición de vídeo no lineal, estando asignado el primer editor a un proyecto de edición de vídeo no lineal (NLE) que comprende:

45 una pluralidad de metadatos contenidos en uno o más archivos o una o más entradas de base de datos, comprendiendo la visualización de información:

50 una pluralidad de indicadores de estado de metadatos que representan la pluralidad de metadatos que definen el proyecto de edición de vídeo no lineal, en donde cada metadato de la pluralidad de metadatos es propiedad o bien de un usuario editor que representa un editor asignado al proyecto de NLE o bien de un usuario de servidor que no representa un editor asignado al proyecto de NLE; y, para cada metadato de la pluralidad de metadatos, un indicador de estado asociado seleccionado de entre un grupo que consiste en:

55 un primer indicador de estado que indica la propiedad de los metadatos por el usuario de servidor; un segundo indicador de estado que indica la propiedad de los metadatos por el primer usuario editor; y un tercer indicador de estado que indica la propiedad de los metadatos por un usuario editor que no es el primer usuario editor;

60 en donde al primer usuario editor solo se le permite un acceso de escritura a los metadatos de la pluralidad de metadatos que son propiedad del primer editor; y en donde la selección del primer indicador de estado o el segundo indicador de estado da lugar a que un dispositivo de servidor transfiera la propiedad de los metadatos asociados con el primer indicador de estado o el segundo indicador de estado seleccionado sin mover los uno o más archivos o las una o más entradas de base de datos.

65 17. El aparato de la reivindicación 16, en donde se puede seleccionar cada uno del primer indicador de estado y el segundo indicador de estado.

18. El aparato de la reivindicación 17, en donde, para cada primer indicador de estado incluido en la visualización, la selección del primer indicador de estado da lugar a que el dispositivo de servidor transfiera, del primer usuario al usuario de servidor, usando al menos un procesador, la propiedad de los metadatos asociados con el primer indicador seleccionado.

19. El aparato de la reivindicación 17, en donde, para cada segundo indicador de estado incluido en la visualización, la selección del segundo indicador de estado da lugar a que el dispositivo de servidor transfiera, del usuario de servidor al primer usuario, usando al menos un procesador, la propiedad de los metadatos asociados con el segundo indicador seleccionado.

20. El aparato de la reivindicación 16, en donde adicionalmente, para cada metadato de la pluralidad de metadatos: el primer indicador de estado indica que el primer editor usuario es propietario de los metadatos y ha extraído los metadatos, el segundo indicador de estado indica que el primer usuario editor no tiene la propiedad de los metadatos y está habilitado para extraer los metadatos, y el tercer indicador de estado indica que el primer usuario editor no tiene la propiedad de los metadatos y no está habilitado para extraer los metadatos.

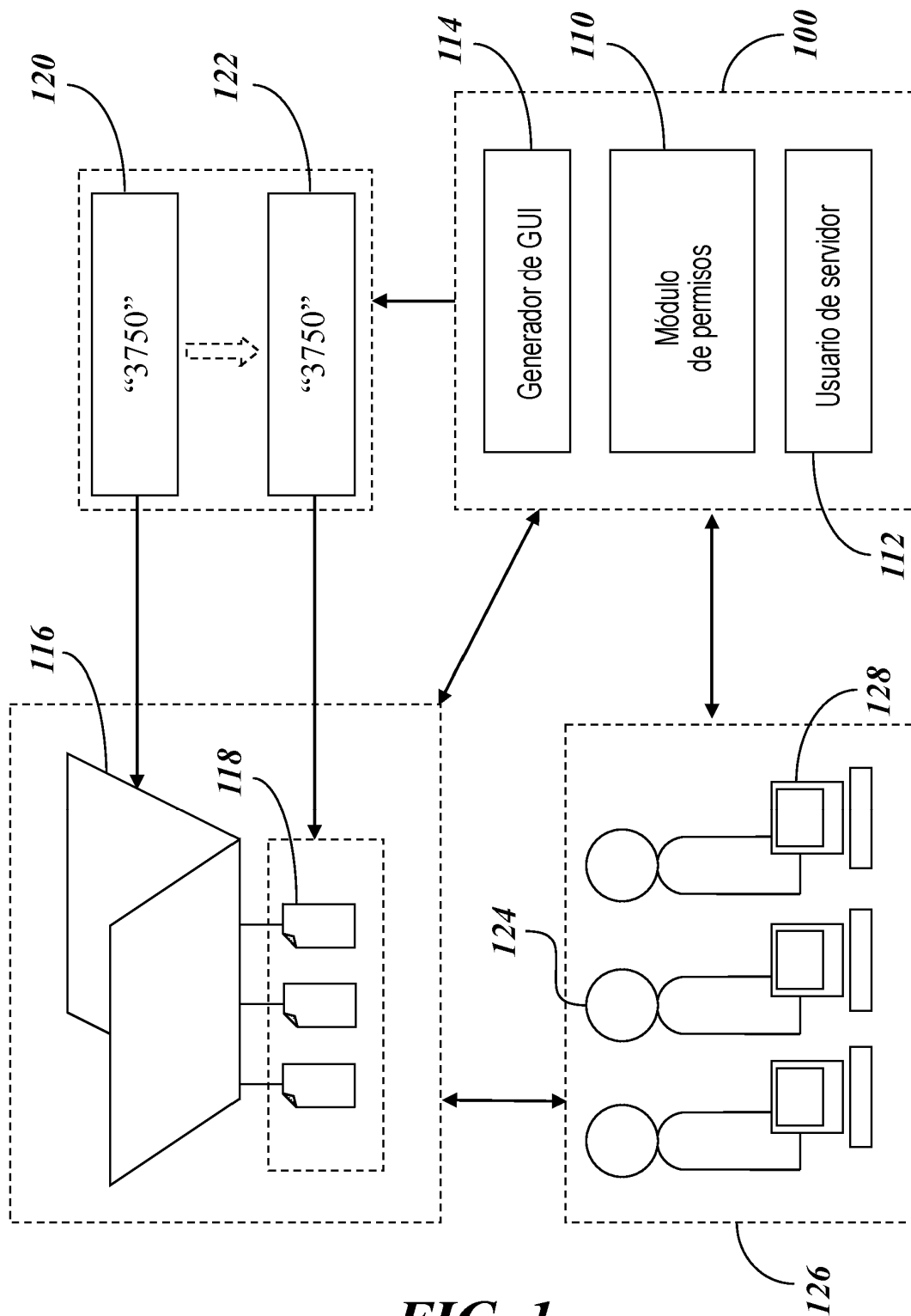


FIG. 1

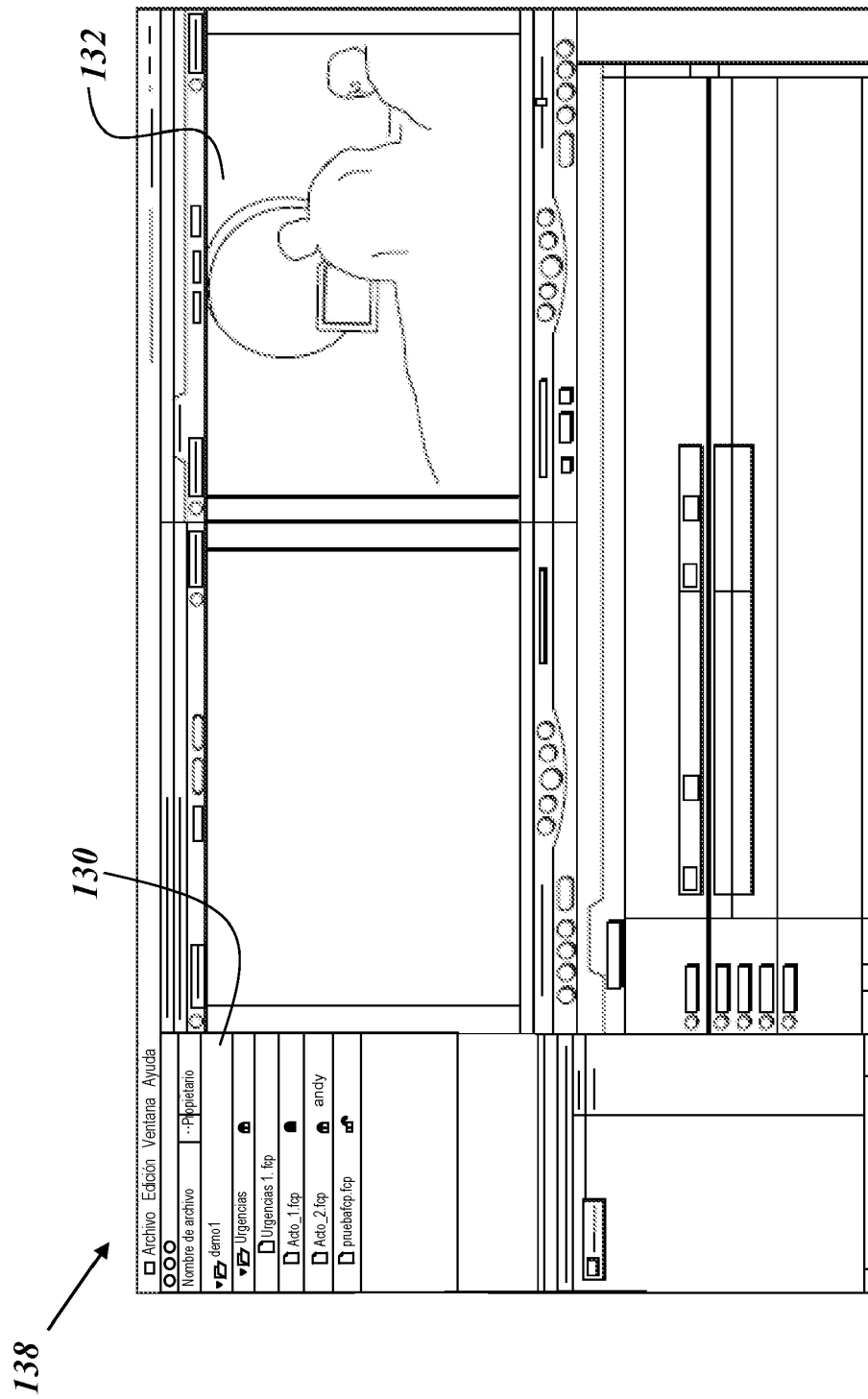


FIG. 2

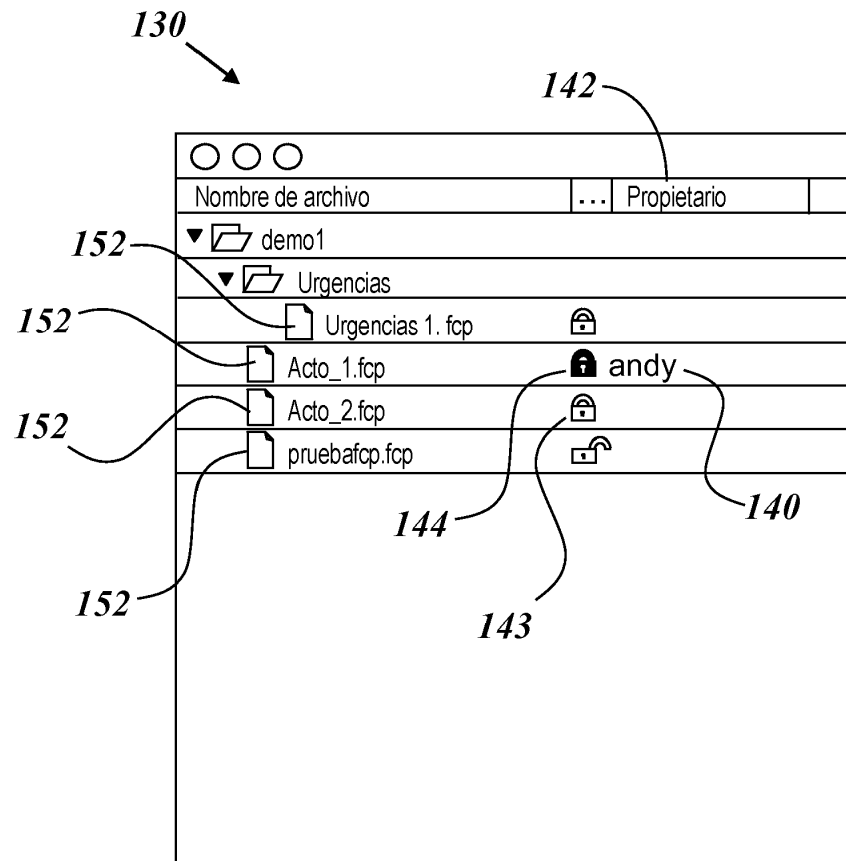


FIG. 3A

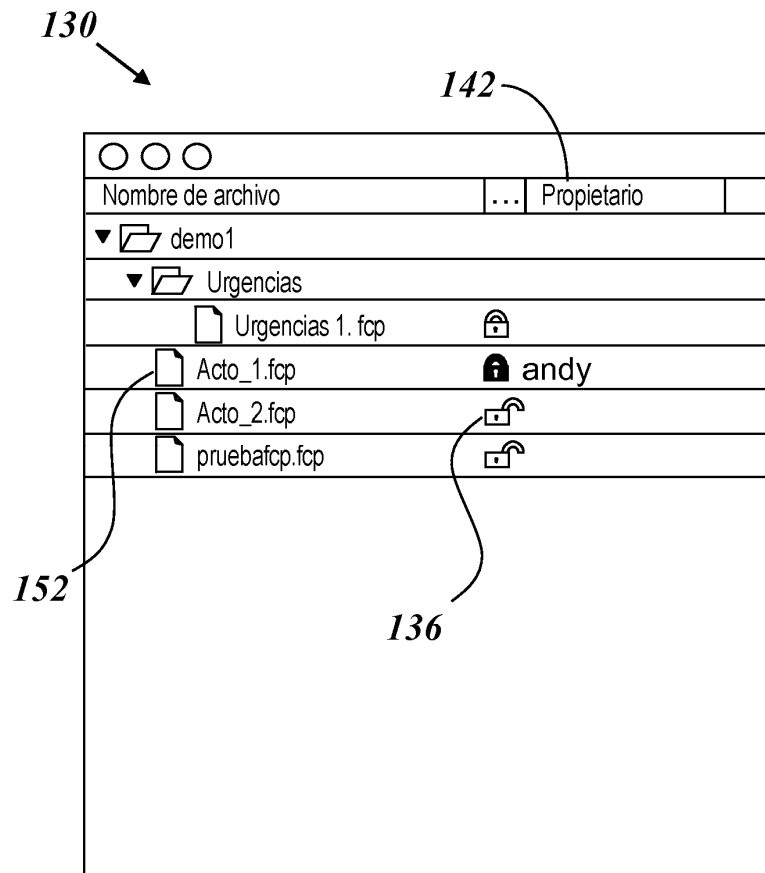


FIG. 3B

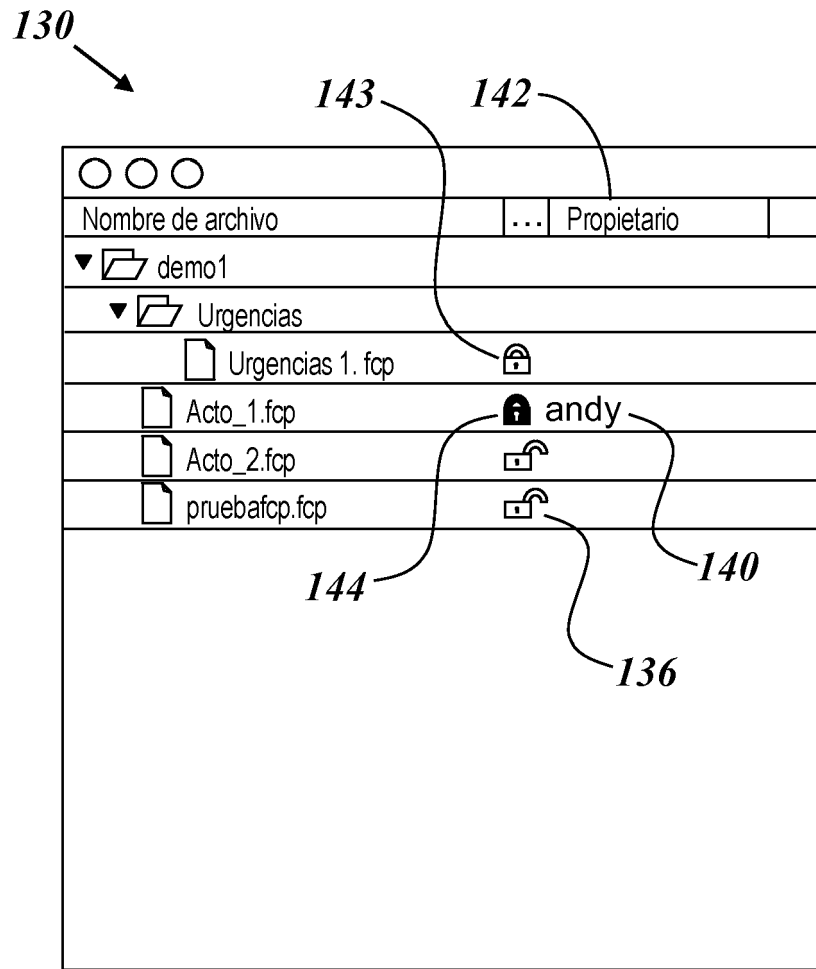


FIG. 4A

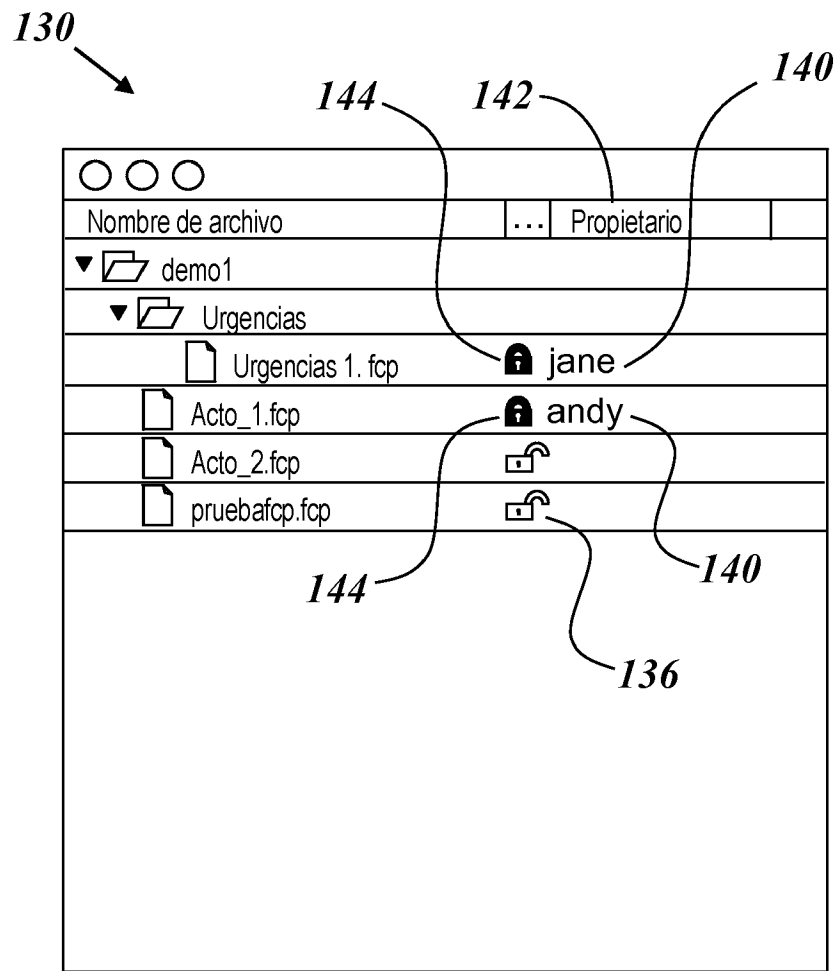


FIG. 4B

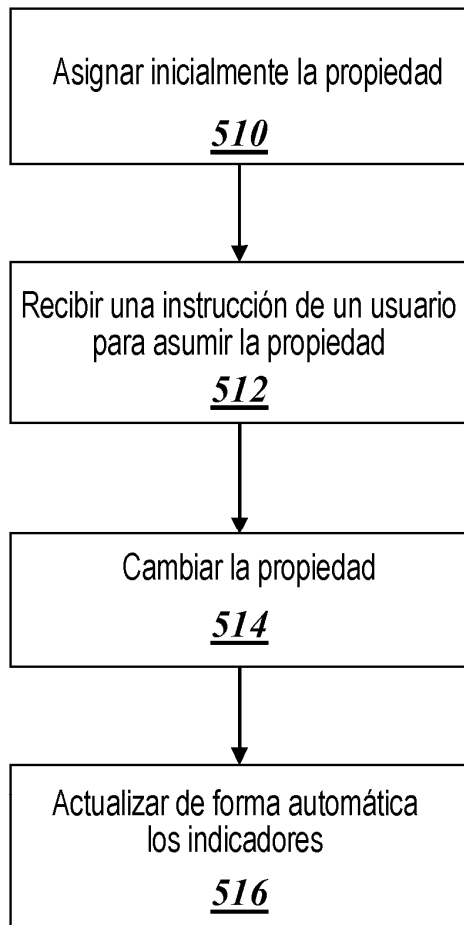


FIG. 5A

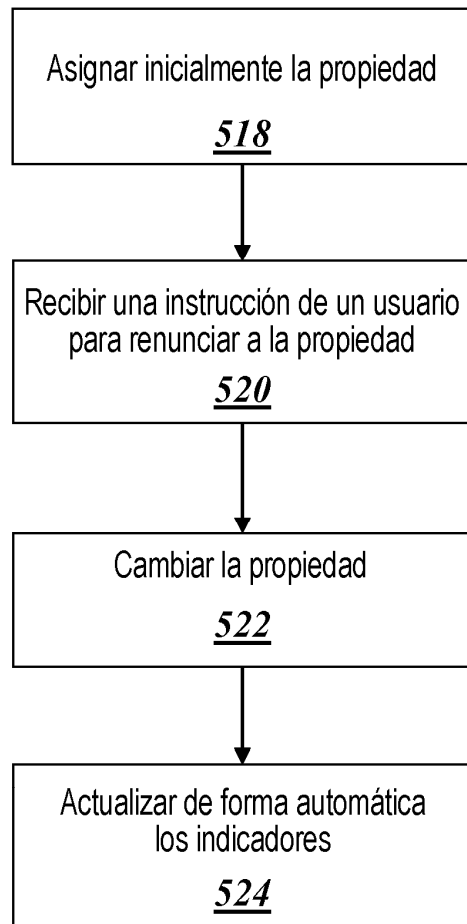


FIG. 5B

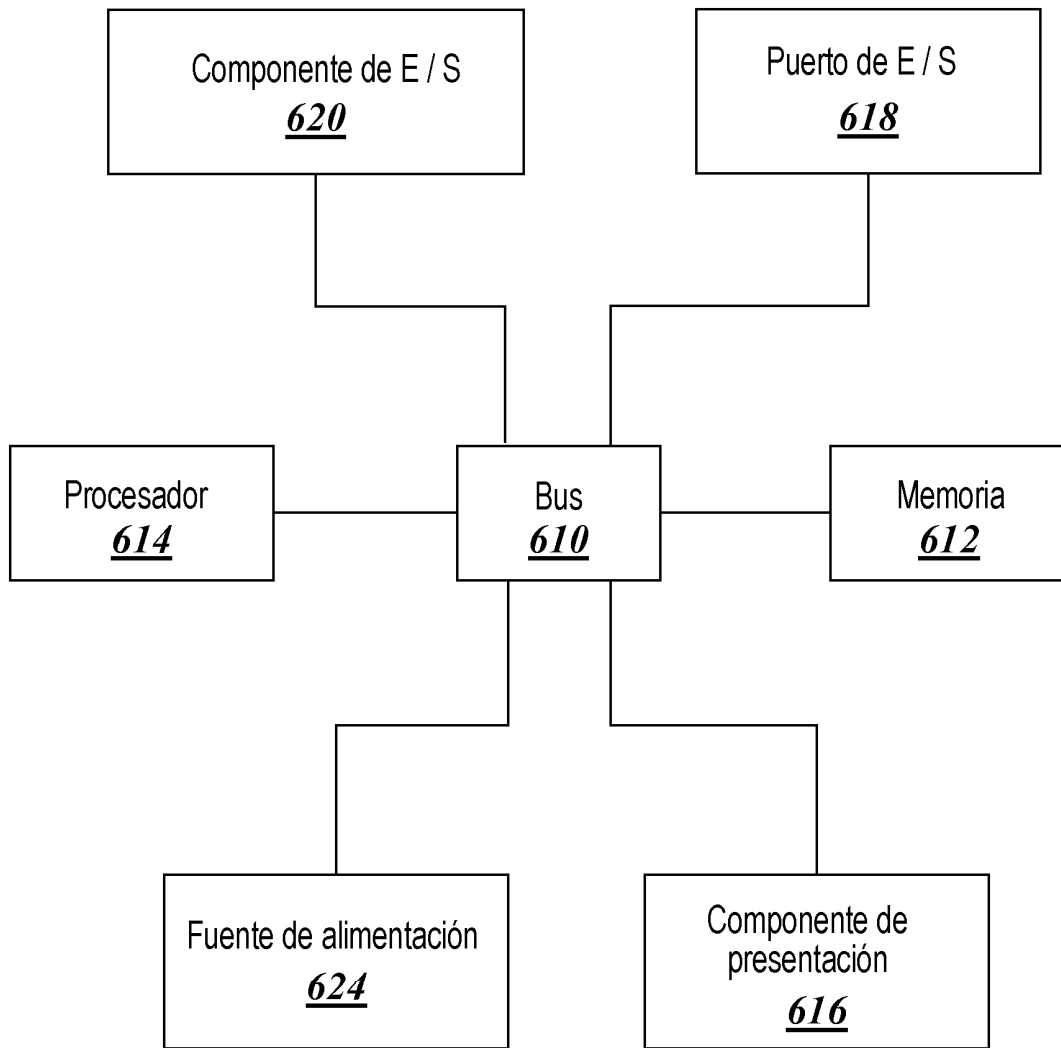


FIG. 6

```

[ raiz @es-vmwaredemo demo1]# ls -lah
total 68K
drwxrws--T 4 editar compartir editores 4,0K 06-04-2011 10:52 ./
drwxrws--T 5 editar compartir editores 4,0K 05-04-2011 14:05 ../
-rw-r----- 1 andy editores 0 05-04-2011 13:35 Acto_1.fcp
-rw-r----- 1 editar compartir editores 0 05-04-2011 13:36 Acto_2.fcp
-rw-r----- 1 jeff editores 45K 05-04-2011 14:07 prueba.fcp
drwxrws--T 2 editar compartir editores 4,0K 05-04-2011 14:07 recurso.frk/
drwxrws--T 3 editar compartir editores 4,0K 05-04-2011 14:18 Urgencias/
-rw-r----- 1 editar compartir editores 20 05-04-2011 13:10 .sps_lp
[ raiz @es-vmwaredemo demo1]# █

```

FIG. 7

```

[raiz @es-vmwaredemo demo1]# ls -lah
total 68K
drwxrws--T 4 editar compartir editores 4,0K 06-04-2011 10:52 ./
drwxrws--T 5 editar compartir editores 4,0K 05-04-2011 14:05 ../
-rw-r----- 1 andy editores 0 05-04-2011 13:35 Acto_1.fcp
-rw-r----- 1 jeff editores 0 05-04-2011 13:36 Acto_2.fcp
-rw-r----- 1 jeff editores 45K 05-04-2011 14:07 prueba.fcp
drwxrws--T 2 editar compartir editores 4,0K 05-04-2011 14:07 recurso.frk/
drwxrws--T 3 editar compartir editores 4,0K 05-04-2011 14:18 Urgencias/
-rw-r----- 1 editar compartir editores 20 05-04-2011 13:10 .sps_lp

```

FIG. 8

```

[raiz @es-vmwaredemo Urgencias]# ls -lah
total 60K
drwxrws--T 3 editar compartir editores 4,0K 05-04-2011 14:18 ./
drwxrws--T 4 editar compartir editores 4,0K 06-04-2011 10:52 ../
drwxrws--T 2 editar compartir editores 4,0K 05-04-2011 14:18 recurso.frk/
-rw-r----- 1 editar compartir editores 47K 05-04-2011 14:18 Urgencias 1.fcp

```

FIG. 9

```

[ raiz @es-vmwaredemo Urgencias]# ls -lah
total 60K
drwxrws--T 3  editar compartir  editores  4,0K 05-04-2011  14:18 ./
drwxrws--T 4  editar compartir  editores  4,0K 06-04-2011  10:52 ../
drwxrws--T 2  editar compartir  editores  4,0K 05-04-2011  14:18 recurso.frk/
-rw-r----- 1 jane             editores  47K 05-04-2011  14:18 Urgencias 1.fcp

```

FIG. 10

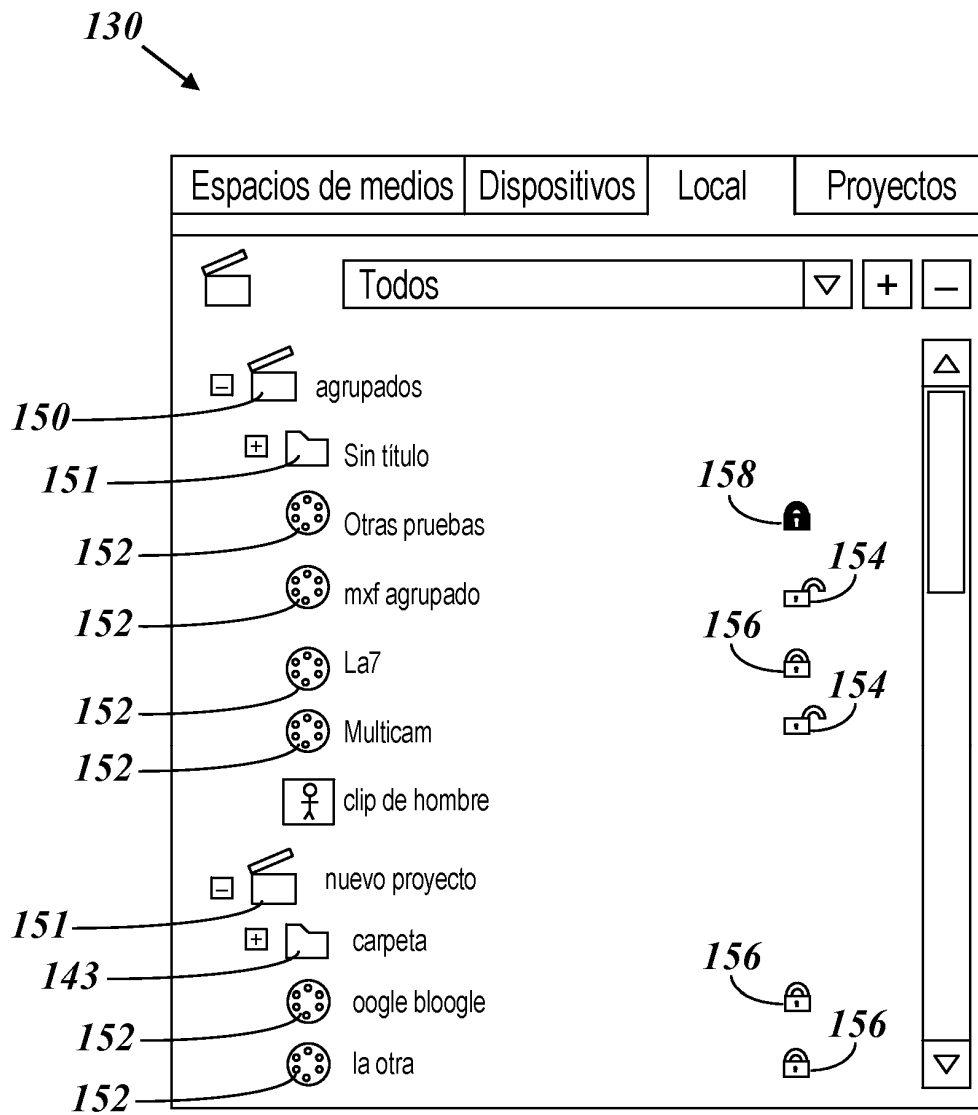


FIG. 11